Extending a Mathematical Theory of the Emergence of Knowledge from the Experience to Capture Learning Dynamics in Transformers

Sabrina Jones (sabjones@stanford.edu)
Neurosciences Interdepartmental Program, Stanford University

James L. McClelland (jlmcc@stanford.edu)
Department of Psychology, Stanford University

Abstract

The Transformer architecture used in LLMs has garnered widespread attention due to these model's human-like conceptual knowledge and language understanding, yet understanding how these models' capabilities result from experience-guided learning, and connecting this learning process with the structure in their training data, can seem intractable. Here we present preliminary steps to characterizing the developmental trajectory of a minimal Transformer trained on a next-token prediction task, using a simple dataset with quantifiable uncertainty and a simple, intuitively characterizable structure that captures some aspects of natural semantic structure learned by LLMs from large datasets. We show how the dynamic learning process of this model is a predictable consequence of the structure of the training data, exhibiting attested features of human semantic development, as captured in a theory of neural network learning dynamics (Saxe *et. al.* 2019) previously used to capture such dynamics in a network originally introduced by Rumelhart & Todd (1993).

Keywords: learning dynamics; semantic cognition; concepts; development; neural networks; language models; transformer

Introduction

What is the nature and format of human concepts; where does this conceptual knowledge originate; and how does the structure of experience affect it? Early approaches to these questions centered around propositional representations (Quillian, 1969; Fig. 1a), but experimental observations challenged this, and graded, prototype-based approaches were proposed (Rosch *et al.,* 1976; See Medin & Smith, 1981 for review). With the advent of the back-propagation learning algorithm, Rumelhart and Todd (1993) suggested that many properties of concepts could be understood as emerging through connection-based learning in neural networks, guided by experience with the conceptual facts in Quillian's propositional hierarchy. Ensuing work by Rogers & McClelland (2004; 2003), (henceforth R&M) explored how this framework captures features of human conceptual knowledge and its development. Saxe, McClelland & Ganguli then provided a mathematical analysis of the developmental process (2019), relating the statistical structure of experience as captured in a model's training data to learning dynamics of deep linear neural networks.

The Rumelhart & Todd (R&T) model exemplifies a connectionist perspective: conceptual knowledge is implicitly stored in network connections shaped by experience-driven learning. However, R&T framed learning as mapping abstract inputs, the first two propositional elements (e.g. 'robin can'), to outputs, third elements of all propositions starting with 'robin can' ('grow, move, fly'). This task setup and the network used to learn it were purpose-built and left open questions about how the relevant learning occurs more naturally. Additionally, their work's relevance to natural human concepts was questioned; critics argued, in commentaries in Rogers & McClelland (2008), that real semantic information cannot emerge solely from mechanisms that learn the statistical structure of data.

This debate continues in the era of transformers (Vaswani *et al* 2017) and large language models (LLMs), such as OpenAI's GPT models (e.g., Brown *et al,* 2020; GPT-3). As in other connectionist models, these models' conceptual knowledge is implicitly stored in connections and arises from experience-driven learning. While trained models' cognitive abilities are widely-studied, their emergence during training remains largely unexplored. We see this as an important gap, since connectionist models capture key aspects of human cognitive and conceptual development. Here we focus on two such aspects: (a) progressive differentiation, an initial tendency to make broad distinctions (e.g. animate vs inanimate), only later differentiating within these categories (as seen in human experiments by, e.g., Keil, 1979 and Mandler *et al.,* 1991), and (b) U-shaped developmental tendencies in patterns of (over)-generalization, such as attributing legs to any animal, including those lacking these typical properties (Gelman & Williams, 1998). Given the above, an exploration of whether these phenomena are captured in todays' performant LLM-based learning systems remains relevant to ongoing debates about what must be built in and what can be learned (Chomsky, 2023; Piantadosi, 2024). Therefore, in this work, we begin to explore whether, and how, transformers exhibit development dynamics consistent with statistical structure in their training data in accordance with the mathematical analysis provided by Saxe *et al.* (2019).

