On the Origins of Linear Representations in Large Language Models

Yibo Jiang [*1] **Goutham Rajendran** [*2] **Pradeep Ravikumar** [2] **Bryon Aragam** [3] **Victor Veitch** [4 5]

Abstract

Recent works have argued that high-level semantic concepts are encoded "linearly" in the representation space of large language models. In this work, we study the origins of such linear representations. To that end, we introduce a simple latent variable model to abstract and formalize the concept dynamics of the next token prediction. We use this formalism to show that the next token prediction objective (softmax with cross-entropy) and the implicit bias of gradient descent together promote the linear representation of concepts. Experiments show that linear representations emerge when learning from data matching the latent variable model, confirming that this simple structure already suffices to yield linear representations. We additionally confirm some predictions of the theory using the LLaMA-2 large language model, giving evidence that the simplified model yields generalizable insights.

1. Introduction

One of the central questions of interpretability research for language models (Mikolov et al., 2013; OpenAI, 2023; Touvron et al., 2023) is to understand how high-level semantic concepts that are meaningful to humans are encoded in the representations of these models. In this context, a surprising observation is that these concepts are often represented *linearly* (e.g., Mikolov et al., 2013; Pennington et al., 2014; Arora et al., 2016; Elhage et al., 2022; Burns et al., 2022; Tigges et al., 2023; Nanda et al., 2023; Moschella et al., 2022; Park et al., 2023; Li et al., 2023; Gurnee et al., 2023). This observation is mainly empirical, leaving open major questions; most importantly: could the apparent "linearity" be illusory?

In this paper, we study the origins of linear representations in large language models. We introduce a mathematical model for next token prediction in which context sentences and next tokens both reflect latent binary *concept variables*. These latent variables give a formalization of underlying human-interpretable concepts. Using this mathematical model, we prove that these latent concepts are indeed linearly represented in the learned representation space. This result comes in two parts. First, similar to earlier findings on word embeddings (Pennington et al., 2014; Arora et al., 2016; Gittens et al., 2017b; Ethayarajh et al., 2018), we show log-odds matching leads to linear structure. Second, we show that the implicit bias of gradient descent leads to the emergence of linear structure even when this (strong) log-odds matching condition fails. Together, these results provide strong support for the linear representation hypothesis.

There are some noteworthy implications of these results. First, linear representation structure is not specific to the choice of model architecture, but a by-product of how the model learns the conditional probabilities of different contexts and corresponding outputs. Second, the simple latent variable model gives rise to representation behavior such as linearity and orthogonality akin to those observed in LLMs. This suggests it may be a useful tool for further theoretical study in interpretability research.

The development is as follows:

1. In Section 2, we present a latent variable model that abstracts the concept dynamics of LLM inference, allowing us to mathematically analyze LLM representations of concepts.
2. Using this model, we show in Section 3 that the next token prediction objective (softmax with cross-entropy) and the implicit bias of gradient descent together promote concepts to admit linear representations.
3. A surprising fact about LLMs is that Euclidean geometry on representations sometimes reasonably encodes semantics, despite the Euclidean inner product being unidentified by the standard LLM objective (Park et al., 2023). In Section 4, we show that the implicit bias of gradient descent can result in Euclidean structure having a privileged status such that independent concepts are represented almost orthogonally.

[*]Equal contribution [1]Department of Computer Science, University of Chicago [2]Machine Learning Department, Carnegie Mellon University [3]Booth School of Business, University of Chicago [4]Department of Statistics, University of Chicago [5]Data Science Institute, University of Chicago. Correspondence to: Yibo Jiang <yiboj@uchicago.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

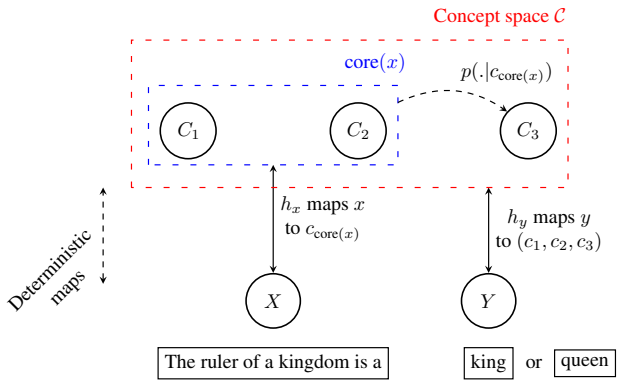


Figure 1. Latent conditional model visualization. An example flow: Suppose X is "The ruler of a kingdom is a". It first maps to a set of core concepts $C_1 = c_1, C_2 = c_2$. This leaves concept $C_3 = \diamond$ unknown, which indicates say "male" or "female". The value of $C_3 = c_3 \in \{0, 1\}$ is determined by a conditional probability $p(.|C_1 = c_1, C_2 = c_2)$ and once it's determined, the entire $c = (c_1, c_2, c_3)$ can be mapped to the next token Y, e.g. "king" (if $c_3 = 0$) or "queen" (if $c_3 = 1$) respectively.

4. Finally, in Section 5, we conduct experiments on simulations from the latent variable model confirming that this structure does indeed yield linear representations. Additionally, we assess the theory's predictions using LLaMA-2 (Touvron et al., 2023), showing that the simple model yields generalizable predictions.

2. Problem setting

In this section, we give a simple latent variable model to study next-token predictions.

2.1. Latent conditional model

We let context sentences and next tokens share the same latent space where the *probabilistic inference* happens. This lets us transform the next token prediction problem into the problem of learning various conditional probabilities among the latent variables. For a visualization of the framework, see Figure 1.

Latent space of concepts We model the latent space with a set of binary random variables: $V_C = \{C_1, ..., C_m\}$. Intuitively, one can view each latent binary random variable as representing the existence of a *concept*, e.g. positive vs negative sentiment, and V_C contains all relevant concepts of interest for language modeling. Notation-wise, we will use $C = (C_1, \ldots, C_m)$ to represent the latent random vector and $c \in \{0, 1\}^m$ to denote realizations of the latent random vector. For a index set $I \subseteq [m]$, C_I (similarly, c_I) denotes a subset of random variables. Note that the latent sample space $\mathcal{C}$ is the collection of binary vectors $\{0, 1\}^m$.

