On the Geometry of Semantics in Next-token Prediction

Anonymous authors
Paper under double-blind review

Abstract

Modern language models demonstrate a remarkable ability to capture linguistic meaning despite being trained solely through next-token prediction (NTP). We investigate how this conceptually simple training objective leads models to extract and encode latent semantic and grammatical concepts. Our analysis reveals that NTP optimization implicitly guides models to encode concepts via singular value decomposition (SVD) factors of a centered data-sparsity matrix that captures next-word co-occurrence patterns. While the model never explicitly constructs this matrix, learned word and context embeddings effectively factor it to capture linguistic structure. We find that the most important SVD factors are learned first during training, motivating using spectral clustering of embeddings to identify human-interpretable semantics, including both classical k-means and a new orthant-based method directly motivated by our interpretation of concepts. Overall, our work bridges distributional semantics, neural collapse geometry, and neural network training dynamics, providing insights into how NTP's implicit biases shape the emergence of meaning representations in language models.

1 Introduction

Next-token prediction (NTP) is conceptually simple: predict the next word given a preceding sequence (aka context). Yet it trains models with remarkable ability to capture meaning.

A foundational step toward understanding how NTP leads to models acquiring semantic information is to first understand how explicit textual inputs—words and contexts—are represented in the models' d-dimensional word and context vector representations (embeddings). Zhao et al. (2024) recently proved that in well-trained models, the geometry of these learned representations is characterized by the factorization of a *data-sparsity matrix* $\widetilde{\boldsymbol{S}}$ whose entries encode whether a particular word follows a given context in the training corpus.

While this characterization explains how models encode explicit training signals, it does not address the deeper question about semantic learning. Language conveys meaning through latent concepts that exist beyond explicit (context, word) pairs, leading us to investigate: *How do NTP-trained models extract and encode these latent concepts?* For instance, how do they capture semantic dichotomies like male/female, or grammatical categories like nouns and verbs? In other words, *how does latent information shape the embedding geometry*, and does this suggest *ways to identify interpretable semantic structures in the embedding space?* These questions are challenging because linguistic concepts are never explicitly given as inputs in the NTP objective. Yet, by analyzing the geometry of learned representations and its relation to the data-sparsity matrix, we show that latent linguistic structure emerges from the latter. Concretely, our contributions are summarized as follows. Please see also a statement on the scope in A.1.

C1. Geometric emergence of latent concepts: We demonstrate that latent linguistic concepts learned by a large neural model through NTP emerge as principal components in the singular value decomposition of a centered data-sparsity matrix $\tilde{\mathbf{S}}$ (defn. in Sec. 3.1, see Fig. 1 row **(A)** for illustration). While $\widetilde{S}$ is never explicitly formed during training, its structure naturally shapes the geometry of learned representations: concepts can be recovered as weighted combinations of word and context embeddings, where the weights are determined by the singular vectors of $\tilde{\mathbf{S}}$. This extends the characterization of the unconstrained-features model (UFM) by Zhao et al. (2024), revealing how embeddings inherently organize around latent concept dimensions.

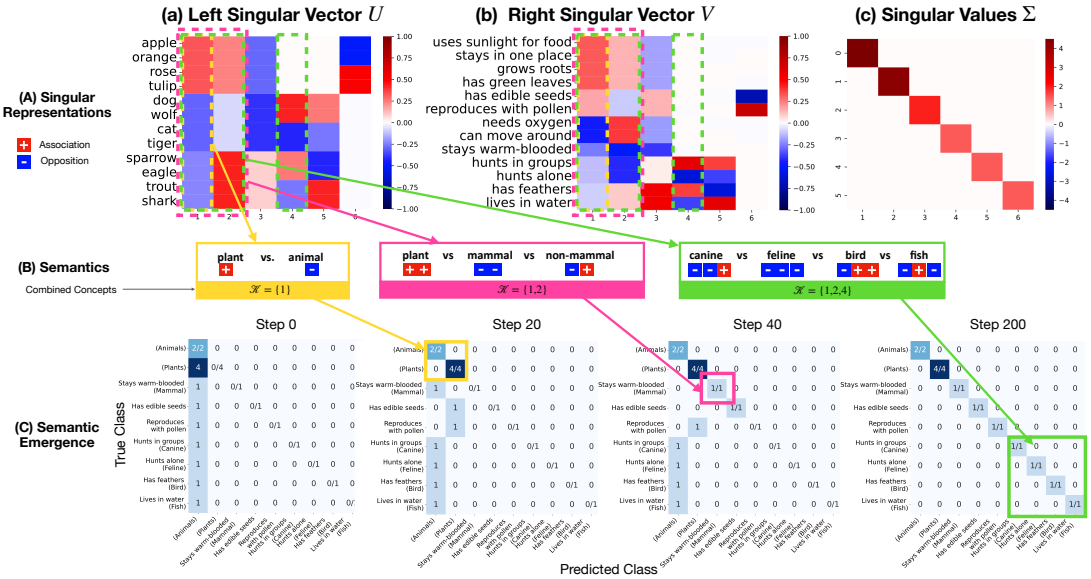

Figure 1: **(A) Singular Representations.** Semantic structure extracted via SVD of the centered data matrix $\widetilde{S}$ (rank = 6). Data is shown in Fig. 2C. (a) Left singular vectors U (word analyzer vectors): the first direction separates "plants" (e.g., *apple*, *tulip*) from "animals" (e.g., *dog*, *shark*); the second further separates mammals (e.g., *dog*, *cat*) from non-mammals (e.g., *eagle*, *trout*). (b) Right singular vectors V (context analyzer vectors): the first direction separates plant-specific contexts (e.g., "uses sunlight for food") from animal ones (e.g., "hunts in groups"); the second splits mammalian traits (e.g., "stays warm-blooded") from aquatic ones (e.g., "lives in water"). (c) Singular values, ordered by magnitude, indicate the relative importance of each semantic direction. **(B) Combined Concepts.** Higher-order combinations of singular directions yield finer-grained semantic partitions. Combining the first and second vectors gives distinction among plant, mammal and non-mammal. Including a fourth separates animal subtypes such as *canine*, *feline*, *bird*, and *fish*. **(C) Semantic Emergence Over Time.** Confusion matrices from a 2-layer transformer (embedding dim. = 128) trained on synthetic structured data at steps 0, 20, 40, and 200. Semantic classes are distinct support sets and are annotated in brackets (e.g., "Animals") when applicable. Early in training (e.g., step 20), broad distinctions like "Plant" vs. "Animal" emerge. Finer features (e.g., "Reproduces with pollen", "Lives in water") appear later, reflecting a coarse-to-fine learning pattern: high-variance (large singular value) concepts are acquired earlier, while specific, low-variance features emerge gradually.

C2. Rate of learning: Singular values of the data-sparsity matrix quantify the significance of their corresponding concepts. By recognizing a connection between UFM with square-loss and the closed-form training dynamics of two-layer neural networks with orthogonal inputs derived by Saxe et al. (2013), we show that concepts associated with larger singular values are learned faster during training.

C3. Semantics[1] as orthant-based clustering: Not all concepts have human-interpretable meaning. Guided by the insight that top concepts are learned first during training, we find that human-interpretable semantic information is captured through specific positive/negative configurations (indicating how present/absent the specific concept is in that word or context) across a few top singular concept dimensions (see Fig. 2B for illustration). Identifying semantics through such sign configurations geometrically manifests as (spectral) clustering of word/context embeddings in orthant slices. Through experiments, we verify that this approach identifies distinct human-interpretable linguistic categories.

C4. Connecting distributional semantics and neural collapse (NC) geometries: On the one hand, linking to classical distributional semantics approaches based on heuristic forms of matrix factorization of data co-occurrences, we find using model abstractions popularized in the NC literature that NTP implicitly favors SVD factorization of the centered data-sparsity matrix. On the other hand, we broaden the scope of NC geometries of embeddings to encompass geometries of latent semantics – making the connection explicit, we show the emergence of semantics even for the default one-hot data setting in the NC literature.

[1]We broadly define "semantic" to encompass linguistic factors that humans consider when producing text, including grammatical (e.g., tense), syntactic (e.g., verb), and semantic (e.g., sentiment) elements.

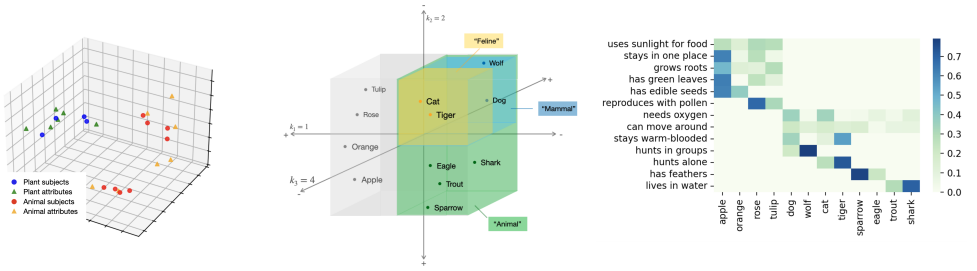

Figure 2: NTP-trained word embeddings naturally organize into semantic clusters. **(C)** Training data (showing $\boldsymbol{P}^\top$) consists of pairs using the template "The organism that [attribute] is [subject]", where attributes are rows and subjects are columns. **(A)** When trained with $d > V > rank(\widetilde{\boldsymbol{S}}) = 6$ and visualized in 3D PCA projection, both words (dots) and contexts (triangular marks) cluster by semantic category—animals (green/blue), and plants (red/orange). This demonstrates how semantic structure emerges naturally from the underlying data sparsity pattern. **(B)** Visualization of orthant-based clustering and hierarchical semantic structure in a 3D subspace of context-analyzer vectors. Selected dimensions $\mathcal{K} = \{k_1, k_2, k_3\} = 1, 2, 4$ correspond to columns of $\boldsymbol{U}$ in **Fig. 1 (A)(a)**. Colored blocks represent different sign configurations: Green ($p = 1$): $c_{k_1} = -1$ captures "animal" semantic. Blue ($p = 2$): $[c_{k_1}, c_{k_2}] = [-1, -1]$ refines to "mammal" semantic. Yellow ($p = 3$): $[c_{k_1}, c_{k_2}, c_{k_3}] = [-1, -1, -1]$ further specifies "feline" semantic. This nested structure illustrates how increasing the number of active dimensions (p) yields progressively finer-grained semantic categories, where "mammal" is a subset of "animal", and "feline" is more specific within "mammal". Contexts shown with same color within each suborthant are example members according to Defn. 1.

2 Related Work

We bring together, in the context of NTP-trained language models, the following two classical ideas: (1) semantic patterns emerge from linear relationships between vector representations of textual inputs, and (2) these learned representations form geometries tied to language statistics via count matrices.

Semantics from Distributed Representations. The idea of extracting semantics from distributed representations, while rooted in earlier works Bengio et al. (2000), was popularized through word embedding research Mikolov et al. (2013a). Two classically popular approaches include: (i) constructing semantic axes from centroids of pole words An et al. (2018); Fast et al. (2016), evolving from lexicon-based sentiment analysis Taboada et al. (2011); Hu and Liu (2004), and (ii) investigating semantic patterns through "word analogies," which reveal that semantic relationships manifest as linear directions in embedding space Mikolov et al. (2013c); Levy et al. (2015); Drozd et al. (2016); Ethayarajh et al. (2019); Allen and Hospedales (2019). In the context of modern LLMs, word analogies have been empirically observed Rezaee and Camacho-Collados (2022); Wijesiriwardene et al. (2024). However, recent focus has shifted to techniques for distilling interpretable semantics through dictionary learning or sparse autoencoders Cunningham et al. (2023); Bricken et al. (2023), further showing these can be used for model steering Turner et al. (2023); Li et al. (2024) (techniques that in fact trace back to early work with static word embeddings, e.g., Murphy et al. (2012); Faruqui et al. (2015); Arora et al. (2018); Zhang et al. (2019)). Alternative approaches include geometric studies showing how linguistic features cluster in embedding spaces Coenen et al. (2019); Hewitt and Liang (2019). Simultaneously, supervised methods such as linear probing have been employed to extract semantic and syntactic information from modern LLMs Alain and Bengio (2017); Hewitt and Manning (2019); Marks and Tegmark (2023), relying on the same principle of linear representation found in word analogies. Unlike these aforementioned works that empirically analyze trained models post-hoc, we focus on formalizing the mechanisms by which semantic structure emerges during NTP training. *Our goal is not to introduce a state-of-the-art method for extracting semantics, but rather to understand, from an optimization viewpoint, how semantics emerge as a consequence of next-token prediction.* While Park et al. (2023; 2024) have explored similar questions, here, rather than relying on a model for concept generation and its link to words/contexts, we directly analyze NTP.

Geometry from Co-occurrence Statistics. Vector space representations emerge from two approaches: (i) traditional count-based models using co-occurrence matrices, and (ii) prediction-based models that optimize token prediction objectives like Skip-gram or NTP. Levy and Goldberg (2014); Pennington et al. (2014) demonstrate that these approaches

are fundamentally similar, as both probe underlying corpus co-occurrence statistics. Zhao et al. (2024) have extended this insight to causal NTP models with a key distinction: the co-occurrence matrix between contexts and words is inherently sparse, unlike the dense word-word co-occurrences in prior work. This distinction makes relevant recent advances in implicit regularization Ji and Telgarsky (2019); Ji et al. (2020) and neural collapse geometries Papyan et al. (2020b); Mixon et al. (2020); Fang et al. (2021), through which Zhao et al. (2024) analyze how word and context embeddings converge via an unconstrained features model, relating them to singular factors of the centered co-occurrence probability matrix's support set (see Eq. (2)). We show that these singular factors encode rich semantic structure. While we argue that the natural language's inherent multilabel nature is responsible for the emergence of rich semantics, making full circle to the neural-collapse literature, we show that the semantic interpretation applies to as simple a setting as one-hot imbalanced classification. Moreover, by connecting to the closed-form analysis of learning dynamics of linear neural networks by Saxe et al. (2013; 2019); Gidel et al. (2019), we show how singular values determine the learning order during training. More details on related works deferred to App. E. See also Remark 1.

3 Background

Notations. For any integer k, $[k] := \{1, \dots, k\}$. Matrices, vectors, and scalars are denoted by $\boldsymbol{A}$, $\boldsymbol{a}$, and a respectively. For matrix $\boldsymbol{A}$, $\boldsymbol{A}[i, j]$ is its (i, j)-th entry, and for vector $\boldsymbol{a}$, $\boldsymbol{a}[i]$ is its i-th entry. $\mathbb{1}$ is the all-ones vector and $\otimes$ is Kronecker product.