Addressing these questions may seem intractable. LLMs are trained on natural, human-generated text datasets that possess immense complexity and scale, reaching ~15 trillion words. Further, these models' immense size and complexity makes analysis difficult. To make a start, we make two key simplifications. First, we start with the small, human-designed database used by R&T and adapt it to form

In D. Barner, N.R. Bramley, A. Ruggeri and C.M. Walker (Eds.), *Proceedings of the 47th Annual Conference of the Cognitive Science Society*

sentences we use to train a transformer-based model on a next-token prediction task. Second, we use a minimal instantiation of the transformer architecture to make exploration of its learning dynamics more tractable, showing how this can be even further reduced to understand learning dynamics. We then explore how the learning dynamics of such systems relates to the basic aspects of the structure of conceptual knowledge that may be implicit in their training data. As we will argue below, both the data and the architecture/learning task capture aspects of the much more complex natural data and architectures/learning tasks faced by human and machine learning systems.

Datasets and Decomposition

We use the dataset from R&T (1993) based on Quillian's hierarchical model (Fig 1a), mapping 8 items and 4 relations (e.g. can) to 36 completions. This *Rumelhart Original* (RO) dataset paired eight one-hot item vectors with four one-hot relation vectors, mapped to multi-hot output vectors for all true propositions (e.g., "robin can [move, fly, grow]".

From this, we create a transformer training corpus of 92 simple sentences, named the *Rumelhart Conditional Probability* (CP) dataset. It has a 12-token input vocabulary (8 items, 4 relations) and a 36-token output vocabulary (all true completions). Each sentence pairs an item and relation (e.g., "robin can") with a single completion (e.g., "move"). Like in natural text, where sequences often have multiple valid completions, this inherent uncertainty introduces irreducible uncertainty in next-token prediction.

The presence of uncertainty takes a step toward greater alignment with human semantic experience, while allowing exact quantification of irreducible uncertainty, something not possible in real-world datasets. Irreducible uncertainty is defined as the cross-entropy (CE) between conditional probabilities and perfect predictions that could be made if the target completion was completely predictable. When training a model to minimize CE, its *irreducible loss (*Arora and Goyal, 2023) equals the irreducible uncertainty. Learning minima are set by item+relation conditioned probabilities; for example, "robin can" has three valid completions (fly, move, grow), each with a ⅓ probability. In the CP dataset, the irreducible loss is ~102.97.

Fully characterizing RO or CP datasets requires a third-order tensor (completion × relation × item). However, Saxe *et al.* (2019) showed that learning dynamics in the R&T model can be approximated using a simpler training set and network, collapsing over the relation dimension and mapping item vectors to multi-hot completion vectors in RO (Fig. 1b). We extend this to item+relation conditional probability vectors in CP (Fig. 1f). This corresponds to a sum over the items of the outer product of its one-hot item vector (x_{item}^{μ}) with its completion vector (y^{μ}), denoted as Σ^{OI}.

In the CP case, these outer products can also be understood as the sum over all 92 sentences of each outer product of the one-hot item and its completion, scaled by its relation-specific conditional probability (1/n), where n is the number of dataset items with a given item+relation input:

$$\Sigma^{OI-CP} = \sum_{\mu=1}^{92} \frac{1}{n} [y^{\mu} (x^{\mu})^{T}] \qquad \textbf{(Eq. 1)}$$

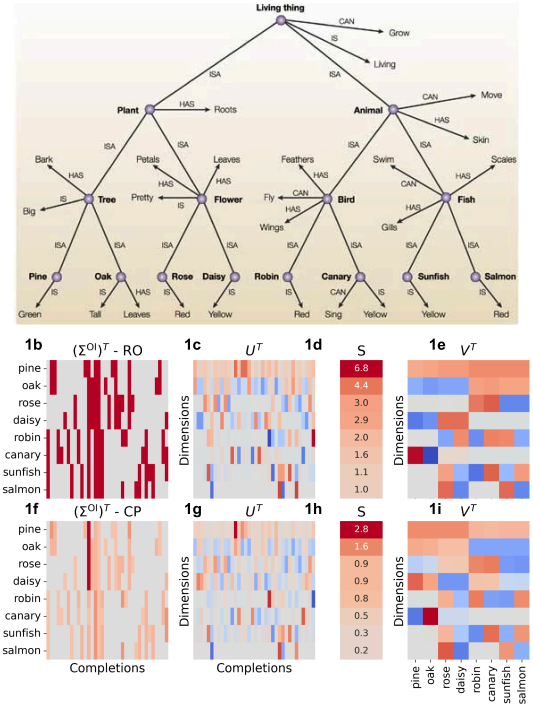

Figure 1: Dataset and its structure. (a) Quillian's propositional hierarchical framework of living things. Arrows represent propositions with the subject, relation, and attribute at the tail, shaft, and point, respectively (from McClelland and Rogers, 2003). (b) RO corresponding to $\Sigma^{OI\text{-}RO}$, transposed for interpretation. Red is used for positive quantities in [0,1]. (c-e) Resulting U,S, and V matrices from SVD of $\Sigma^{OI\text{-}RO}$ arranged to show each dimension's feature synthesizing *u* vector, strength, and item analyzing *v* vector on the same row. (f-i) show the same as (b-e) for CP.