These binary random variables are not necessarily independent. To model their dependencies, we assume that the variables form a Markov random field. That is, there is an undirected graph $G_C = (V_C, E_C)$ such that $p(C_i|C_{[m]\setminus i}) = p(C_i|C_{\text{ne}(i)})$ for all $i \in [m]$ (where $\text{ne}(i)$ denotes the set of neighbors of i), i.e. a variable is conditionally independent of all other variables given its neighbors. Markov random fields capture a wide variety of probability distributions, which include independent random variables as a special case.

Remark 2.1. The theory readily handles directed acyclic graphical models (DAGs) (Koller & Friedman, 2009), but we work with undirected graphs for notational simplicity.

Latent space to next token (y) Let $Y \in \mathcal{Y}$ be the random variable denoting the next token with sample space $\mathcal{Y}$. We assume there exists an injective measurable function from concept space $\mathcal{C}$ to token space $\mathcal{Y}$. That is, the value of C can always be read off from Y. This map induces a distribution on the next token (via the pushforward measure) that the LLM tries to learn. We denote the inverse of this map as h_y (mapping tokens to concepts). The injectivity assumption is made for simplicity—with the injectivity assumption, for any given concept, one can identify pairs of tokens that only differ in that concept. This is useful for the theoretical analysis.

Context sentence (x) to latent space Let $X \in \mathcal{X}$ be the random variable denoting context sentence where $\mathcal{X}$ is the sample space, e.g. "The ruler of a kingdom is a" is an element of $\mathcal{X}$. Intuitively, X contains partial information about the latent concepts that determine the next token Y. This is similar to a hidden Markov model where one can use previous observations to infer the current latent state. In other words, if one models language generation as transitions/drifts in the latent space (Arora et al., 2016), then given the context, one cannot fully determine but can make educated guesses on where the latent space might be transitioned into. Note that there's inherent ambiguity involving the next token prediction, e.g., multiple tokens could follow the sentence "I am a ".

Now, we will describe precisely how contexts X and concepts C relate to each other. First, observe that not every concept combination can induce the context sentence of interest, e.g., not every human-interpretable concept is associated with the sentence "The ruler of a kingdom is a". To capture the notion of concepts relevant to a specific prompt x we define:

$$\mathcal{C}^x = \{c \in \mathcal{C} = \{0, 1\}^m : P(x|C = c) \neq 0\} \tag{1}$$

$$\text{core}(x) = \{i \in [m] : (c_i = 1, \forall c \in \mathcal{C}^x) \text{ or } (c_i = 0, \forall c \in \mathcal{C}^x)\}. \tag{2}$$

Here, $\mathcal{C}^x$ is the set of all realizations of latent concept vectors

in $\mathcal{C} = \{0,1\}^m$ that can induce x. In other words, $\mathcal{C}^x$ contains all concept vectors c that are related to x. Within this subset, $\text{core}(x)$ indicates the concepts that are always "on" or "off". See Figure 1 for an illustration. In other words, given x, the values of "core concepts" are fixed and the rest of the non-core concepts are left to be determined. This represents the determinacy in the relationship between X and C. To capture this determinancy relationship we define the map $h_x : \mathcal{X} \to \{\diamond, 0, 1\}^m$ (where $\diamond$ stands for "unknown" and is to be read as "unknown") as

$$h_x(x)_i = \begin{cases} c_i & \text{if } i \in \text{core}(x) \text{ and } c \in \mathcal{C}^x \\ \diamond & \text{if } i \notin \text{core}(x) \end{cases}$$

where c_i is the known quantity for the core concept of index i. Let's denote $\mathcal{D} = \{\diamond, 0, 1\}^m$, the set of all possible conditioning contexts.

Predictive distribution We make the modeling choice $p(c|x) = p(c|c_{\text{core}(x)})$—the relationship between x to c factorizes via the core concepts. This captures the idea that only the core concepts present in a context sentence are relevant for the next token prediction. So, e.g., the sentence "The ruler of a kingdom is a " has an induced probability of the gender concept. This induced probability is determined by the conditional prob of core concepts (e.g. `is_royalty` $= 1$) of the sentence. In other words, given a context sentence x, the joint posterior distribution of latent concepts outside of "core concepts"—which are fixed by x—is determined by the latent space. Thus, we have

$$p(y|x) = p(c|x) = p(c|c_{\text{core}(x)})$$

where $c = h_y(y)$ and the first equality follows from injectivity.

Notation We summarize our notation here.

- $\mathcal{X}$ – Set of all context sentences
- $\mathcal{Y}$ – Set of all tokens
- $\mathcal{C} = \{0,1\}^m$ – The set of binary concept vectors
- $\mathcal{D} = \{\diamond, 0, 1\}^m$ – The set of context vectors with some unknown concepts
- $\text{core}(x) \subseteq [m]$ – The core concepts in x
- h_x – Deterministic map from X to $\mathcal{D}$
- h_y – Deterministic map from Y to $\mathcal{C}$

2.2. Next token prediction

The goal of the next token prediction is to learn $p(y|x)$. Such probabilities are outputs by autoregressive large language models. In particular, LLMs learn two functions, the embedding $\bar{f} : \mathcal{X} \to \mathbb{R}^d$ and the unembedding $\bar{g} : \mathcal{Y} \to \mathbb{R}^d$ such that

$$p(y|x) \approx \hat{p}(y|x) = \frac{\exp(\bar{f}(x)^T \bar{g}(y))}{\sum_y \exp(\bar{f}(x)^T \bar{g}(y))}$$

Note that under our model, $p(y|x) = p(c|c_{\text{core}(x)})$. Therefore, we can assume $\bar{f}, \bar{g}$ depends on the latents c as well through the functions f, g. In particular, define $\bar{f} = f \circ h_x$ and $\bar{g} = g \circ h_y$, i.e. $\bar{f}(x) = f(h_x(x))$ and $\bar{g}(y) = g(h_y(y))$.

