Closed-Form Training Dynamics Reveal Learned Features and Linear Structure in `Word2Vec`-like Models

Dhruva Karkada[*] UC Berkeley
James B. Simon Imbue and UC Berkeley
Yasaman Bahri Google DeepMind
Michael R. DeWeese UC Berkeley

Abstract

Self-supervised word embedding algorithms such as `word2vec` provide a minimal setting for studying representation learning in language modeling. We examine the quartic Taylor approximation of the `word2vec` loss around the origin, and we show that both the resulting training dynamics and the final performance on downstream tasks are empirically very similar to those of `word2vec`. Our main contribution is to analytically solve for both the gradient flow training dynamics and the final word embeddings in terms of only the corpus statistics and training hyperparameters. The solutions reveal that these models learn orthogonal linear subspaces one at a time, each one incrementing the effective rank of the embeddings until model capacity is saturated. Training on Wikipedia, we find that each of the top linear subspaces represents an interpretable topic-level concept. Finally, we apply our theory to describe how linear representations of more abstract semantic concepts emerge during training; these can be used to complete analogies via vector addition.

1 Introduction

Modern machine learning models achieve impressive performance on complex tasks in large part due to their ability to automatically extract useful features from data. Despite rapid strides in engineering, a scientific theory describing this process remains elusive. The challenges in developing such a theory include the complexity of model architectures, the nonconvexity of the optimization, and the difficulty of data characterization. To make progress, it is prudent to turn to simple models that admit theoretical analysis while still capturing phenomena of interest.

Word embedding algorithms are a class of self-supervised algorithms that learn word representations with task-relevant vector structure. One example is `word2vec`, a contrastive algorithm that learns to model the probability of finding two given words co-occurring in natural text using a two-layer linear network (Mikolov et al., 2013). Despite its simplicity, the resulting models succeed on a variety of semantic understanding tasks. One striking ability exhibited by these embeddings is analogy completion: most famously, $\mathbf{man} - \mathbf{woman} \approx \mathbf{king} - \mathbf{queen}$, where $\mathbf{man}$ is the embedding for the word "man" and so on. Importantly, this ability is not explicitly promoted by the optimization objective; instead, it emerges spontaneously from the process of representation learning.

Contributions. In this work, we give a closed-form description of representation learning in models trained to minimize the quartic Maclaurin approximation of the `word2vec` loss. We prove that the learning problem reduces to matrix factorization with quadratic loss (Theorem 1), so we call these models *quadratic word embedding models* (QWEMs). We derive analytic solutions for their training dynamics and final embeddings under gradient flow and vanishing initialization (Result 3). Training on Wikipedia, we show that QWEMs closely match `word2vec`-trained models in their training dynamics, learned features, and performance on standard benchmarks (Figure 3). We apply our results to show that the dynamical formation of abstract linear representations is well-described by quantities taken from random matrix theory (Figure 4). Taken together, our results give a clear picture of the learning dynamics in contrastive word embedding algorithms such as `word2vec`.

[*]dkarkada@berkeley.edu
Code to reproduce all experiments available at `https://github.com/dkarkada/qwem`.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

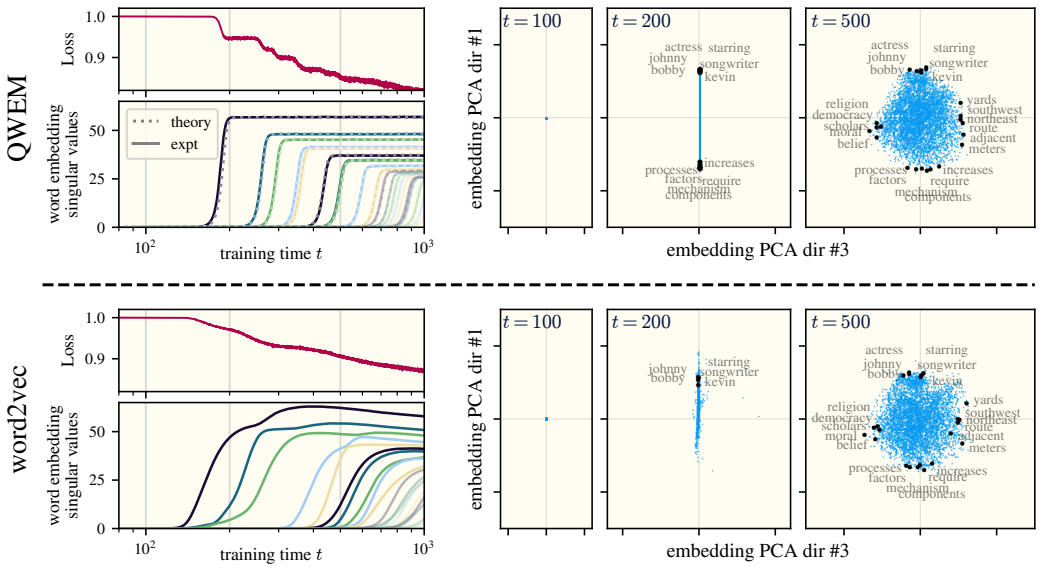

Figure 1: Quadratic word embedding models are a faithful and analytically solvable proxy for `word2vec`. We compare the time course of learning in QWEMs (top) and `word2vec` (bottom), finding striking similarities in their training dynamics and learned representations. Analytically, we solve for the optimization dynamics of QWEMs under gradient flow from small initialization, revealing discrete, rank-incrementing learning steps corresponding to stepwise decreases in the loss (top left). In latent space (right side plots), embedding vectors expand into subspaces of increasing dimension at each learning step. These PCA directions are the model's learned features, and they can be extracted from our theory in *closed form* given only the corpus statistics and hyperparameters (Theorem 1). Empirically, QWEMs yield high-quality embeddings very similar to `word2vec`'s in terms of their learned features and performance on benchmarks (Figure 3). See Appendix A for details.

Relation to previous work. `word2vec` is a self-supervised contrastive word embedding algorithm widely used for its simplicity and performance (Mikolov et al., 2013; Levy et al., 2015). Although the resulting models are known to implicitly factorize a target matrix (Levy & Goldberg, 2014), it is not known *which* low-rank approximation of the target is learned (Arora et al., 2016). We provide the answer in a close approximation of the task, solving for the final word embeddings directly in terms of the statistics of the data and the training hyperparameters.

Our result connects deeply with previous works on the gradient descent dynamics of linear models. For two-layer linear feedforward networks trained on a supervised learning task with square loss, whitened inputs, and weights initialized to be aligned with the target, the singular values of the weights undergo sigmoidal dynamics; each singular direction is learned independently with a distinct learning timescale (Saxe et al., 2014, 2019; Gidel et al., 2019; Atanasov et al., 2022). These results either rely on assumptions on the data (e.g., input covariance $\Sigma_{xx} = \boldsymbol{I}$ or Σ_{xx} commutes with Σ_{xy}) or are restricted to scalar outputs. Similarly, supervised matrix factorization models are known to exhibit rank-incremental training dynamics in some settings (Li et al., 2018; Arora et al., 2019; Gissin et al., 2019; Li et al., 2021; Jacot et al., 2021; Jiang et al., 2023; Chou et al., 2024). Many of these results rely on over-parameterization or special initialization schemes. While our result is consistent with these works, our derivation does not require assumptions on the data distribution, nor does it require special structure in the initial weights. Another key difference is that our result is the first to solve for the training dynamics of a natural language task learned by a *self-supervised* contrastive algorithm. This directly expands the scope of matrix factorization theory to new settings of interest.

Closest to our work, HaoChen et al. (2021); Tian et al. (2021); Simon et al. (2023b) study linearized contrastive learning in vision tasks. Our work differs in several ways: we study a natural language task using a different contrastive loss function, we do not linearize a nonlinear model architecture, we obtain closed-form solutions for the learning dynamics, and we do not require assumptions on the data distribution (e.g., special graph structure in the image augmentations, or isotropic image data). Saunshi et al. (2022) stress that a theory of contrastive learning must account for both the true data distribution and the optimization dynamics; to our knowledge, our result is the first to do so.

2 Preliminaries

Notation. We use capital boldface to denote matrices and lowercase boldface for vectors. Subscripts denote elements of vectors and tensors ($\boldsymbol{A}_{ij}$ is a scalar). We use the "economy-sized" singular value decomposition (SVD) $\boldsymbol{A} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^\top$, where $\boldsymbol{S}$ is square. We denote the rank-r truncated SVD as $\boldsymbol{A}_{[r]} = \boldsymbol{U}_{[:,:r]}\boldsymbol{S}_{[:r,:r]}\boldsymbol{V}_{[:,:r]}{}^\top$.

Setup. The training corpus is a long sequence of words drawn from a finite vocabulary of cardinality V. A *context* is any contiguous length-L subsequence of the corpus. Let i and j index the vocabulary. Let $\Pr(i)$ be the empirical unigram distribution, and let $\Pr(j|i)$ be the proportion of occurrences of word j in contexts containing word i. Define $\Pr(i,j) \coloneqq \Pr(j|i)\Pr(i)$ to be the *skip-gram distribution*. We use the shorthand $P_{ij} \coloneqq \Pr(i,j)$ and $P_i \coloneqq \Pr(i)$.

Let $\boldsymbol{W} \in \mathbb{R}^{V\times d}$ be a trainable weight matrix whose i^{th} row $\boldsymbol{w}_i$ is the d-dimensional embedding vector for word i. We restrict our focus to the underparameterized regime $d \ll V$, in accordance with practical settings. The goal is to imbue $\boldsymbol{W}$ with semantic structure so that the inner products between word embeddings capture semantic similarity. To do this, one often uses an iterative procedure that aligns frequently co-occurring words and repels unrelated words. The principle underlying this method is the *distributional hypothesis*, which posits that semantic structure in natural language can be discovered from the co-occurrence statistics of the words (Harris, 1954).

Primer on `word2vec`. In `word2vec` "skip-gram with negative sampling" (SGNS),[3] two embedding matrices $\boldsymbol{W}$ and $\boldsymbol{W}'$ are trained[4] using stochastic gradient descent to minimize the contrastive loss

$$\mathcal{L}_{\texttt{w2v}}(\boldsymbol{W},\boldsymbol{W}') = \mathop{\mathbb{E}}_{i,j\sim\Pr(\cdot,\cdot)}\left[\Psi^+_{ij}\log\left(1+e^{-\boldsymbol{w}_i^\top\boldsymbol{w}'_j}\right)\right] + \mathop{\mathbb{E}}_{\substack{i\sim\Pr(\cdot)\\ j\sim\Pr(\cdot)}}\left[\Psi^-_{ij}\log\left(1+e^{\boldsymbol{w}_i^\top\boldsymbol{w}'_j}\right)\right]. \quad (1)$$

The averages are estimated by drawing samples from the corpus. The nonnegative hyperparameters $\{\Psi^+_{ij}\}$ and $\{\Psi^-_{ij}\}$ are reweighting coefficients for word pairs; we use them here to capture the effect of several of `word2vec`'s implementation details, including subsampling (i.e., probabilistically discarding frequent words during iteration), dynamic window sizes, and different negative sampling distributions. All of these can be seen as preprocessing techniques that directly modify the unigram and skip-gram distributions. In Appendix A.3 we provide more detail about these engineering tricks and discuss how one can encode their effects in Ψ^+ and Ψ^-.

The quality of the resulting embeddings are evaluated using standard semantic understanding benchmarks. For example, the Google analogy test set measures how well the model can complete analogies (e.g., man:woman::king:?) via vector addition (Mikolov et al., 2013). Importantly, this benchmark is distinct from the optimization task, and performing well on it requires representation learning.

The global minimizer of $\mathcal{L}_{\texttt{w2v}}$ is the *pointwise mutual information* (PMI) matrix

$$\mathop{\arg\min}_{\boldsymbol{w}_i^\top\boldsymbol{w}'_j} \mathcal{L}_{\texttt{w2v}}(\boldsymbol{W},\boldsymbol{W}') = \log\left(\frac{\Psi^+_{ij}P_{ij}}{\Psi^-_{ij}P_iP_j}\right), \quad (2)$$

where the minimization is over the inner products (Levy & Goldberg, 2014). Crucially, the PMI minimizer can only be realized ($\boldsymbol{W}\boldsymbol{W}'^\top = \text{PMI}$) if there is no rank constraint ($d \geq \text{rank}(\text{PMI})$). This condition is always violated in practice. It is crucial, then, to determine *which* low-rank approximation of the PMI matrix is learned by `word2vec`. It is *not* the least-squares approximation; the resulting embeddings are known to perform significantly worse on downstream tasks such as analogy completion (Levy et al., 2015). This is because the divergence at $P_{ij}/P_iP_j \to 0$ causes least squares to over-allocate fitting power to these rarely co-occurring word pairs. Various alternatives have been proposed, including the *positive PMI*, $\text{PPMI}_{ij} = \max(0, \text{PMI}_{ij})$, but we find that these still differ from the embeddings learned by `word2vec`, both in character and in performance (Figure 3).

Our approach is different: rather than approximate the minimizer of $\mathcal{L}_{\texttt{w2v}}$, we obtain the *exact* minimizer of a (Taylor) approximation of $\mathcal{L}_{\texttt{w2v}}$. Though this may seem to be a coarser approximation, we are well-compensated by the ability to analytically treat the implicit bias of gradient descent, which enables us to give a full theory of *how* and *which* low-rank embeddings are learned.

[3] Throughout this paper, we use the abbreviated "`word2vec`" to refer to the `word2vec` SGNS algorithm.

[4] $\boldsymbol{W}$ is for "center" words, and $\boldsymbol{W}'$ is for "context" words. For simplicity, we consider the setting $\boldsymbol{W}' = \boldsymbol{W}$; in Figure 3 we show that this is sufficient for good performance on semantic understanding tasks.

3 Quadratic Word Embedding Models

We set $\boldsymbol{W}' = \boldsymbol{W}$ and study the quartic approximation of $\mathcal{L}_{\texttt{w2v}}$ around the origin:

$$\mathcal{L}(\boldsymbol{W}) := \mathop{\mathbb{E}}_{i,j\sim\Pr(\cdot,\cdot)}\left[\Psi^+_{ij}\left(\frac{(\boldsymbol{w}_i^\top \boldsymbol{w}_j)^2}{4} - \boldsymbol{w}_i^\top \boldsymbol{w}_j\right)\right] + \mathop{\mathbb{E}}_{\substack{i\sim\Pr(\cdot)\\ j\sim\Pr(\cdot)}}\left[\Psi^-_{ij}\left(\frac{(\boldsymbol{w}_i^\top \boldsymbol{w}_j)^2}{4} + \boldsymbol{w}_i^\top \boldsymbol{w}_j\right)\right]. \quad (3)$$

Note that the key quantities are the inner products between embeddings. There is no privileged coordinate basis in embedding space. Since the objective is quadratic in these inner products, we refer to the resulting models *quadratic word embedding models* (QWEMs).[5]

Model equivalence classes. Since $\mathcal{L}(\boldsymbol{W})$ is invariant under orthogonal transformations of the right singular vectors of $\boldsymbol{W}$, we define the right orthogonal equivalence class

$$\mathrm{REquiv}(\boldsymbol{W}) := \left\{\boldsymbol{W}\boldsymbol{U} \;\middle|\; \boldsymbol{U} \in \mathbb{R}^{d\times d}, \boldsymbol{U}^\top\boldsymbol{U} = \boldsymbol{I}\right\}. \quad (4)$$

3.1 QWEM is equivalent to matrix factorization with square loss.

Target matrix. We start by introducing a matrix $\boldsymbol{M}^\star$. We will show in Theorem 1 that $\boldsymbol{M}^\star$ is the optimization target for QWEM, just as the PMI matrix is the optimization target for `word2vec`.

$$\boldsymbol{M}^\star_{ij} := \frac{\Psi^+_{ij}P_{ij} - \Psi^-_{ij}P_iP_j}{\frac{1}{2}(\Psi^+_{ij}P_{ij} + \Psi^-_{ij}P_iP_j)}. \quad (5)$$

To understand this quantity, first note that if language were a stochastic process with independently sampled words, the co-occurrence statistics would be structureless, i.e., $P_{ij} - P_iP_j = 0$. The distributional hypothesis then suggests that algorithms may learn semantics by modeling the statistical deviations from independence. It is exactly these (relative) deviations that comprise the optimization target $\boldsymbol{M}^\star$ and serve as the central statistics of interest in our theory. Furthermore, $\boldsymbol{M}^\star$ can be seen as an approximation the PMI matrix; see Appendix D.