3.1 NTP Objective as Sparse Soft-Label Classification

Let vocabulary $\mathcal{V} = [V]$, where $z_t \in \mathcal{V}$ are tokens/words within sequences $\boldsymbol{z}_{1:t} = (z_1, \dots, z_t)$. NTP predicts a target token $z := z_t$ from context $\boldsymbol{x} := \boldsymbol{z}_{1:t-1}$ using training data $\mathcal{T}_n := \{(\boldsymbol{x}_i, z_i)\}_{i \in [n]}$, where $\boldsymbol{x}_i \in \mathcal{V}^{t-1}$ and $z_i \in \mathcal{V}$ for each $i \in [n]$, and the context length $t-1$ ranges from 0 to $T-1$. For the prediction, a model with logits $\boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\boldsymbol{x})$, where $\boldsymbol{W} \in \mathbb{R}^{V \times d}$ is the decoding matrix and $\boldsymbol{\theta}$ parameterizes (using either MLP, LSTM, Transformer (TF), or state-space models) the context-embedding map $\boldsymbol{h}_{\boldsymbol{\theta}} : \mathcal{X} \to \mathbb{R}^d$, is trained to minimize the empirical cross-entropy (CE) loss. Following Zhao et al. (2024), we cast the NTP objective as classifying among $m \leq n$ *unique* contexts $\bar{\boldsymbol{x}}_j, j \in [m]$, each associated with a *sparse* label vector $\hat{\boldsymbol{p}}_j \in \Delta^{V-1}, j \in [m]$ in the probability simplex, representing the conditional distribution of next tokens. The sparsity of the conditional distributions is both a sampling artifact and inherent at the population level since not all tokens from the vocabulary are valid next-tokens of a given context in natural language data. We further let $\hat{\pi}_j$ denote the empirical probability for context $\bar{\boldsymbol{x}}_j$. With these, the NTP objective becomes

$$\mathcal{L}(\boldsymbol{W}, \boldsymbol{\theta}) = \textstyle\sum_{j \in [m]} \hat{\pi}_j \cdot \ell\left(\boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\bar{\boldsymbol{x}}_j); \hat{\boldsymbol{p}}_j\right) , \tag{1}$$

where the loss ℓ penalizes deviations between the model's logits $\boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\bar{\boldsymbol{x}}_j)$ for context j and its corresponding soft-label vector $\hat{\boldsymbol{p}}_j$. Unless otherwise specified, we use the standard cross-entropy loss we set the loss ℓ to be the standard CE loss, i.e., $\mathrm{CE}\left(\mathbb{S}(\boldsymbol{W}\boldsymbol{h}_{\boldsymbol{\theta}}(\bar{\boldsymbol{x}}_j)) \,||\, \boldsymbol{p}_j\right)$, where $\mathbb{S}()$ denotes the softmax map of logits to the probability simplex.

For later use, define the **support matrix** $\boldsymbol{S} \in \{0, 1\}^{V \times m}$ of the conditional probability matrix $\boldsymbol{P} = [\hat{\boldsymbol{p}}_1, \dots, \hat{\boldsymbol{p}}_m] \in \mathbb{R}^{V \times m}$. Formally, $\boldsymbol{S}[z, j] = 1 \Leftrightarrow \boldsymbol{P}[z, j] > 0$. For context j, we refer to tokens z for which $\boldsymbol{S}[z, j] = 1$ as in-support tokens, and refer to all others as off-support. Central to our analysis is the centered support matrix

$$\widetilde{\boldsymbol{S}} := (\mathbb{I}_V - \frac{1}{V}\mathbb{1}_V\mathbb{1}_V^\top)\boldsymbol{S} . \tag{2}$$

For convenience, we refer to $\widetilde{\boldsymbol{S}}$ as the **data-sparsity matrix**, though note that unlike the support matrix $\boldsymbol{S}$, its entries are not binary due to the centering operation:

$$\widetilde{\boldsymbol{S}}[z, j] = \begin{cases} 1 - |\mathcal{S}_j|/V & \text{if } \boldsymbol{S}[z, j] = 1 , \\ -|\mathcal{S}_j|/V & \text{if } \boldsymbol{S}[z, j] = 0 . \end{cases}$$

3.2 Geometry of Words and Contexts

Following Yang et al. (2017), we assume the model is sufficiently expressive to allow optimizing context embeddings in (1) freely, instead of abiding by their architecture-specific parameterization. Concretely, this modeling abstraction yields a simplified training objective

$$\min_{\boldsymbol{W},\boldsymbol{H}} \quad \mathcal{L}(\boldsymbol{W}\boldsymbol{H}) + \frac{\lambda}{2}\|\boldsymbol{W}\|^2 + \frac{\lambda}{2}\|\boldsymbol{H}\|^2, \tag{NTP-UFM}$$

which jointly optimizes $\boldsymbol{W} \in \mathbb{R}^{V\times d}$ and $\boldsymbol{H} := [\boldsymbol{h}_1, \ldots, \boldsymbol{h}_m] \in \mathbb{R}^{d\times m}$, where $\boldsymbol{h}_j$ is the freely optimized embedding of context $\bar{\boldsymbol{x}}_j$. Since the minimization is unconstrained for both variables, we follow Mixon et al. (2022); Wojtowytsch (2021); Fang et al. (2021) in referring to this as the unconstrained features model (UFM) for NTP training. We refer to $\boldsymbol{W}$ and $\boldsymbol{H}$ as the **word** and **context embedding matrices**.

Eq. (NTP-UFM) includes ridge-regularization with weight $\lambda > 0$. Zhao et al. (2024) have analyzed the geometry of solutions to (NTP-UFM) when $\lambda \to 0$. This limit (referred to in the literature as the regularization-path) serves as a proxy for the limiting behavior of gradient descent (GD) training as the number of iterations approaches infinity Ji et al. (2020). For large embedding dimensions $d \geq V$, [2] as $\lambda \to 0$, Zhao et al. (2024) show the following:

1. Logits Convergence: The logit matrix $\boldsymbol{L} = \boldsymbol{W}\boldsymbol{H}$ decomposes into two orthogonal components: one that grows unboundedly during training and another that remains finite. After normalization (required to characterize the convergence of diverging logits), only the unbounded component becomes dominant, denoted as $\boldsymbol{L}^{\text{mm}}$. Crucially, $\boldsymbol{L}^{\text{mm}}$ depends only on the data support matrix $\boldsymbol{S}$ (see Zhao et al. (2024) for details).

2. SVD factors of $\boldsymbol{L}^{\text{mm}}$: Like logits, word and context embeddings grow unboundedly in magnitude, but converge in direction to $\boldsymbol{W}^{\text{mm}} = \boldsymbol{U}\boldsymbol{\Sigma}^{1/2}\mathbf{R}$ and $\boldsymbol{H}^{\text{mm}} = \mathbf{R}^\top\boldsymbol{\Sigma}^{1/2}\boldsymbol{V}^\top$, respectively. Here, $\boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ is the SVD of $\boldsymbol{L}^{\text{mm}}$ and $\mathbf{R}$ is a partial orthogonal matrix.

3. Data-sparsity matrix as proxy: The matrix $\widetilde{\boldsymbol{S}}$ (see Eq. (2)) is a good proxy for $\boldsymbol{L}^{\text{mm}}$. [3] Put together, the singular factors of $\widetilde{\boldsymbol{S}}$ play a crucial role in determining the geometry of word and context embeddings learnt by NTP training. Denote this SVD as

$$\widetilde{\boldsymbol{S}} := \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top, \tag{3}$$

where $\boldsymbol{U} \in \mathbb{R}^{V\times r}$, $\boldsymbol{V} \in \mathbb{R}^{m\times r}$ with $\boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{V}^\top\boldsymbol{V} = \mathbb{I}_r$, the singular values $\boldsymbol{\Sigma} = \text{diag}(\sigma_1, \ldots, \sigma_r)$ are ordered: $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0$, and $r \leq V - 1$ denotes the rank.

4 Geometry of Concepts

4.1 Motivating Questions

While prior work explicitly describes the geometry of word and context embeddings, it leaves open two key questions: (i) *What is the latent information that drives the word/context embeddings to this geometry?* And, (ii) *does the resulting geometry form structures (e.g. in the form of clusters) that lend themselves to human-interpretable semantics?*

To illustrate this, Fig. 2A shows a 3D projection of trained word and context embeddings on a simplified dataset for visualization. The conditional probability matrix $\boldsymbol{P}$ is shown on the right. The geometry reveals clear structure: animal-related words and contexts cluster on the left, plant-related ones on the right. This suggests the emergence of latent concept information—specifically, a positive/negative semantic distinction.

From Sec. 3.2, we know word embeddings take the form $\boldsymbol{W} = \boldsymbol{U}\sqrt{\boldsymbol{\Sigma}}\mathbf{R}$, where $\boldsymbol{U}$ encodes the row-space of $\widetilde{\boldsymbol{S}}$ (Eq. (2)). Since rows of $\widetilde{\boldsymbol{S}}$ correspond to words, and words that appear in similar contexts (e.g., "the organism that grows roots is..." vs. "the organism that has green leaves is...") have similar patterns, $\boldsymbol{U}$ may naturally encode semantic relationships. Its columns can be interpreted as a "concept basis", where a word's projection reflects its association with these latent concepts.

[2] See Sec. A.2 for a discussion on this assumption.

[3] This is formally proven in symmetric cases and supported by extensive empirical evidence.

4.2 Principal Components as Concepts

We term each dimension $k \in [r]$ resulting from the SVD of $\widetilde{S}$ as a *concept*. Adopting terminology from Saxe et al. (2019), we call the columns $\mathbf{u}_k \in \mathbb{R}^V$ and $\mathbf{v}_k \in \mathbb{R}^m$ of matrices $\mathbf{U}$ and $\mathbf{V}$ the "word analyzer vectors" and "context analyzer vectors," respectively. These vectors represent the alignment of tokens and contexts with each concept: for word z and concept k, the sign of $\mathbf{u}_k[z]$ indicates *association* ($\mathbf{u}_k[z] > 0$) or *opposition* ($\mathbf{u}_k[z] < 0$) to the concept, while its magnitude quantifies the strength of this relationship. [4] Words for which $|\boldsymbol{u}_k[z]|$ is small are *neutral* to this concept. The same interpretation applies to context components $\mathbf{v}_k[j]$, measuring each context's alignment with the respective concept. Fig. 1(A) verifies the SVD of $\widetilde{S}$ captures the dataset's semantic information for the simplified[5] dataset with syntax "The organism that [attribute] is [subject]" (see Fig. 2).

To represent concepts in the embedding space and thus directly characterize their relationships with word and context embeddings, we define the d-dimensional "word-concept representations" $\boldsymbol{u}_k^{\mathrm{d}}$ and "context-concept representations" $\boldsymbol{v}_k^{\mathrm{d}}$ for $k \in [r]$ as projections onto the respective spaces of words and contexts, that is (see also Sec. B for details), $\boldsymbol{u}_k^{\mathrm{d}} = \boldsymbol{W}^\top \boldsymbol{u}_k$ and $\boldsymbol{v}_k^{\mathrm{d}} = \boldsymbol{H}\boldsymbol{v}_k$. This ensures that words or contexts more closely aligned with a specific concept have embeddings closer to the concept's representation. Similar to classical semantic axes formed by averaging relevant words, these representations are weighted averages where weights ($\boldsymbol{u}_k$) reflect concept relevance. Fig. 8 visualizes two most significant concept representations and their relationships with word/context embeddings. For background and related developments on matrix-based semantic representations and their connection to our framework, see Remark 1.

4.3 Orthant-based Clustering in Singular Spaces

Concepts extracted from $\widetilde{S}$, represent significant components from data. However, as noted, for example, by Chersoni et al. (2021); Piantadosi et al. (2024), not all concepts individually correspond to linguistically interpretable factors. Instead, we posit that human-interpretable semantics can be identified by specific combinations of several concepts. Here, we formalize this hypothesis by relating it to an orthant-based clustering of word/context representations.

Recall from Sec. 4.2 that the k-th word analyzer vector $\boldsymbol{u}_k$ categorizes words z based on $\mathtt{sign}(\boldsymbol{u}_k[z])$: words with large $|u_k[z]|$ exhibit strong correlation with concept k, with $\mathtt{sign}(\boldsymbol{u}_k[z])$ indicating whether this correlation is positive/negative. We extend this categorization beyond single dimensions to multiple concept dimensions simultaneously.

Let $\mathcal{K} = \{k_1, \ldots, k_p\}$ denote a set of $p \leq r$ selected dimensions $k_i \in [r], i \in [p]$. Given $\mathcal{K}$, we define 2^p possible signature configurations $C = C(\mathcal{K}) = [c_{k_1}, \ldots, c_{k_p}] \in \{\pm 1\}^p$, where $c_{k_i} \in \{\pm 1\}$ indicates whether dimension k_i contributes positively or negatively to the semantic category. Each signature configuration represents a potential semantic category, characterized by its member words and their degrees of association with the category.

Definition 1. *For $C = [c_{k_1}, \ldots, c_{k_p}]$ with sign configurations $c_{k_i} \in \{\pm 1\}$ for dimensions $k_i \in [r]$:*

- *A word z is a member of C (denoted $z \in C$) if and only if $\mathtt{sign}(\boldsymbol{u}_{k_i}[z]) = c_{k_i}$ for all $k_i, i \in [p]$.*
- *The typicality of a member $z \in C$ is defined as: Typicality$(z; C) = \sum_{i \in [p]} \left|\boldsymbol{u}_{k_i}[z]\right|$.*

Analogous definitions hold for contexts.

Geometrically, each configuration corresponds to a p-dimensional orthant (or orthant sheet) in the d-dimensional embedding space. Membership indicates which word/context embeddings lie within this orthant, effectively creating an orthant-based clustering in the subspace spanned by the selected concept dimensions $\mathcal{K}$. The typicality of a member measures the

[4] Note that each word analyzer vector $\boldsymbol{u}_k, k \in [r]$ has both positive and negative entries (corresponding to association and opposition), because $\boldsymbol{u}_k$ is orthogonal to $\mathbb{1}_V$ (which is in the nullspace of $\widetilde{S}$).

[5] This design isolates semantic concept extraction by minimizing grammatical influences. These simplifications are for visualization clarity only—the principles apply to complex datasets with combined grammatical/semantic attributes and to full autoregressive training (see Secs. 5 and A.3).

L1-norm (other choices are also possible) of its representation's projection (the z-th row of $\boldsymbol{U}$ for words, j-th row of $\boldsymbol{V}$ for contexts) onto the selected concept dimensions. Fig. 2B and Sec. 5 demonstrate that this orthant-based clustering reveals interpretable semantic categories, while also inducing a hierarchical structure, where the number of concept dimensions determines the semantic granularity: broad linguistic categories (e.g., verbs) require fewer dimensions, while more specific categories (e.g., past-tense verbs) need more dimensions for precise characterization. See also Sec. C and Fig. 9 in the appendix.

4.4 Rate of Learning

We show that the hierarchical structure of semantics discussed in Sec. 4.3 is reflected in their emergence over the course of training: broad distinctions are learned earlier than fine-grained features. We study this both in synthetic language and in a controlled imbalanced setting, and provide theoretical support via training dynamics under square loss.