Recall that $\mathcal{C} = \{0,1\}^m$ denotes the collection of all binary concept vectors, and $\mathcal{D} = \{\diamond, 0, 1\}^m$ is the set encompassing all conditions. Then, equivalently, under the latent conditional model, f and g are trained such that their inner product can be used to estimate various conditional probabilities of the following form,

$$p(c|d) \approx \hat{p}(c|d) = \text{softmax}(f(d)^T g(c))$$

for all $c \in \widehat{\mathcal{C}}$ and $d \in \widehat{\mathcal{D}}$ where $\widehat{\mathcal{C}} \subseteq \mathcal{C}$ and $\widehat{\mathcal{D}} \subseteq \mathcal{D}$. $\widehat{\mathcal{C}}$ and $\widehat{\mathcal{D}}$ represent the binary vectors and contexts present in the training dataset. To prove the full slate of our results, unless otherwise stated, we assume $\widehat{\mathcal{C}} = \mathcal{C}$ and $\widehat{\mathcal{D}} = \mathcal{D}$. We will show in Section 5 what happens when this does not hold.

In our analysis, without loss of generality, we represent elements of $\widehat{\mathcal{C}}$ and $\widehat{\mathcal{D}}$ as one-hot encodings. We let f and g assign unique vectors to each element (i.e., the functions are embedding lookups).

3. Linearity

In this section, we study the phenomenon of linear representations, which we now formally define. Adapting (Park et al., 2023), we introduce the following definition within the latent conditional model. Recall that the cone of a vector v is defined as $\text{Cone}(v) = \{\alpha v : \alpha > 0\}$. For a concept $c \in \mathcal{C}$ and $t \in \{0,1\}$, let $c_{(i \to t)}$ indicate the concept c with the ith concept set to t, i.e. the jth concept of $c_{(i \to t)}$ is c_j if $j \neq i$ and t otherwise. A similar notation can be defined for vectors in $\mathcal{D}$ as well. In addition, we refer to vectors that only differ in one latent concept as counterfactual pairs (e.g. $c_{(i \to 1)}, c_{(i \to 0)}$) and the differences between their representations as steering vectors (e.g. $g(c_{(i \to 1)}) - g(c_{(i \to 0)})$).

Definition 3.1 (Linearly encoded representation)**.** A latent concept C_i is said to have *linearly encoded representation in the unembedding space* if there exists a unit vector u such that $g(c_{(i \to 1)}) - g(c_{(i \to 0)}) \in \text{Cone}(u)$ for all $c \in \widehat{\mathcal{C}}$. Similarly, we say that C_i has a *linearly encoded representation in the embedding space* if there exists a unit vector v such that $f(d_{(i \to 1)}) - f(d_{(i \to 0)}) \in \text{Cone}(v)$ for all $d \in \widehat{\mathcal{D}}$. In addition, we say C_i has *a matched*-representation if $u = v$.

We first prove that linearly encoded representations will arise in a subspace as a result of log-odds matching (Section 3.1) which relies on assumptions on the training data distributions. However, empirical observations (Section 5) reveal that, within the latent conditional model, linear representations in both the embedding and unembedding space can emerge without depending on the underlying graphical structures or the true conditional probabilities. This leads us

to establish a connection between the linearity phenomenon and the implicit bias of gradient descent on exponential loss in Section 3.2, which forms the main technical contribution of our work.

3.1. Linearity from log-odds

In this section, we show that if the log odds of the learned probabilities is a constant that depends only on the concept of interest, then we must have linearity of representations for that concept in a subspace. This result is in line with many existing works on word embeddings (Pennington et al., 2014; Arora et al., 2016; Gittens et al., 2017a; Ethayarajh et al., 2018; Allen et al., 2019; Ri et al., 2023). This shows that the latent conditional model is a viable theoretical framework for studying LLMs. Going beyond prior works, our findings additionally connect linearity with the graphical structure of the latent space. An illustrative example is the case of independent concepts, discussed below.

Concretely, the goal of this section is to argue that for any concept $i \in [m]$, the steering vectors in the unembedding space $\Delta_{c,i} = g(c_{(i\to 1)}) - g(c_{(i\to 0)})$ will all be parallel for all $c \in \mathcal{C}$ (up to an ambiguous subspace we cannot control).

Before we state the main theorem, we motivate the assumptions. Firstly, we can reasonably assume the log-odds condition, which states that the model has been trained well enough to have learned correct log-odds for any concept i, under all contexts not conditioning on i. Secondly, since we cannot take the logarithm of 0 conditional probabilities, it's difficult to predict how the steering vector behaves in the directions $f(d)$ for $d_i \neq \diamond$. Therefore, in this section, we project out this ambiguous subspace for now. However, as we will show in Section 3.2, learning these 0 conditional probabilities with gradient descent promotes linear representations overall.

As motivated above, define $\overline{\Delta_{c,i}} = \Pi_i \Delta_{c,i}$ where Π_i is the projection operator onto the space $\text{span}\{f(d) | d_i = \diamond\}$, i.e., we project onto the space of contexts that does not condition on C_i.

Theorem 3.2 (Log-odds implies linearity)**.** *Fix a concept $i \in [m]$. Suppose for any concept vector $c \in \mathcal{C}$ and context $d \in \mathcal{D}$ such that $d_i = \diamond$, we have*

$$\ln \frac{\hat{p}(c_{(i\to 0)}|d)}{\hat{p}(c_{(i\to 1)}|d)} = \ln \frac{p(C_i = 0)}{p(C_i = 1)}$$

Then, the vectors $\overline{\Delta_{c,i}}$ over all $c \in \mathcal{C}$ are parallel.