With these 2D dataset representations, we characterize the underlying structure that guides experience- driven learning. Using Σ^{OI}, we apply a singular value decomposition (SVD), yielding three matrices: U, S, and V. V contains object-analyzing vectors, distinguishing item categories (e.g., birds vs. fish) while U consists of feature-synthesizing vectors. S is a diagonal matrix representing each dimension's strength, a measure of how much of the total dataset variance is captured by a dimension. The SVD of Σ^{OI} for both RO and CP produces similar decompositions (Fig. 1c-e; Fig. 1g-i) though their alignment with Quillians' hierarchy is less pure in the CP case. Prior work showed this decomposition enabled analysis of experience-driven learning of the RO dataset, showing how features of human conceptual development emerge in networks trained via backpropagation (Saxe *et al.*, 2013; 2019). We extend this analysis to minimal transformers trained on the CP dataset.

Model, Training and Loss Measures

We use a minimal decoder-only transformer (Fig 2) consisting of an embedding layer; one transformer module (Vaswani *et al.* 2017); and a final projection to an output layer with a softmax function that transforms the output into a vector with elements of predicted probabilities (*p)*, one for each possible completion. Inside the transformer module (dashed box), there is an attention module (AM) with one attention head and a feedforward network (FFN). Around the AM and FFN are residual connections (red arrows) that add the input to each module into its output via vector addition (circles) and layernorm operations. Weight matrices are initialized to span the full dimensionality of each matrix (Saxe *et al,* 2014) with orthogonal dimensions of magnitude 0.5 (default initial values of 0 and 1 are used for the learnable *beta* and *gamma* parameters of the layernorm operation). Inputs are one-hot vectors for each item and relation of a single sentence (e.g. 'robin has') and, with no additional context, the model is trained to predict the final word of the sentence (e.g. 'wings') when the preceding word is the item and the current word is the relation. Given this setup, the output is determined predominantly by the relation via the residual connections and by attention to the preceding item via the AM (light blue box in Fig. 2) (we shall see that in our setup, the FFN does not come to play a role). Models were trained using gradient descent with one gradient update after each epoch (i.e., full sweep through the training dataset) with a categorical cross-entropy (CCE) loss function, the standard loss for next token prediction.

We decompose the total loss into the *loss-between* and *loss-within*. Loss-between reflects predictions of completions inconsistent with the relation, defined in *Eq. 2,* where sp_c is the sum of the p's of relation-consistent completions (indexed by k) for an example sentence.

$$L_{btwn} = \sum_{n=1}^{92} [-\ log(sp_c)]\ ;\ sp_c = \sum_k p_k \qquad \textbf{(Eq. 2)}$$

To understand *Eq. 2*, note that for an example input such as "Robin has", any predicted probability of "fly" or other relation-inconsistent completion is implicitly reflected in sp_c, since the sum of all p's must add to 1. If the sum of relation-consistent probabilities is less than 1, the remainder must come from relation-inconsistent completions. This loss goes to 0 when the sum of the relation-consistent probabilities (sp_c) is 1, since log(1)=0.

Loss-within refers to loss produced by activating relation-relevant completions that are not the target for a given sentence. This quantity depends on the relation-normalized predicted probability of the target, rnp_t as given by *Eq. 3,* where p_t is the predicted probability of the target completion and sp_c is as previously defined:

$$L_{wthn} = \sum_{n=1}^{92} [-\ log(rnp_t)]_n\ ;\ rnp_t = \frac{p_t}{sp_c} \qquad \textbf{(Eq. 3)}$$

Intuitively, rnp_t goes to 1 (and its log goes to 0) when the sum of the p's of all relation-consistent completions other than the target is 0. *P's* of relation-but-not-item appropriate completions, such as "roots" or "gills" for "Robin has", lower rnp_t from its max value of 1. Predictions of other true completions of the item-relation pair (here "skin" or "feathers") that are not the target in a given sentence also reduce rnp_t. Due to the irreducible uncertainty (that is, the fact that there is no way to tell which valid completions is the target), loss-between converges to the irreducible loss, as observed in results below.