Reweighting hyperparameters. Our goal now is to directly convert Equation (3) into a matrix factorization problem. To do this, we make some judicious choices for the hyperparameters. We first define the quantity $\boldsymbol{G}_{ij} := \Psi^+_{ij}P_{ij} + \Psi^-_{ij}P_iP_j$, which captures the aggregate effect of the reweighting hyperparameters on the optimization. Then we establish the following hyperparameter setting.

Setting 3.1 (Symmetric Ψ^+, Ψ^- and constant $\boldsymbol{G}_{ij}$)**.** Let $\Psi^+_{ij} = \Psi^+_{ji}$ and $\Psi^-_{ij} = \Psi^-_{ji}$ so that, by symmetry, the eigendecomposition $\boldsymbol{M}^\star = \boldsymbol{V}^\star\boldsymbol{\Lambda}\boldsymbol{V}^{\star\top}$ exists. Let $\boldsymbol{G}_{ij} = g$ for some constant g.

Note that infinitely many choices of Ψ^+ and Ψ^- are encompassed by this setting. Let us study a concrete example: $\Psi^+_{ij} = \Psi^-_{ij} = (P_{ij} + P_iP_j)^{-1}$, so that $\boldsymbol{G}_{ij} = g = 1$. This has the effect of down-weighting frequently appearing words and word pairs, which hastens optimization and prevents the model from over-allocating fitting power to words such as "the" or "and" which may not individually carry much semantic content. This is exactly the justification given for subsampling in `word2vec`, which motivates Setting 3.1. Indeed, in Appendix A.3 we discuss how these choices of Ψ^+ and Ψ^- can be seen as approximating several of the implementation details and engineering tricks in `word2vec`. In Figures 2 and 3, we show that this simplified hyperparameter setting does not wash out the relevant structure, thus retaining realism.

With these definitions, we state our key result: rank-constrained quadratic word embedding models trained under Setting 3.1 learn the top d eigendirections of $\boldsymbol{M}^\star$.

Theorem 1 (QWEM = unweighted matrix factorization)**.** *Under Setting 3.1, the contrastive loss Equation (3) can be rewritten as the unweighted matrix factorization problem*

$$\mathcal{L}(\boldsymbol{W}) = \frac{g}{4}\left\|\boldsymbol{W}\boldsymbol{W}^\top - \boldsymbol{M}^\star\right\|_{\mathrm{F}}^2 + \mathrm{const}. \quad (6)$$

If $\boldsymbol{\Lambda}_{[:d,:d]}$ is positive semidefinite, then the set of global minima of $\mathcal{L}$ is given by

$$\arg\min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}) = \mathrm{REquiv}\left(\boldsymbol{V}^\star_{[:,:d]}\boldsymbol{\Lambda}^{1/2}_{[:d,:d]}\right). \quad (7)$$

Proof. Equation (6) follows from completing the square in Equation (3). Equation (7) follows from the Eckart-Young-Mirsky theorem. In Appendix A.2, we note that the PSD assumption is easily satisfied in practice.

[5]We use "QWEM" as shorthand for minimizing the quartic approximation of the `word2vec` loss, and "QWEMs" for the resulting embeddings. We emphasize that QWEMs do not refer to a new model architecture.

We emphasize that although previous results prove that word embedding models find low-rank approximations to some target matrix (e.g., PMI for `word2vec`), previous results do not establish *which* low-rank factorization is learned. To our knowledge, our result is the first to solve for the rank-constrained minimizer of a practical word embedding task. Furthermore, our solution is given in terms of only the corpus statistics and the hyperparameters Ψ^+ and Ψ^-. In particular, we show that QWEMs learn compressed representations of the relative deviations between the true co-occurrence statistics and the baseline of independently distributed words. Unlike previous work, we do not require stringent assumptions on the data distribution (e.g., spherically symmetric latent vectors, etc.).

The main limitation of Theorem 1 is its restriction to Setting 3.1. What happens in the general case?

Proposition 2. *Equation (3) can be rewritten as the weighted matrix factorization problem*

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{4}\sum_{i,j} \boldsymbol{G}_{ij}(\boldsymbol{W}\boldsymbol{W}^\top - \boldsymbol{M}^\star)_{ij}^2 + \text{const.} \tag{8}$$

Let Ψ^+ and Ψ^- each be symmetric in i, j, let $\boldsymbol{g} \in \mathbb{R}^V$ be an arbitrary vector with non-negative elements, and let $\boldsymbol{\Gamma} = \text{diag}(\boldsymbol{g})^{1/2}$. Then the eigendecomposition of $\boldsymbol{\Gamma}\boldsymbol{M}^\star\boldsymbol{\Gamma} = \boldsymbol{V}_\Gamma^\star \boldsymbol{\Lambda}_\Gamma \boldsymbol{V}_\Gamma^{\star\top}$ exists. If $\boldsymbol{G} = \boldsymbol{g}\boldsymbol{g}^\top$ and $\boldsymbol{\Lambda}_{[:d,:d]}$ is positive semidefinite, then the set of global minima of $\mathcal{L}$ is given by

$$\arg\min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}) = \text{REquiv}\left(\boldsymbol{\Gamma}^{-1}\boldsymbol{V}^\star_{\Gamma,[:,:d]}\boldsymbol{\Lambda}^{1/2}_{\Gamma,[:d,:d]}\right). \tag{9}$$

Proof. See Appendix C. Equation (8) says that in the general case, the form of the target matrix remains unchanged. Equation (9) says that if the hyperparameters are chosen so that $\boldsymbol{G}$ is rank-1, then we can characterize the minimizer. However, if $\boldsymbol{G}$ is not rank-1, then we do not know what low-rank factorization is learned, nor can we describe the training dynamics; indeed, weighted matrix factorization is known to be NP-hard (Gillis & Glineur, 2011). For this reason, we do not revisit these more general settings, and we hereafter restrict our focus to Setting 3.1. This is ultimately justified by the empirically close match between our theory and the embeddings learned by `word2vec` (Figure 3).

3.2 Training dynamics of QWEMs reveal implicit bias towards low rank.

Note that despite the simplification afforded by Theorem 1, the minimization problem Equation (6) is still nonconvex since $\mathcal{L}(\boldsymbol{W})$ is quartic in $\boldsymbol{W}$. Thus, there remain questions regarding convergence time and the effect of early stopping regularization. To study them, we examine the training trajectories induced by gradient flow. The central variables of our theory will be the economy-sized SVD of the embeddings, $\boldsymbol{W}(t)^\top = \boldsymbol{U}(t)\boldsymbol{S}(t)\boldsymbol{V}(t)^\top$, and the eigendecomposition of the target, $\boldsymbol{M}^\star = \boldsymbol{V}^\star\boldsymbol{\Lambda}\boldsymbol{V}^{\star\top}$. For convenience, we define $\lambda_k \coloneqq \boldsymbol{\Lambda}_{kk}$ and $s_k \coloneqq \boldsymbol{S}_{kk}$.

In general, it is difficult to solve the gradient descent dynamics of Equation (3) from arbitrary initialization. We first consider a simple toy case in which the initial embeddings are already aligned with the top d eigenvectors of $\boldsymbol{M}^\star$.

Lemma 3.1 (Training dynamics, aligned initialization)**.** *If $\tilde{\boldsymbol{\Lambda}} \coloneqq \boldsymbol{\Lambda}_{[:d,:d]}$ is positive semidefinite and $\boldsymbol{V}(0) = \boldsymbol{V}^\star_{[:,:d]}$, then under Setting 3.1, the gradient flow dynamics $\frac{\mathrm{d}\boldsymbol{W}}{\mathrm{d}t} = -\frac{1}{2g}\nabla\mathcal{L}(\boldsymbol{W})$ yields*

$$\boldsymbol{U}(t) = \boldsymbol{U}(0) \tag{10}$$

$$\boldsymbol{S}(t) = \boldsymbol{S}(0)\left(e^{-\tilde{\boldsymbol{\Lambda}}t} + \boldsymbol{S}^2(0)\tilde{\boldsymbol{\Lambda}}^{-1}\left(\boldsymbol{I} - e^{-\tilde{\boldsymbol{\Lambda}}t}\right)\right)^{-1/2} \tag{11}$$

$$\boldsymbol{V}(t) = \boldsymbol{V}(0) = \boldsymbol{V}^\star_{[:,:d]} \tag{12}$$

The proof is given in Appendix C. See Saxe et al. (2014) for a proof in an equivalent learning problem.

This result states that the final embeddings' PCA directions are given by the top d eigenvectors of $\boldsymbol{M}^\star$, that the dynamics are decoupled in this basis, and that the population variance of the embeddings along the k^{th} principal direction increases sigmoidally from $s_k^2(0)$ to λ_k in a characteristic time $t = \tau_k \coloneqq (1/\lambda_k)\ln\big(\lambda_k/s_k^2(0)\big)$. These training dynamics have been discovered in a variety of other learning problems (Saxe et al., 2014; Gidel et al., 2019; Atanasov et al., 2022; Simon et al., 2023b; Dominé et al., 2023). Our result adds self-supervised quadratic word embedding models to the list.

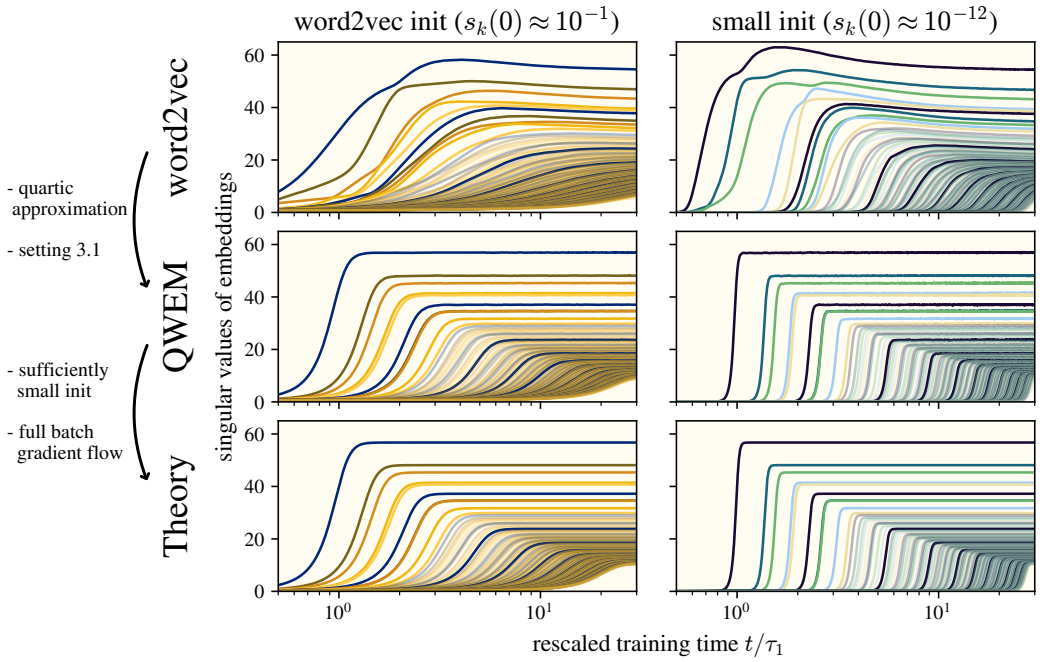

Figure 2: Theory matches experiment. We make two simplifications to the `word2vec` algorithm: a quartic approximation of the loss, and a restriction on the reweighting hyperparameters. We train these QWEMs on 2 billion tokens of English Wikipedia (see Appendix A for details) and compare to `word2vec`. We find good qualitative match in the singular value dynamics, both with the standard `word2vec` initialization scheme and with small random initialization. (For evidence that the singular vectors match as well, see Figure 3.) We compare the dynamics to the prediction of Result 3, which is derived in the vanishing initialization limit with full-batch gradient flow. Even though the experiment uses stochastic mini-batching, non-vanishing learning rate, and large initialization, we find excellent agreement even up to constant factors.

It is restrictive to require perfect alignment $\boldsymbol{V} = \boldsymbol{V}^\star$ at initialization. To lift this assumption, we show that in the limit of vanishing random initialization, the dynamics are indistinguishable from the aligned case, up to orthogonal transformations of the right singular vectors of $\boldsymbol{W}$. To our knowledge, previous works have not derived this equivalence for under-parameterized matrix factorization.

Result 3 (Training dynamics, vanishing random initialization)**.** *Initialize $\boldsymbol{W}_{ij}(0) \sim \mathcal{N}(0, \sigma^2)$, and let $\boldsymbol{S}(0)$ denote the singular values of $\boldsymbol{W}(0)$. Define the characteristic time $\tau_1 := \lambda_1^{-1} \ln(\lambda_1/\sigma^2)$ and the rescaled time variable $\tilde{t} = t/\tau_1$. Define $\boldsymbol{W}^\star(0) := \boldsymbol{V}^\star_{[:,:d]} \boldsymbol{S}(0)$ and let $\boldsymbol{W}^\star(t)$ be its gradient flow trajectory given by Lemma 3.1. If $\boldsymbol{\Lambda}_{[:d,:d]}$ is positive semidefinite, then under Setting 3.1 we have with high probability that*

$$\forall \tilde{t} \geq 0, \quad \lim_{\sigma^2 \to 0} \min_{\boldsymbol{W}' \in \mathrm{REquiv}(\boldsymbol{W}(\tilde{t}\tau_1))} \|\boldsymbol{W}' - \boldsymbol{W}^\star(\tilde{t}\tau_1)\|_{\mathrm{F}} = 0. \tag{13}$$

Derivation. See Appendix C. The main idea is to study the dynamics of the $\boldsymbol{QR}$ factorization of $\boldsymbol{W}^\top \boldsymbol{V}^\star$. We write the equation of motion, discard terms that become small in the limit, solve the resulting equation, and show that the discarded terms remain small. We conjecture that our argument can be made rigorous by appropriately bounding the discarded terms. We leave this to future work.

Result 3 generalizes previous work that establishes a *silent alignment* phenomenon for linear networks with scalar outputs (Atanasov et al., 2022). Here, for all $k \leq d$, $\boldsymbol{V}_{[:,:k]}$ quickly aligns with $\boldsymbol{V}^\star_{[:,:k]}$ while s_k remains near initialization; thus the alignment assumption is quickly near-satisfied and Lemma 3.1 becomes applicable. We conclude that quadratic word embedding models trained from small random initialization are inherently greedy spectral methods. The principal components of the embeddings learn a one-to-one correspondence with the eigenvectors of $\boldsymbol{M}^\star$, and each component is realized independently and sequentially. Thus, early stopping acts as an implicit regularizer, constraining model capacity in terms of rank rather than weight norm.