The proof is deferred to Appendix A. This theorem states that for a fixed concept C_i, all concept steering vectors are parallel to each other, regardless of the other concepts (if we project out the ambiguous subspace). This can help to explain that in the representation space, one can have $\bar{g}$("king") $-$ $\bar{g}$("queen") (approximately) parallel to $\bar{g}$("man") $-$ $\bar{g}$("woman") (Mikolov et al., 2013; Park et al., 2023). In this case, the two pairs ("king", "queen") and ("man", "woman") only differ in the gender concept.

The case of independent concepts We now justify that the assumption in Theorem 3.2 holds when the concepts are jointly independent in the training distributions, therefore the theorem applies to such concepts as a special case. Indeed, in the jointly independent case, the true distribution p is a product distribution, therefore the expression on the left-hand side must match the right-hand side above and the assumption holds.

In summary, Theorem 3.2 applies to the case of jointly independent concepts (and also more generally), implying that matching log-odds formally implies linearity of concept representations in a subspace. In Appendix B, we further generalize this result to the case when the concepts are not necessarily independent and instead come from an MRF (or a DAG), where we show using log-odds that concept representations lie in a subspace of small dimension. However, the assumption of matching log-odds may be restrictive. In order to go beyond, we invoke ideas from the optimization literature on the implicit bias of gradient descent, which we outline in the next section.

3.2. Linearity from implicit bias of gradient descent

The goal of this section is to argue that implicit bias of gradient descent also promotes linearity in the entire space of representations. We have seen in the previous section that log-odds matching can lead to linearity. However, in the context of language modeling, it is stringent to require concepts to have matching log-odds under every possible conditioning. Furthermore, the peculiar case of linearity in the latent conditional model is that the phenomenon can occur even for randomly generated graphs and parameters of the underlying distributions (Section 5). This raises the question of whether there are other factors influencing the representation of concepts.

We now study the role of gradient descent in this phenomenon. However, because the next token prediction is a highly complex nonconvex optimization problem, rather than delving into the entire optimization process, our focus is on identifying *subproblems* within this optimization that can result in linearly encoded representations. The goal here is to highlight the underlying dynamics that prefer linear representations. It is worth noting that we choose to study the role of gradient descent instead of the more computational tractable stochastic gradient descent used in experiments for analytic simplicity. However, according to classical stochastic approximation theory (Kushner & Yin, 2003), with a sufficiently small learning rate, the additional stochasticity

is negligible, and stochastic gradient descent will behave highly similarly to gradient descent (Wu et al., 2020).

The key observation is that there is a hidden binary classification task. Let's consider the simple three-variable example. Suppose instead of predicting all possible conditional distributions of latents, the model only learns $p(\cdot|c_1 = 1)$ and $p(\cdot|c_1 = 0)$. Denoting $v = g(1,1,1) - g(0,1,1), w_0 = f(0,\diamond,\diamond), w_1 = f(1,\diamond,\diamond)$,

$$\frac{\hat{p}(0,1,1|c_1 = 1)}{\hat{p}(1,1,1|c_1 = 1)} = \exp(-w_1^T v) \approx 0,$$
$$\frac{\hat{p}(1,1,1|c_1 = 0)}{\hat{p}(0,1,1|c_1 = 0)} = \exp(w_0^T v) \approx 0$$

Equivalent, the model is trying to optimize this loss:

$$L(w_1, w_0, v) = \sum_{i=0}^{1} \ell(-y_i w_i^T v)$$

where $y_0 = 1, y_1 = -1$ and $\ell(x) = \exp(-x)$. Intuitively, if w_0 and w_1 are fixed, then this is a binary classification task with exponential loss. It is known that gradient descent under this setting would converge to the max-margin solution (Soudry et al., 2018), which makes the direction of v unique.

The following theorem makes this intuition precise. It states that by optimizing a specific subproblem of the next token predictions using gradient descent with *embeddings fixed*, latent unembedding representations will be encoded linearly. For vectors u, v, denote $\cos(u, v) = \frac{\langle u,v \rangle}{||u||\cdot||v||}$.

Theorem 3.3 (Gradient descent with fixed embeddings)**.** *Fix $i \in [m]$. Let $\widehat{\mathcal{D}} = \{d^{\diamond}_{(i\to 1)}, d^{\diamond}_{(i\to 0)}\}$ where $d^{\diamond} = [\diamond, ..., \diamond]$ and $\Delta_{c,i} = g(c_{(i\to 1)}) - g(c_{(i\to 0)})$. Suppose the loss function is the following:*

$$L(\{\Delta_{c,i}\}_c, f(d^{\diamond}_{(i\to 1)}), f(d^{\diamond}_{(i\to 0)})) = \sum_{c\in\overline{\mathcal{C}}} \left(\exp(-\Delta_{c,i}^T f(d^{\diamond}_{(i\to 1)})) + \exp(\Delta_{c,i}^T f(d^{\diamond}_{(i\to 0)}))\right)$$

where for all $c \in \overline{\mathcal{C}}$, $c_i = 1$ and $f(d^{\diamond}_{(i\to 1)})) \neq f(d^{\diamond}_{(i\to 0)})$. Then fixing f and training g using gradient descent with the appropriate step size, we have

$$\lim_{t\to\infty} \cos(\Delta^t_{c^1,i}, \Delta^t_{c^2,i}) = 1$$

for any $c^1, c^2 \in \overline{\mathcal{C}}$ where the superscript t is meant to represent vectors after t number of iterations.

All proofs are deferred to Appendix C. The theorem says that under gradient descent with the exponential loss function and embeddings fixed, the unembedding vector differences for a fixed concept are all aligned in the limit. It is worth mentioning that although the loss function is a bit simplified, it does represent the significant subproblem of the optimization in terms of linear geometry. The reasons are two-fold: (1) Because conditional probabilities are learned with the softmax function, there is an extra degree of freedom. In other words, one does not need to use all the logits to estimate distributions, just the differences between them. (2) Due to the nature of the exponential function, the closer the ratios between conditional distributions are to zero, the larger the distances between unembeddings need to be. Therefore, learning to predict these zero conditional probabilities would dominate the direction of concept representations.