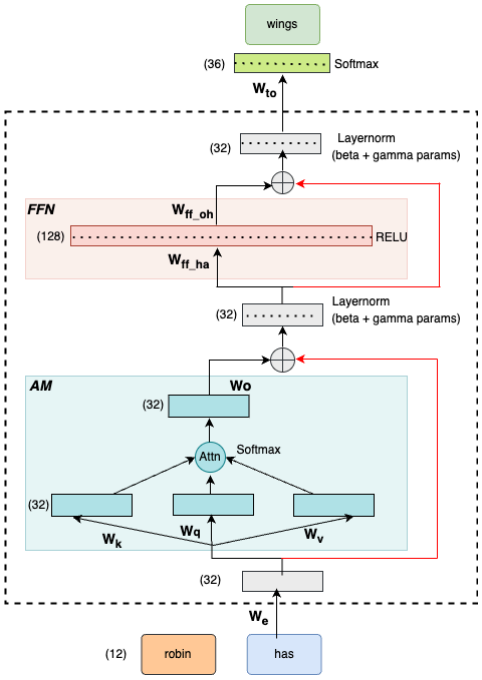

Figure 2: Minimal Transformer. Numbers show layer size. Modifiable parameters include weight matrices (e.g.$\mathbf{W_e}$) and Layernorm unit-wise beta and gamma parameters.

Results

Approximation to LLMs. Before exploring learning in our minimal setup, we confirmed that, despite the small scale of our model and our toy-dataset dataset, its learning captures aspects of the conceptual similarity structure learned in LLMs: we compared the embeddings of the 8 items from our dataset to the embeddings of these words in 10 open-weights, pre-trained LLMs, accessed through the Hugging Face API, and found that the similarity matrix of these LLM's embeddings has an average correlation of 0.73 with those learned in our model. Though our analysis will focus on the dynamics in the network's outputs, the embeddings also reflect the similarity structure in the data set and exhibit similar developmental trajectories; they approximately correspond to 'representation layer' in the R&T model; their dynamics were described by R&M.

Biphasic Learning

To obtain an understanding of the learning dynamics in our setup, we first explore how the time course of reduction of the total loss (L_{eval}) can be understood in terms of the loss between and within. We see that learning largely occurs in

two consecutive phases: a fast initial phase reflecting mainly a decrease in the loss-between (L_{btwn}) and a slower later phase reflecting a gradual decrease in loss-within (L_{wthn}).

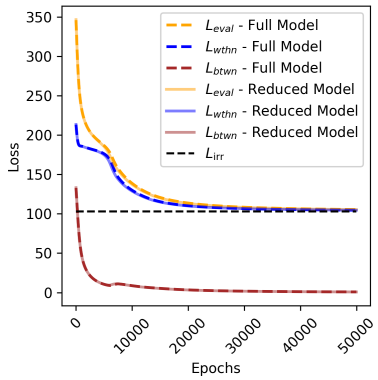

Figure 3: Transformer training loss for the full model (dashed lines) and reduced model of *Eq. 4* (solid lines). Curves show results using the same random initialization in one instance of each version of the model.

During the first phase, the network relies almost exclusively on the relation input. L_{btwn} can be reduced to 0 simply by taking the relation – the word immediately preceding the completion – into account. The relation also supports learning relation-specific, but item independent, predicted probabilities, which vary considerably ('grow' follows 'can' far more frequently that 'sing' for example), similar to the way words partially predict their immediate successors, resulting in the small initial drop in L_{wthn}. This occurs through adjustments in the embedding matrix $\mathbf{W_e}$ and the output matrix $\mathbf{W_{to}}$, bypassing the FFN and AM. In the second phase, to further decrease L_{wthn}, the model must incorporate the item in its prediction to maximize np_t (*Eq. 3)*. In the transformer, this occurs through the AM. In our setup, adjusting only $\mathbf{W_v}$ and $\mathbf{W_o}$ is sufficient to allow the item preceding the relation to affect the output. Minimal learning occurs in $\mathbf{W_q}$ and $\mathbf{W_k}$; attention is initially evenly distributed across the item and the following relation, and, in our task, there is little pressure to change this given that there is only one item in context. The item information via the AM is merged with the relation information from the residual stream and normed above the AM, then proceeds to the output through the second residual connection, not needing to engage the FFN, to the output layer. Here, the model learns to condition the output on both inputs, decreasing the loss-within to the irreducible loss (L_{irr}).