`word2vec` feature neighbors	QWEM feature neighbors	svd(PPMI) feature neighbors
(PC1) jones scott gary frank robinson terry michael david eric kelly	**(PC1)** tom jones david frank michael scott robinson kelly tony	**(PC1)** eric cooper jones dennis oliver sam tom robinson
(PC2) government council national established in state united republic	**(PC2)** government council establishment appointed republic union	**(PC2)** jones dennis eric robinson scott michael oliver taylor
(PC3) adjacent located built surface powered bay meters road near	**(PC3)** bay adjacent located north junction southwest road northeast	**(PC3)** government establishment foreign authorities leaders
(PC11) combat enemy offensive deployed naval artillery narrative	**(PC10)** offensive combat war artillery enemy naval defensive	**(PC11)** deployed force combat patrol command naval squadron
(PC12) whilst trained skills competitions studying teaching artistic	**(PC11)** trained skills whilst studying competitions aged honours	**(PC14)** piano vocal orchestra solo music instrumental recordings
(PC13) britain produced anglo ltd welsh australian scottish sold	**(PC12)** britain produced considerable price notably industry sold	**(PC15)** england thus great price meant liverpool share earl enjoyed
(PC100) il northern di worker laid contributions ireland down oak	**(PC100)** doctor bar lives disc oregon credited ultimate split serial	**(PC100)** org figure standing riding with http green date www parent

(a) Words with smallest cosine distance to embedding principal components

	`word2vec`	QWEM	svd($\boldsymbol{M}^\star$)	svd(PPMI)	svd(PMI)
Google Analogies	68.0%	65.1%	66.3%	50.6%	8.4%
MEN test set	0.744	0.755	0.755	0.740	0.448
WordSim353	0.698	0.682	0.683	0.690	0.221

(b) Performance on standard word embedding benchmarks

Figure 3: QWEMs and `word2vec` learn similar features, whereas PMI (and variants) differ qualitatively. For the following, all models $\boldsymbol{W} \in \mathbb{R}^{V \times d}$ are trained with $V = 10,000$ and $d = 200$ on 2 billion tokens of Wikipedia. See Appendix A for experimental details. We denote $\mathrm{svd}(\boldsymbol{M}) \coloneqq \arg\min_{\boldsymbol{W}} \|\boldsymbol{W}\boldsymbol{W}^\top - \boldsymbol{M}\|_{\mathrm{F}}^2$. **(Top.)** We compute the principal components of the final embeddings and list the closest embeddings. See Appendix B for a quantitative plot of subspace overlaps. Top section: top three components of `word2vec` and QWEM represent topic-level concepts (biography, government, geography) corresponding to common topics on Wikipedia. Middle section: components closest to `word2vec` components 11, 12, 13. Up to reordering, QWEMs match closely and remain interpretable, while positive PMI deviates. Bottom row: late components lose their interpretability. **(Bottom.)** Since QWEMs and `word2vec` learn similar features, they perform similarly on the vector addition analogy completion task. Explicit factorization of $\boldsymbol{M}^\star$ almost matches the performance of `word2vec`, much better than the best previously-known SVD embeddings.

3.3 Empirically, QWEMs are a good proxy for `word2vec`.

Our rationale for studying QWEMs is to gain analytical tractability (e.g., Result 3) in a setting that is "close to" the true setting of interest: `word2vec`. In Figure 3, we check whether QWEMs are in fact a faithful representative for `word2vec`. We find that not only do QWEMs nearly match `word2vec` on the analogy completion task, they also learn very similar representations, as measured by the alignment between their singular vectors. Importantly, QWEMs are closer to `word2vec` than the least-squares approximations of either PMI or PPMI. This underscores the importance of accounting for the implicit bias of gradient descent.

To run the experiments in Figures 1 to 3, we implemented a GPU-enabled training algorithm for both `word2vec` and QWEM. The corpus data is streamed from the hard drive in chunks to avoid memory overhead and excessive disk I/O, and the loops for batching positive and negative word pairs are compiled for efficiency. Notably, it is often faster to construct and explicitly factorize $\boldsymbol{M}^\star$ (e.g., with vocabulary size $V = 10,000$, it takes $\sim$10 minutes in total on a single GeForce GTX 1660 GPU).

Figure 3 suggests that if one is interested in understanding the learning behaviors of `word2vec`, it suffices to study QWEMs. Then Theorem 1 and Result 3 state that if one is interested in understanding the learning behaviors of QWEMs, it suffices to study $\boldsymbol{M}^\star$. In this spirit, we will now study $\boldsymbol{M}^\star$ to investigate certain aspects of representation learning in word embedding models.

4 Linear Structure in Latent Space

Result 3 reveals that, for QWEMs, the "natural" basis of the learning dynamics is simply the eigenbasis of $\boldsymbol{M}^\star$. Figure 3 suggests that this basis already encodes concepts interpretable to humans. These are the fundamental features learned by the model: each word embedding is naturally expressed as a linear combination of these orthogonal latents. Both early stopping and small embedding dimension regularize the model by constraining the number (but not the character) of available latents.

We may conclude that natural language contains linear semantic structure in its co-occurrence statistics, and that it is easily extracted by word embedding algorithms. It is not unreasonable to expect, then, that other semantic concepts may be encoded as linear subspaces. This idea is the *linear representation hypothesis*, and it motivates modern research in more sophisticated language models, including representation learning (Park et al., 2023; Jiang et al., 2024; Wang et al., 2024), mechanistic interpretability (Li et al., 2023; Nanda et al., 2023; Lee et al., 2024), and LLM alignment (Lauscher et al., 2020; Zou et al., 2023; Li et al., 2024). To make these efforts more precise, it is important to develop a quantitative understanding of these linear representations in simple models. Our Theorem 1 provides a new lens for this analysis: we may gain insight by studying the properties of $\boldsymbol{M}^\star$.

In word embedding models, a natural category of abstract linear representations are the difference vectors between semantic pairs, e.g., $\{\boldsymbol{r}^{(n)}\}_n = \{\mathbf{man} - \mathbf{woman}, \mathbf{king} - \mathbf{queen}, \dots\}$ for gender binaries. If the $\boldsymbol{r}^{(n)}$ are all approximately equal, then the model can effectively complete analogies via vector addition (see Appendix A.4). Following Ilharco et al. (2022), we call the $\boldsymbol{r}^{(n)}$ *task vectors*.

Here, we provide empirical evidence for the following dynamical picture of learning. The model internally builds task vectors in a sequence of noisy learning steps, and the geometry of the task vectors is well-described by a spiked random matrix model. Early in training, semantic signal dominates; however, later in training, noise may begin to dominate, causing a degradation of the model's ability to resolve the task vector. We validate this picture by studying the task vectors in a standard analogy completion benchmark.

We emphasize that, due to our Result 3, any result comparing the final embeddings of many models of different sizes d is *fully equivalent* to a result considering the time course of learning for a single (sufficiently large) model. Therefore, although we vary the model dimension in our plots, these can be viewed as results concerning the dynamics of learning in word embedding models.

Task vectors are often concentrated on a few dominant eigen-features. Task vectors derived from the analogy dataset are neither strongly aligned with a single model eigen-feature, nor are they random vectors. Instead, they exhibit *localization*: a handful of the top eigen-features are primarily responsible for the task vector. In some cases, these dominant eigen-features are interpretable and clearly correlate with the abstract semantic concept associated with the task vector (Figure 4).

Task vectors are well-described by a spiked random matrix model. To study the geometry of the task vectors within a class of semantic binaries, we consider stacking task vectors to produce the matrix $\boldsymbol{R}_d \in \mathbb{R}^{N\times d}$, where N is the number of word pairs and d is the embedding dimension. We note that $\boldsymbol{R}_d$ can be computed in closed form using our Theorem 1. If all the task vectors align (as desired for analogy completion), then $\boldsymbol{R}_d$ will be a rank-1 matrix; if the task vectors are all unrelated, then $\boldsymbol{R}_d$ will have a broad spectrum. This observation suggests that a useful object to study is the empirical spectrum of the Gram matrix $\boldsymbol{G}_d \coloneqq \boldsymbol{R}_d\boldsymbol{R}_d^\top \in \mathbb{R}^{N\times N}$.

We find that this spectrum is very well-described by the *spiked covariance model*, a well-known distribution of random matrices. In the model, one studies the spectrum of $\boldsymbol{Z} = \boldsymbol{\Xi}\boldsymbol{\Xi}^\top + \mu\boldsymbol{a}\boldsymbol{a}^\top$, where $\boldsymbol{\Xi}$ is a random matrix with i.i.d. mean-zero entries with variance σ^2 and $\mu\boldsymbol{a}\boldsymbol{a}^\top$ is a deterministic rank-1 perturbation with strength μ. If μ is sufficiently large compared to σ, then in the asymptotic regime the spectrum of $\boldsymbol{Z}$ is known to approach the Marchenko-Pastur distribution with a single outlier eigenvalue (Baik et al., 2005). We consistently observe this structure in the real empirical data, across both model dimension and semantic families (Figure 4). We note that $N \approx 30$ is fairly small, and so it is somewhat surprising that results from the asymptotic regime visually appear to hold.

We interpret this observation as evidence that task vectors are well-described as being random vectors with a strong deterministic signal (e.g., Gaussian random vectors with nonzero mean). One expects that in a high-quality linear representation, the signal is simply the mean task vector, and that it overwhelms the random components. To understand whether a model can effectively utilize the task vectors, then, we examine the relative strength of the spike compared to the noisy bulk.

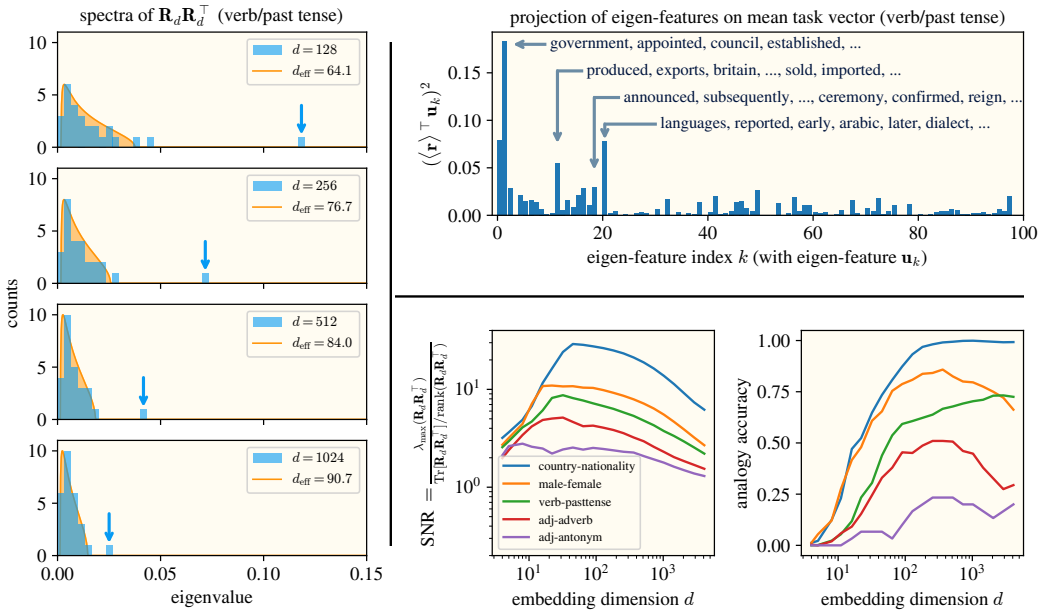

Figure 4: Models build linear representations from a few informative and many noisy eigen-features. In the left and upper plots, we examine task vectors between verb past tenses and their participle (e.g., $\mathbf{went} - \mathbf{going}$). In Appendix B we show that these observations hold for other semantic binaries. **(Left.)** The spectrum of the Gram matrix (histogram) is well-described by a Marchenko-Pastur distribution (orange) plus an outlier "spike," across model sizes d. See Appendix A.6 for details. **(Top.)** The spike corresponds to the average task vector, which comprises a few dominant eigen-features. Many of these features correspond to concepts related to history or temporal change, consistent with this semantic category. **(Bottom.)** We measure the strength of the spike across model size d for various semantic categories. We find that the spike strength correlates strongly with the model's ability to use the task vectors for analogy completion.

The strength of the spike perturbation corresponds to the robustness of the task vector. We measure the relative strength of the signal-containing spike using an empirical measure of the signal-to-noise ratio: $\text{snr} := \lambda_{\max}(\boldsymbol{G}) \cdot \text{rank}(\boldsymbol{G}) / \text{Tr}[\boldsymbol{G}]$. This quantity is simply the ratio between the maximum eigenvalue (i.e., the signal strength) and the mean nonzero eigenvalue (i.e., the typical variation due to noise). We find that as the model learns representations of increasing rank during learning, it first captures an increasing fraction of the signal (consistent with the previous observation regarding the localization of the task vector). Later in training, the noise begins to dominate, and the model's ability to resolve the signal degrades. Finally, the maximum achieved SNR serves as a coarse predictor for how effectively the model can use the representation for downstream tasks: the model achieves higher analogy completion scores on semantic directions with higher SNR.

Together, these observations provide evidence that useful linear representations arise primarily from the model's ability to resolve signal-containing eigen-features without capturing excess noise from extraneous eigen-features. Furthermore, our results suggest that tools from random matrix theory may be fruitfully applied to understand and characterize this interplay. We leave this to future work.

5 FAQ

Can the assumptions be relaxed? Our theoretical results require only four assumptions: quartic approximation of the loss, our Setting 3.1, small initial weights, and population gradient flow. The latter two are technical conveniences that simplify the analysis; Figure 2 suggests that they may be relaxed (or possibly eliminated) with additional effort. The first two are genuine approximations of the `word2vec` algorithm; their validity is supported by our empirical checks (see Figures 1, 3 and 5 and Appendix A.3). To further understand why these two approximations are technically useful, and what happens if they are relaxed, we perform empirical ablation tests in Figure 7.[6]

[6]We note that these auxiliary experiments use a different training corpus, model size, and hyperparameter choices, indicating that our theory is not sensitive to these choices.

The clear stepwise learning and decoupled dynamics in our theory originate from Setting 3.1. This condition turns the weighted factorization problem into an unweighted factorization. As a consequence, the singular value/vector dynamics quickly become "untangled" in the early stage of training; see Proposition 2 and the subsequent discussion. Somewhat surprisingly, this simplification is "close enough" to `word2vec`. Though the `word2vec` singular vectors do exhibit mild "mixing" in time (Figure 2), the overall learning dynamics are well-described by the simplified setting. This agreement is partly because the recommended `word2vec` hyperparameters approximate satisfy Setting 3.1 (see Appendix A.3). Our results suggest that future efforts to understand more complex learning systems may find purchase in identifying useful approximations of this kind.

Why do QWEM factorizations describe `word2vec` better than PMI factorizations? The crux of the issue lies in adequately handling the rank constraint. The idea behind factorizing PMI via SVD is to first solve the unconstrained minimizer of the loss (i.e., the PMI matrix) and then, at the end, apply the rank constraint by choosing the closest (in Frobenius norm) rank-d matrix. This does not work well because the loss basin is extremely wide and shallow, so the global unconstrained minimum is actually very far from where the model actually ends up via gradient descent in finite time. We take a different approach: by first approximating the loss landscape itself, we can account for the rank constraint throughout the entire optimization trajectory. As a result, our prediction for what `word2vec` learns is significantly more accurate.

Why does decreasing SNR sometimes still yield high performance in Figure 4? This is an ancillary effect of using top-1 accuracy as the performance metric. As a concrete example, consider the analogy "France : Paris :: Japan : Tokyo," satisfying $\texttt{japan} + (\texttt{paris} - \texttt{france}) \approx \texttt{tokyo}$. As the effective embedding dimension increases, the SNR may decrease; at the same time, the typical separation between embeddings increases (due to the increased available volume in latent space). This means that there are fewer nearby "competitors" for `tokyo`. In the large d regime, we find that the embeddings are sufficiently spaced, and `tokyo` is still the nearest embedding to $\texttt{japan} + (\texttt{paris} - \texttt{france})$ despite an increase in absolute error (see Figure 8). Thus, top-1 performance does not degrade with decreasing SNR. We note that unintuitive effects associated with using top-k accuracy have been observed in LLMs as well (Schaeffer et al., 2024).