The above theorem assumes that the embedding space is fixed. It turns out that gradient descent will also have the tendency to align embedding and unembedding space when they're not fixed and are trained jointly, as we will show next. First, one can observe that if the 0 conditional probabilities are learned to a reasonable approximation, then the inner product of embedding and unembedding steering vectors of the same concept should be large (Proposition C.1) or in other words, the exponential of the inner product is approaching infinity. But since there are different ways that this inner product can reach infinity, a natural question to ask is what happens when one purely optimizes this exponential learning objective. Theorem C.2 shows that gradient descent, when trained to minimize the exponential of negative inner product of two vectors, would align these two vectors.

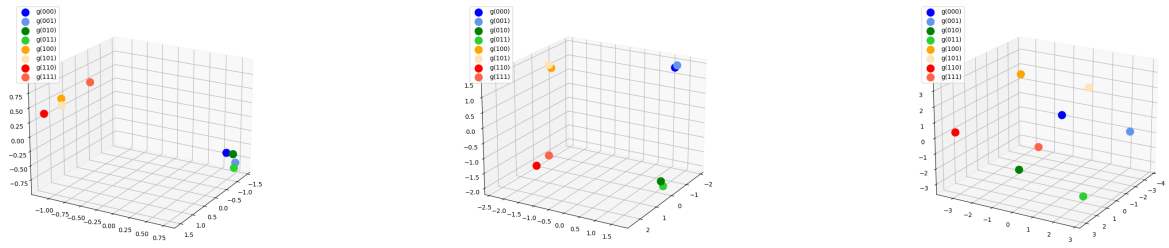
Finally, using Theorem 3.3 and Theorem C.2, we obtain the following statement, which is one of our core contributions.

Theorem 3.4 (Gradient descent aligns representations)**.** *Under the conditions of Theorem 3.3, if one further assumes that $\{\Delta_{c,i}\}_c, f(d^{\diamond}_{(i\to 1)})), f(d^{\diamond}_{(i\to 0)})$ are initialized to be mutually orthogonal with the same norm, then optimizing f and g using gradient descent, we have*

$$\lim_{t\to\infty} \cos(\Delta^t_{c^1,i}, \Delta^t_{c^2,i}) = 1$$
$$\lim_{t\to\infty} \cos(\Delta^t_{c^1,i}, f^t(d^{\diamond}_{(i\to 1)}) - f^t(d^{\diamond}_{(i\to 0)})) = 1$$

for any $c^1, c^2 \in \overline{\mathcal{C}}$ where the superscript t is meant to represent vectors after t number of iterations.

While stated generally as a result of the properties of gradient descent, Theorem 3.4 says that when the latent conditional models are trained on a subproblem of next token prediction with reasonable initializations (see below remark), implicit bias of gradient descent will align the embedding and unembedding vectors as well as aligning among the unembedding vectors. This can manifest as linear representations in the final trained LLMs such as LLaMA-2 (Touvron et al., 2023). Although the theorem addresses only a specific subproblem, it notably suggests, as discussed earlier, the underlying bias and dynamics that contribute to the emergence



(a) $\widehat{\mathcal{D}} = \{(0, \diamond, \diamond), (1, \diamond, \diamond)\}$ (b) $\widehat{\mathcal{D}} = \{(0, 0, \diamond), (0, 1, \diamond), (1, 0, \diamond), (1, 1, \diamond)\}$ (c) $\widehat{\mathcal{D}} = \mathcal{D}$

Figure 2. Unembedding representations form different clusters after training on various conditioning sets. This figure shows unembedding representations after learning different sets of conditional distributions.

of linear representations. That is, if one solely optimizes this subproblem, one can get perfect linear representations. We verify this result empirically in Section 5.

Remark 3.5. The initial condition assumption in the theorem statement seems unnecessary as evidenced by our experiments in Section 5. On the other hand, if one initializes vectors with high dimensional multivariate Gaussians, then the assumption will be approximately satisfied with high probability (Dasgupta, 1999).

The role of learning various conditional distributions So far, we have only discussed learning conditional distributions for a single concept. In general, due to the presence of zero probabilities in various conditioning contexts, learning these conditional probabilities is, in some sense, learning different forces to push sets of representations far apart because one needs relatively large inner products to get exponents to be close to zero.

Figure 2 gives an illustrative example with three concept variables that shows how learning different subsets of conditional probabilities would push various subsets of unembeddings in different ways. For example, if the model exclusively learns conditional probabilities for C_1, the unembedding representations would form clusters based on the values of C_1. Similarly, if the model learns conditional probabilities for both C_1 and C_2, the unembedding representations would cluster based on the values of both C_1 and C_2. However, in the second case, the concept directions of C_1 and C_2 are coupled. As shown in Figure 2(b), $g(110) - g(100)$ is parallel with $g(111) - g(101)$ but not with $g(010) - g(000)$. To decouple them, one needs a bigger $\widehat{\mathcal{D}}$ as in Figure 2(c). That said, it seems that a complete set of $\mathcal{D}$ is not always necessary (Section 5).

An analogy would be to view each binary vector as a combination of "electrons" where the values represent positive or negative charges. Learning probabilities under different conditioning contexts can be likened to applying various forces to manipulate the positions of these electrons. This is similar to the Thomson Problem in chemistry and occurs in other contexts of representation learning as well (Elhage et al., 2022).

4. Orthogonality

A surprising phenomenon observed in LLMs like LLaMA-2 (Touvron et al., 2023) is that the Euclidean geometry can somewhat capture semantic structures, despite the fact that the Euclidean inner product is not identified by the training objective (Park et al., 2023). Consequently, one notable outcome is the tendency for semantically unrelated concepts to be represented as almost orthogonal vectors in the unembedding space. In this section, we study this phenomenon under our latent conditional model. Based on our underlying Markov random field distribution, we define unrelated concepts to be those separated in G_C.