To support the description we provide above, we show in Fig. 3 that the reduced model described by *Eq 4,* where $\hat{y}$ is the output for a given item-relation pair (x_i,x_r), captures the learning trajectories of the full model:

$$\hat{y} = Sm(W_{to}[[W_o(.5W_vW_ex_i + .5W_vW_ex_r) + W_ex_r]]) \quad \textbf{(Eq. 4)}$$

Here the 0.5's represent the fixed, evenly distributed attention scores assigned to the item and relation in the AM, *Sm* stands for softmax, and []'s denote the add-norm operation. *Eq.4* can be further reduced because the weights in W_v that could propagate x_r through the AM stay small making the contribution of $.5W_vW_ex_r$ negligible:

$$\hat{y} = Sm(W_{to}[[W_o.5W_vW_ex_i + W_ex_r]]) \quad \textbf{(Eq. 5)}$$

From this, we can define a *short pathway*, propagating information about the relation through the product of 2 weight matrices (W_{to}, W_e) and a *long pathway*, propagating information about the item through 4 matrices (W_e ,W_v ,W_o, W_{to}). Extensions of these ideas may be relevant to learning in large LLMs trained with larger datasets (see *Discussion*).

Decomposing the Model Activations

In order to understand the relationship between the developmental trajectory of the model and the structure of the dataset more fully, we apply a decomposition to the network activations like we did with the training data. Thus, at evaluation epochs, network activations were recorded for all examples. Then, we substitute $\hat{y}$, the network activation for input μ at time *t,* for the target *y* in *Eq. 1* producing a matrix we call Σ^{YX}. This representation of activations is arranged the same way as $\Sigma^{OI\text{-}CP}$ in Fig. 1 and is shown for one seed at two points during training (Fig. 4a,b, left). We then apply SVD to these matrices, allowing us to observe the structure in model outputs at different time points.

Early in training, some of the dimensions reflect random structure due to random weights. To observe the emergence of $\Sigma^{OI\text{-}CP}$'s dimensions in Σ^{YX}, we compute the outer product of the vectors of U and V^T of $\Sigma^{OI\text{-}CP}$ and project Σ^{YX} onto these dimensions at successive time points to observe the trajectories of these dimensions (Fig. 4c). Each curve represents the median, across seeds, of the per-seed projection strengths, for each dimension.

Our understanding of loss-within and between is strengthened by considering the dataset's dimensions. The rapid decrease we observe in loss-between is a consequence of learning an item-independent representation of D_1, the base rates (Fig 4a, U^T,S, and V^T row 1). The activations for each example will approximate the *u* vector for D_1 scaled by its strength. Using *Eqns. 2* and *3*, we see that learning this portion of D_1 drives a dramatic decrease in loss-between to ~5.6 and a moderate reduction in loss-within to ~188. The curve for D_1 starts with a non-zero strength because the initial uniform output of the network has a positive projection onto the relation-conditioned base rates captured by this dimension, and rises quickly in part because it depends only on the short pathway in *Eq 5*.

To further decrease the loss, item information must be used in the output computation via the AM. The remaining item-dependent dimensions (Fig 4b V^T rows) can be learned progressively, driving this gradual decrease of loss-within to the irreducible loss. The gradual nature of learning $D_{2\text{-}8}$ results from their weaker strengths and the long pathway's greater complexity (*Eq 5*), further explained below.

We can further understand these projection strengths' trajectories, first noting they asymptotically approach the true strengths of $\Sigma^{OI\text{-}CP}$'s dimensions. As noted above, D_1

corresponds to relation-specific base rates, and arises quickly during the first learning phase. $D_{2\text{-}8}$ exhibit a stage like emergence; stronger dimensions "come online" first.