Spectral progression in natural language. We track the singular values of the logit matrix $\boldsymbol{L}_t = \boldsymbol{W}_t \boldsymbol{H}_t$ at training step t, where $(\boldsymbol{W}_t, \boldsymbol{H}_t)$ evolve under the unconstrained objective (NTP-UFM). Figure 16 shows that singular values increase at different rates: those with larger final magnitudes begin diverging earlier. This suggests that concepts aligned with dominant singular directions of the data are acquired earlier. This trend is evident in natural language experiments. As shown in Figure 1(C), confusion matrices from a 2-layer transformer trained on structured synthetic language reveal that coarse categories (e.g., "Animals" vs. "Plants") are learned in early epochs (step 20), while finer semantic distinctions (e.g., "Lives in water", "Reproduces with pollen") emerge later (step 40 onward). This coarse-to-fine learning order aligns with the ordering of singular values in the support matrix.

Controlled validation via class imbalance. To isolate this effect and remove semantic confounds, we construct a step-imbalanced one-hot classification task where two majority classes appear in many contexts, while two minority classes occur only once. This setting allows us to directly manipulate the spectrum of the support matrix $\widetilde{S}$. As shown in Figure 7, the model first predicts all tokens as the dominant majority class (Step 0), then separates the two majority classes (Step 13), begins identifying the minority classes (Step 30), and only later resolves confusion between them (Step 55). This progression follows the order of singular values, despite the absence of semantic structure, confirming that the model prioritizes directions with higher spectral mass. The imbalanced setting thus provides strong causal evidence for spectral bias in learning dynamics.

Theoretical justification. To analyze this behavior formally, we consider the square loss version of (NTP-UFM). As discussed in Appendix D, this setting reduces to the well-studied linear training dynamics of Saxe et al. (2013), where each singular direction k exhibits a sharp transition in $\boldsymbol{L}_t$ at time $T_k \propto 1/\sigma_k$, with σ_k the k-th singular value of $\widetilde{S}$. Hence, directions aligned with larger singular values (i.e., broader or more frequent concepts) are learned earlier. Although this analysis is limited to square loss, we observe similar learning patterns under cross-entropy empirically. Extending this theory to CE loss remains an open question.

5 Experiments

5.1 Setup

Datasets. We conduct experiments on both synthetic data and pretrained models. For our synthetic experiments, we constructed two datasets by subsampling from two distinct corpora: TinyStories Eldan and Li (2023) and WikiText-2 Merity et al. (2017). For both datasets, we fix the vocabulary size to $V = 1,000$ and extract $m = 10,000$ contexts, creating the "Simplified TinyStories/WikiText" datasets. We extracted the contexts by sampling frequent contexts of lengths 2-6 from the entire datasets. This variation in context-length emulates the autoregressive nature of training with sliding window, which yields rich data-sparsity patterns that in turn result in richer semantic information (compared to the last-token visuals in Sec. 4). See Sec. F.2 for details on OpenWebText data with GPT2 embeddings.

Concepts and semantics. Following Sec. 4, for the simplified datasets where it is computationally feasible, we construct the $V \times m$ data-sparsity matrix $\widetilde{\boldsymbol{S}}$ and compute its SVD.

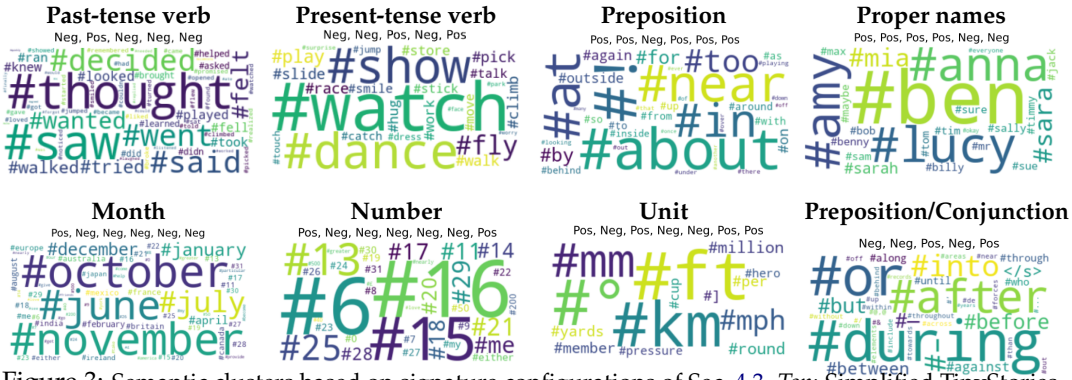

Figure 3: Semantic clusters based on signature configurations of Sec. 4.3. *Top:* Simplified TinyStories dataset showing Past-tense verbs, Present-tense verbs, Prepositions, and Proper names. *Bottom:* Simplified Wikitext dataset showing Months, Numbers, Units, and Prepositions/Conjunctions. Each configuration is shown in its title.

The left/right singular vectors serve as our word/context concepts. As per Sec. 4.3, we also consider signature-based concept combinations, which we examine for semantic interpretability as discussed in *Visuals* below. For completeness, we also experiment with k-means spectral clustering as an alternative way to derive interpretable semantics from top important concepts. For larger-scale experiments, where direct SVD of $\widetilde{S}$ is infeasible, we work with the embeddings from pretrained models. In particular, for GPT-2 trained on OpenWebText, we analyze the spectral structure of its learned token embeddings (Sec. F.2). To test generality across architectures and objectives, we additionally evaluate orthant-based clustering on BERT (a masked language model) and Qwen-7B (a multilingual model with 4096-dimensional embeddings). In these cases, we approximate concept axes via SVD on filtered subsets of embeddings. See Secs. F.4 and F.5 for experimental details.

Visuals. To interpret the semantic information, we visualize the top 40 words/contexts associated with each concept. "Top" here is measured with respect to the typicality score in Defn. 1. For visualization, we use word clouds where size is proportional to its typicality score. We often use the pound sign (#) to indicate token start.

5.2 Findings

Individual concepts are not interpretable. We have seen in Sec. 4 that not all concepts individually correspond to linguistically interpretable factors. For completeness, we verify this conclusion for the Simplified TinyStories/WikiText datasets in Fig. 12 in the appendix. This observation is well-reported in the literature, e.g., Chersoni et al. (2021), and is also related to the recently popularized superposition hypothesis Elhage et al. (2022).

Combining concepts with signature-configurations reveals semantics. We demonstrate that combining non-interpretable concepts can reveal human-interpretable semantic information. Here, we show this by implementing the idea of orthant-based clustering of Sec. 4.3. Specifically, we combine $p = 5$ to 7 concepts in successive order starting from the most important one, i.e. $\mathcal{K} = \{1, \ldots, p\}$. Fig. 3 shows human-interpretable semantics identified by orthant-based clustering in Simplified TinyStories and WikiText-2 datasets. In Fig. 9, we show how orthant-based clustering progressively reveals hierarchical semantic structure as we combine more concept dimensions, illustrated through the example of verb forms organized by grammatical distinctions. Starting with a single dimension ($\mathcal{K} = \{1\}$), the categories remain broad and difficult to interpret. However, as we incorporate additional dimensions, increasingly refined semantic categories emerge, ultimately distinguishing between different verb forms such as past/present tense, and present continuous forms.

As a complementary method to orthant-based clustering, we also explore k-means spectral clustering over concept subspaces. This also reveals interpretable groupings such as past-tense verbs, proper names, and numerical tokens. For methodology and full results, see Appendix F.7.

Relation of words and context semantics. Figure 2A shows that words and contexts sharing semantic categories (e.g., plant vs. animal) cluster closely in embedding space—a pattern that extends to richer datasets. To verify this, we apply orthant-based clustering to words and contexts using the same signature pattern across a subset of concept dimensions. In

Simplified TinyStories (Fig. 4), contexts with modals like "want/like/love/have to" align with clusters of infinitive verbs; intensifiers like "was so/very" correspond to descriptive adjectives. This duality reflects how contexts encode meaning both via their surface words and via the semantics of likely next tokens—a view naturally supported by the predictive nature of NTP.

Figure 4: Context-word cluster pairs sharing identical sign patterns across top 5 concepts. Left and right pair illustrates the "verb+to infinitive" and "was so/very + adj" structures, confirming semantic alignment between contexts and their in-support words.

Semantics in pretrained models. Simplified datasets primarily reveal grammatical and syntactic structure due to limited size and context diversity—making them analytically tractable but semantically constrained. To test whether richer sparsity induces richer semantics, we analyze pretrained models.

In **GPT-2** (Fig. 5), orthant-based clustering recovers categories such as medical terms, emotional tone, entertainment, and political figures. In **BERT** (Fig. 17), we observe clusters reflecting numerical patterns (e.g., 3-digit numbers), action verbs, administrative phrases, and morphological forms. Applying the same method to **Qwen-7B**, a 4096-dimensional multilingual model, yields clusters like UI phrases, programming keywords, numeric tokens, and functional Chinese expressions. Qwen requires more concept directions (7–11 vs. 5–7 for GPT-2) to isolate coherent groups (Table 1), suggesting a more distributed semantic encoding. These results suggest that spectral semantic structure generalizes across objectives and embedding scales.

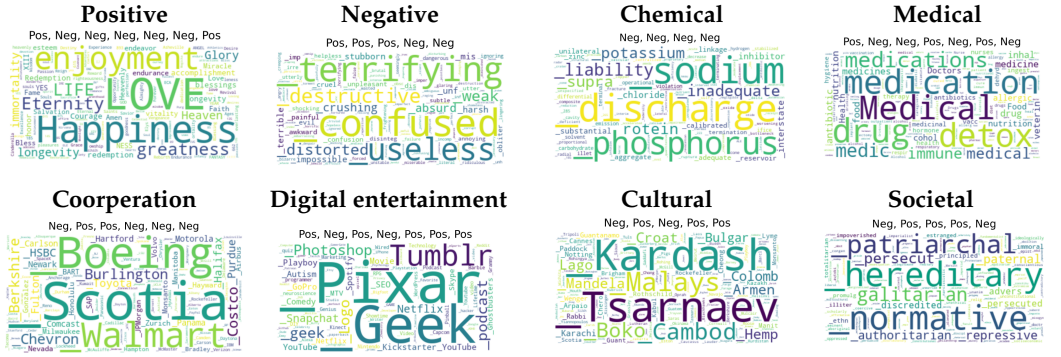

Figure 5: Semantics identified by orthant-based clustering on GPT2's word embeddings.

6 Conclusion and Limitations

While continuing to advance LLMs' capabilities, understanding their inner workings is crucial. We study how NTP optimization guides the geometry of word and context representations to organize around latent semantic information. Without explicitly computing it, the model encodes such information in the SVD factors of an implicit data-sparsity matrix. As an early contribution in this direction, our analysis motivates several future directions: (1) While the assumption $d \geq V$ allows for rich geometric structures, a formal investigation is needed for the $d < V$ case prevalent in practice (see Sec. A.2). (2) Our analysis assumes sufficient expressivity and training time, which may become increasingly relevant with improved compute or in data-limited regimes, but relaxing them or experimentally validating their applicability in the spirit of our GPT-2 experiments (e.g., in the spirit of Wu and Papyan (2024)) is important. (3) Understanding how transformer architectures specifically fit into our analytical framework remains an open challenge. (4) Some conclusions rely on square loss and our work highlights a significant gap in understanding CE loss dynamics, even in simple bilinear models. (5) While our approach differs from concurrent efforts to explain semantic emergence in LLMs via probing or sparse-dictionary learning, future work could explore connections and potential applications to model improvement techniques.

References

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations*, 2017.

Carl Allen and Timothy Hospedales. Analogies explained: Towards understanding word embeddings. *Proceedings of the 36th International Conference on Machine Learning*, pages 223–231, 2019. URL https://proceedings.mlr.press/v97/allen19a.html.

Jisun An, Haewoon Kwak, and Yong-Yeol Ahn. Semaxis: A lightweight framework to characterize domain-specific word semantics beyond sentiment. *arXiv preprint arXiv:1806.05521*, 2018.

Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. Linear algebraic structure of word senses, with applications to polysemy. *Transactions of the Association for Computational Linguistics*, 6:483–495, 2018.

Marco Baroni and Alessandro Lenci. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721, 2010.

Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. *Advances in neural information processing systems*, 13, 2000.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, Shauna Kravec, Nicholas Schiefer, Tim Maxwell, Nicholas Joseph, Zac Hatfield-Dodds, Alex Tamkin, Karina Nguyen, Brayden McLean, Josiah E Burke, Tristan Hume, Shan Carter, Tom Henighan, and Christopher Olah. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023. https://transformer-circuits.pub/2023/monosemantic-features/index.html.

Emmanuele Chersoni, Enrico Santus, Chu-Ren Huang, and Alessandro Lenci. Decoding word embeddings with brain-based semantic features. *Computational Linguistics*, 47(3): 663–698, November 2021. doi: 10.1162/coli_a_00412. URL https://aclanthology.org/2021.cl-3.20/.

Andy Coenen, Emily Reif, Ann Yuan, Been Kim, Adam Pearce, Fernanda B Viégas, and Martin Wattenberg. Visualizing and measuring the geometry of bert. In *Advances in Neural Information Processing Systems*, 2019.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*, 2023.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

Ahmet Demirkaya, Jiasi Chen, and Samet Oymak. Exploring the role of loss functions in multiclass classification. In *2020 54th Annual Conference on Information Sciences and Systems (CISS)*, pages 1–5, 2020. doi: 10.1109/CISS48834.2020.1570627167.

Chris Ding, Tao Li, and Wei Peng. Nonnegative matrix factorization and probabilistic latent semantic indexing: Equivalence chi-square statistic, and a hybrid method. In *AAAI*, volume 42, pages 137–43, 2006.

Aleksandr Drozd, Anna Gladkova, and Satoshi Matsuoka. Word embeddings, analogies, and machine learning: Beyond king - man + woman = queen. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3519–3530, Osaka, Japan, 2016. The COLING 2016 Organizing Committee. URL https://aclanthology.org/C16-1332/.

Esin Durmus, Alex Tamkin, Jack Clark, Jerry Wei, Jonathan Marcus, Joshua Batson, Kunal Handa, Liane Lovitt, Meg Tong, Miles McCain, Oliver Rausch, Saffron Huang, Sam Bowman, Stuart Ritchie, Tom Henighan, and Deep Ganguli. Evaluating feature steering: A case study in mitigating social biases, 2024. URL https://anthropic.com/research/evaluating-feature-steering.

Ronen Eldan and Yuanzhi Li. Tinystories: How small can language models be and still speak coherent english? *arXiv preprint arXiv:2305.07759*, 2023.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.

Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. Towards understanding linear word analogies. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3253–3262, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1315. URL https://aclanthology.org/P19-1315/.

Cong Fang, Hangfeng He, Qi Long, and Weijie J Su. Exploring deep neural networks via layer-peeled model: Minority collapse in imbalanced training. *Proceedings of the National Academy of Sciences*, 118(43), 2021.

Manaal Faruqui, Yulia Tsvetkov, Dani Yogatama, Chris Dyer, and Noah Smith. Sparse overcomplete word vector representations. *arXiv preprint arXiv:1506.02004*, 2015.

Ethan Fast, Binbin Chen, and Michael S. Bernstein. Empath: Understanding topic signals in large-scale text. In *Proceedings of the 2016 CHI conference on human factors in computing systems*, pages 4647–4657. ACM, 2016.