Theorem 4.1. *Let $\widehat{\mathcal{C}} = \mathcal{C}$ and $\widehat{\mathcal{D}} = \mathcal{D}$. Assuming $p(c) > 0$ for any $c \in \mathcal{C}$ and C_i and C_j are two latent variables separated in G_C. Given any binary vector $c \in \mathcal{C}$, there exists a subset $\mathcal{D}_c \subset \mathcal{D}$ such that $d_i = \diamond$ and $p(c|d) > 0$ for any $d \in \mathcal{D}_c$. If one further assume that $\hat{p}(c|d) = p(c|d)$ for any $d \in \mathcal{D}_c$, then*

$$g(c_{(i\to 1)}) - g(c_{(i\to 0)}) \perp f(d_{(j\to c_j)}) - f(d_{(j\to \diamond)})$$

for any $d \in \mathcal{D}_c$.

All proofs are deferred to Appendix D. As suggested in Section 3.2 and verified empirically in Section 5, under the latent conditional model, representations are not only linearly encoded but also frequently aligned between embedding and unembedding spaces, which implies the following corollary on the orthogonal representation of unrelated concepts.

Corollary 4.2. *Under the conditions of Theorem 4.1, if concept C_i and C_j have matched-representations, then*

$$g(c_{(i\to 1)}) - g(c_{(i\to 0)}) \perp g(c_{(j\to 1)}) - g(c_{(j\to 0)})$$

for any $c \in \mathcal{C}$.

In simulation, given enough dimensions, the latent conditional model tends to learn near orthogonal representations

regardless of graphical dependencies. One plausible rationale behind this phenomenon is the tendency for steering vectors in both unembedding and embedding spaces to possess large norms because, by Proposition C.1, the inner product of embedding and unembedding steering vectors of the same concept is large. However, for steering vectors representing distinct concepts, it's often necessary for the inner product to be small as dictated by what's given in the training distributions. To accommodate for this, their cosine similarities tend to be close to zero. As a consequence, unembedding representations of different concepts often end up being near orthogonal as well.

Remark 4.3. It is worth noting that the conditions for orthogonal representations presented here can break in practice for LLMs. For instance, the injectivity assumption might not hold, and embedding and unembedding representations might not have perfectly matched representations.

5. Experiments

In this section, we present two slates of experiments to validate and augment our theoretical contributions. Specifically, in Section 5.1, we run experiments on simulated data from the latent conditional model to verify the existence of linear and orthogonal representations in this theoretical framework and how training on smaller dimensions, incomplete sets of contexts, and concept vectors can affect the representations. Then, we run experiments on large language models in Section 5.2 to show nontrivial alignment between embedding and unembedding representations of the same concept as predicted by our theory.

5.1. Simulated experiments

Simulation setup For simulation experiments, we first create simulated datasets by initially creating random DAGs (see Remark 2.1) with m variables/concepts. For a specific random DAG, the conditional probabilities of one variable given its parents are modeled by Bernoulli distributions where the parameters are sampled uniformly from $[0.3, 0.7]$. For example, suppose the DAG is $C_1 \to C_2$. Then $C_1 \sim \text{Bern}(p_1)$, $(C_2|C_1 = 0) \sim \text{Bern}(p_2)$, $(C_2|C_1 = 1) \sim \text{Bern}(p_3)$ where $p_1, p_2, p_3 \sim \text{Unif}\ ([0.3, 0.7])$. In other words, we generate random graphical models wherein both the structures and distributions are created randomly. From these random graphical models, we can sample values of variables which are binary vectors as our datasets.

To let models learn conditional distributions, we train them to make predictions under cross-entropy loss after randomly masking the sampled binary vectors. Unless otherwise stated, the model is trained using stochastic gradient descent with a learning rate of 0.1 and batch size 100. More details are deferred to Appendix E.

Table 1. When the model is trained to learn the complete set of conditionals, the m latent variables are represented linearly, and the embedding and unembedding representations are matched. The table shows average cosine similarities among and between steering vectors of unembeddings and embeddings. Standard errors are over 100 runs for 3 variables and 4 variables, $50, 20$ and 10 runs for $5, 6$ and 7 variables respectively.

m	UNEMBEDDING	EMBEDDING	UNEMBEDDING AND EMBEDDING
3	0.972±0.006	0.982±0.005	0.980±0.005
4	0.975±0.005	0.971±0.005	0.973±0.005
5	0.988±0.004	0.981±0.004	0.984±0.004
6	0.997±0.000	0.985±0.002	0.990±0.001
7	0.995±0.001	0.972±0.004	0.981±0.003

Complete set of conditionals ($\widehat{\mathcal{C}} = \mathcal{C}, \widehat{\mathcal{D}} = \mathcal{D}$) We first run experiments where the model is trained to learn the complete set of conditional distributions. The representation dimension is set to be the same as the number of latent variables. Given a concept, to measure linearity, we calculate the average cosine similarities among steering vectors in the unembedding space and in the embedding space as well as between steering vectors in these two spaces. The result is shown in Table 1 which suggests that under the latent conditional model with the complete set of contexts, learned representations are indeed linearly encoded. In addition, Figure 5 in Appendix E shows that as loss decreases, the cosine similarities increase rapidly. Due to the exponential number of vectors that need to be trained, one cannot directly simulate a large number of latent variables. However, as we will show later (Appendix E), one can use incomplete sets $\widehat{\mathcal{C}}$ and $\widehat{\mathcal{D}}$ and simulate with more latent variables.

Orthogonality Section 4 argues that, under the latent conditional model, unembedding representations of separated concepts will be orthogonal. In practice, Figure 7(a) (Appendix E) shows that this happens for simulated data even without latent variables being separated in the latent graph. We test the orthogonalities of representations in the latent conditional model by calculating the average cosine similarities between different sets of steering vectors. To see how well the theoretical framework fits LLMs, we similarly calculate the average cosine similarities between different sets of steering vectors in LLaMA-2 (Touvron et al., 2023) using the counterfactual pairs in (Park et al., 2023) and they're reported in Figure 3. In particular, Figure 7 (Appendix E) shows that both simulated experiments and LLaMA-2 experiments exhibit similar behaviors where steering vectors of the same concept are more aligned while steering vectors of different concepts are almost orthogonal. This validates the claim that the latent conditional model is a suitable proxy to study LLMs.