How might one apply these results to other self-supervised learning tasks? Since our results are distribution agnostic, Theorem 1 can be extended to establish an *explicit equivalence* between self-supervised contrastive learning and supervised matrix factorization. In particular, if the contrastive objective has the functional form $\mathcal{L}(\boldsymbol{W}) = \mathbb{E}_{\text{positive pairs}}[f^{+}(\boldsymbol{w}^T\boldsymbol{w}')] + \mathbb{E}_{\text{negative pairs}}[f^{-}(\boldsymbol{w}^T\boldsymbol{w}')]$ for some differentiable $f^{+}(\cdot)$ and $f^{-}(\cdot)$, then one can simply take the quadratic Taylor approximation, complete the square, and obtain a target matrix in terms of the positive and negative distributions of the learning problem. See Appendix D.2 for a demonstration of this idea for the SimCLR loss. To obtain closed-form learning dynamics, one needs to find an equivalent of our Setting 3.1. This may be challenging depending on the form of the input data; for instance, when the inputs are not one-hot, it may require a whitened data covariance.

What do these results tell us about feature learning? Two operational notions of *feature learning* are 1) optimization trajectories must escape the local vicinity of the (typically small) initialization (Yang & Hu, 2021; Jacot et al., 2021; Zhu et al., 2022; Atanasov et al., 2024; Kunin et al., 2025), and 2) learned weight matrices must project data onto target-relevant subspaces (Damian et al., 2022; Radhakrishnan et al., 2024). Our message complements this line of research by offering a practical yet solvable setting in which both behaviors appear: `word2vec` escapes its near-isotropic initialization region to learn a dense compression of the relative excess co-occurrence probability $\boldsymbol{M}^{\star}$, thereby aligning the embedding geometry with the most salient semantic linear subspaces. Our result is thus a step forward in the broader scientific project of obtaining quantitative, predictive descriptions of learning in practical algorithms.

Limitations. Our results are limited to the symmetric setup (tied encoder/decoder weights, i.e., $\boldsymbol{W} = \boldsymbol{W}'$). We did not train and evaluate at scales that are considered state-of-the-art, nor did we compare against other embedding models such as GloVe (Pennington et al., 2014).

Author contributions. DK developed the analytical results, ran all experiments, and wrote the manuscript with input from all authors. JS proposed the initial line of investigation and provided insight at key points in the analysis. YB and MD helped shape research objectives and gave oversight and feedback throughout the project's execution.

Acknowledgements. DK is grateful to Daniel Kunin and Carl Allen for early conversations about learning linear analogy structure from small initialization. DK thanks Google DeepMind and BAIR for funding support and compute access. MRD thanks the U.S. Army Research Laboratory and the U.S. Army Research Office for supporting this work under Contract No. W911NF-20-1-0151.

References

Arora, S., Li, Y., Liang, Y., Ma, T., and Risteski, A. A latent variable model approach to pmi-based word embeddings. *Transactions of the Association for Computational Linguistics*, 4:385–399, 2016. Cited on page 2.

Arora, S., Cohen, N., Hu, W., and Luo, Y. Implicit regularization in deep matrix factorization. *Advances in Neural Information Processing Systems*, 32, 2019. Cited on page 2.

Atanasov, A., Bordelon, B., and Pehlevan, C. Neural networks as kernel learners: The silent alignment effect. In *International Conference on Learning Representations*, 2022. Cited on pages 2, 5, and 6.

Atanasov, A., Meterez, A., Simon, J. B., and Pehlevan, C. The optimization landscape of sgd across the feature learning strength. *arXiv preprint arXiv:2410.04642*, 2024. Cited on page 10.

Baik, J., Ben Arous, G., and Péché, S. Phase transition of the largest eigenvalue for nonnull complex sample covariance matrices. *Annals of Probability*, 2005. Cited on page 8.

Baldi, P. and Hornik, K. Neural networks and principal component analysis: Learning from examples without local minima. *Neural networks*, 2(1):53–58, 1989. Cited on page 34.

Bordelon, B., Canatar, A., and Pehlevan, C. Spectrum dependent learning curves in kernel regression and wide neural networks. In *International Conference on Machine Learning*, pp. 1024–1034. PMLR, 2020. Cited on page 34.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., and Zhang, Q. JAX: composable transformations of Python+NumPy programs, 2018. URL `http://github.com/jax-ml/jax`. Cited on page 23.

Bruni, E., Tran, N.-K., and Baroni, M. Multimodal distributional semantics. *Journal of artificial intelligence research*, 49:1–47, 2014. Cited on page 24.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020. Cited on page 33.

Chizat, L., Oyallon, E., and Bach, F. On lazy training in differentiable programming. *Advances in neural information processing systems*, 32, 2019. Cited on page 34.

Chou, H.-H., Gieshoff, C., Maly, J., and Rauhut, H. Gradient descent for deep matrix factorization: Dynamics and implicit bias towards low rank. *Applied and Computational Harmonic Analysis*, 68: 101595, 2024. Cited on page 2.

Church, K. and Hanks, P. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29, 1990. Cited on page 33.

Damian, A., Lee, J., and Soltanolkotabi, M. Neural networks can learn representations with gradient descent. In *Conference on Learning Theory*, pp. 5413–5452. PMLR, 2022. Cited on page 10.

Dominé, C. C., Braun, L., Fitzgerald, J. E., and Saxe, A. M. Exact learning dynamics of deep linear networks with prior knowledge. *Journal of Statistical Mechanics: Theory and Experiment*, 2023 (11):114004, 2023. Cited on page 5.

Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., and Ruppin, E. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pp. 406–414, 2001. Cited on page 24.

Gidel, G., Bach, F., and Lacoste-Julien, S. Implicit regularization of discrete gradient dynamics in linear neural networks. *Advances in Neural Information Processing Systems*, 32, 2019. Cited on pages 2 and 5.

Gillis, N. and Glineur, F. Low-rank matrix approximation with weights or missing data is np-hard. *SIAM Journal on Matrix Analysis and Applications*, 32(4):1149–1165, 2011. Cited on page 5.

Gissin, D., Shalev-Shwartz, S., and Daniely, A. The implicit bias of depth: How incremental learning drives generalization. *arXiv preprint arXiv:1909.12051*, 2019. Cited on page 2.

HaoChen, J. Z., Wei, C., Gaidon, A., and Ma, T. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *Advances in Neural Information Processing Systems*, 34: 5000–5011, 2021. Cited on page 2.

Harris, Z. S. Distributional structure, 1954. Cited on page 3.

Huang, J. and Yau, H.-T. Dynamics of deep neural networks and neural tangent hierarchy. In *International conference on machine learning*, pp. 4542–4551. PMLR, 2020. Cited on page 34.

Ilharco, G., Ribeiro, M. T., Wortsman, M., Gururangan, S., Schmidt, L., Hajishirzi, H., and Farhadi, A. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022. Cited on page 8.

Jacot, A., Gabriel, F., and Hongler, C. Neural tangent kernel: Convergence and generalization in neural networks. *Advances in neural information processing systems*, 31, 2018. Cited on page 34.

Jacot, A., Ged, F., Şimşek, B., Hongler, C., and Gabriel, F. Saddle-to-saddle dynamics in deep linear networks: Small initialization training, symmetry, and sparsity. *arXiv preprint arXiv:2106.15933*, 2021. Cited on pages 2, 10, and 34.

Jiang, L., Chen, Y., and Ding, L. Algorithmic regularization in model-free overparametrized asymmetric matrix factorization. *SIAM Journal on Mathematics of Data Science*, 5(3):723–744, 2023. Cited on page 2.

Jiang, Y., Rajendran, G., Ravikumar, P., Aragam, B., and Veitch, V. On the origins of linear representations in large language models. In *Proceedings of the 41st International Conference on Machine Learning*, 2024. Cited on page 8.

Karkada, D. The lazy (ntk) and rich (µp) regimes: a gentle tutorial. *arXiv preprint arXiv:2404.19719*, 2024. Cited on page 34.

Kunin, D., Marchetti, G. L., Chen, F., Karkada, D., Simon, J. B., DeWeese, M. R., Ganguli, S., and Miolane, N. Alternating gradient flows: A theory of feature learning in two-layer neural networks. *arXiv preprint arXiv:2506.06489*, 2025. Cited on page 10.

Lauscher, A., Glavaš, G., Ponzetto, S. P., and Vulić, I. A general framework for implicit and explicit debiasing of distributional word vector spaces. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8131–8138, 2020. Cited on page 8.

Lee, A., Bai, X., Pres, I., Wattenberg, M., Kummerfeld, J. K., and Mihalcea, R. A mechanistic understanding of alignment algorithms: A case study on dpo and toxicity. *arXiv preprint arXiv:2401.01967*, 2024. Cited on page 8.

Lee, J., Xiao, L., Schoenholz, S., Bahri, Y., Novak, R., Sohl-Dickstein, J., and Pennington, J. Wide neural networks of any depth evolve as linear models under gradient descent. *Advances in neural information processing systems*, 32, 2019. Cited on page 34.

Levy, O. and Goldberg, Y. Neural word embedding as implicit matrix factorization. *Advances in neural information processing systems*, 27, 2014. Cited on pages 2, 3, and 33.

Levy, O., Goldberg, Y., and Dagan, I. Improving distributional similarity with lessons learned from word embeddings. *Transactions of the association for computational linguistics*, 3:211–225, 2015. Cited on pages 2 and 3.

Li, K., Patel, O., Viégas, F., Pfister, H., and Wattenberg, M. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36, 2024. Cited on page 8.

Li, Y., Ma, T., and Zhang, H. Algorithmic regularization in over-parameterized matrix sensing and neural networks with quadratic activations. In *Conference On Learning Theory*, pp. 2–47. PMLR, 2018. Cited on page 2.

Li, Y., Li, Y., and Risteski, A. How do transformers learn topic structure: Towards a mechanistic understanding. In *International Conference on Machine Learning*, pp. 19689–19729. PMLR, 2023. Cited on page 8.

Li, Z., Luo, Y., and Lyu, K. Towards resolving the implicit bias of gradient descent for matrix factorization: Greedy low-rank learning. In *International Conference on Learning Representations*, 2021. Cited on page 2.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013. Cited on pages 1, 2, 3, 23, and 24.

Nanda, N., Lee, A., and Wattenberg, M. Emergent linear representations in world models of self-supervised sequence models. In *Proceedings of the 6th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*. Association for Computational Linguistics, 2023. Cited on page 8.

Park, K., Choe, Y. J., and Veitch, V. The linear representation hypothesis and the geometry of large language models. *arXiv preprint arXiv:2311.03658*, 2023. Cited on page 8.

Pennington, J., Socher, R., and Manning, C. D. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, 2014. Cited on page 10.

Radhakrishnan, A., Beaglehole, D., Pandit, P., and Belkin, M. Mechanism for feature learning in neural networks and backpropagation-free machine learning models. *Science*, 383(6690): 1461–1467, 2024. Cited on page 10.

Rumelhart, D. E. and Abrahamson, A. A. A model for analogical reasoning. *Cognitive Psychology*, 5 (1):1–28, 1973. Cited on page 24.

Saunshi, N., Ash, J., Goel, S., Misra, D., Zhang, C., Arora, S., Kakade, S., and Krishnamurthy, A. Understanding contrastive learning requires incorporating inductive biases. In *International Conference on Machine Learning*, pp. 19250–19286. PMLR, 2022. Cited on page 2.

Saxe, A., McClelland, J., and Ganguli, S. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *Proceedings of the International Conference on Learning Represenatations 2014*. International Conference on Learning Represenatations 2014, 2014. Cited on pages 2, 5, and 29.

Saxe, A. M., McClelland, J. L., and Ganguli, S. A mathematical theory of semantic development in deep neural networks. *Proceedings of the National Academy of Sciences*, 116(23):11537–11546, 2019. Cited on page 2.

Schaeffer, R., Miranda, B., and Koyejo, S. Are emergent abilities of large language models a mirage? *Advances in Neural Information Processing Systems*, 36, 2024. Cited on page 10.

Simon, J. B., Dickens, M., Karkada, D., and Deweese, M. The eigenlearning framework: A conservation law perspective on kernel ridge regression and wide neural networks. *Transactions on Machine Learning Research*, 2023a. Cited on page 34.

Simon, J. B., Knutins, M., Ziyin, L., Geisz, D., Fetterman, A. J., and Albrecht, J. On the stepwise nature of self-supervised learning. In *International Conference on Machine Learning*, pp. 31852–31876. PMLR, 2023b. Cited on pages 2, 5, and 34.

Tian, Y., Chen, X., and Ganguli, S. Understanding self-supervised learning dynamics without contrastive pairs. In *International Conference on Machine Learning*, pp. 10268–10278. PMLR, 2021. Cited on page 2.

Vyas, N., Atanasov, A., Bordelon, B., Morwani, D., Sainathan, S., and Pehlevan, C. Feature-learning networks are consistent across widths at realistic scales. *Advances in Neural Information Processing Systems*, 36:1036–1060, 2023. Cited on page 34.

Wang, Z., Gui, L., Negrea, J., and Veitch, V. Concept algebra for (score-based) text-controlled generative models. *Advances in Neural Information Processing Systems*, 36, 2024. Cited on page 8.

Yang, G. and Hu, E. J. Tensor programs iv: Feature learning in infinite-width neural networks. In *International Conference on Machine Learning*, pp. 11727–11737. PMLR, 2021. Cited on page 10.

Zhu, L., Liu, C., Radhakrishnan, A., and Belkin, M. Quadratic models for understanding catapult dynamics of neural networks. *arXiv preprint arXiv:2205.11787*, 2022. Cited on page 10.

Zou, A., Phan, L., Chen, S., Campbell, J., Guo, P., Ren, R., Pan, A., Yin, X., Mazeika, M., Dombrowski, A.-K., et al. Representation engineering: A top-down approach to ai transparency. *arXiv preprint arXiv:2310.01405*, 2023. Cited on page 8.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and intro state the contributions of the paper and frame them in relation to the larger goal of developing theory for machine learning.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: The limitations are discussed in the final section.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Proofs can be found in the appendices.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Experimental details can be found in the appendices. Code is provided in supplemental material.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code to run experiments is provided in the supplemental material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.

- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The experimental details can be found in the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: The main contribution of this paper is theoretical; the experiment simply confirms the analytic theory. Our sanity checks revealed that our results are not sensitive to particular initializations or data subsets.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Given in appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: No ethics violations.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: This paper is primarily theoretical.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: No high risk for misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We train on Wikipedia, which has a CC 4.0 license.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.

A Experimental details

All our implementations use `jax` (Bradbury et al., 2018). In all reported experiments we use a vocabulary size $V = 10^4$, model dimension $d = 200$, and a context length $L = 32$ neighbors for each word. These were chosen fairly arbitrarily and our robustness checks indicated that our main empirical results were not sensitive to these choices.

A.1 Training corpus

We train all word embedding models on the November 2023 dump of English Wikipedia, downloaded from `https://huggingface.co/datasets/wikimedia/wikipedia`. We tokenize by replacing all non-alphabetical characters (including numerals and punctuation) with whitespace and splitting by whitespace. The full corpus contains 6.4 million articles and 4.34 billion tokens in total. The number of tokens per article follows a long-tailed distribution: the (50%, 90%, 95%, 99%) quantiles of the article token counts are (364, 1470, 2233, 5362) tokens. We discard all articles with fewer than 500 tokens, leaving a training corpus of 1.58 million articles with 2.00 billion tokens in total.