The dynamics that govern the trajectories of dimensions in a deep linear network with one hidden layer and two learnable weight matrices trained using SGD with mean squared error (MSE) were described analytically by Saxe *et al.* (2013,2014,2019). This equation describes the trajectory a of the strength of Σ^{YX} dimension α:

$$a(t, s^{\alpha}, a_0^{\alpha}) = s^{\alpha}\left[\frac{e^{2s^{\alpha}t/\tau}}{e^{2s^{\alpha}t/\tau} + \frac{s^{\alpha}}{a_0^{\alpha}} - 1}\right]; \tau = \frac{1}{c * lr} \quad \textbf{(Eq.6)}$$

Eq. 6 describes a sigmoidal trajectory that is influenced by s^{α}, Σ^{OI} dimension α's strength, and a^{α}_0, the initial strength of dimension α in Σ^{YX}. τ is a constant where *lr* is the learning rate and *c* is the average contribution of an example to the loss. In MSE loss, $c = 1/P$ as loss is averaged over P training items. The presence of s^{α} as a multiplier and as a factor in the exponential term captures these qualitative signature properties of these equations: as s increases, the asymptote and steepest slope of each curve increases while the timing of the sigmoid transition (captured in the time to reach half of the asymptotic value) decreases.

Though this theory was derived for a linear network, unpublished simulations show that the signature properties are preserved for $D_{2\text{-}8}$ in the more complex R&T network trained with the RO dataset. Here, we show that these signature properties are also preserved in our minimal Transformer, despite differences in loss function, number of learnable weight matrices, added non-linearities, and the shift to the CP dataset. That is, the asymptote and maximal slope of each curve increases with s^{α} ,while the time to reach half of the asymptotic value decreases (Fig. 4c- see legend).

Developmental Implications

Having extended an analytical understanding of the relationship between training data structure and learning dynamics to a minimal transformer, we explore how this explains model behavior during development mirroring features of human semantic development.

Progressive Differentiation. In the SVD of $\Sigma^{OI\text{-}CP}$, we can see the feature-synthesizing vectors of U and the item-analyzing vectors of V^T become progressively more focused from dimension 1-8 (Fig. 1g-i). This is because the dimensions progressively encode finer differentiations. For example, in D_2's item-analyzing vector, plants are positively represented and animals negatively, and the corresponding feature-synthesizing vector has positive entries for completions that all plants share and negative entries for those of animals. Thus, D_2 separates plants and animals. In contrast, D_5 and D_6, are more specific: D_5 captures the shared red-yellow color alternation exhibited by three of the item pairs, while D_6 predominantly represents the differentiation between the oak and pine. D's 3 and 4 lie between, predominantly distinguishing the birds from the fish and the trees from the flowers, respectively.

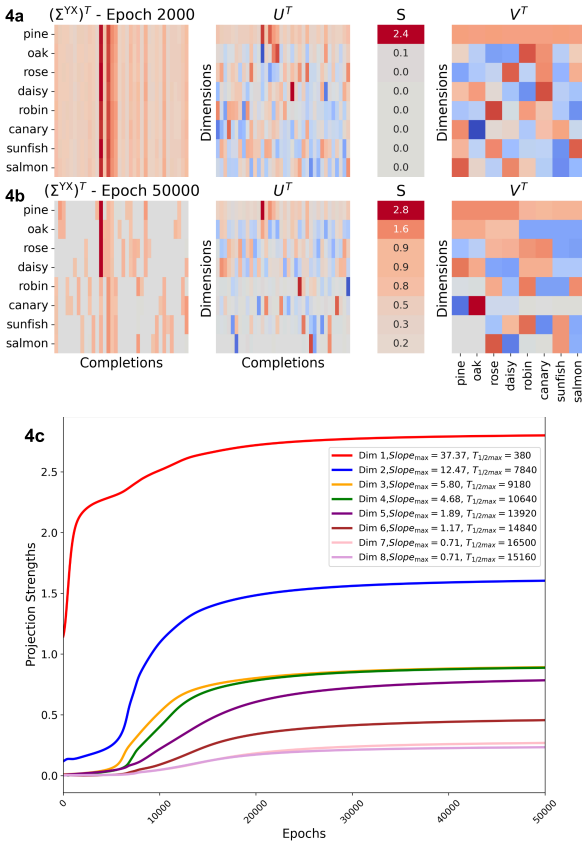

Figure 4: Transformer Activations Decomposition. (a,b) SVD on activations at epochs 2k and 50k. (c) Strengths of projections of Σ^{YX} onto dimensions of $\Sigma^{OI\text{-}CP}$ for dimension curve with the median max slope, across seeds. Max slope (multiplied by τ) and time of half max (in epochs) in legend.