In real-life datasets, not every concept or context vector has a natural correspondence to a token or sentence. On the other hand, the number of tokens and sentences grows exponentially with the number of latent concepts. Therefore, it is not realistic to always have $\widehat{\mathcal{D}} = \mathcal{D}$ or $\widehat{\mathcal{C}} = \mathcal{C}$. In other words, one does not need the next token prediction to learn all possible conditional probabilities perfectly. Because we are in a simulated setup, it's easy to probe the behavior in these situations. Therefore, we also perform additional experiments to observe these aspects of our simulated setup, in particular (i) Training on an incomplete set of contexts ($\widehat{\mathcal{D}} \subset \mathcal{D}$), (ii) Incomplete set of concept vectors ($\widehat{\mathcal{C}} \subset \mathcal{C}$). These are deferred to Appendix E. The experiments show that linearity is robust to these changes.

Moreover, in practice, the representation dimension is typically much smaller than the number of concepts represented. Thus, we run additional experiments in Appendix E with decreasing dimensions and observe one can still get reasonable linear representations.

Finally, we show in Appendix E that gradient descent or stochastic gradient descent is not the only algorithm that can induce linear representation, running other first-order methods like Adam (Kingma & Ba, 2015) can lead to a similar pattern.

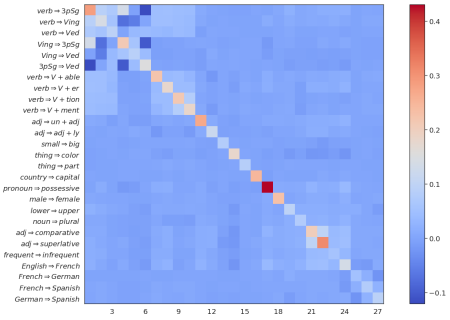


Figure 3. Unembedding steering vectors of the same concept in LLaMA-2 have nontrivial alignment, but steering vectors of different concepts are represented almost orthogonally.

5.2. Experiments with large language models

In this section, we will conduct experiments on pre-trained large language models. We first emphasize that prior works (Elhage et al., 2022; Burns et al., 2022; Tigges et al., 2023; Nanda et al., 2023; Moschella et al., 2022; Park et al., 2023; Gurnee et al., 2023) have already exhibited certain geometries of LLM representations related to linearity and orthogonality, via experiments. Therefore, in this section, we probe the geometry of the LLM representations, in particular that of the interplay between the embeddings and unmbeddings of the contexts and the tokens, that have not been explored in prior works. As (Park et al., 2023) remark, for a given binary concept, it's hard to generate pairs of contexts that differ in precisely this context, partly because of nuances of natural language and mainly because such counterfactual sentences are hard to construct even for human beings. In this section, we try to recreate these sets of missing experiments in the literature using existing open-source datasets.

Multilingual embedding geometry We first consider language translation concepts. For the embedding vectors, we consider pairs of contexts (x^0, x^1) where x^0, x^1 are the same sentences but in different languages. We consider four language pairs French–Spanish, French–German, English–French, and German–Spanish from the OPUS Books dataset (Tiedemann, 2012). We take 150 random samples and filter out contexts with less than 20 or more than 150 tokens. For the unembedding concept vectors, we use the 27 concepts as described in (Park et al., 2023), which were built on top of the Big Analogy Test dataset (Gladkova et al., 2016). Examples of both datasets and a list of the 27 concepts are shown in Appendix F.

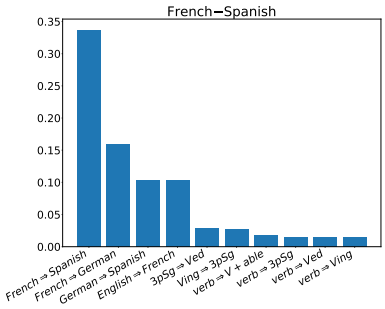


Figure 4. The French–Spanish concept is highly correlated with similar token concepts relative to others. Figure shows cosine similarities between the French–Spanish concept and token concepts.

We consider embeddings and unembeddings from LLaMA-2-7B (Touvron et al., 2023). We compute the absolute cosine similarity between the average embedding steering vector and the average unembedding steering vector. A barplot for French–Spanish with top 10 similarities is shown in Figure 4. The entire barplot with 27 concepts and the barplots for the other language translation concepts are in Appendix F. As we can see, there is alignment between the embedding and unembedding representations for matching concepts relative to unmatched concepts, as Theorem 3.4 predicts.

Winograd Schema Next, we consider counterfactual context pairs arising from the Winograd Schema dataset (Levesque et al., 2012), which is a dataset of pairs of sentences that differ in only one or two words and which further contain an ambiguity that can only be resolved with world knowledge and reasoning. We run experiments similar to the multilingual embedding geometry experiments and also run additional experiments on similarities between Wino-

grad context pairs and the 27 concepts from (Park et al., 2023). The experiment details are in Appendix F.

These LLM experiments show that matching contexts are better aligned with the corresponding unembedding steering vectors than non-matching contexts, as predicted from our theory. Note that although the final cosine similarities are lower than what was exhibited in the simulated experiments, this is not surprising due to the complexity and nuances of natural language and LLMs. Another reason is that abstract concepts do not necessarily lie in a one-dimensional space, so standard cosine similarity metrics may not be an optimal choice here. However, the partial alignment already serves to strongly validate our theoretical insights.

6. Literature review

Linear representations Linear representations have been observed empirically in word embeddings (Mikolov et al., 2013; Pennington et al., 2014; Arora et al., 2016) and large language models (Elhage et al., 2022; Burns et al., 2022; Tigges et al., 2023; Nanda et al., 2023; Moschella et al., 2022; Park et al., 2023; Gurnee et al., 2023). Many works in the pre-LLM era, also attempt to explain this phenomenon theoretically. For instance, Arora et al. (2015) and their follow-up works (Arora et al., 2018; Frandsen & Ge, 2019) study the RAND-WALK model in which latent vectors undergo continuous drift on the unit sphere. A similar notion can be found also in dynamic topic modeling (Blei & Lafferty, 2006) and its subsequent works (Rudolph et al., 2016; Rudolph & Blei, 2017). In contrast, the latent conditional model in this paper studies discrete latent concept variables that can capture semantic meanings.