We use a vocabulary consisting of the $V = 10^4$ most frequently appearing words. We discard out-of-vocabulary words from both the corpus and the benchmarks. Our robustness checks indicated that as long as the corpus is sufficiently large (as is the case here), it does not matter practically whether out-of-vocabulary words are removed or simply masked. This choice of V retains 87% of the tokens in the training corpus and 53% of the analogy pairs.

A.2 Optimization

We optimize using stochastic gradient descent with no momentum nor weight decay. Each minibatch consists of 100,000 word pairs (50,000 positive pairs and 50,000 negative pairs). For benchmark evaluations (Figure 3) we use a stepwise learning rate schedule in which the base learning rate is decreased by 90% at $t = 0.75t_{\max}$. We found that this was very beneficial, especially for QWEMs, which appear more sensitive to large learning rates. The finite-stepsize gradient descent dynamics of matrix factorization problems remains an interesting area for future research.

We directly train the tensor of embedding weights $\boldsymbol{W} \in \mathbb{R}^{d\times V}$. One potential concern is that since the model $\boldsymbol{W}\boldsymbol{W}^\top$ is positive semidefinite, it cannot reconstruct the eigenmodes of $\boldsymbol{M}^\star \in \mathbb{R}^{V\times V}$ with negative eigenvalue. However, we empirically find that with $V = 10^4$, $\boldsymbol{M}^\star$ has 4795 non-negative eigenvalues. Therefore, in the underparameterized regime $d \ll V$, the model lacks the capacity to attempt fitting the negative eigenmodes. Thus, the PSD-ness of our model poses no problem.

A.3 Reweighting hyperparameters

Taking from the original `word2vec` implementation, we use the following engineering tricks to improve performance.

Dynamic context windows. Rather than using a fixed context window length L, `word2vec` uniformly samples the context length between 1 and L at each center word. In aggregate, this has the effect of more frequently sampling word pairs with less separation. Let d_{ij} be the mean distance between words i and j, when found co-occurring in contexts of length L. (Thus, d_i is small for the words "phase" and "transition" since they are a linguistic collocation, but large for "proved" and "derived" since verbs are typically separated by many words.) Then it is not hard to calculate the effect of uniformly-distributed dynamic context window lengths; it is equivalent to setting

$$\Psi^+_{ij} = \frac{L - d_{ij}}{\sum_{i'j'}(L - d_{i'j'})P_{i'j'}}. \tag{14}$$

Subsampling frequent words. Mikolov et al. (2013) suggest discarding very frequent words with a frequency-dependent discard probability. This is akin to rejection sampling, with the desired unigram distribution flatter than the true Zipfian distribution. The original `word2vec` implementation uses an acceptance probability of

$$P_{acc}(i) = \min\left(1, \frac{10^{-3}}{P_i} + \sqrt{\frac{10^{-3}}{P_i}}\right). \tag{15}$$

This is equivalent to setting the hyperparameters

$$\Psi^+_{ij} = \Psi^-_{ij} = P_{acc}(i)P_{acc}(j). \tag{16}$$

Different negative sampling rate. Finally, Mikolov et al. (2013) draw negative samples from a different distribution (empirically, they find that $P_j^{3/4}$ is particularly performant) and upweight the negative sampling term by a hyperparameter k (they recommend $k \approx 5$ for large corpora). This is equivalent to setting

$$\Psi_{ij}^{-} = \frac{kP_j^{-1/4}}{V^{-1}\sum_{j'} P_{j'}^{-1/4}}. \tag{17}$$

Note that this is an asymmetric choice, so it is not covered by Setting 3.1. However, both this and the previous subsampling technique are methods for modifying and flattening the word distributions, as seen by the training algorithm. Our Setting 3.1 accomplishes the same thing. We speculate that these manipulations accomplish for language data what spectral whitening does for image data.

Taken together, these give a prescription for cleanly and concisely capturing several of the implementation details of `word2vec` in a set of training hyperparameters. In our experiments, when training `word2vec`, we use all of these settings – they can be combined by multiplying the different Ψ^{+} and likewise for Ψ^{-}. For QWEMs, we use $\Psi_{ij}^{+} = \Psi_{ij}^{-} = (P_{ij} + P_i P_j)^{-1}$, and modify Ψ^{+} by combining with the setting for dynamic context windows. For fair comparison, we use $k = 1$ for the SGNS experiments in the main text.

A.4 Benchmarks

We use the Google analogies described in Mikolov et al. (2013) for the analogy completion benchmark, `https://github.com/tmikolov/word2vec/blob/master/questions-words.txt`. We then compute the analogy accuracy using

$$\mathrm{acc}(\boldsymbol{W}) \coloneqq \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{a},\boldsymbol{b},\boldsymbol{a}',\boldsymbol{b}')\in\mathcal{D}} \mathbf{1}_{\{\boldsymbol{b}'\}} \left(\underset{\boldsymbol{w}\in\{\boldsymbol{W}\}\setminus\{\boldsymbol{a},\boldsymbol{b},\boldsymbol{a}'\}}{\arg\max} \hat{\boldsymbol{w}}^{\top} \left(\hat{\boldsymbol{a}}' + (\hat{\boldsymbol{b}} - \hat{\boldsymbol{a}}) \right) \right) \tag{18}$$

$$= \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{a},\boldsymbol{b},\boldsymbol{a}',\boldsymbol{b}')\in\mathcal{D}} \mathbf{1}_{\{\boldsymbol{b}'\}} \left(\underset{\boldsymbol{w}\in\{\boldsymbol{W}\}\setminus\{\boldsymbol{a},\boldsymbol{b},\boldsymbol{a}'\}}{\arg\min} \left\| \hat{\boldsymbol{a}} - \hat{\boldsymbol{b}} - \hat{\boldsymbol{a}}' + \hat{\boldsymbol{w}} \right\|_{\mathrm{F}}^{2} \right), \tag{19}$$

where the 4-tuple of embeddings $(\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{a}', \boldsymbol{b}')$ constitute an analogy from the benchmark data $\mathcal{D}$, hats (e.g., $\hat{\boldsymbol{w}}$) denote normalized unit vectors, $\mathbf{1}$ is the indicator function, and $\{\boldsymbol{W}\}$ is the set containing the word embeddings. The first expression measures cosine alignment between embeddings and the "expected" representation obtained by summing the query word and the task vector. The second expression measures the degree to which four embeddings form a closed parallelogram (Rumelhart & Abrahamson, 1973). The two forms of the accuracy are equivalent since

$$\underset{\boldsymbol{w}}{\arg\min} \left\| \hat{\boldsymbol{a}} - \hat{\boldsymbol{b}} - \hat{\boldsymbol{a}}' + \hat{\boldsymbol{w}} \right\|_{\mathrm{F}}^{2} = \underset{\boldsymbol{w}}{\arg\min} \left(2\hat{\boldsymbol{w}}^{\top}(\hat{\boldsymbol{a}} - \hat{\boldsymbol{b}} - \hat{\boldsymbol{a}}') + \mathrm{const.} \right) \tag{20}$$

$$= \underset{\boldsymbol{w}}{\arg\max}\, \hat{\boldsymbol{w}}^{\top}(\hat{\boldsymbol{a}}' + \hat{\boldsymbol{b}} - \hat{\boldsymbol{a}}). \tag{21}$$

To understand the role of normalization, we compared this metric with one in which only the candidate embedding is normalized:

$$\mathrm{winner} = \underset{\boldsymbol{w}\in\{\boldsymbol{W}\}\setminus\{\boldsymbol{a},\boldsymbol{b},\boldsymbol{a}'\}}{\arg\max} \hat{\boldsymbol{w}}^{\top} \left(\boldsymbol{a}' + (\boldsymbol{b} - \boldsymbol{a}) \right). \tag{22}$$

We found that the accuracy metric using this selection criterion yields performance measurements that are nearly identical to those given by the standard fully-normalized accuracy metric. We conclude that the primary role of embedding normalization is to prevent longer $\boldsymbol{w}$'s from "winning" the $\arg\max$ just by virtue of their length. The lengths of $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{a}'$ are only important if there is significant *angular* discrepancy between $(\hat{\boldsymbol{a}}' + \hat{\boldsymbol{b}} - \hat{\boldsymbol{a}})$ and $(\boldsymbol{a}' + \boldsymbol{b} - \boldsymbol{a})$; in the high-dimensional regime with relatively small variations in embedding length, we expect such discrepancies to be negligible.

For semantic similarity benchmarks, we use the MEN dataset (Bruni et al., 2014) and the WordSim353 dataset (Finkelstein et al., 2001). These datasets consist of a set of N word pairs ranked by humans in order of increasing perceived semantic similarity. The model rankings are generated by computing the inner product between the embeddings in each pair and sorting them. The model is scored by computing the Spearman's rank correlation coefficient between its rankings and the human rankings:

$$\rho(\boldsymbol{W}) \coloneqq \mathrm{Corr}[r_{\mathrm{human}}(i), r_{\mathrm{model}}(i)] \tag{23}$$

where $r(i)$ is the rank of pair i and Corr is the standard Pearson's correlation coefficient.

A.5 Computational resources

Our implementations are relatively lightweight. The models can be trained on a home desktop computer with an i7 4-core processor, 32GB RAM, and a consumer-grade NVIDIA GTX 1660 graphics card, in about 2 hours. For the experiments in Figure 3, we train for 12 hours with a much lower learning rate. Our code is publicly available at `https://github.com/dkarkada/qwem`.

A.6 Empirics of task vectors

We construct the theoretical embeddings $\boldsymbol{W}$ by constructing $\boldsymbol{M}^\star$ according to Equation (5), diagonalizing it, and applying Theorem 1. Due to right orthogonal symmetry, we are free to apply any right orthogonal transformation; we choose the identity as the right singular matrix, so that the components of each embedding in the canonical basis are simply its projections on the eigen-features. With this setup, it is easy to extract the embeddings resulting from a smaller model: simply truncate the extraneous columns of $\boldsymbol{W}$.

We construct the task vectors by first collecting a dataset of semantic binaries. These can, for example, be extracted from the analogy dataset. The task vectors are then simply the difference vectors between the embeddings in each binary. In our reported results, we do not normalize either the word embeddings nor the task vectors. However, our robustness checks indicated that choosing to normalize does not qualitatively change our results.

Fitting the empirical spectra with the Marchenko-Pastur distribution. The Marchenko-Pastur distribution is a limiting empirical spectral distribution given in terms of an *aspect ratio* $\gamma := N/d$ and an overall scale σ^2. In particular, if $N < d$ and $\boldsymbol{\Xi} \in \mathbb{R}^{N\times d}$ is a random matrix with i.i.d. elements of mean zero and variance σ^2, then as $N, d \to \infty$ with $N/d = \gamma$ fixed, the MP law for $d^{-1}\boldsymbol{\Xi}\boldsymbol{\Xi}^\top$ is

$$\mathrm{d}\mu(\lambda) = \frac{1}{2\pi\sigma^2}\frac{\sqrt{(\lambda_+ - \lambda)(\lambda - \lambda_-)}}{\gamma\lambda}\mathrm{d}\lambda \tag{24}$$

$$\lambda_\pm := \sigma^2(1 \pm \sqrt{\gamma})^2 \tag{25}$$

where the support is $\lambda_- \le \lambda \le \lambda_+$, and the density is zero outside. Thus, the MP law gives the expected distribution of eigenvalues for a large noisy covariance (or Gram) matrix.

To fit the MP law to the Gram matrix of task vectors, we first subtract off the mean task vector. Then we compute the population variance of the elements of the centered matrix – we use this to set the σ^2 parameter. We manually fit the aspect ratio γ, and we find that the best-fitting $\hat{\gamma} = N/d_{\text{eff}}$ corresponds to an effective dimension which is often much lower than the true embedding dimension d. We hypothesize that $d_{\text{eff}} < d$ due to anisotropic noise in the task vectors. We report the d_{eff} resulting from the fit in Figure 4.

Composition of mean task vector. We measure the degree to which the spike captures the mean task vector by computing the ratio

$$\text{signal in mean} = \frac{\mathbf{1}^\top \boldsymbol{R}_d \boldsymbol{R}_d^\top \mathbf{1}}{N \cdot \lambda_{\max}(\boldsymbol{R}_d \boldsymbol{R}_d^\top)} \le 1, \tag{26}$$

where $\mathbf{1}$ is the ones-vector. If this quantity is (close to) 1, then the mean task vector is a large component of the spike, and vice versa. We find that this quantity is consistently greater than 0.9 for all the semantic categories in the analogy benchmark.

B Additional figures

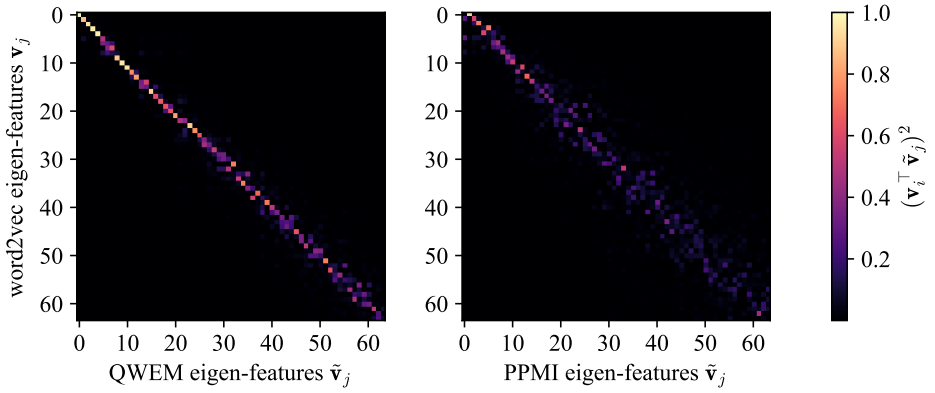

Figure 5: `word2vec` eigen-features align very closely with QWEMs' and less with PPMI. The heatmap indicates the squared overlaps between the latent feature vectors (i.e., the left singular vectors) between the three models considered. This is a quantitative version of the table in Figure 3.

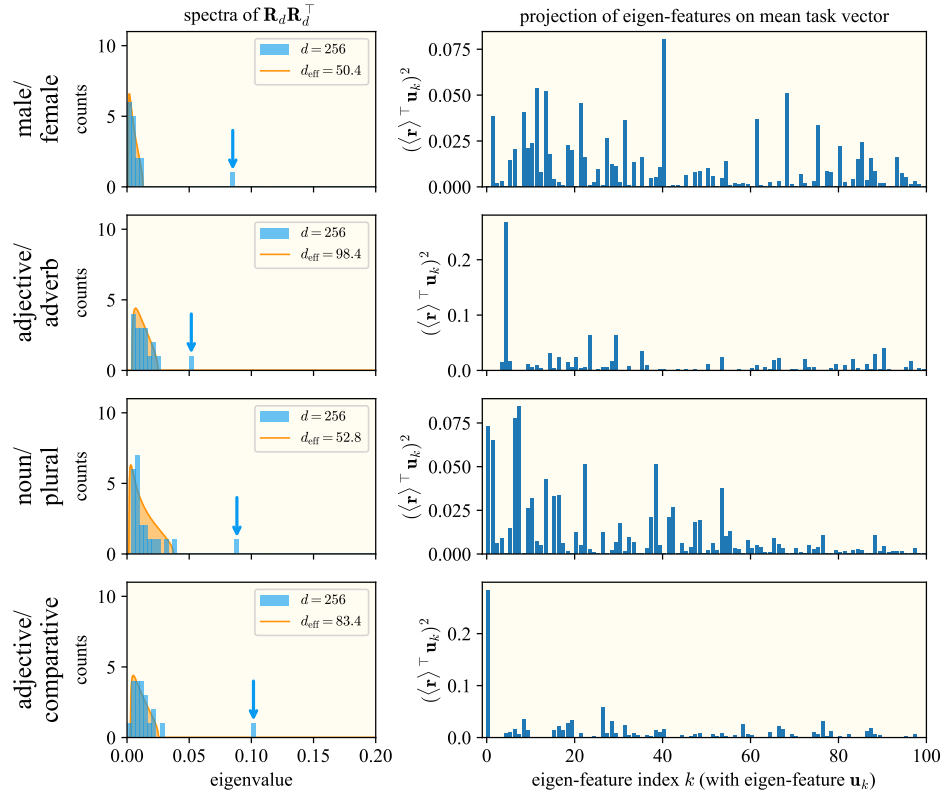

Figure 6: The empirical observations regarding linear representations extend across the semantic classes in the analogy dataset. We show that across analogy categories, the corresponding task vectors exhibit the geometric structure discussed in Figure 4.