The content of each dimension, together with the relationship between its time course of learning from *Eq. 6,* implies that the model's behavior will show progressive differentiation, moving down the dataset's hierarchical structure, capturing the early differentiation of broad conceptual distinctions and their subsequent refinement (Keil 1978; Mandler *et al.* 1991). We examined differentiation in the model's outputs at three levels (Fig 5a). We see that the Euclidean distance (ED) between average completion vectors for plants vs animals increases before that between trees and flowers, and the separation of individual trees occurs last. By superimposing the curves for the projection strengths of $D_{2,4,6}$, which represent these hierarchical levels, we see that the trajectories of these curves nearly coincide. Thus, differentiation of categories results from the progressive learning of dimensions, as captured by the signature properties described above. The slight differences between the ED and corresponding D curves reflects that the dimensions in the CP data set are not as purely aligned with the taxonomy of those in RO (see Fig. 1e, i). Similar points apply to the relationship of the tree-flower distinction to D_4 and the pine-oak distinction to D_6. This imperfect alignment is characteristic of small-scale

natural datasets (McClelland, *et al.* 2016) and seem likely to apply to larger natural datasets as well.

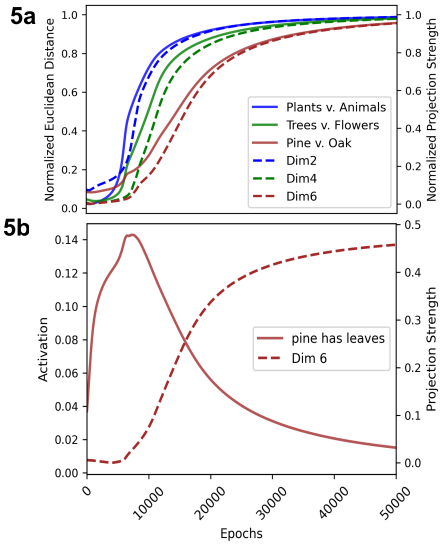

Fig 5. Model Developmental Behavior. (a) Progressive differentiation of concepts. Euclidean distance between different hierarchical classes over time (solid lines), normalized by max separation and normalized trajectories of relevant singular dimension strengths (dashed lines). (b) U-shaped activation of "leaves" for "pine has" and projection strength of dimension that separates trees.

U-Shaped Learning. Another characteristic of human semantic development are inverted "U-shaped" developmental trends. These capture phenomena such as children in a certain age range assigning features typical of a class to all class members (e.g. ascribing legs to animals without visible legs, Gelman & Williams, 1998). In our dataset, a case in point is the pine tree, the only plant that does not have leaves. The activations for the completion "leaves" for the input "pine has" exhibit this inverted U-shaped trajectory, with the model first increasing its prediction of leaves for "pine" and later reversing this (Fig. 5b). We can understand this as a direct result of the progressive learning of $\Sigma^{\text{OI-CP}}$'s dimensions. The information that separates pine from oak is contained in D_6, and the vertex of this inverted U co-occurs as D_6 begins to come on line. Again, the developmental behavioral patterns displayed by the model follow from the way the structure of the dataset influences learning dynamics.

Discussion

Do transformers learn in a unique way, or does experience guide their learning and behavior in a way that mirrors human semantic development? LLMs, based on the Transformer architecture, have set new standards for neural networks due to seemingly human-like cognitive abilities, and they often behave in human-like ways. Thus, the question of *how* these models come to know what they do is of great intrigue. It can also seem intractable, given the scale and complexity of the models and their training data.

We hope our work has taken some small steps toward shedding some light on this *how* question. We began with a simple human-designed data set capturing semantic information in a set of simple sentences like '*Robin has wings',* using it as training data for a minimal Transformer, and observed substantial similarities between the word representations (embeddings) learned in larger transformers trained with larger datasets and the representations learned in our much simpler setting. Analysis of our simple dataset enabled quantification of the irreducible uncertainty (equivalent to the irreducible loss). We introduced two novel loss metrics, associated with distinct phases of the model's learning, one primarily driven by exploiting the relation term immediately preceding each sentence prediction via a short pathway through the network, and another driven by recruiting the attention module to guide predictions using the item presented as the preceding word in context.

We then applied the theory of deep linear networks (Saxe *et al.* 2013, 2014, 2019) to further understand the model's learning. This analysis characterises learning in deep linear networks as the progressive mastery of singular dimensions that capture the (approximately) hierarchical structure in the dataset. Despite the additional complexities of the transformer, this theory serves as a strong method for understanding the dynamics of the model's predictions as it learns while also capturing characteristic patterns of human semantic development.