Connall Garrod and Jonathan P Keating. The persistence of neural collapse despite low-rank bias: An analytic perspective through unconstrained features. *arXiv preprint arXiv:2410.23169*, 2024.

Gauthier Gidel, Francis Bach, and Simon Lacoste-Julien. Implicit regularization of discrete gradient dynamics in linear neural networks. *Advances in Neural Information Processing Systems*, 32, 2019.

Florian Graf, Christoph Hofer, Marc Niethammer, and Roland Kwitt. Dissecting supervised constrastive learning. In *International Conference on Machine Learning*, pages 3821–3830. PMLR, 2021.

XY Han, Vardan Papyan, and David L Donoho. Neural collapse under mse loss: Proximity to and dynamics on the central path. *arXiv preprint arXiv:2106.02073*, 2021.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. *Transactions of the Association for Computational Linguistics*, 7:453–470, 2019.

John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 4124–4135, 2019.

Wanli Hong and Shuyang Ling. Neural collapse for unconstrained feature model under cross-entropy loss with imbalanced data. *arXiv preprint arXiv:2309.09725*, 2023.

Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 168–177. ACM, 2004.

Like Hui and Mikhail Belkin. Evaluation of neural architectures trained with square loss vs cross-entropy in classification tasks. *arXiv preprint arXiv:2006.07322*, 2020.

Ziwei Ji and Matus Telgarsky. The implicit bias of gradient descent on nonseparable data. In *Conference on Learning Theory*, pages 1772–1798. PMLR, 2019.

Ziwei Ji, Miroslav Dudík, Robert E Schapire, and Matus Telgarsky. Gradient descent follows the regularization path for general losses. In *Conference on Learning Theory*, pages 2109–2136. PMLR, 2020.

Jiachen Jiang, Jinxin Zhou, Peng Wang, Qing Qu, Dustin Mixon, Chong You, and Zhihui Zhu. Generalized neural collapse for a large number of classes. *arXiv preprint arXiv:2310.05351*, 2023.

Ganesh Ramachandra Kini, Vala Vakilian, Tina Behnia, Jaidev Gill, and Christos Thrampoulidis. Symmetric neural-collapse representations with supervised contrastive loss: The impact of relu and batching. In *The Twelfth International Conference on Learning Representations*.

Kai Konen, Sophie Jentzsch, Diaoulé Diallo, Peer Schütt, Oliver Bensch, Roxanne El Baff, Dominik Opitz, and Tobias Hecking. Style vectors for steering generative large language model. *arXiv preprint arXiv:2402.01618*, 2024.

Thomas K Landauer. The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2):211–240, 1997.

Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *nature*, 401(6755):788–791, 1999.

Omer Levy and Yoav Goldberg. Neural word embedding as implicit matrix factorization. *Advances in neural information processing systems*, 27, 2014.

Omer Levy, Yoav Goldberg, and Ido Dagan. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the association for computational linguistics*, 3:211–225, 2015.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36, 2024.

Pengyu Li, Xiao Li, Yutong Wang, and Qing Qu. Neural collapse in multi-label learning with pick-all-label loss. *arXiv preprint arXiv:2310.15903*, 2023.

Haixia Liu. The exploration of neural collapse under imbalanced data. *arXiv preprint arXiv:2411.17278*, 2024.

Weiyang Liu, Longhui Yu, Adrian Weller, and Bernhard Schölkopf. Generalizing and decoupling neural collapse via hyperspherical uniformity gap. *arXiv preprint arXiv:2303.06484*, 2023.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. *arXiv preprint arXiv:2310.06824*, 2023.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In *Proceedings of the 5th International Conference on Learning Representations (ICLR)*, 2017. URL https://arxiv.org/abs/1609.07843.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeff Dean. Efficient estimation of word representations in vector space. In *Workshop at ICLR*, 2013a.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013b.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013c.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013d.

Dustin G Mixon, Hans Parshall, and Jianzong Pi. Neural collapse with unconstrained features. *arXiv preprint arXiv:2011.11619*, 2020.

Dustin G Mixon, Hans Parshall, and Jianzong Pi. Neural collapse with unconstrained features. *Sampling Theory, Signal Processing, and Data Analysis*, 20(2):11, 2022.

Brian Murphy, Partha Pratim Talukdar, and Tom Mitchell. Learning effective and interpretable semantic models using non-negative sparse embedding. In *International Conference on Computational Linguistics (COLING 2012), Mumbai, India*, pages 1933–1949. Association for Computational Linguistics, 2012.

Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020a.

Vardan Papyan, XY Han, and David L Donoho. Prevalence of neural collapse during the terminal phase of deep learning training. *Proceedings of the National Academy of Sciences*, 117(40):24652–24663, 2020b.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023.

Kiho Park, Yo Joong Choe, Yibo Jiang, and Victor Veitch. The geometry of categorical and hierarchical concepts in large language models. *arXiv preprint arXiv:2406.01506*, 2024.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

Steven T Piantadosi, Dyana CY Muller, Joshua S Rule, Karthikeya Kaushik, Mark Gorenstein, Elena R Leib, and Emily Sanford. Why concepts are (probably) vectors. *Trends in Cognitive Sciences*, 2024.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Kiamehr Rezaee and Jose Camacho-Collados. Probing relational knowledge in language models via word analogies. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3930–3936, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-emnlp.289. URL https://aclanthology.org/2022.findings-emnlp.289/.

Andrew M Saxe, James L McClelland, and Surya Ganguli. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*, 2013.

Andrew M Saxe, James L McClelland, and Surya Ganguli. A mathematical theory of semantic development in deep neural networks. *Proceedings of the National Academy of Sciences*, 116(23):11537–11546, 2019.

Peter Súkeník, Marco Mondelli, and Christoph H Lampert. Deep neural collapse is provably optimal for the deep unconstrained features model. *Advances in Neural Information Processing Systems*, 36:52991–53024, 2023.

Peter Súkeník, Marco Mondelli, and Christoph Lampert. Neural collapse versus low-rank bias: Is deep neural collapse really optimal? *arXiv preprint arXiv:2405.14468*, 2024.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. Lexicon-based methods for sentiment analysis. *Computational Linguistics*, 37(2):267–307, 2011.

Christos Thrampoulidis, Ganesh R Kini, Vala Vakilian, and Tina Behnia. Imbalance trouble: Revisiting neural-collapse geometry. *arXiv preprint arXiv:2208.05512*, 2022.

Tom Tirer and Joan Bruna. Extended unconstrained features model for exploring deep neural collapse. *arXiv preprint arXiv:2202.08087*, 2022.

Alexander Matt Turner, Lisa Thiergart, Gavin Leech, David Udell, Juan J Vazquez, Ulisse Mini, and Monte MacDiarmid. Activation addition: Steering language models without optimization. *arXiv e-prints*, pages arXiv–2308, 2023.

Peter D Turney and Patrick Pantel. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37:141–188, 2010.

Thilini Wijesiriwardene, Ruwan Wickramarachchi, Aishwarya Naresh Reganti, Vinija Jain, Aman Chadha, Amit Sheth, and Amitava Das. On the relationship between sentence analogy identification and sentence structure encoding in large language models. In Yvette Graham and Matthew Purver, editors, *Findings of the Association for Computational Linguistics: EACL 2024*, pages 451–457, St. Julian's, Malta, March 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-eacl.31/.

Stephan Wojtowytsch. On the emergence of simplex symmetry in the final and penultimate layers of neural network classifiers. *Proceedings of Machine Learning Research vol*, 145:1–21, 2021.

Robert Wu and Vardan Papyan. Linguistic collapse: Neural collapse in (large) language models. *arXiv preprint arXiv:2405.17767*, 2024.

Zhilin Yang, Zihang Dai, Ruslan Salakhutdinov, and William W Cohen. Breaking the softmax bottleneck: A high-rank rnn language model. *arXiv preprint arXiv:1711.03953*, 2017.

Zeyu Yun, Yubei Chen, Bruno A Olshausen, and Yann LeCun. Transformer visualization via dictionary learning: contextualized embedding as a linear superposition of transformer factors, 2023.

Juexiao Zhang, Yubei Chen, Brian Cheung, and Bruno A Olshausen. Word embedding visualization via dictionary learning. *arXiv preprint arXiv:1910.03833*, 2019.

Yize Zhao, Tina Behnia, Vala Vakilian, and Christos Thrampoulidis. Implicit geometry of next-token prediction: From language sparsity patterns to model representations. *arXiv preprint arXiv:2408.15417*, 2024.

Jinxin Zhou, Xiao Li, Tianyu Ding, Chong You, Qing Qu, and Zhihui Zhu. On the optimization landscape of neural collapse under mse loss: Global optimality with unconstrained features. *arXiv preprint arXiv:2203.01238*, 2022a.

Jinxin Zhou, Chong You, Xiao Li, Kangning Liu, Sheng Liu, Qing Qu, and Zhihui Zhu. Are all losses created equal: A neural collapse perspective. *arXiv preprint arXiv:2210.02192*, 2022b.

Zhihui Zhu, Tianyu Ding, Jinxin Zhou, Xiao Li, Chong You, Jeremias Sulam, and Qing Qu. A geometric analysis of neural collapse with unconstrained features. *Advances in Neural Information Processing Systems*, 34, 2021.

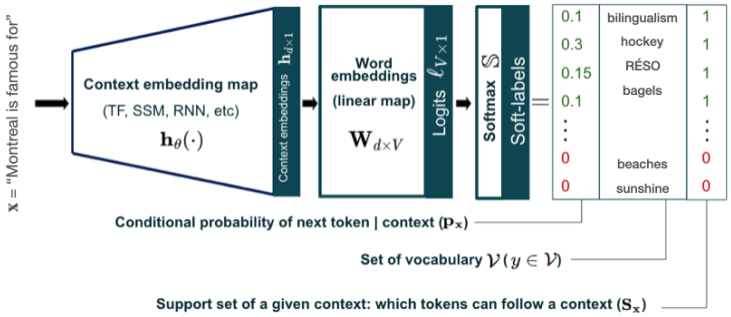

Figure 6: Illustration of setup and terminology.

A Additional Discussions

A.1 Positioning and Scope.

This work adopts a first principles and bottom up approach to understanding how semantic structure emerges during next token prediction training. Rather than analyzing large language models directly, we focus on simplified and analytically tractable settings such as synthetic datasets and the unconstrained feature model. These abstractions allow us to isolate and study fundamental mechanisms with mathematical clarity. While they do not capture the full complexity of real models, they reveal core geometric and learning dynamics that are otherwise difficult to observe. As we later show, insights gained from this analysis extend qualitatively to more realistic settings, including pretrained GPT-2 and BERT embeddings, and can inform future empirical investigations.

A.2 On the Assumption $d \geq V$

Our analysis has assumed the embedding dimension d is at least as large as the rank of the data-sparsity matrix (guaranteed when $d \geq V$), allowing word and context embeddings to form geometric structures representing all concepts encoded in the matrix's singular factors Zhao et al. (2024). As noted in Zhao et al. (2024), while this assumption differs from current practice in LLMs, the geometry of embeddings remains rich in this setting. Importantly, this assumption is less restrictive than requiring $d > C$ in one-hot classification with C classes. There, due to collapse of embeddings from the same class, $d > C$ effectively requires the dimension to exceed the number of training examples. In contrast, for NTP, the number of contexts m can be (and typically is) much larger than d, allowing for rich geometric arrangements of context embeddings in the lower-dimensional space and non-trivial emergence of concepts, which is confirmed throughout our case studies.

That said, the current trend of modern language models typically employing $d < V$, raises the question: Which concepts are prioritized when the model cannot capture all of them? We hypothesize that during NTP training, models learn to represent the d most significant concepts, corresponding to the largest singular values of the data-sparsity matrix. While this requires a separate study that is beyond our current scope, our preliminary investigation using NTP-UFM with square-loss on a synthetic dataset supports this hypothesis. Fig. 18 shows the learned logit matrix's singular values converging to the d largest singular values of the data-sparsity matrix during training. Following Sec. 4.4, we used square loss rather than CE to ensure bounded singular values. While a rigorous analysis for $d < V$ remains future work, since meaningful semantic categories emerge from combinations of concepts, even a reduced set of core concepts could enable rich semantic representations through the orthant clustering mechanism of Sec. 4.3.

A.3 Role of Autoregression

We have shown that concepts emerge during NTP training as principal directions of $\widetilde{\boldsymbol{S}}$, a centered version of the support matrix $\boldsymbol{S}$. Each column j of $\boldsymbol{S}$ corresponds to a context and can be viewed as a binary (multi-)label vector $\boldsymbol{s}_j$, where entries of 1 indicate tokens that appear as next-tokens for that context in the training data. While concepts are determined by the principal directions of this label matrix, this might seem limiting: contexts can relate not only through shared next-tokens but also through their intrinsic structure (e.g., overlapping constituent tokens).

The autoregressive nature of training naturally captures these structural relationships as follows. As the model processes labels for progressively longer contexts, each context $(z_1, \ldots, z_t)$ contributes to concept formation both through its distribution over next-tokens z_{t+1} and through the labels of its shorter subsequences $(z_1, \ldots, z_{t-1})$, $(z_1, \ldots, z_{t-2})$, etc.—each representing distinct columns in $\boldsymbol{S}$. This overlapping window structure produces fine-grained label information, creating rich sparsity patterns in $\boldsymbol{S}$ (and consequently in $\widetilde{\boldsymbol{S}}$) that yield nontrivial concepts with varying significance, as reflected in word and context embeddings.

A.4 Relationship to Spectral Clustering

The orthant-based clustering approach introduced in this work shares foundational similarities with spectral clustering while providing distinct advantages for linguistic interpretation. Both methods operate in reduced-dimensional spaces and leverage embedding geometry to identify meaningful clusters. First, while both approaches leverage singular factors, the orthant-based method provides a more direct link (in view of concept definition in Sec. 4) between NTP and the emergence of semantic structure by explicitly interpreting the sign patterns of SVD dimensions as semantic indicators.

An interesting property of our method is its selective nature: while p dimensions generate 2^p potential orthants, only a subset contains semantically coherent word groupings. This selectivity reflects linguistic reality, where not all mathematical combinations yield meaningful semantic categories. In contrast, spectral clustering typically partitions the space completely, which may not always align with natural semantic boundaries in language. Finally, the orthant-based method offers semantic interpretation of membership and typicality, and dimensionality control through parameter p. This aligns with language model training dynamics, as the most important singular value components are learned first during training, making the resulting clusters reflective of the model's knowledge acquisition process. To this respect, note that unlike orthant-based clustering producing a potential maximum of 2^p clusters when selecting p top concepts, typical clustering algorithms such as vanilla k-means on principal components would result in p-clusters with the same number of selected concepts. Thus, when we evaluate spectral methods in our experiments, we choose k on the order of 2^p for the value of p that we find orthant-based clustering reveals semantically informative orthants.