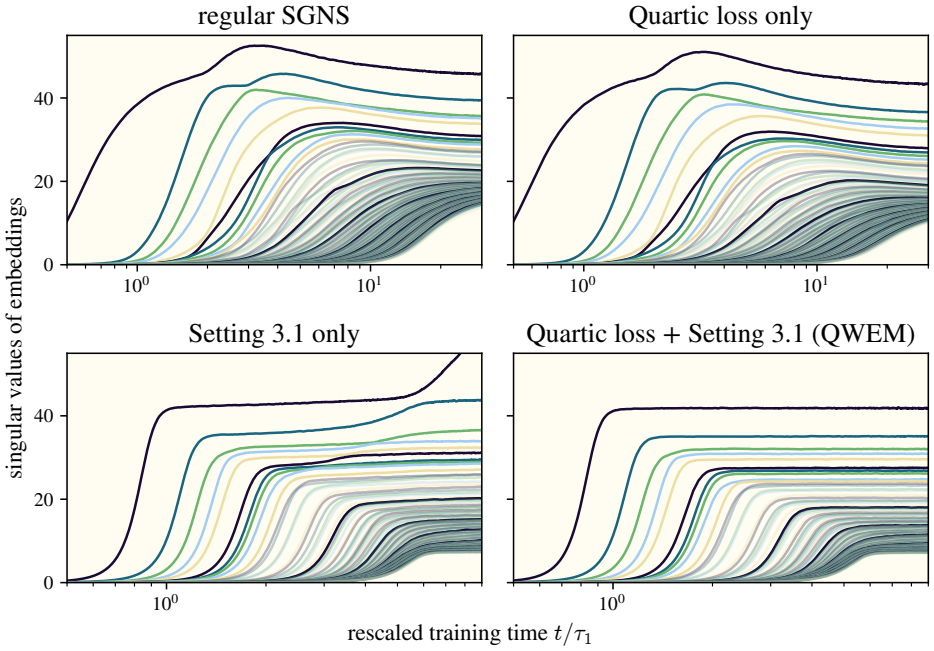

Figure 7: Ablation tests. Starting from vanilla `word2vec` (top left), we separately add in both the quartic approximation (top right) and the reweighting condition Setting 3.1 (bottom left). We include QWEM on the bottom right (i.e., both effects). The quartic approximation hardly changes the singular value dynamics at all; the clean mode separation is due to the reweighting. However, as is common with logistic losses, the weights begin to diverge at late times. Using the quartic loss prevents this. The experiments here use a 41-million-token mixture of the Corpus of Contemporary American English and the News on the Web dataset. We use a vocabulary size of 20,000; the embedding dimension is 150; 2 negative samples per positive sample; and a context length of 16.

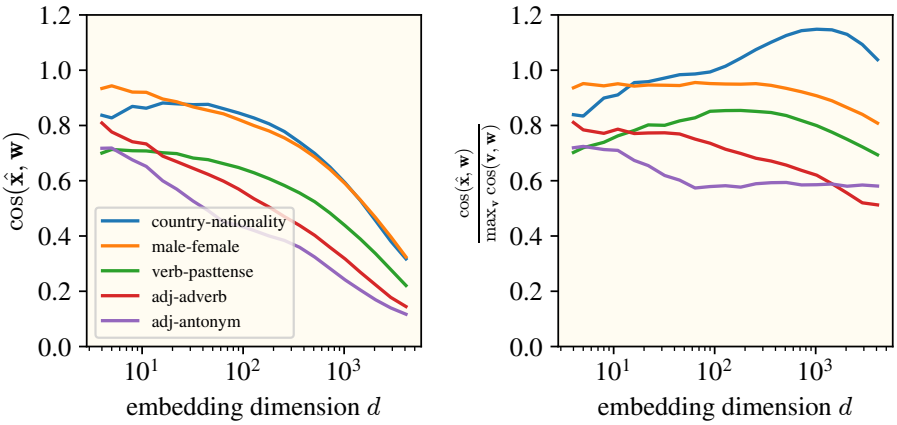

Figure 8: Although the analogy geometry degrades with d, relative scores remain high. We plot two different measures of analogy geometry across d to illustrate subtleties associated with the top-1 accuracy metric. For an analogy $a : b :: c : w$, we denote $\hat{\boldsymbol{x}} := \boldsymbol{c} + (\boldsymbol{b} - \boldsymbol{a})$ the prediction for the fourth word. We plot the average cosine similarity between $\hat{\boldsymbol{x}}$ and the true $\boldsymbol{w}$ over the analogy family. The prediction degrades dramatically as the embedding dimension increases, complementing the observation in Figure 4 that the SNR decreases with d. However, this geometric breakdown fails to capture the fact that *all* embedding vectors separate as d increases; if we normalize by the maximum cosine similarity between $\boldsymbol{w}$ and non-analogy embeddings, the score remains roughly stable at large d. This provides an explanation for why top-1 accuracy often remains high despite the breakdown of geometric analogical structure.

C Proofs

Theorem 1 (QWEM = unweighted matrix factorization)**.** *Under Setting 3.1, the contrastive loss Equation (3) can be rewritten as the unweighted matrix factorization problem*

$$\mathcal{L}(\boldsymbol{W}) = \frac{g}{4}\left\|\boldsymbol{W}\boldsymbol{W}^\top - \boldsymbol{M}^\star\right\|_{\mathrm{F}}^2 + \text{const.} \tag{6}$$

If $\boldsymbol{\Lambda}_{[:d,:d]}$ is positive semidefinite, then the set of global minima of $\mathcal{L}$ is given by

$$\arg\min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}) = \mathrm{REquiv}\left(\boldsymbol{V}^\star_{[:,:d]}\boldsymbol{\Lambda}^{1/2}_{[:d,:d]}\right). \tag{7}$$

Proof. Define $\boldsymbol{M} \coloneqq \boldsymbol{W}\boldsymbol{W}^\top$. Rewriting Equation (3) and plugging in Equation (5) and Setting 3.1,

$$\begin{aligned}
\mathcal{L}(\boldsymbol{W}) &= \sum_{i,j} P_{ij}\Psi^+_{ij}\left(\frac{1}{4}\boldsymbol{M}_{ij}^2 - \boldsymbol{M}_{ij}\right) + P_iP_j\Psi^-_{ij}\left(\frac{1}{4}\boldsymbol{M}_{ij}^2 + \boldsymbol{M}_{ij}\right) && (27)\\
&= \frac{1}{4}\sum_{ij}\boldsymbol{G}_{ij}\boldsymbol{M}_{ij}^2 - 4(P_{ij}\Psi^+_{ij} - P_iP_j\Psi^-_{ij})\boldsymbol{M}_{ij} && (28)\\
&= \frac{1}{4}\sum_{ij}\boldsymbol{G}_{ij}\left(\boldsymbol{M}_{ij}^2 - 2\frac{P_{ij}\Psi^+_{ij} - P_iP_j\Psi^-_{ij}}{\frac{1}{2}(P_{ij}\Psi^+_{ij} + P_iP_j\Psi^-_{ij})}\boldsymbol{M}_{ij}\right) && (29)\\
&= \frac{1}{4}\sum_{ij}\boldsymbol{G}_{ij}\left(\boldsymbol{M}_{ij}^2 - 2\boldsymbol{M}^\star_{ij}\boldsymbol{M}_{ij} + \boldsymbol{M}^{\star\,2}_{ij} - \boldsymbol{M}^{\star\,2}_{ij}\right) && (30)\\
&= \frac{g}{4}\left(\|\boldsymbol{M} - \boldsymbol{M}^\star\|_{\mathrm{F}}^2 + \|\boldsymbol{M}^\star\|_{\mathrm{F}}^2\right). && (31)
\end{aligned}$$

Since P_{ij}, P_iP_j, Ψ^+_{ij}, Ψ^-_{ij} are all real symmetric, so is $\boldsymbol{M}^\star$, so it has an eigendecomposition. By the Eckart-Young-Mirsky theorem, the loss-minimizing $\boldsymbol{M}$ must be the truncated SVD $\boldsymbol{M}^\star_{[d]}$, whose symmetric factors are exactly given by Equation (7). ■

Proposition 2. *Equation (3) can be rewritten as the weighted matrix factorization problem*

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{4}\sum_{i,j}\boldsymbol{G}_{ij}(\boldsymbol{W}\boldsymbol{W}^\top - \boldsymbol{M}^\star)_{ij}^2 + \text{const.} \tag{8}$$

Let Ψ^+ and Ψ^- each be symmetric in i, j, let $\boldsymbol{g} \in \mathbb{R}^V$ be an arbitrary vector with non-negative elements, and let $\boldsymbol{\Gamma} = \mathrm{diag}(\boldsymbol{g})^{1/2}$. Then the eigendecomposition of $\boldsymbol{\Gamma}\boldsymbol{M}^\star\boldsymbol{\Gamma} = \boldsymbol{V}^\star_\Gamma\boldsymbol{\Lambda}_\Gamma\boldsymbol{V}^{\star\top}_\Gamma$ exists. If $\boldsymbol{G} = \boldsymbol{g}\boldsymbol{g}^\top$ and $\boldsymbol{\Lambda}_{[:d,:d]}$ is positive semidefinite, then the set of global minima of $\mathcal{L}$ is given by

$$\arg\min_{\boldsymbol{W}} \mathcal{L}(\boldsymbol{W}) = \mathrm{REquiv}\left(\boldsymbol{\Gamma}^{-1}\boldsymbol{V}^\star_{\Gamma,[:,:d]}\boldsymbol{\Lambda}^{1/2}_{\Gamma,[:d,:d]}\right). \tag{9}$$

Proof. The formulation as a weighted matrix factorization follows from Equation (30). In the case that $\boldsymbol{G}$ is rank 1, substituting in $\boldsymbol{\Gamma}$, we find

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{4}\|\boldsymbol{\Gamma}(\boldsymbol{M} - \boldsymbol{M}^\star)\boldsymbol{\Gamma}\|_{\mathrm{F}}^2. \tag{32}$$

After distributing factors and invoking the Eckart-Young-Mirsky theorem, we conclude that the rank-d minimizer is

$$\boldsymbol{M}_{\min} = \boldsymbol{\Gamma}^{-1}\left(\boldsymbol{\Gamma}\boldsymbol{M}^\star\boldsymbol{\Gamma}\right)_{[d]}\boldsymbol{\Gamma}^{-1} \tag{33}$$

whose symmetric factors are exactly given by Equation (9). ■

Lemma 3.1 (Training dynamics, aligned initialization)**.** *If $\tilde{\boldsymbol{\Lambda}} \coloneqq \boldsymbol{\Lambda}_{[:d,:d]}$ is positive semidefinite and $\boldsymbol{V}(0) = \boldsymbol{V}^\star_{[:,:d]}$, then under Setting 3.1, the gradient flow dynamics $\frac{\mathrm{d}\boldsymbol{W}}{\mathrm{d}t} = -\frac{1}{2g}\nabla\mathcal{L}(\boldsymbol{W})$ yields*

$$\boldsymbol{U}(t) = \boldsymbol{U}(0) \tag{10}$$

$$\boldsymbol{S}(t) = \boldsymbol{S}(0)\left(e^{-\tilde{\boldsymbol{\Lambda}}t} + \boldsymbol{S}^2(0)\tilde{\boldsymbol{\Lambda}}^{-1}\left(\boldsymbol{I} - e^{-\tilde{\boldsymbol{\Lambda}}t}\right)\right)^{-1/2} \tag{11}$$

$$\boldsymbol{V}(t) = \boldsymbol{V}(0) = \boldsymbol{V}^\star_{[:,:d]} \tag{12}$$

Proof. By Theorem 1, we write the loss (Equation (3)) as

$$\mathcal{L}(\boldsymbol{W}) = \frac{g}{4}\,\mathrm{Tr}\left[\left(\boldsymbol{W}\boldsymbol{W}^\top - \boldsymbol{M}^\star\right)^\top(\boldsymbol{W}\boldsymbol{W}^\top - \boldsymbol{M}^\star)\right]. \tag{34}$$

We neglect constant terms since they do not affect the gradient descent dynamics. Under the stated gradient flow, the equation of motion is

$$\frac{\mathrm{d}\boldsymbol{W}}{\mathrm{d}t} = -\frac{1}{2g}\nabla\mathcal{L}(\boldsymbol{W}) \tag{35}$$

$$= -\frac{1}{8}\left(2(\boldsymbol{W}\boldsymbol{W}^\top - \boldsymbol{M}^\star)(2\boldsymbol{W})\right) \tag{36}$$

$$= \frac{1}{2}(\boldsymbol{M}^\star - \boldsymbol{W}\boldsymbol{W}^\top)\boldsymbol{W}. \tag{37}$$

From here on we adopt the dot notation for time derivatives. We use the spectral decompositions $\boldsymbol{W}(t)^\top = \boldsymbol{U}(t)\boldsymbol{S}(t)\boldsymbol{V}(t)^\top$ and $\boldsymbol{M}^\star = \boldsymbol{V}^\star\boldsymbol{\Lambda}\boldsymbol{V}^{\star\top}$. Then the above can be written

$$\dot{\boldsymbol{W}} = \dot{\boldsymbol{V}}\boldsymbol{S}\boldsymbol{U}^\top + \boldsymbol{V}\dot{\boldsymbol{S}}\boldsymbol{U}^\top + \boldsymbol{V}\boldsymbol{S}\dot{\boldsymbol{U}}^\top = \frac{1}{2}(\boldsymbol{V}^\star\boldsymbol{\Lambda}\boldsymbol{V}^{\star\top}\boldsymbol{V} - \boldsymbol{V}\boldsymbol{S}^2)\boldsymbol{S}\boldsymbol{U}^\top. \tag{38}$$

Left-multiplying by $\boldsymbol{V}^\top$ and right-multiplying by $\boldsymbol{U}$, we obtain

$$\boldsymbol{V}^\top\dot{\boldsymbol{V}}\boldsymbol{S} + \dot{\boldsymbol{S}} + \boldsymbol{S}\dot{\boldsymbol{U}}^\top\boldsymbol{U} = \frac{1}{2}(\tilde{\boldsymbol{\Lambda}} - \boldsymbol{S}^2)\boldsymbol{S}. \tag{39}$$

where we now use the alignment assumption, $\boldsymbol{V}^\top\boldsymbol{V}^\star = \boldsymbol{I}$. Note that since we use the economy SVD in our notation, we use a non-standard notation where $\boldsymbol{I} \in \mathbb{R}^{d\times V}$ is a rectangular matrix with ones on the main diagonal and zeros elsewhere. Now, since the RHS is diagonal, we must have that

$$(\boldsymbol{V}^\top\dot{\boldsymbol{V}}\boldsymbol{S} + \boldsymbol{S}\dot{\boldsymbol{U}}^\top\boldsymbol{U})_{ij} = 0 \qquad \text{for} \qquad i \neq j. \tag{40}$$

Furthermore, since $\boldsymbol{V}$ and $\boldsymbol{U}$ are orthogonal, $\boldsymbol{V}^\top\dot{\boldsymbol{V}}$ and $\dot{\boldsymbol{U}}^\top\boldsymbol{U}$ must both be antisymmetric (since for any orthogonal matrix $\boldsymbol{Q}$, $\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{Q}^\top\boldsymbol{Q} = \dot{\boldsymbol{Q}}^\top\boldsymbol{Q} + \boldsymbol{Q}^\top\dot{\boldsymbol{Q}} = \dot{\boldsymbol{I}} = 0$). It follows that, for all $i \neq j$,

$$(\boldsymbol{V}^\top\dot{\boldsymbol{V}})_{ij}s_i + (\dot{\boldsymbol{U}}^\top\boldsymbol{U})_{ij}s_j = (\boldsymbol{V}^\top\dot{\boldsymbol{V}})_{ij}s_j + (\dot{\boldsymbol{U}}^\top\boldsymbol{U})_{ij}s_i = 0. \tag{41}$$

Isolating $(\boldsymbol{V}^\top\dot{\boldsymbol{V}})_{ij}$, we see that this can only hold if $s_i = s_j$ or if $(\boldsymbol{V}^\top\dot{\boldsymbol{V}})_{ij} = (\dot{\boldsymbol{U}}^\top\boldsymbol{U})_{ij} = 0$. The former is ruled out by level repulsion in Gaussian random matrices; with probability 1 we have that $\boldsymbol{S}$ contains distinct singular values. We conclude that $\dot{\boldsymbol{U}} = 0$ and $\dot{\boldsymbol{V}} = 0$.