We see our model as capturing aspects of learning likely to be relevant to much larger models. For example, we think it likely that larger models rapidly exploit predictions from a given word to its immediate successor via residual connections, and progressively differentiate their predictions based on preceding words by progressively acquiring conceptual distinctions that are associated with recurring semantic motifs in their training data.

There is, to be sure, much more that remains to be captured. To perform as well as they do in exploiting information available to them in context, transformers must deploy attention selectively, and this is central to their impressive capabilities. Relevant contextual information may be separated from the current word context by many intervening words (Ollson *et al.* 2022), as in a sentence like 'The robin in the tree has wings.' Addressing such dependencies requires a model to represent conceptual and grammatical roles and learn attentional strategies to exploit these roles, requiring exploitation of the key and query matrices in transformer's attention modules. Other work with minimal transformers has shed light on how transformers learn these structures (Reddy, 2023) in an abstract-item-label binding task. We are intrigued to extend our models and our training sets to allow the exploration of how these characteristics emerge in data sets more aligned with the demands of extracting meaning from natural language.

References

Arora, S., & Goyal, A. (2023). A theory for emergence of complex skills in language models (preprint). *arXiv.* https://arxiv.org/abs/2307.16443

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

Chomsky, N. (2023). The false promise of ChatGPT. *The New York Times,* March 8, 2023.

Collins, A. M., & Quillian, M. R. (1969). Retrieval time from semantic memory. *Journal of Verbal Learning & Verbal Behavior, 8*(2), 240–247.

Gelman, R., & Williams, E. M. (1998). Enabling constraints for cognitive development and learning: Domain specificity and epigenesis. In W. Damon (Ed.), *Handbook of child psychology: Vol. 2. Cognition, perception, and language* (pp. 575–630). John Wiley & Sons, Inc..

Keil, F. C. (1979). Semantic and Conceptual Development: An Ontological Perspective. Harvard University Press.

Mandler, J. M., Bauer, P. J., & McDonough, L. (1991). Separating the sheep from the goats: Differentiating global categories. *Cognitive Psychology*, *23*(2), 263-298.

McClelland, J. L., & Rogers, T. T. (2003). The Parallel Distributed Processing Approach to Semantic Cognition. *Nature Reviews Neuroscience, 4*(4), 310–322.

McClelland, J. L., Sadeghi, Z. & Saxe, A. M. (2016). A Critique of pure hierarchy: Uncovering cross-cutting structure in a natural dataset. *Neurocomputational Models of Cognitive Development and Processing*, pp. 51-68. World Scientific.

Olsson C, Elhage N, Nanda N, et al. (2022). In-context learning and induction heads. *arXiv preprint.* arXiv:2209.11895.

Piantadosi, S. (2024). Modern language models refute Chomsky's approach to language. *Empirically Oriented Theoretical Morphology and Syntax 15*, 353–414. Berlin: Language Science Press.

Smith, E. E., & Medin, D. L. (1981). *Categories and concepts.* De Greuter.

Reddy, G. (2023). The mechanistic basis of data dependence and abrupt learning in an in-context classification task. In *The Twelfth International Conference on Learning Representations*.

Rogers, T. T., & McClelland, J. L. (2004). *Semantic cognition: A parallel distributed processing approach.* MIT Press.

Rogers, T. T., & McClelland, J. L. (2008). Précis of *Semantic cognition: A parallel distributed processing approach*. *Behavioral and Brain Sciences, 31*(6), 689–714.

Rosch, E., Mervis, C. B., Gray, W. D., Johnson, D. M., & Boyes-Braem, P. (1976). Basic objects in natural categories. *Cognitive psychology*, *8*(3), 382-439.

Rumelhart, D. E., & Todd, P. M. (1993). Learning and connectionist representations. In D. E. Meyer & S. Kornblum (Eds.), *Attention and performance 14.* (pp. 3–30). The MIT Press.

Saxe, A. M, McClelland, J. L, & Ganguli, S. (2013). Learning hierarchical categories in deep neural networks. *Proceedings of the Annual Meeting of the Cognitive Science Society,* 35.

Saxe, A. M., McClelland, J. L., & Ganguli, S. (2014). Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv.* https://doi.org/10.48550/arXiv.1312.6120

Saxe A.M., McClelland J.L., & Ganguli S.(2019). A mathematical theory of semantic development in deep neural networks, *Proc. Natl. Acad. Sci. U.S.A.* 116 (23)

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.