A.5 Neural Collapse Meets Semantics

The emergence of concepts with rich semantic meanings stems from the interplay of labels (encoded in $\boldsymbol{S}$ or its centered version $\widetilde{\boldsymbol{S}}$) across different contexts. Consider the minimal label richness case: each context has exactly one next-token, with contexts distributed equally across the vocabulary. Here, $\boldsymbol{S}$ can be rearranged as $\mathbb{I}_V \otimes \mathbb{1}_{m/V}^\top$, yielding trivial concepts where all singular directions contribute equally. This setting parallels standard balanced one-hot classification studied in neural collapse literature, where last-layer embeddings and weights form highly symmetric aligned structures Papyan et al. (2020a), reflecting the symmetric nature of $\boldsymbol{S}$ where labels induce no interesting conceptual structure.

However, such balanced settings never occur in autoregressive NTP, which as discussed in Sec. A.3, produces rich label formations. In these richer settings, $\boldsymbol{S}$ induces a geometry of concepts that provides semantic interpretation to the embedding geometry studied in

the neural collapse literature. To illustrate this new perspective, we demonstrate that meaningful concepts emerge even in one-hot classification—the traditional focus of neural collapse literature—given minimal deviation from perfect balance. Specifically, consider STEP imbalances with ratio R: $V/2$ majority classes each have $R > 1$ times more samples than minority classes.

Here, Thrampoulidis et al. (2022) shows the learned logit matrix converges to $\widetilde{S}$, whose singular values exhibit a three-tier structure:

$\sigma_1 = \ldots = \sigma_{V/2-1} = \sqrt{R} > \sigma_{V/2} = \sqrt{(R+1)/2} > \sigma_{V/2+1} = \ldots = \sigma_V = 1$. The left singular vectors matrix $\boldsymbol{U}$ takes a sparse block form:

$$\boldsymbol{U} = \begin{bmatrix} \mathbb{F} & -\sqrt{\frac{1}{V}}\mathbf{1} & \mathbf{0} \\ \mathbf{0} & \sqrt{\frac{1}{V}}\mathbf{1} & \mathbb{F} \end{bmatrix} \in \mathbb{R}^{V\times(V-1)},$$

where $\mathbb{F} \in \mathbb{R}^{V/2\times(V/2-1)}$ is an orthonormal basis of the subspace orthogonal to $\mathbb{1}_{V/2}$.[6] The structure of $\boldsymbol{U}$ reveals three distinct types of concepts, corresponding to the three tiers of singular values: (1) First $V/2-1$ columns (non-zero only for majority classes) represent distinctions among majority classes; (2) Middle column (singular value $\sqrt{(R+1)/2}$) has opposite-signed entries for majority versus minority classes, encoding the majority-minority dichotomy; (3) Last $V/2-1$ columns (non-zero only for minority classes) capture distinctions among minority classes. This hierarchical structure shows the network learns concepts in order: first majority class distinctions, then the majority-minority split, and finally minority class differences. See Fig. 11 for a visualization of $\boldsymbol{U}$ and Fig. 7 for an experiment that confirms the above semantic interpretation of concepts (columns of $\boldsymbol{U}$).

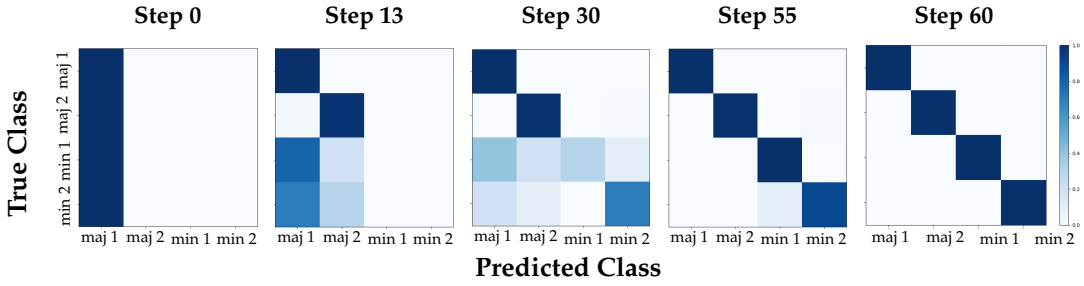

Figure 7: Experimental illustration of the fact that important concepts are learned first. Confusion matrix evolution during training of a 3-layer convolutional network on an imbalanced MNIST data. The matrices show five snapshots at different training steps (0, 13, 30, 55, 65) for four classes: two majority classes (Maj1, Maj2) with 100 samples each and two minority classes (Min1, Min2) with 10 samples each. Training progresses from left to right: initially classifying all data as Maj1 (Step 0), then correctly identifying majority classes while misclassifying minority classes (Step 13), gradually improving minority class recognition (Step 30), confusion between minority classes only (Step 55), and finally achieving perfect classification by Step 60.

B From Sparsity Language Pattern to Concepts Geometry

Recall the centered data-sparsity matrix $\widetilde{S}$ and its SVD

$$\widetilde{S} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top, \quad \text{where} \quad \boldsymbol{U} \in \mathbb{R}^{V\times r}, \boldsymbol{\Sigma} \in \mathbb{R}^{r\times r}, \boldsymbol{V} \in \mathbb{R}^{m\times r} \quad \text{and} \quad \boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{V}^\top\boldsymbol{V} = \mathbb{I}_r,$$

and the singular values $\boldsymbol{\Sigma} = \operatorname{diag}(\sigma_1, \ldots, \sigma_r)$ are ordered:

$$\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r > 0.$$

[6] For concreteness, $\mathbb{F}$ can be constructed using the discrete cosine transform matrix, excluding the constant column: $\mathbb{F}[i,j] = \sqrt{\frac{4}{V}} \cdot \cos\left(\frac{\pi(2i-1)j}{V}\right)$ for $i \in [V/2], j \in [V/2-1]$

Adopting terminology from Saxe et al. (2019), we interpret the columns $\boldsymbol{u}_k \in \mathbb{R}^V, \boldsymbol{v}_k \in \mathbb{R}^m, k \in [r]$ of $\boldsymbol{U}, \boldsymbol{V}$ as word and context analyzer vectors for concept k. For each word $z \in \mathcal{V}$ and each word-concept $k \in [r]$, the component $\boldsymbol{u}_k[z]$ represents how *present* or *absent* is a word z in context k. Respectively for contexts.

Specifically, we think of column dimensions of $\boldsymbol{U}$ as semantic dimensions that capture semantic categories.

Q: How do we define word-concept and context-concept representations, i.e. d-dimensional representations of word and context analyzer vectors for various concepts?

Let $\boldsymbol{W} \in \mathbb{R}^{V \times d}$ and $\boldsymbol{H} \in \mathbb{R}^{d \times m}$ be the representations of words and contexts. We then define **word-concept representations** $\boldsymbol{u}_k^{\mathrm{d}}$ and **context-concept representations** $\boldsymbol{v}_k^{\mathrm{d}}$ for $k \in [r]$ as projections onto the spaces of word and context representations, respectively. Specifically, for projection matrices

$$\mathbb{P}_{\boldsymbol{W}} = \boldsymbol{W}^\top (\boldsymbol{W}\boldsymbol{W}^\top)^{-\dagger} \boldsymbol{W} \qquad \text{and} \qquad \mathbb{P}_{\boldsymbol{H}} = \boldsymbol{H}(\boldsymbol{H}^\top \boldsymbol{H})^{-\dagger} \boldsymbol{H}^\top ,$$

let

$$\begin{aligned} \boldsymbol{u}_k^{\mathrm{d}} &= \mathbb{P}_{\boldsymbol{W}} \boldsymbol{W}^\top \boldsymbol{u}_k \\ \boldsymbol{v}_k^{\mathrm{d}} &= \mathbb{P}_{\boldsymbol{H}} \boldsymbol{H} \boldsymbol{v}_k \,. \end{aligned}$$

Let's now simplify these by using the known SVD representation of $\boldsymbol{W}$ and $\boldsymbol{H}$. Using this representation (i.e. use $\boldsymbol{W} \leftarrow \boldsymbol{W}^{\mathrm{mm}}$, $\boldsymbol{H} \leftarrow \boldsymbol{H}^{\mathrm{mm}}$) we compute

$$\mathbb{P}_{\boldsymbol{W}} = \mathbf{R}\mathbf{R}^\top = \mathbb{P}_{\boldsymbol{H}}.$$

Thus,

$$\boldsymbol{u}_k^{\mathrm{d}} = \mathbf{R}\mathbf{R}^\top \mathbf{R}\sqrt{\boldsymbol{\Sigma}}\boldsymbol{U}^\top \boldsymbol{u}_k = \sigma_k \mathbf{R}\boldsymbol{e}_k = \boldsymbol{W}^\top \boldsymbol{u}_k \tag{4a}$$

$$\boldsymbol{v}_k^{\mathrm{d}} = \sigma_k \mathbf{R}\boldsymbol{e}_k = \boldsymbol{H}\boldsymbol{v}_k = \sum_{j \in [m]} \boldsymbol{v}_k[j] \cdot \boldsymbol{h}_j \,. \tag{4b}$$

We conclude that the d-dimensional representations of word and context analyzer vectors are the same. This is intuitive since concepts are categories in a certain sense broader than words/contexts, where the latter can be thought of as realizations of the concept in the form of explicit constituents of natural language. We thus refer to $\boldsymbol{u}_k^{\mathrm{d}} = \boldsymbol{v}_k^{\mathrm{d}} = \mathbf{R}\boldsymbol{e}_k$ as the representation of concept k. See Fig. 8 for a visualization.

Finally, observe from Eq. (4) that the k-th concept representation is given by a weighted average of word or context embeddings with weights taken by the resppective context analyzer vectors.

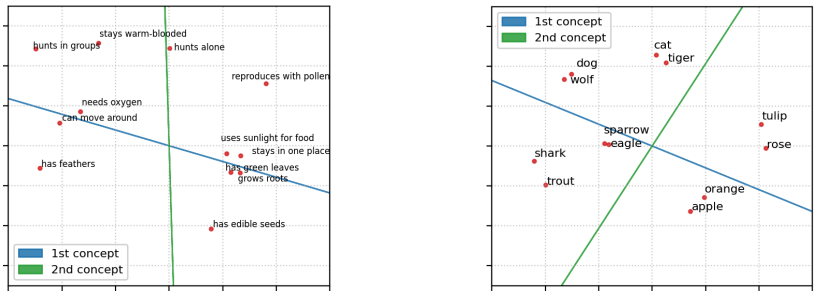

Figure 8: Left: Context embeddings; Right: Word embeddings. In both plots, concept representations (blue, green) are computed using Eq. (4). Projections onto these concept axes (v_1^{d}, v_2^{d}) quantify semantic relevance. The first concept (v_1^{d}, blue line) defines a spectrum from plant-related to animal-related elements, with animals ("hunts in groups", "dog") projecting positively and plants ("uses sunlight for food", "apple") projecting negatively along the same axis. The second concept (v_2^{d}, green line) represents a continuum from non-mammalian to mammalian traits, with mammalian words and features ("cat", "stays warm-blooded") having positive projections and non-mammalian ones ("trout", "has feathers") having negative projections. Please note that the concepts are not orthogonal in the plots because it is PCA projection to 2D space.

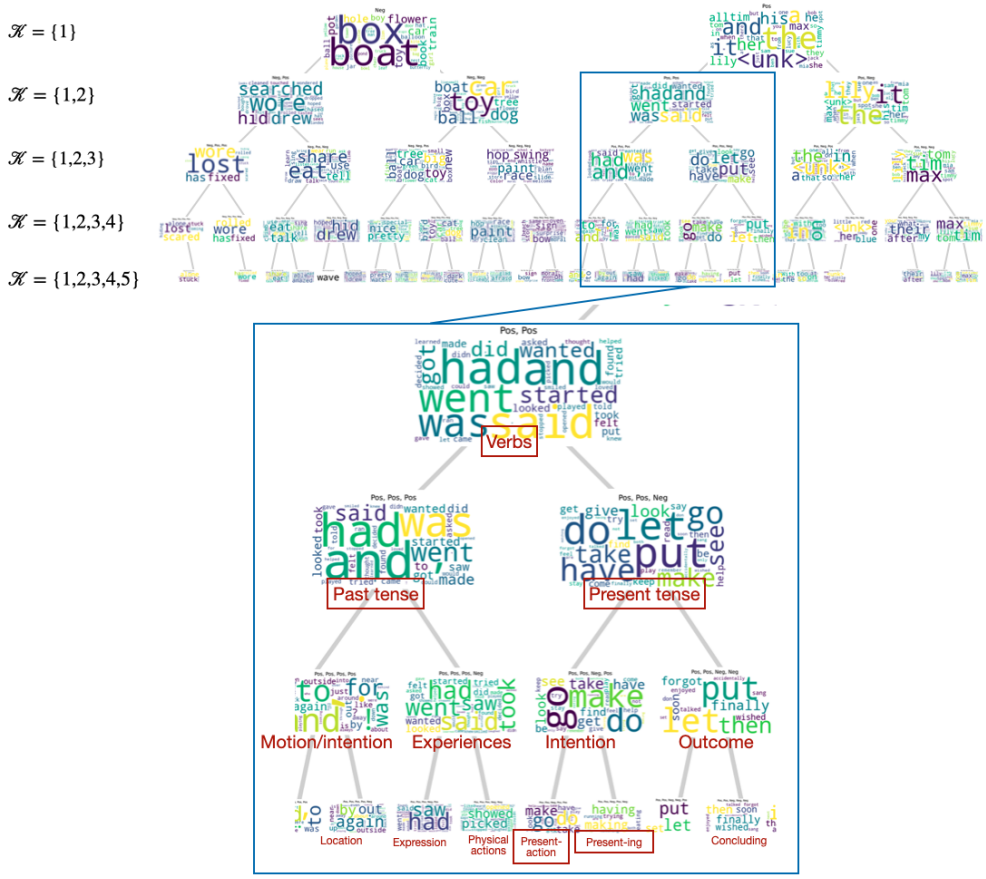

Figure 9: Hierarchical semantic structure revealed by orthant-Based clustering on Simplified TinyStories. **Top:** Hierarchical semantic structure emerges as we combine concepts. Each row represents clusters formed by combining an increasing number of concept dimensions (from 1 to 5), with each word cloud displaying the most typical words for a specific sign configuration. Semantic categories become progressively more fine-grained moving downward as additional concept dimensions refine the classifications. **Bottom:** Detailed view of verb semantics. The combination of the first two concept dimensions ($\mathcal{K} = \{1,2\}$ with configuration $C_{\mathcal{K}} = \{+1,+1\}$) identifies verbs. Adding the third dimension further distinguishes between past tense ($C_{\mathcal{K}} = \{+1,+1,+1\}$) and present tense ($C_{\mathcal{K}} = \{+1,+1,-1\}$). The present tense category is further refined by the fifth dimension to differentiate present action verbs from present continuous (-ing) forms. This hierarchical organization mirrors linguistic grammar structures and supports our hypothesis that language models learn fundamental grammatical distinctions (verb vs. non-verb) before more specific ones (tense and aspect distinctions).