Returning to Equation (39), we have that

$$\dot{\boldsymbol{S}} = \frac{1}{2}(\tilde{\boldsymbol{\Lambda}} - \boldsymbol{S}^2)\boldsymbol{S}. \tag{42}$$

These are precisely the dynamics studied in Saxe et al. (2014). These dynamics are now decoupled, so we may solve each component separately. The solution to this equation is

$$s_k^2(t) = \frac{s_k^2(0)\ \lambda_k\ e^{\lambda_k t}}{\lambda_k + s_k^2(0)\left(e^{\lambda_k t} - 1\right)}. \tag{43}$$

Thus, the each singular direction of the embeddings is realized in a characteristic time

$$\tau_k = \frac{1}{\lambda_k}\ln\frac{\lambda_k}{s_k^2(0)}. \qquad \blacksquare \tag{44}$$

Result 3 (Training dynamics, vanishing random initialization)**.** *Initialize $\boldsymbol{W}_{ij}(0) \sim \mathcal{N}(0, \sigma^2)$, and let $\boldsymbol{S}(0)$ denote the singular values of $\boldsymbol{W}(0)$. Define the characteristic time $\tau_1 \coloneqq \lambda_1^{-1} \ln\left(\lambda_1/\sigma^2\right)$ and the rescaled time variable $\tilde{t} = t/\tau_1$. Define $\boldsymbol{W}^\star(0) \coloneqq \boldsymbol{V}^\star_{[:,:d]}\boldsymbol{S}(0)$ and let $\boldsymbol{W}^\star(t)$ be its gradient flow trajectory given by Lemma 3.1. If $\boldsymbol{\Lambda}_{[:d,:d]}$ is positive semidefinite, then under Setting 3.1 we have with high probability that*

$$\forall \tilde{t} \geq 0, \quad \lim_{\sigma^2 \to 0} \min_{\boldsymbol{W}' \in \mathrm{REquiv}(\boldsymbol{W}(\tilde{t}\tau_1))} \|\boldsymbol{W}' - \boldsymbol{W}^\star(\tilde{t}\tau_1)\|_{\mathrm{F}} = 0. \tag{13}$$

Derivation. Before starting the main derivation, we give a qualitative argument for why one expects the result of Lemma 3.1 to hold in the small initialization limit despite a lack of initial alignment.

Warmup. We begin with the equation of motion for $\boldsymbol{M} \coloneqq \boldsymbol{W}\boldsymbol{W}^\top$:

$$\dot{\boldsymbol{M}} = \boldsymbol{W}\dot{\boldsymbol{W}}^\top + \dot{\boldsymbol{W}}\boldsymbol{W}^\top = \frac{1}{2}\left(\boldsymbol{M}\boldsymbol{M}^\star + \boldsymbol{M}^\star\boldsymbol{M} - 2\boldsymbol{M}^2\right). \tag{45}$$

We again consider the dynamics in terms of the spectral decompositions $\boldsymbol{M}(t) = \boldsymbol{V}(t)\boldsymbol{S}^2(t)\boldsymbol{V}(t)^\top$ and $\boldsymbol{M}^\star = \boldsymbol{V}^\star\boldsymbol{\Lambda}^\star\boldsymbol{V}^{\star\top}$. Note that here $\boldsymbol{V}, \boldsymbol{S} \in \mathbb{R}^{V \times V}$ are square. We define the eigenbasis overlap $\boldsymbol{O} \coloneqq \boldsymbol{V}^{\star\top}\boldsymbol{V}$. After transforming coordinates to the target eigenbasis, we find

$$\begin{aligned}
\boldsymbol{V}^{\star\top}\dot{\boldsymbol{M}}\boldsymbol{V}^\star &= \boldsymbol{V}^{\star\top}(\dot{\boldsymbol{V}}\boldsymbol{S}^2\boldsymbol{V}^\top + \boldsymbol{V}(2\boldsymbol{S}\dot{\boldsymbol{S}})\boldsymbol{V}^\top + \boldsymbol{V}\boldsymbol{S}^2\dot{\boldsymbol{V}}^\top)\boldsymbol{V}^\star && (46)\\
&= \dot{\boldsymbol{O}}\boldsymbol{S}^2\boldsymbol{O}^\top + 2\boldsymbol{O}\boldsymbol{S}\dot{\boldsymbol{S}}\boldsymbol{O}^\top + \boldsymbol{O}\boldsymbol{S}^2\dot{\boldsymbol{O}}^\top && (47)\\
&= \frac{\boldsymbol{\Lambda}\boldsymbol{O}\boldsymbol{S}^2\boldsymbol{O}^\top + \boldsymbol{O}\boldsymbol{S}^2\boldsymbol{O}^\top\boldsymbol{\Lambda}}{2} - \boldsymbol{O}\boldsymbol{S}^4\boldsymbol{O}^\top. && (48)
\end{aligned}$$

For clarity, we rotate coordinates again into the $\boldsymbol{O}$ basis and find

$$\boldsymbol{S}^2\dot{\boldsymbol{O}}^\top\boldsymbol{O} + \boldsymbol{O}^\top\dot{\boldsymbol{O}}\boldsymbol{S}^2 + 2\boldsymbol{S}\dot{\boldsymbol{S}} = \frac{\boldsymbol{S}^2\boldsymbol{O}^\top\boldsymbol{\Lambda}\boldsymbol{O} + \boldsymbol{O}^\top\boldsymbol{\Lambda}\boldsymbol{O}\boldsymbol{S}^2}{2} - \boldsymbol{S}^4. \tag{49}$$

Since $\boldsymbol{O}$ is orthogonal, it satisfies $\dot{\boldsymbol{O}}^\top\boldsymbol{O} + \boldsymbol{O}^\top\dot{\boldsymbol{O}} = \boldsymbol{0}$ (this follows from differentiating the identity $\boldsymbol{O}^\top\boldsymbol{O} = \boldsymbol{I}$). Therefore the first two terms on the LHS of Equation (49), which concern the singular vector dynamics, have zero diagonal; the third term, which concerns singular value dynamics, has zero off-diagonal. This implies

$$2\boldsymbol{S}\dot{\boldsymbol{S}} = \left(\mathrm{diag}(\boldsymbol{O}^\top\boldsymbol{\Lambda}\boldsymbol{O}) - \boldsymbol{S}^2\right)\boldsymbol{S}^2, \tag{50}$$

where $\mathrm{diag}(\cdot)$ is the diagonal matrix formed from the diagonal of the argument. From Equation (50), we see that at initialization $2\boldsymbol{S}\dot{\boldsymbol{S}}$ scales with the initialization size σ^2 since $\boldsymbol{S}^2(0) \sim \sigma^2$. On the other hand, from the off-diagonal of Equation (49), we see that $\dot{\boldsymbol{O}}$ is order 1 (since the scale of $\boldsymbol{O}$ is fixed by orthonormality). Therefore, in the limit of small initialization, we expect the model to align quickly compared to the dynamics of $\boldsymbol{S}^2$. This motivates the *silent alignment ansatz*, which informally posits that with high probability, the top $d \times d$ submatrix of $\boldsymbol{O}$ converges towards the identity matrix well before $\boldsymbol{S}^2$ reaches the scale of $\boldsymbol{\Lambda}$. As $\boldsymbol{O} \to \boldsymbol{I}$, the dynamics decouple and enter a regime well-described by Lemma 3.1. We formalize this idea in our concrete derivation of Result 3.

Main derivation. We are interested in showing that

$$\forall \tilde{t} \geq 0, \quad \lim_{\sigma^2 \to 0} \min_{\boldsymbol{W}' \in \mathrm{REquiv}(\boldsymbol{W}(\tilde{t}\tau_1))} \|\boldsymbol{W}' - \boldsymbol{W}^\star(\tilde{t}\tau_1)\|_{\mathrm{F}} = 0, \tag{51}$$

where we express the statement in terms of rescaled time $\tilde{t} = t/\tau_1$ since we anticipate that $\tau_1 \to \infty$ as $\sigma^2 \to 0$. For notational clarity, though, let us switch back to the original time variable. Then

$$\begin{aligned}
\min_{\boldsymbol{W}'(t) \in \mathrm{REquiv}(\boldsymbol{W}(t))} \|\boldsymbol{W}'(t) - \boldsymbol{W}^\star(t)\|_{\mathrm{F}} &= \min_{\boldsymbol{W}'(t) \in \mathrm{REquiv}(\boldsymbol{W}(t))} \|\boldsymbol{W}'(t)^\top - \boldsymbol{W}^\star(t)^\top\|_{\mathrm{F}} && (52)\\
&= \min_{\boldsymbol{U}'(t) \in \mathcal{O}} \|\boldsymbol{U}'(t)\boldsymbol{U}(t)\boldsymbol{S}(t)\boldsymbol{V}^\top(t) - \boldsymbol{S}^\star(t)\boldsymbol{V}^{\star\top}\|_{\mathrm{F}} && (53)\\
&= \min_{\boldsymbol{U}'(t) \in \mathcal{O}} \|\boldsymbol{U}'(t)\boldsymbol{U}(t)\boldsymbol{S}(t)\boldsymbol{O}^\top(t) - \boldsymbol{S}^\star(t)\|_{\mathrm{F}} && (54)
\end{aligned}$$

where we again define the eigenbasis overlap $\boldsymbol{O} \coloneqq \boldsymbol{V}^{\star\top}\boldsymbol{V}$.

Motivated by the expectation that we will see sequential learning dynamics starting from the top mode and descending into lower modes, we seek a change of variables in which the dynamics are expressed in an upper-triangular matrix. We can achieve this reparameterization using a QR factorization: $\boldsymbol{USO}^\top \to \boldsymbol{QR}$, where $\boldsymbol{Q}$ is orthogonal and $\boldsymbol{R}$ is upper triangular. Then

$$\min_{\boldsymbol{W}'(t)\in \mathrm{REquiv}(\boldsymbol{W}(t))} \|\boldsymbol{W}'(t) - \boldsymbol{W}^\star(t)\|_{\mathrm{F}} = \min_{\boldsymbol{U}'(t)\in\mathcal{O}} \|\boldsymbol{U}'(t)\boldsymbol{Q}(t)\boldsymbol{R}(t) - \boldsymbol{S}^\star(t)\|_{\mathrm{F}} \tag{55}$$

$$= \min_{\boldsymbol{U}'(t)\in\mathcal{O}} \|\boldsymbol{U}'(t)\boldsymbol{R}(t) - \boldsymbol{S}^\star(t)\|_{\mathrm{F}}, \tag{56}$$

where the second equation follows since $\boldsymbol{U}'$ is a variable to be minimized, and it can simply absorb $\boldsymbol{Q}$. We emphasize that this convenience follows from the right orthogonal symmetry of the embeddings. If we can show that $\|\boldsymbol{R}(t) - \boldsymbol{S}^\star(t)\|_{\mathrm{F}} \to 0$ for all t in the vanishing initialization limit, then we will have completed the derivation.

Our starting point will be transpose of Equation (37), right-multiplied by $\boldsymbol{V}^\star$:

$$\frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{USO}^\top) = \frac{1}{2}\boldsymbol{USO}^\top(\boldsymbol{\Lambda} - \boldsymbol{OS}^2\boldsymbol{O}^\top) \implies \dot{\boldsymbol{Q}}\boldsymbol{R} + \boldsymbol{Q}\dot{\boldsymbol{R}} = \frac{1}{2}\boldsymbol{QR}\left(\boldsymbol{\Lambda} - \boldsymbol{R}^\top\boldsymbol{R}\right). \tag{57}$$

Left-multiplying by $2\boldsymbol{Q}^\top$ and rearranging, we find

$$2\dot{\boldsymbol{R}} = \boldsymbol{R}\left(\boldsymbol{\Lambda} - \boldsymbol{R}^\top\boldsymbol{R}\right) - 2\boldsymbol{Q}^\top\dot{\boldsymbol{Q}}\boldsymbol{R} \tag{58}$$

$$= \boldsymbol{R}\boldsymbol{\Lambda} - (\boldsymbol{R}\boldsymbol{R}^\top + 2\boldsymbol{Q}^\top\dot{\boldsymbol{Q}})\boldsymbol{R} \tag{59}$$

$$= \boldsymbol{R}\boldsymbol{\Lambda} - \tilde{\boldsymbol{R}}\boldsymbol{R}, \tag{60}$$

where we define $\tilde{\boldsymbol{R}} \coloneqq \boldsymbol{R}\boldsymbol{R}^\top + 2\boldsymbol{Q}^\top\dot{\boldsymbol{Q}}$. Note that $\tilde{\boldsymbol{R}}$ must be upper triangular: since the other terms in the equation are upper triangular, so must be $\tilde{\boldsymbol{R}}\boldsymbol{R}$; and since $\boldsymbol{R}$ and $\tilde{\boldsymbol{R}}\boldsymbol{R}$ are both upper triangular, then $\tilde{\boldsymbol{R}}$ must be upper triangular.