C On the hierarchical structure of language

As discussed in Sec. 4.3, language semantics exhibit a hierarchical structure, with more comprehensive semantics involving numerous concepts being inherently more detailed. Each concept's significance is indicated by its associated singular values; higher values denote concepts critical for differentiating words or contexts. Moreover, as discussed in Sec. 4.4, broader semantic categories are acquired more rapidly, indicating a sequential learning progression from general to more specific concepts. Fig. 9 below provides an illustration of this hierarchical structure.

D Connections to closed-form training dynamics Saxe et al. (2013)

Consider the following square-loss NTP-UFM proxy:

$$\min_{\boldsymbol{W},\boldsymbol{H}} \left\{ \sum_{j=1}^{m} \|\widetilde{\boldsymbol{s}}_j - \boldsymbol{W}\boldsymbol{h}_j\|^2 = \|\widetilde{\boldsymbol{S}} - \boldsymbol{W}\boldsymbol{H}\|^2 \right\}. \tag{5}$$

which fits logits $\boldsymbol{W}\boldsymbol{H}$ to the centered sparsity matrix $\widetilde{\boldsymbol{S}}$. For one-hot labels, this reduces to standard square-loss classification, which has shown competitive performance to CE minimization in various settings (Hui and Belkin (2020); Demirkaya et al. (2020)). However, in our soft-label setting, the choice of loss function requires more careful consideration. While one could follow Glove's approach Pennington et al. (2014) by using $\log(\boldsymbol{P})$ instead of $\widetilde{\boldsymbol{s}}$, this creates issues with $\boldsymbol{P}$'s zero entries. Since CE loss minimization leads $\boldsymbol{W}$ and $\boldsymbol{H}$ to factor $\widetilde{\boldsymbol{s}}$, we thus maintain $\widetilde{\boldsymbol{s}}$ as the target in (5).

The neural-collapse literature has extensively studied Eq. (5) for one-hot $\widetilde{\boldsymbol{s}}$, primarily in the balanced case (e.g., Mixon et al. (2020); Han et al. (2021); Súkeník et al. (2023); Tirer and Bruna (2022); **?**) but recently also for imbalanced data (e.g., Liu (2024); Hong and Ling (2023)). Most works focus on global minima of regularized UFM, with less attention to unregularized cases or training dynamics. While some landscape analyses provide partial answers about global convergence (e.g.,), they are limited to regularized cases and don't characterize dynamics. Even for square loss, where Mixon et al. (2022); Han et al. (2021) study training dynamics—notably Han et al. (2021)'s analysis of the 'central path' in balanced one-hot cases—these results rely on approximations. Thus, a significant gap remains in understanding UFM training dynamics, even for simple balanced one-hot data with square loss.

By interpreting the UFM with square loss in Eq. (5) as a two-layer linear network with orthogonal inputs, we identify an unexplored connection to Saxe et al. (2013); Gidel et al. (2019)'s analysis. They provide explicit characterization of gradient descent dynamics (with small initialization) for square-loss UFM. The key insight is rewriting (5) as $\sum_{j\in[m]} |\widetilde{\boldsymbol{s}}_j - \boldsymbol{W}\boldsymbol{H}\boldsymbol{e}_j|^2$ with orthogonal inputs $\boldsymbol{e}_j \in \mathbb{R}^m$. This enables direct application of their result, originally stated in Saxe et al. (2013) and formalized in Gidel et al. (2019). For completeness, we state this here in our setting and terminology as a corollary below.

Proposition 1. *Consider gradient flow (GF) dynamics for minimizing the square-loss NTP-UFM (5). Recall the SVD $\widetilde{\boldsymbol{S}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$. Assume weight initialization*

$$\boldsymbol{W}(0) = e^{-\delta}\boldsymbol{U}\mathbf{R}^\top \quad \textit{and} \quad \boldsymbol{H}(0) = e^{-\delta}\mathbf{R}\boldsymbol{V}^\top$$

for some partial orthogonal matrix $\mathbf{R} \in \mathbb{R}^{d\times r}$ ($\mathbf{R}^\top\mathbf{R} = \mathbb{I}_r$) and initialization scale $e^{-\delta}$. Then the iterates $\boldsymbol{W}(t), \boldsymbol{H}(t)$ of GF are as follows:

$$\boldsymbol{W}(t) = \boldsymbol{U}\sqrt{\boldsymbol{\Sigma}}\sqrt{\boldsymbol{A}(t)}\mathbf{R}^\top \qquad \textit{and} \qquad \boldsymbol{H}(t) = \mathbf{R}\sqrt{\boldsymbol{\Sigma}}\sqrt{\boldsymbol{A}(t)}\boldsymbol{V}^\top \tag{6}$$

for $\boldsymbol{A}(t) = \operatorname{diag}(a_1(t), \dots, a_r(t))$ with

$$a_i(t) = \frac{1}{1 + (\sigma_i e^{2\delta} - 1)e^{-2\sigma_i t}},\ i \in [r]. \tag{7}$$

Moreover, the time-rescaled factors $a_i(\delta t)$ converge to a step function as $\delta \to \infty$ (limit of vanishing initialization):

$$a_i(\delta t) \to \frac{1}{1+\sigma_i}\mathbb{1}[t = T_i] + \mathbb{1}[t > T_i], \tag{8}$$

where $T_i = 1/\sigma_i$ and $\mathbb{1}[A]$ is the indicator function for event A. Thus, the i-th component is learned at time T_i inversely proportional to σ_i.

Proof. After having set up the analogy of our setting to that of Saxe et al. (2013); Gidel et al. (2019), this is a direct application of (Gidel et al., 2019, Thm. 1). Specifically, this is made

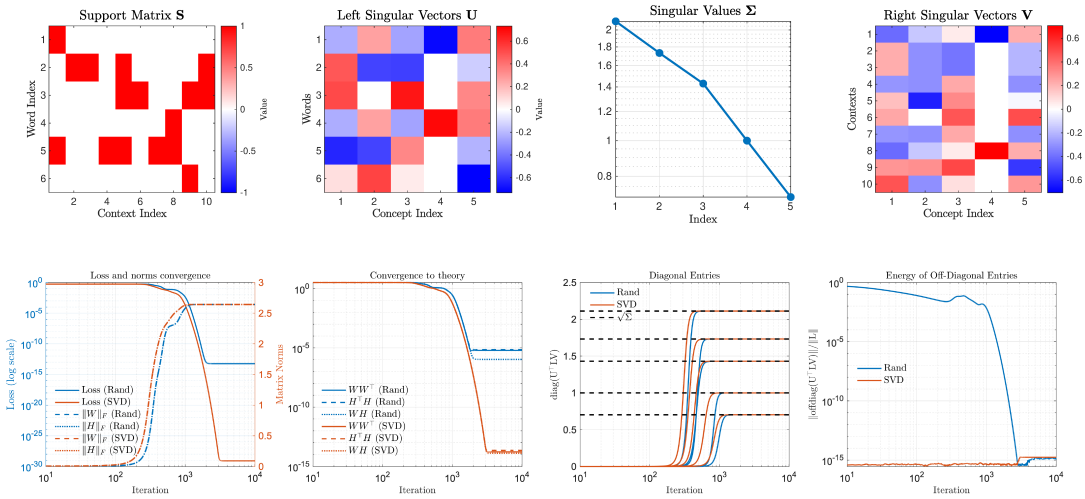

Figure 10: **(Top)** Support matrix and SVD factors of centered support matrix for a synthetic example. **(Bottom)** Training dynamics of GD minimization of NTP-UFM with square loss (Eq. (5) for two initializations: (i) SVD: initialize $\boldsymbol{W}$ and $\boldsymbol{H}$ as per Thm. 1 for $\delta = 8$. (ii) Rand: intialize $\boldsymbol{W}$ and $\boldsymbol{H}$ random Gaussian scaled to match the norm of SVD initialization. Dynamics with the two initialization are shown in red (SVD) and blue (Rand), respectively. Qualitatively the behavior is similar. *Left:* Training loss and norms of paramteres. *Middle-Left:* Convergence of word and context gram-matrices and of logits to the theory predicted by Thm. 1. *Middle-Right:* Convergence of singular values of logit matrix to those of $\boldsymbol{\Sigma}$ (see Thm. 1. *Right:* Projection of logits to subspace orthogonal to $\boldsymbol{U}$ and $\boldsymbol{V}$; Logits with Rand initialization initially have non-zero projection but it becomes zero as training progresses.

possible in our setting by: (i) interpreting the UFM with square-loss in (5) as a two-layer linear network (ii) recognizing that the covariance of the inputs (which here are standard basis vectors $\boldsymbol{e}_j, j \in \mathbb{R}^m$) is the identity matrix, hence the (almost) orthogonality assumption (see (Saxe et al., 2013) and (Gidel et al., 2019, Sec. 4.1)) holds. □

The result requires initializing word/context embeddings aligned with the SVD factors of the data-sparsity matrix. While this might appear as a strong assumption, Saxe et al. (2013; 2019) conjectured and verified experimentally that the characterization remains qualitatively accurate under small random initialization. Our experiments with the data-sparsity matrix confirm this - Fig. 16 shows the singular values of the logit matrix during training closely follow the predicted exponential trend in Eq. (7). This reveals that dominant singular factors, corresponding to primary semantic concepts, are learned first. In the limit $t \to \infty$, the theorem shows convergence to:

$$\boldsymbol{W}(t) \to \boldsymbol{W}_\infty := \boldsymbol{U}\sqrt{\boldsymbol{\Sigma}}\mathbf{R}^\top \qquad \text{and} \qquad \boldsymbol{H}(t) \to \boldsymbol{H}_\infty := \mathbf{R}\sqrt{\boldsymbol{\Sigma}}\boldsymbol{V}^\top, . \tag{9}$$

This aligns with Zhao et al. (2024)'s regularization-path analysis of NTP-UFM with CE loss, where normalized quantities converge as $\lambda \to 0$ (recall Section 3.2):

$$\overline{\boldsymbol{W}}(\lambda) \to \boldsymbol{U}\sqrt{\overline{\boldsymbol{\Sigma}}}\mathbf{R}^\top \qquad \text{and} \qquad \overline{\boldsymbol{H}}(\lambda) \to \mathbf{R}\sqrt{\overline{\boldsymbol{\Sigma}}}\boldsymbol{V}^\top.$$

D.1 An example of controlling the rate of learning via reweighting

Consider minimizing the following weighted version of (5):

$$\min_{\boldsymbol{W},\boldsymbol{H}} \left\| \left(\widetilde{\boldsymbol{S}} - \boldsymbol{W}\boldsymbol{H} \right) \boldsymbol{\Omega} \right\|^2, \tag{10}$$

where $\mathbf{\Omega} = \mathrm{diag}([\omega_1, \dots, \omega_m])$ is a diagonal matrix of weights, one for each context. Here, we consider the STEP-imbalanced one-hot classification setting described in Sec. A.5. Concretely, let the support matrix be

$$\boldsymbol{S} = \begin{bmatrix} \mathbb{I}_{V/2} \otimes \mathbb{1}_R^\top & \mathbf{0}_{V/2 \times V/2} \\ \mathbf{0}_{V/2 \times RV/2} & \mathbb{I}_{V/2} \end{bmatrix} \tag{11}$$

where R is the imbalance ratio and without loss of generality we assumed that the first $V/2$ classes are majorities and that minorities have 1 example (in our language: context) each. (Thus, the total number of examples is $m = RV/2 + V/2 = (R+1)V/2$.) Recall that $\widetilde{\boldsymbol{S}} = (\mathbb{I}_V - \frac{1}{V}\mathbb{1}_V\mathbb{1}_V^\top)\boldsymbol{S}$.

Recall from Sec. A.5 that $\widetilde{\boldsymbol{S}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ where the singular values and the left singular vectors are explicitly given by:

$$\boldsymbol{\Sigma} = \mathrm{diag}(\begin{bmatrix}\sqrt{R}\mathbb{1}_{V/2-1} & \sqrt{(R+1)/2} & \mathbb{1}_{V/2-1}\end{bmatrix}), \qquad \boldsymbol{U} = \begin{bmatrix} \mathbb{F} & -\sqrt{\frac{1}{V}}\mathbb{1} & \mathbf{0} \\ \mathbf{0} & \sqrt{\frac{1}{V}}\mathbb{1} & \mathbb{F} \end{bmatrix} \in \mathbb{R}^{V \times (V-1)},$$

with $\mathbb{F} \in \mathbb{R}^{V/2 \times (V/2-1)}$ an orthonormal basis of the subspace orthogonal to $\mathbb{1}_{V/2}$. According to (Thrampoulidis et al., 2022, Lem. A.3) we also have

$$\boldsymbol{V}^\top = \begin{bmatrix} \mathbb{F}^\top \otimes \mathbb{1}_R^\top & \mathbf{0} \\ -\sqrt{\frac{2}{(R+1)V}}\mathbb{1}_{RV/2}^\top & \sqrt{\frac{2}{(R+1)V}}\mathbb{1}_{V/2}^\top \\ \mathbf{0} & \mathbb{F}^\top \end{bmatrix} \in \mathbb{R}^{(V-1)\times m}.$$

We now set the weight matrix as follows:

$$\mathbf{\Omega} := \mathrm{diag}(\begin{bmatrix}\sqrt{\frac{1}{R}}\,\mathbb{1}_{RV/2}^\top & \mathbb{1}_{V/2}^\top\end{bmatrix}). \tag{12}$$

Direct calculation yields

$$\begin{aligned} \boldsymbol{V}^\top\mathbf{\Omega} &= \begin{bmatrix} \sqrt{\frac{1}{R}}\mathbb{F}^\top \otimes \mathbb{1}_R^\top & \mathbf{0} \\ -\sqrt{\frac{2}{R(R+1)V}}\mathbb{1}_{RV/2}^\top & \sqrt{\frac{2}{(R+1)V}}\mathbb{1}_{V/2}^\top \\ \mathbf{0} & \mathbb{F}^\top \end{bmatrix} \\ &= \mathrm{diag}(\begin{bmatrix}\sqrt{\frac{1}{R}}\mathbb{1}_{V/2-1}^\top & \sqrt{\frac{2}{R+1}} & \mathbb{1}_{V/2-1}^\top\end{bmatrix}) \underbrace{\begin{bmatrix} \mathbb{F}^\top \otimes \mathbb{1}_R^\top & \mathbf{0} \\ -\sqrt{\frac{1}{RV}}\mathbb{1}_{RV/2}^\top & \sqrt{\frac{1}{V}}\mathbb{1}_{V/2}^\top \\ \mathbf{0} & \mathbb{F}^\top \end{bmatrix}}_{=:\widetilde{\boldsymbol{V}}^\top} = \boldsymbol{\Sigma}^{-1}\widetilde{\boldsymbol{V}}^\top. \end{aligned} \tag{13}$$

where in the last equation, we recognized the diagonal matrix is equal to $\boldsymbol{\Sigma}^{-1}$ and called the other matrix $\widetilde{\boldsymbol{V}} \in \mathbb{R}^{m \times (V-1)}$. Thus, the *effective* sparsity matrix $\widetilde{\boldsymbol{S}}_{\mathbf{\Omega}} =: \widetilde{\boldsymbol{S}}\mathbf{\Omega}$ is equal to

$$\widetilde{\boldsymbol{S}}_{\mathbf{\Omega}} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top\mathbf{\Omega} = \boldsymbol{U}\widetilde{\boldsymbol{V}}^\top.$$

It is now easy to check that $\widetilde{\boldsymbol{V}}^\top\widetilde{\boldsymbol{V}} = \mathbb{I}_{V-1}$. Thus, the display above is the SVD factorization of $\widetilde{\boldsymbol{S}}_{\mathbf{\Omega}}$. Note that the singular values of this *effective* sparsity matrix are all equal, unlike the singular values $\boldsymbol{\Sigma}$ of the original data-sparsity matrix $\widetilde{\boldsymbol{S}}$. With this at hand, we can show analogous to Proposition 1 that the eigenvalues of $\boldsymbol{W}(t)$ and $\boldsymbol{H}(t)$ are now all learned at the same time $T = 1$. We also confirm this experimentally in Fig. 11.

E Extended Discussion on Related Works

Word embeddings and semantic analysis in neural probabilistic language models. The word2vec architecture Mikolov et al. (2013d;b) and its variants, notably GloVe Pennington

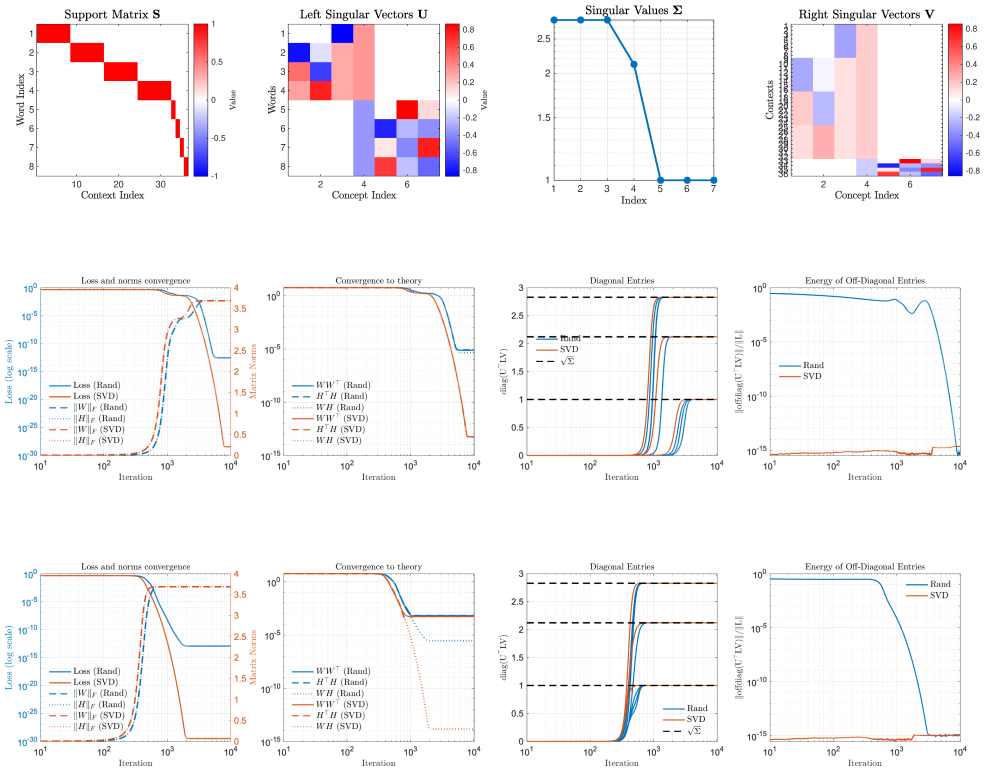

Figure 11: **(Top)** Support matrix and SVD factors of centered support matrix for a one-hot STEP-imbalanced example. **(Middle)** Training dynamics of GD minimization of NTP-UFM with square loss (Eq. (5); same setup and visualizations as Fig. 10. **(Bottom)** Training dynamics of GD minimization of NTP-UFM with weighted square loss (Eq. (10)) with weights as in Eq. (12). Note that thanks to the weighting all singular factors are now learned simultaneuously.

et al. (2014), represent seminal early neural probabilistic language models. These simple log-bilinear models, trained on large text corpora, revolutionized word embedding learning. As noted in Zhao et al. (2024), NTP-UFM shares structural similarities with these early models, though in both their work and ours, it serves as a tractable abstraction rather than a practical architecture. Our approach differs by learning both context and word embeddings, following modern practice. The foundational work of Levy and Goldberg (2014) connected word2vec's geometry to matrix factorization of the pointwise mutual information (PMI) matrix—a specialized word co-occurrence matrix. Subsequent works Levy et al. (2015); Turney and Pantel (2010); Baroni and Lenci (2010) empirically demonstrated semantic interpretations of the PMI matrix's singular factors and principal components. Building on Zhao et al. (2024), which formalizes a modern version of Levy and Goldberg (2014)'s results, our investigation of concepts differs from this classical literature in two key aspects: (a) We study the NTP setting where both context and word embeddings are learned, yielding concepts that relate to both words and contexts; (b) Our data-sparsity matrix differs fundamentally from classical PMI matrices: it is a centered version of the data support matrix (independent of specific next-token probabilities) and has different structural properties—being orthogonal and non-square, unlike the word2vec setting.

Superposition and feature steering. Our work was partly motivated by recent compelling literature suggesting that embeddings can be decomposed into linear combinations of a finite set of semantic concepts Bricken et al. (2023); Yun et al. (2023); Park et al. (2023). These insights from mechanistic interpretability have led to practical applications in "feature steering"—where model behavior can be controlled by manipulating concept representations

through addition or subtraction Durmus et al. (2024); Konen et al. (2024). Our analysis complements the mechanistic interpretability approach by providing a systematic framework for understanding how concepts emerge naturally as principal components from training data statistics. Exploring deeper connections between our theoretical framework and the mechanistic interpretability literature remains an intriguing direction for future work. For completeness, we note that Park et al. (2023; 2024) also investigate geometric properties of concept directions, albeit through fundamentally different technical approaches, assumptions, and perspectives.

Saxe et al.'s closed-form dynamics of two-layer linear network training. Our work draws inspiration from Saxe et al. (2019). Conceptually, Saxe et al. (2019) uses a two-layer linear neural network as a theoretical proxy to study the emergence of semantic knowledge in human cognition, providing mathematical justifications for phenomena observed in cognitive semantics literature. A key insight from their work is that even a simple two-layer linear network with orthogonal inputs can yield rich and meaningful conclusions about semantic learning. While two-layer neural networks represent perhaps the simplest instances of non-linear learning, their training dynamics generally remain analytically intractable. However, Saxe et al. (2013) (with aspects later formalized in Gidel et al. (2019)) demonstrated that with square loss, orthogonal inputs, and sufficiently small initialization, these dynamics admit exact closed-form solutions. This mathematical characterization underlies their results on semantic information development through singular factors of the network's input-output correlation matrix. We make a novel connection to this line of work: the UFM fits perfectly within the framework studied by Saxe et al. (2013; 2019). Specifically, the UFM can be viewed as a linear two-layer network where the input dimension equals the number of input examples (in our case, the number of contexts m). This connection is valuable in two directions: First, the UFM—recently popularized through neural collapse literature (see below)—provides perhaps the most natural and practical setting satisfying Saxe et al.'s seemingly restrictive orthogonal input assumptions. Second, this connection allows us to leverage Saxe et al.'s earlier results in the evolving neural collapse literature. Despite these methodological similarities with Saxe et al. (2019), our work differs in motivation and interpretation. We focus specifically on NTP and how semantic and grammatical concepts emerge from natural language data, rather than general cognitive development. Additionally, we primarily focus on CE loss which is typically used in NTP training.

Neural-collapse geometries. Our results contribute to the recent literature on the neural collapse (NC) phenomenon Papyan et al. (2020a). Originally observed in one-hot classification training of DNNs, neural collapse describes two key properties of well-trained, sufficiently expressive DNNs: (1) NC: embeddings of examples from the same class collapse to their class mean, and (2) ETF-geometry: class-mean embeddings form a simplex equiangular tight frame (ETF), being equinorm and maximally separated, with classifier weights exhibiting the same structure and aligning with their respective class-mean embeddings. This phenomenon, consistently observed across diverse datasets and architectures, has sparked extensive research interest, generating hundreds of publications a complete review of which is beyond our scope. Instead, we discuss below the most closely-related line of works. One fundamental direction, which forms the basis for many extensions, focuses on explaining NC's emergence through the UFM. This model abstracts training as joint optimization of last-layer embeddings (unconstrained by architecture) and classifier weights Mixon et al. (2022); Fang et al. (2021). Multiple influential works have proven NC emergence by analyzing the UFM's global optima (e.g., Zhu et al. (2021); Garrod and Keating (2024); Jiang et al. (2023)), with extensions to various loss functions beyond cross-entropy, including square loss Zhou et al. (2022a); Han et al. (2021); Tirer and Bruna (2022); Súkeník et al. (2023; 2024) and supervised contrastive loss Graf et al. (2021); Zhou et al. (2022b); Kini et al.. Most early works maintained the original assumptions from Papyan et al. (2020a): balanced data (equal examples per class) and embedding dimension d exceeding the number of classes C. Recent work has explored $d < C$ settings, though often requiring additional assumptions on the loss function Jiang et al. (2023); Liu et al. (2023). More substantial progress has emerged in the $d > C$ regime with unbalanced data, where Thrampoulidis et al. (2022) provided a complete characterization for step-imbalanced data (where examples

are distributed equally within minority classes and equally within majority classes). They introduced the SELI (simplex-encoding labels interpolation) geometry, showing that logits interpolate a simplex-encoding matrix—a centered version of the one-hot encoding matrix. The embeddings and classifier vectors are then determined, up to rotation and scaling, by the singular vectors of this matrix. The SELI geometry emerges as a special case of the richer geometries characterized in the NTP setting by Zhao et al. (2024). Together with Li et al. (2023), these works stand alone in extending geometric characterization beyond one-hot encoding—to soft-label and multilabel settings respectively. Specifically, Zhao et al. (2024) analyzes the soft-label setting arising in NTP training on natural language, showing that word and context embeddings are determined by the singular factors of the data sparsity matrix. Our work deepens this understanding by revealing that these SVD factors encode conceptual meaning, thereby extending neural collapse geometry to capture not only the structure of embeddings but also the organization of latent concepts.

Connections to Classical Co-occurrence-Based Semantics

Remark 1. *The idea that SVD factors of a co-occurrence-type matrix convey latent semantic information dates back at least to latent semantic analysis of word-document co-occurrence matrices Landauer (1997); Deerwester et al. (1990) and later to pointwise mutual information matrices of word-word co-occurrences Levy and Goldberg (2014); Levy et al. (2015). Beyond SVD factorization, researchers have explored alternative factorization techniques, such as non-negative matrix factorization Lee and Seung (1999); Ding et al. (2006) and sparse dictionary learning Murphy et al. (2012); Faruqui et al. (2015), and have applied them to various transformations of co-occurrence data, including modifications that address the sparsity of PMI matrices by eliminating negative entries (e.g., PPMI) or applying probability smoothing techniques Levy et al. (2015). Our key conceptual contribution is that we let the training dynamics NTP optimization determine both the matrix representation of the data and the factorization method used to encode latent semantics: the model naturally utilizes the centered support matrix $\widetilde{S}$ and its SVD factorization. We validate through structured examples in this section and through experiments in Sec. 5, that the model's emergent representational choices successfully encode linguistic components without explicit engineering of the representation space.*

F Additional Experiment Results

F.1 Semantics for Individual Concept

In this section we include the plots illustrating that individual concept do not contain interpretable semantics as discussed in sec 5

Figure 12: Word clouds of top words from individual concept dimensions across datasets, illustrating their lack of human-interpretable semantics without combination. Shown: positive 2nd dimension (Simplified TinyStories), negative 6th dimension (Simplified TinyStories), negative 2nd dimension (Simplified WikiText), positive 4th dimension (Simplified WikiText).

F.2 Experiment on the Semantics of Pretrained Context Embeddings

To identify the semantics in context embeddings from pretrained transformers, we chose GPT-2 Radford et al. (2019) and extracted 50,000 text sequences, each 11 tokens in length from GPT-2's training corpus, the OpenWebText dataset. By segmenting these sequences into contexts of varying lengths (1-10 tokens), we generated 500,000 contexts, which we designate as the "simplified OpenWebText" dataset.

Figure 13: orthant-based clustering on Pretrained word embedding, $p = 4$, GPT2

To obtain concepts from pretrained models, instead of performing SVD on $\widetilde{S}$ (which is computationally expensive to construct and factorize), we work directly with the embeddings of the pretrained model. For word concepts, we extract $\boldsymbol{W}$ from GPT-2's decoder and take the matrix $\boldsymbol{U}$ of word concepts to be its left singular matrix. For context concepts, we form a $d \times m$ matrix $\boldsymbol{H}$ by concatenating the embeddings (last-MLP layer representations before the decoder) of GPT-2 for each context in our subset. This matrix corresponds to only a portion of the "true" embedding matrix for the entire dataset. To ensure the coupling between $\boldsymbol{W}$ and $\boldsymbol{H}$ that is inherent to the SVD of $\widetilde{S}$ (which we don't have direct access to), we compute the matrix $\boldsymbol{V}$ of context concepts as $\boldsymbol{V} = \boldsymbol{H}^\top \mathbf{R} \boldsymbol{\Sigma}^{-1/2}$, where $\boldsymbol{U}\boldsymbol{\Sigma}^{1/2}\mathbf{R}$ denotes the SVD of $\boldsymbol{W}$ that we already computed.

We include the orthant-clustering result on GPT2's pretrained word embedding with $p = 4, 5, 6$. (Fig. 13, 14, 15)

F.3 Rate of learning

Fig. 16 shows the evolution of singular values during training for both squared loss and cross-entropy loss as described in Section 4.4.

F.4 Extension to Masked Language Models

While our primary analysis focuses on next-token prediction (NTP), the underlying principles of our framework are also applicable to masked language modeling (MLM) objectives.

Figure 14: Semantics on Pretrained word embedding, $p = 5$, GPT2

Figure 15: orthant-based clustering on Pretrained word embedding, $p = 6$, GPT2

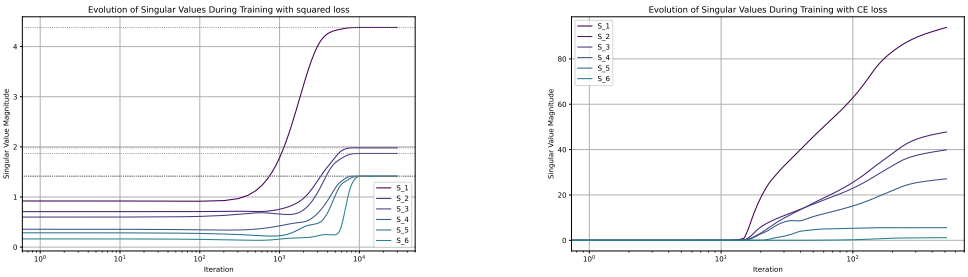

Figure 16: Evolution of singular values of the logit matrix during training. Both plots show that dominant concepts (corresponding to larger singular values) are learned first. **(Left)** With squared loss, singular values converge to those of the optimal solution, demonstrating learning saturation. **(Right)** With CE loss, singular values grow unboundedly while maintaining their relative ordering, reflecting the continuous growth of embedding norms characteristic of CE training.