In fact, this is enough to fully determine the elements of $\tilde{\boldsymbol{R}}$. We know that $\boldsymbol{Q}^\top\dot{\boldsymbol{Q}}$ is antisymmetric (since $\boldsymbol{Q}^\top\boldsymbol{Q} = \boldsymbol{I}$ by orthogonality, $\boldsymbol{Q}^\top\dot{\boldsymbol{Q}} + \dot{\boldsymbol{Q}}^\top\boldsymbol{Q} = \boldsymbol{0}$). Additionally using the fact that $\boldsymbol{R}\boldsymbol{R}^\top$ is symmetric and imposing upper-triangularity on the sum, we have that

$$\tilde{\boldsymbol{R}}_{ij} = \begin{cases} 2(\boldsymbol{R}\boldsymbol{R}^\top)_{ij} & \text{if} \quad i < j \\ (\boldsymbol{R}\boldsymbol{R}^\top)_{ii} & \text{if} \quad i = j \\ 0 & \text{if} \quad i > j \end{cases}. \tag{61}$$

Here, we take a moment to examine the dynamics in Equation (60). Treating the initialization scale σ as a scaling variable, we expect that $\boldsymbol{R}_{ij} \sim \sigma$. Thus, in the small initialization limit, we expect the second term (which scales like σ^3) to contribute negligibly until late times; initially, we will see an exponential growth in the elements of $\boldsymbol{R}$ with growth rates given by $\boldsymbol{\Lambda}$. Later, $\boldsymbol{R}$ will (roughly speaking) reach the scale of $\boldsymbol{\Lambda}^{1/2}$, and there will be competitive dynamics between the two terms. We now write out the element-wise dynamics of $\boldsymbol{R}$ to see this precisely.

$$2\dot{\boldsymbol{R}}_{ij} = \boldsymbol{R}_{ij}\lambda_j - \sum_{j\geq k\geq i} \tilde{\boldsymbol{R}}_{ik}\boldsymbol{R}_{kj} \tag{62}$$

$$= \boldsymbol{R}_{ij}\lambda_j - \sum_{j\geq k\geq i}\sum_{\ell\geq k}(2-\delta_{ik})\boldsymbol{R}_{i\ell}\boldsymbol{R}_{k\ell}\boldsymbol{R}_{kj} \tag{63}$$

$$= \boldsymbol{R}_{ij}\lambda_j - \sum_{\ell\geq i}\boldsymbol{R}_{i\ell}^2\boldsymbol{R}_{ij} - 2\sum_{j\geq k>i}\sum_{\ell\geq k}\boldsymbol{R}_{i\ell}\boldsymbol{R}_{k\ell}\boldsymbol{R}_{kj} \tag{64}$$

$$= \boldsymbol{R}_{ij}\lambda_j - \sum_{\ell\geq i}\boldsymbol{R}_{i\ell}^2\boldsymbol{R}_{ij} - 2\sum_{j\geq k>i}\boldsymbol{R}_{ij}\boldsymbol{R}_{kj}^2 - 2\sum_{j\geq k>i}\sum_{j>\ell\geq k}\boldsymbol{R}_{i\ell}\boldsymbol{R}_{k\ell}\boldsymbol{R}_{kj} \tag{65}$$

$$= \left(\lambda_j - \sum_{\ell\geq i}\boldsymbol{R}_{i\ell}^2 - 2\sum_{j\geq k>i}\boldsymbol{R}_{kj}^2\right)\boldsymbol{R}_{ij} - 2\sum_{j\geq k>i}\sum_{j>\ell\geq k}\boldsymbol{R}_{i\ell}\boldsymbol{R}_{k\ell}\boldsymbol{R}_{kj}. \tag{66}$$

We have separated the dynamics of $\boldsymbol{R}_{ij}$ into a part that is explicitly linear in $\boldsymbol{R}_{ij}$ and a part which has no explicit dependence on $\boldsymbol{R}_{ij}$. (Of course, there is coupling between all the elements of $\boldsymbol{R}$ and $\boldsymbol{R}_{ij}$ through their own dynamical equations.) So far, everything we have done is exact.

Our main approximation is to argue that at all times, only the diagonal elements of $\boldsymbol{R}$ contribute non-negligibly to the dynamics. This holds if the off-diagonal elements are initialized vanishingly small and if they remain vanishingly small throughout. In this case, we may discard any terms that include couplings between off-diagonal elements.

With this approximation, we may discard the entire second term on the RHS, as well as some of the summands in the first prefactor for $\boldsymbol{R}_{ij}$. This leaves the approximate dynamics

$$2\dot{\boldsymbol{R}}_{ij} = \left(\lambda_j - \boldsymbol{R}_{ii}^2 - 2(1-\delta_{ij})\boldsymbol{R}_{jj}^2\right)\boldsymbol{R}_{ij}. \tag{67}$$

We may now directly solve for the diagonal dynamics. Letting $r_k \coloneqq \boldsymbol{R}_{kk}$,

$$\dot{r}_k = \frac{1}{2}\left(\lambda_k - r_k^2\right) r_k \quad \Longrightarrow \quad r_k^2(t) = \frac{r_k^2(0)\ \lambda_k\ e^{\lambda_k t}}{\lambda_k + r_k^2(0)\left(e^{\lambda_k t} - 1\right)}. \tag{68}$$

We obtain the same sigmoidal dynamics as in Lemma 3.1. If we show that the off-diagonal elements remain vanishingly small in the limit $\sigma^2 \to 0$, then: a) our approximation is justified, and b) it follows that $\|\boldsymbol{R}(t) - \boldsymbol{S}^\star(t)\|_{\mathrm{F}} \to 0$ for all t, completing the derivation.

To do this, we examine the dynamics of the off-diagonals and show that the maximum scale they achieve (at any time) decays to zero as $\sigma^2 \to 0$. For $i < j$ we have

$$2\dot{\boldsymbol{R}}_{ij} = \left(\lambda_j - r_i^2 - 2r_j^2\right)\boldsymbol{R}_{ij} \tag{69}$$

This is a linear first-order homogeneous ODE with a time-dependent coefficient, and thus it can be solved exactly:

$$\boldsymbol{R}_{ij}^2(t) = \boldsymbol{R}_{ij}^2(0)\ \cdot \left(\frac{\lambda_j}{\lambda_j + r_j^2(0)\ (e^{\lambda_j t} - 1)}\right)^2 \cdot \frac{\lambda_i}{\lambda_i + r_i^2(0)\ (e^{\lambda_i t} - 1)} \cdot e^{\lambda_j t} \tag{70}$$

$$= \boldsymbol{R}_{ij}^2(0) \cdot \frac{r_j^4(t)}{r_j^4(0)} \cdot \frac{r_i^2(t)}{r_i^2(0)} \cdot e^{-(\lambda_i + \lambda_j)t}. \tag{71}$$

This product contains two factors with sigmoidal dynamics of different timescales, and one factor with an exponential decay to the dynamics. Thus, as $t \to \infty$, the first two factors saturate, and the decay drives the off-diagonal elements $\boldsymbol{R}_{ij}$ to zero. We now show that $\max_t \boldsymbol{R}_{ij}^2(t)$ vanishes as $\sigma^2 \to 0$. Focusing on the scaling w.r.t. the initialization scale, we may approximate $\boldsymbol{R}_{k\ell}^2(0) \approx \sigma^2$ for all $k \leq \ell$. Discarding $O(\sigma^4)$ terms and solving $\dot{\boldsymbol{R}}_{ij} = 0$, we find

$$\max_t \boldsymbol{R}_{ij}^2 \approx \left(\frac{\lambda_i \lambda_j}{\lambda_i - \lambda_j}\right)^{\lambda_j/\lambda_i} \sigma^{2(\lambda_i - \lambda_j)/\lambda_i} \quad \text{when} \quad t = \frac{1}{\lambda_i} \log \frac{\lambda_i \lambda_j}{\sigma^2(\lambda_i - \lambda_j)}. \tag{72}$$

Therefore, for $i < j$, assuming $\lambda_i \neq \lambda_j$,

$$\lim_{\sigma^2 \to 0} \max_t \boldsymbol{R}_{ij}^2 = 0 \tag{73}$$

and we conclude that $\|\boldsymbol{R}(t) - \boldsymbol{S}^\star(t)\|_{\mathrm{F}} \to 0$ as desired.

In fact, under these approximations, as long as the initialization scale satisfies

$$\log \sigma^2 \ll -\frac{\lambda_j}{\lambda_i - \lambda_j} \log\left(\frac{\lambda_i \lambda_j}{\lambda_i - \lambda_j}\right) \tag{74}$$

for all i and j, the off-diagonal elements will remain much smaller than the diagonal elements, and we may view the diagonal element dynamics as simply being the singular value dynamics. This follows from Weyl's inequality for matrix perturbations. Thus we may expect that our results hold in the small-but-finite initialization regime (e.g., the regime accessed by our experiments). ■

D Relation between QWEMs and known algorithms

D.1 Relation to PMI

Early word embedding algorithms obtained low-dimensional embeddings by explicitly constructing some target matrix and employing a dimensionality reduction algorithm. One popular choice was the *pointwise mutual information* (PMI) matrix (Church & Hanks, 1990), defined

$$\mathrm{PMI}_{ij} = \log \frac{P_{ij}}{P_i P_j}. \tag{75}$$

Later, Levy & Goldberg (2014) showed that PMI is the rank-unconstrained minimizer of $\mathcal{L}_{\texttt{w2v}}$. To see the relation between the QWEM target $\boldsymbol{M}^\star$ and PMI, let us write

$$\frac{P_{ij}}{P_i P_j} = 1 + \Delta(x_{ij}), \tag{76}$$

where the function $\Delta(x)$ yields the fractional deviation away from independent word statistics, in terms of some small parameter x of our choosing (so that $\Delta(0) = 0$). This setup allows us to Taylor expand quantities of interest around $x = 0$. A judicious choice will produce terms that cancel the $-\frac{1}{2}\Delta^2$ that arises from the Taylor expansion of $\log(1 + \Delta)$, leaving only third-order corrections. One such example is $\Delta(x) = 2x/(2 - x)$, which yields

$$x_{ij} = \frac{P_{ij} - P_i P_j}{\frac{1}{2}(P_{ij} + P_i P_j)} = \boldsymbol{M}^\star_{ij} \tag{77}$$

and

$$\mathrm{PMI} = \log\left(1 + \frac{2x}{2 - x}\right) = x + \frac{x^3}{12} + \frac{x^5}{80} + \cdots \tag{78}$$

This calculation reveals that $\boldsymbol{M}^\star$ learns a very close approximation to the PMI matrix; the leading correction is third order. However, $\boldsymbol{M}^\star$ is much friendlier to least-squares approximation, since x is bounded ($-2 \le \boldsymbol{M}^\star_{ij} \le 2$).

D.2 Relation to SimCLR

SimCLR is a widely-used contrastive learning algorithm for learning visual representations (Chen et al., 2020). It uses a deep convolutional encoder to produce latent representations from input images. Data augmentation is used to construct positive pairs; negative pairs are drawn uniformly from the dataset. The encoder is then trained using the *normalized temperature-scaled cross entropy loss*:

$$\mathcal{L}(\boldsymbol{f}_\theta) = \mathop{\mathbb{E}}_{i,j\sim\Pr(\cdot,\cdot)}\left[-\log \frac{\exp\Big(\beta \boldsymbol{f}_\theta(x_i)^\top \boldsymbol{f}_\theta(x_j)\Big)}{\sum_{k\neq j}^{B} \exp\Big(\beta \boldsymbol{f}_\theta(x_i)^\top \boldsymbol{f}_\theta(x_k)\Big)}\right], \tag{79}$$

where $\Pr(\cdot,\cdot)$ is the positive pair distribution, $\boldsymbol{f}_\theta x_i$ is the learned representation of x_i, β is an inverse temperature hyperparameter, and B is the batch size. Defining $\boldsymbol{S}_\theta(i,j) = \boldsymbol{f}_\theta(x_i)^\top \boldsymbol{f}_\theta(x_j)$, in the limit of large batch size, we can Taylor expand this objective function around the origin:

$$\begin{aligned}
\mathcal{L}(\boldsymbol{S}_\theta) &= \mathop{\mathbb{E}}_{i,j\sim\Pr(\cdot,\cdot)}\left[-\beta \boldsymbol{S}_\theta(i,j) + \log\left(\mathop{\mathbb{E}}_{k\sim\Pr(\cdot)}\big[\exp(\beta \boldsymbol{S}_\theta(i,k))\big]\right) + \log B\right] && (80)\\
&\approx \mathop{\mathbb{E}}_{i,j\sim\Pr(\cdot,\cdot)}\left[-\beta \boldsymbol{S}_\theta(i,j) + \mathop{\mathbb{E}}_{k\sim\Pr(\cdot)}\big[\exp(\beta \boldsymbol{S}_\theta(i,k))\big] - 1\right] + \log B && (81)\\
&\approx \mathop{\mathbb{E}}_{i,j\sim\Pr(\cdot,\cdot)}\left[-\beta \boldsymbol{S}_\theta(i,j)\right] + \mathop{\mathbb{E}}_{\substack{i\sim\Pr(\cdot)\\ k\sim\Pr(\cdot)}}\left[1 + \beta \boldsymbol{S}_\theta(i,k) + \frac{1}{2}\beta^2 \boldsymbol{S}^2_\theta(i,k)\right] - 1 + \log B && (82)\\
&\approx \beta\left(\mathop{\mathbb{E}}_{i,j\sim\Pr(\cdot,\cdot)}\left[-\boldsymbol{S}_\theta(i,j)\right] + \mathop{\mathbb{E}}_{\substack{i\sim\Pr(\cdot)\\ j\sim\Pr(\cdot)}}\left[\boldsymbol{S}_\theta(i,j) + \frac{\beta}{2}\boldsymbol{S}^2_\theta(i,j)\right]\right) + \text{const.} && (83)\\
& && (84)
\end{aligned}$$

If the model is in a linearized regime, we may approximate $\boldsymbol{S}_\theta(i,j) \approx \boldsymbol{f}_i^\top \boldsymbol{f}_j$ for some linearized feature vectors $\boldsymbol{f}$. Then the loss can be written as an unweighted matrix factorization problem using exactly the same argument as in Theorem 1. Thus, we expect that vision models trained under the SimCLR loss in a linearized regime will undergo stepwise sigmoidal dynamics. This provides an explanation for the previously unresolved observation in Simon et al. (2023b) that vision models trained with SimCLR from small initialization exhibit stepwise learning.

D.3 Relation to next-token prediction.

Word embedding targets are order-2 tensors $\boldsymbol{M}^\star$ that captures two-token (skip-gram) statistics. These two-token statistics are sufficient for coarse semantic understanding tasks such as analogy completion. To perform well on more sophisticated tasks, however, requires modeling more sophisticated language distributions.

The current LLM paradigm demonstrates that the next-token distribution is largely sufficient for most downstream tasks of interest. The next-token prediction (NTP) task aims to model the probability of finding word i given a preceding window of context tokens of length $L-1$. Therefore, the NTP target is an order-L tensor that captures the joint distribution of length-L contexts. NTP thus *generalizes* the word embedding task. Both QWEM and LLMs are underparameterized models that learn internal representations with interpretable and task-relevant vector structure. Both are trained using self-supervised gradient descent algorithms, *implicitly* learning a compression of natural language statistics by iterating through the corpus.

Although the size of the NTP solution space is exponential in L (i.e., much larger than that of QWEM), LLMs succeed because the sparsity of the target tensor increases with L. We conjecture, then, that a dynamical description of learning sparse high-dimensional tensors is necessary for a general scientific theory of when and how LLMs succeed on reasoning tasks and exhibit failures such as hallucinations or prompt attack vulnerabilities.

D.4 Relation to neural tangent kernel.

Our result describing an implicit bias towards low rank directly contrasts the well-studied neural tangent kernel training (NTK) regime (Jacot et al., 2018; Chizat et al., 2019; Karkada, 2024). Here we compare the two learning regimes.

Learning with NTK.

- Learning dynamics and generalization performance can be solved (Lee et al., 2019; Bordelon et al., 2020; Simon et al., 2023a).
- Extreme over-parameterization is required; large finite-width corrections at practical widths (Huang & Yau, 2020).
- Model weights do not learn task-relevant features.
- Optimization remains in a locally convex region of the loss landscape (Chizat et al., 2019).

Learning from small initialization.

- Learning dynamics are generally complicated; can be solved in very simple cases (linear networks, special data distributions, etc.).
- Behavior is qualitatively consistent across network widths, with only moderate finite-width corrections (Vyas et al., 2023).
- Model weights learn task-relevant features.
- Optimization tends to pass near a sequence of saddle points. (Baldi & Hornik, 1989; Jacot et al., 2021).