Both settings rely on context–token co-occurrence patterns, with the key difference being how contexts and targets are defined. In NTP, the context is a sequence of input tokens, and the label is the next token. In MLMs, the context is formed by masking tokens within a sequence, and the objective is to predict those masked tokens given their surrounding context.

To explore this connection, we applied orthant-based clustering to token embeddings from a pretrained BERT model. As shown in Fig. 17, the method recovers a range of interpretable semantic categories, including numerical patterns (e.g., 3-digit numbers, historical years), functional word types (e.g., action verbs, administrative terms), and morphological structures (e.g., verb suffixes, name suffixes, given names).

This example illustrates the broader applicability of our framework and opens up further questions on how co-occurrence structure affects semantic emergence under different training objectives.

Figure 17: Semantics identified by orthant-based clustering on BERT's word embeddings. The sign configuration (e.g., Pos, Neg, Neg, Pos) indicates the directionality of the concept in the top-p embedding axes. In BERT's tokenizer, tokens prefixed with ## represent subword units that appear within or at the end of a word.

F.5 Extension to Higher-Dimensional Embeddings

To assess the applicability of our framework in large-scale settings, we extend our analysis to the Qwen-7B model, which features a 4096-dimensional embedding space. Despite the increased dimensionality, we find that orthant-based clustering on the learned embeddings continues to reveal semantically meaningful structure.

Given that Qwen's tokenizer includes a large number of Chinese characters, many of which cannot be reliably visualized in our word cloud plots, we instead summarize representative semantic categories for selected orthant configurations in Table 1. These configurations correspond to sign patterns along the most significant concept directions (i.e., the top singular vectors) derived from the word embedding matrix $\boldsymbol{W}$.

The discovered clusters include interpretable groupings such as punctuation symbols, programming-related terms, popular acronyms, and functional Chinese words or prompt-like expressions commonly found in UI or news content. Interestingly, we observe that Qwen requires more concept directions to isolate some specific semantic categories (e.g., grammatical function words) compared to smaller models like GPT-2 (5-7 concepts for GPT-2 vs. 7-11 concepts for Qwen). We hypothesize that this may be due to the higher embedding dimensionality and the broader linguistic coverage of Qwen.

This result provides further evidence that the spectral geometry we study has the potential to generalize to more complex, multilingual, higher-dimensional language models.

Configuration $C = [c_{k_1}, \ldots, c_{k_p}]$	**Possible Semantic**	**Language**	**Examples**
$[+1,+1,+1,-1,-1]$	Punctuation / Code Symbols	Mixed (mostly code symbols)	`:`, `(`, `$`, `->`, `//`
$[+1,+1,+1,-1,+1]$	Code / Programming Identifiers	English	`in`, `id`, `app`, `list`, `data`
$[-1,-1,-1,-1,-1,+1]$	Programming Frameworks and Objects	Mixed	`minecraft`, `ScrollView`, `UIImagePickerController`, `PIXI`, `kontrol`
$[-1,-1,+1,-1,+1,-1,-1]$	Two-letter tech acronyms / platforms	English	`mt`, `vc`, `AI`, `ML`, `ios`
$[-1,+1,-1,+1,-1,+1,+1]$	Chinese UI/Content Descriptions	Chinese	可以更好, 游戏操作, 名列前, 友情链接, 网站地图
$[-1,-1,+1,+1,-1,+1,-1,+1]$	Programming / GUI Class Names	English	`PIXEL`, `MainMenu`, `Snackbar`, `PyQt`, `StringUtil`
$[-1,+1,+1,+1,-1,-1,+1,+1]$	Colloquial Chinese Expressions	Chinese	的乐趣, 有意思的, 的情绪, 了下来, 了好多
$[+1,+1,+1,+1,-1,-1,-1,+1,-1,-1]$	Digits + Operators	Mixed	`2`, `1`, `3`, `^`, `+`
$[-1,+1,-1,-1,-1,-1,+1,+1,-1,-1,+1]$	News / Official Terms	Chinese	汶川, 新时期, 正文, 解放军, 中共中央, 十四
$[+1,-1,-1,-1,-1,+1,-1,-1,+1,-1,-1]$	Chinese Function Words / Grammar	Chinese	的, 和, 在, 与, 或

Table 1: Examples of semantic categories recovered from Qwen-7B using orthant-based clustering on its 4096-dimensional word embeddings. Each row corresponds to a specific sign configuration $C = [c_{k_1}, \ldots, c_{k_p}]$ along the top singular concept directions.

F.6 $d > V$

Fig. 18 shows the evolution of singular values for squared loss when the network's embedding dimension is smaller than the vocabulary size, i.e., $d < V$, as described in Section A.2.

F.7 Semantics from K-means Spectral Clustering on Simplified TinyStories and Wiki-Text

k-means spectral clustering. As a complementary approach to our orthant-based clustering, we also investigate k-means spectral clustering. Specifically, we perform k-means clustering according to pairwise distances of analyzer vectors with their top $p = \log_2(k)$ dimensions (corresponding to top-p most important concepts). Vanilla k-means spectral clustering selects k principal dimensions. The modification here is inspired by the orthant-based

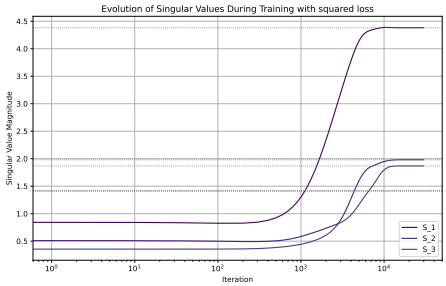

Figure 18: evolution of the learned logit matrix's singular values during training for $3 = d < V$, note that the 3 singular values are converging to the d largest singular values

clustering for which p concepts correspond to 2^p orthants. Fig. 19 shows semantic meaning in a subset of the recovered clusters with $k = 32$ and $k = 64$ for Simplified TinyStories and WikiText-2, respectively; see Sec. F.7 for full results. See also discussion in Sec. A.4.

Figure 19: k-means cluster with top-$p = \log_2(k)$ dimensions of word-analyzer vectors. *Left:* 2 of 32-means clusters with $\mathcal{K} = \{1, \ldots, 5\}$ on Simplified TinyStories representing Past-tense verbs and Proper names; *Right:* 2 of 64-means cluster with $\mathcal{K} = \{1, \ldots, 6\}$ on simplified WikiText, representing Numbers and Conjunctions.

Please also see Fig. 20 and 21 for full result.

F.8 More orthant-based clustering result on Simplified TinyStories and Wiki-Text

We include some examples of orthant-based clustering results in Fig. 22 to 25. We selected $p = 4$, $p = 5$ for Simplified TinyStories and $p = 5$, $p = 6$ for Simplified WikiText because we observe the most human-interpretable concepts.

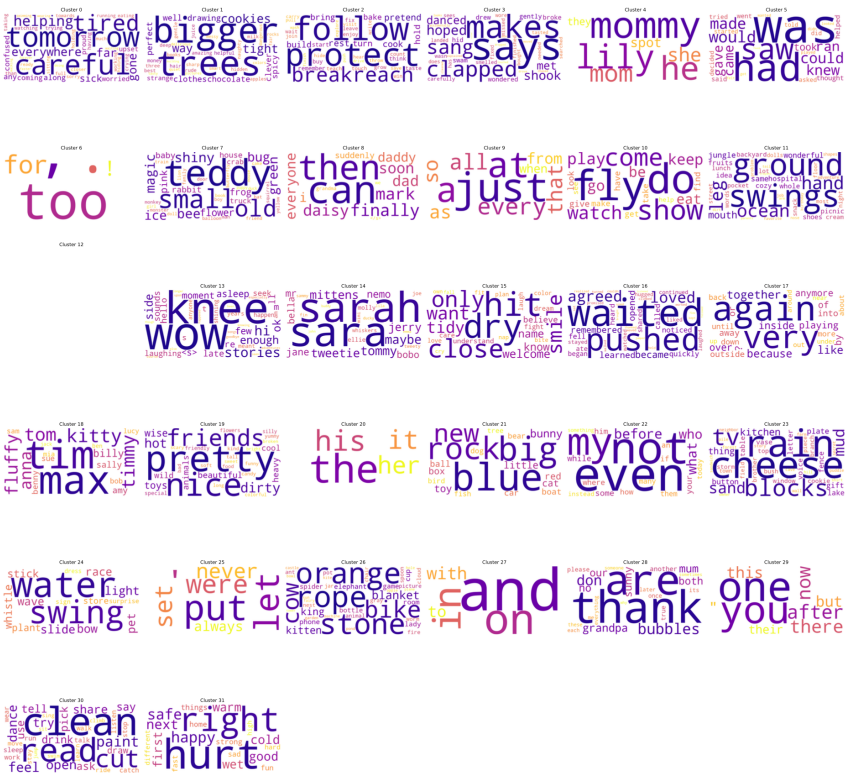

Figure 20: 32-means spectral clustering, $p = 5$, Simplified TinyStories

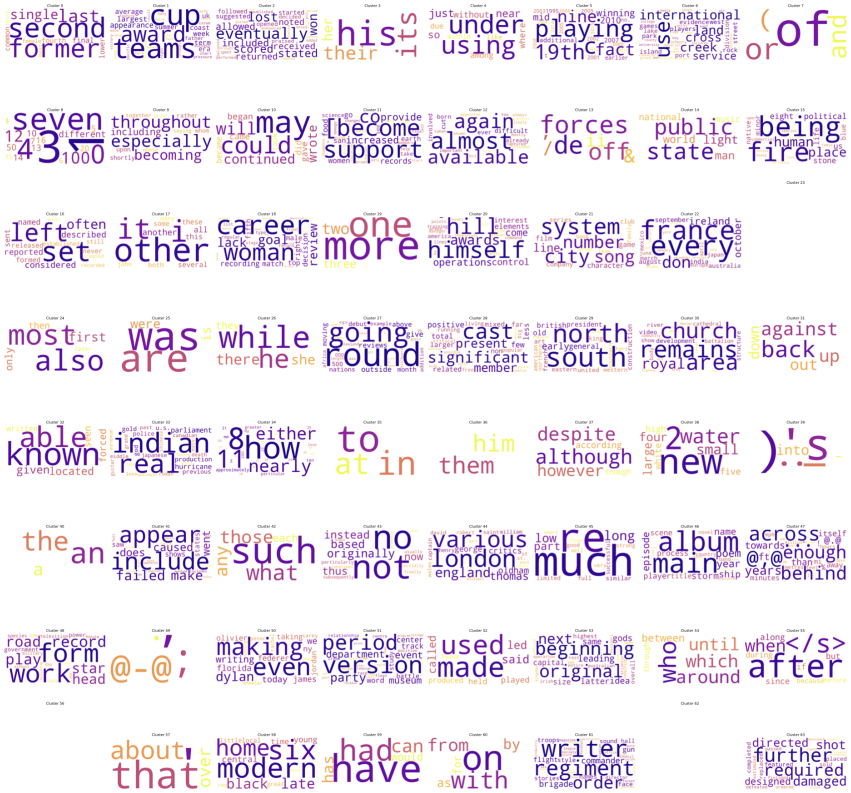

Figure 21: 64-means spectral clustering, $p = 6$, Simplified WikiText

Figure 22: Orthant-based clustering, $p = 4$, Simplified TinyStories

Figure 23: Orthant-based clustering, $p = 5$, Simplified TinyStories

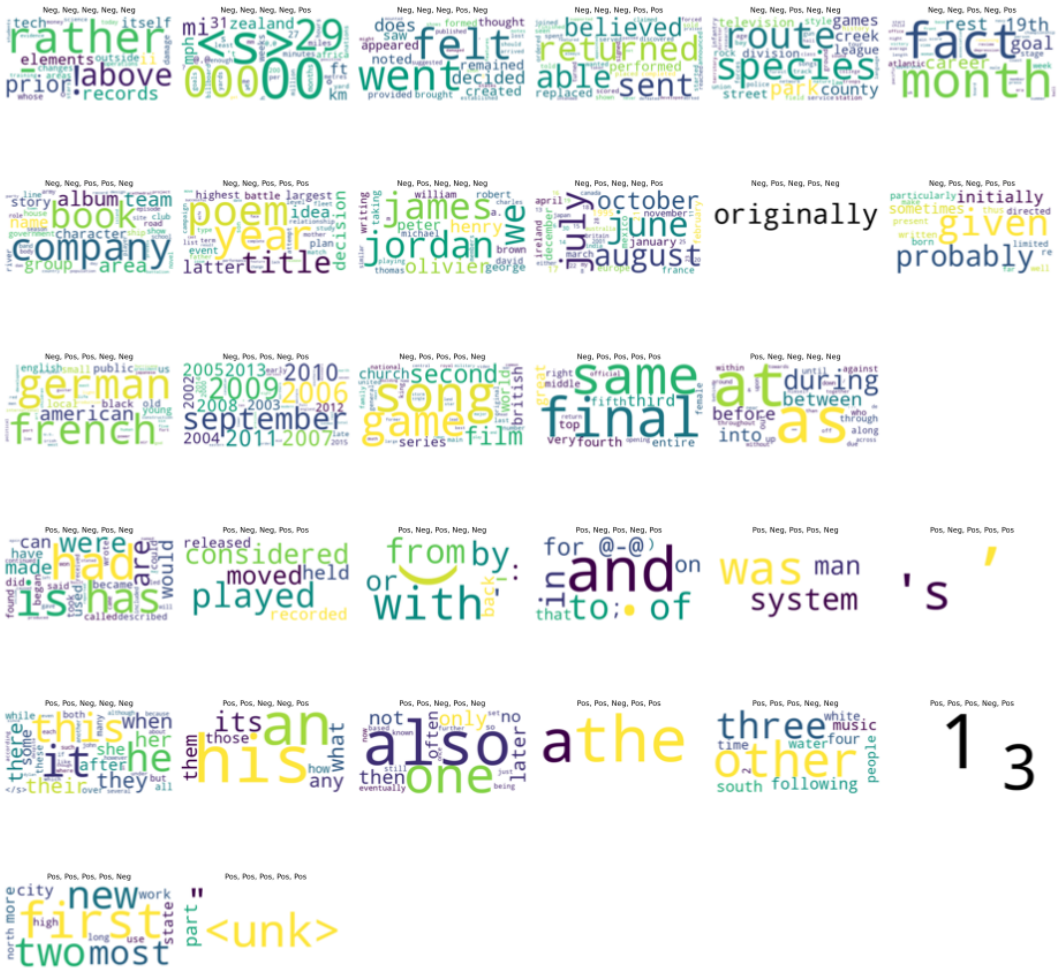

Figure 24: Orthant-based clustering, $p = 5$, Simplified Wiki-Text

Figure 25: Orthant-based clustering, $p = 6$, Simplified Wiki-Text