Other works include the paraphrasing model (Gittens et al., 2017a; Allen & Hospedales, 2019; Allen et al., 2019), which proposes to study a subset of words that are semantically equivalent to a single word, however the uniformity assumption is somewhat unrealistic. Ethayarajh et al. (2018) explore how the linearity property can arise by decomposing pointwise mutual information matrix and Ri et al. (2023) examine the phenomenon from the perspective of contrastive loss which all rely on assumptions on the matching of probability ratios.

Linearity has also been observed in other domains such as computer vision (Radford et al., 2015; Raghu et al., 2017; Bau et al., 2017; Engel et al., 2017; Kim et al., 2018; Trager et al., 2023; Wang et al., 2023) and other intelligent systems (McGrath et al., 2022; Schut et al., 2023). We expect our ideas to extend to other domains, however we restrict our attention to language modeling in this work.

Geometry of representations There's a body of work on studying the geometry of word and sentence representations (Mimno & Thompson, 2017; Reif et al., 2019; Volpi & Malagò, 2021; 2020; Li et al., 2020; Chen et al., 2021; Chang et al., 2022; Liang et al., 2022; Jiang et al., 2023; Park et al., 2023). In particular, Mimno & Thompson (2017) and Liang et al. (2022) discover representation gaps in the context of word embeddings and vision-language models. Park et al. (2023) attempts to define a suitable inner product that captures semantic meanings. Jiang et al. (2023) study the connection between independence and orthogonal representation by adopting the abstract notion of independence models. One can also view embeddings in the context of information geometry (Volpi & Malagò, 2021; 2020).

Causal representation learning There is a subtle connection between our work and the field of causal representation learning that we now briefly comment on. Causal representation learning (Schölkopf et al., 2021; Schölkopf & von Kügelgen, 2022) is an emerging field that exploits ideas from the theory of latent variable modeling (Arora et al., 2013; Kivva et al., 2021; Hyvärinen et al., 2023) along with causality (Spirtes et al., 2000; Pearl, 2009; Rajendran et al., 2021; Squires & Uhler, 2022) to build generative models for various domains. The main goal is to build the true generative models that led to the creation of a dataset. This field has made exciting advances in recent years (Khemakhem et al., 2020; Falck et al., 2021; Zimmermann et al., 2021; Kivva et al., 2022; Lachapelle et al., 2022; Rajendran et al., 2023; Varici et al., 2023; Rajendran et al., 2024; Jiang & Aragam, 2023; Buchholz et al., 2023; Hyvärinen et al., 2023). However, to study large foundation models, a purely statistical notion of building the true model may not reflect the entire story as one should also take into account the implicit bias of optimization as well (Section 3.2). Moreover, identifying the entire underlying latent distributions may also not be necessary for certain practical purposes and it might be sufficient to represent only certain structure information in the geometry of representations, such as linearly encoded representations. Another example of this has been recently explored by (Jiang et al., 2023) namely independence preserving embedding which studies how independence structure can be stored in representations.

7. Conclusion

In this work, we presented a simple latent variable model and showed that using a standard LLM pipeline to learn the distribution results in concepts being linearly represented. We observed that this linear structure is promoted in (at least) two ways—(i) matching log-odds, similar to prior works on word embeddings, and (ii) implicit bias of gradient descent. Additionally, experimental results show that—as predicted—the linear structure emerges from the simple latent variable model. We also saw that, as predicted in the simple model, LLaMA-2 representations exhibit alignment between embedding and unembedding representations.

Acknowledgements This work is supported by ONR grant N00014-23-1-2591, Open Philanthropy, NSF IIS-1956330, NIH R01GM140467, and the Robert H. Topel Faculty Research Fund at the University of Chicago Booth School of Business. We also acknowledge the support of AFRL and DARPA via FA8750-23-2-1015, ONR via N00014-23-1-2368, and NSF via IIS-1909816, IIS-1955532.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Allen, C. and Hospedales, T. Analogies explained: Towards understanding word embeddings. In *International Conference on Machine Learning*, pp. 223–231. PMLR, 2019.

Allen, C., Balazevic, I., and Hospedales, T. What the vec? towards probabilistically grounded embeddings. *Advances in neural information processing systems*, 32, 2019.

Arora, S., Ge, R., Halpern, Y., Mimno, D. M., Moitra, A., Sontag, D. A., Wu, Y., and Zhu, M. A practical algorithm for topic modeling with provable guarantees. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 280–288. JMLR.org, 2013.

Arora, S., Li, Y., Liang, Y., Ma, T., and Risteski, A. Random walks on context spaces: Towards an explanation of the mysteries of semantic word embeddings. *arXiv preprint arXiv:1502.03520*, pp. 385–399, 2015.

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.

Arora, S., Li, Y., Liang, Y., Ma, T., and Risteski, A. Linear algebraic structure of word senses, with applications to polysemy. *Transactions of the Association for Computational Linguistics*, 6:483–495, 2018.

Bau, D., Zhou, B., Khosla, A., Oliva, A., and Torralba, A. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6541–6549, 2017.

Blei, D. M. and Lafferty, J. D. Dynamic topic models. In *Proceedings of the 23rd international conference on Machine learning*, pp. 113–120, 2006.

Buchholz, S., Rajendran, G., Rosenfeld, E., Aragam, B., Schölkopf, B., and Ravikumar, P. Learning linear causal representations from interventions under general nonlinear mixing. *arXiv preprint arXiv:2306.02235*, 2023.

Burns, C., Ye, H., Klein, D., and Steinhardt, J. Discovering latent knowledge in language models without supervision. *arXiv preprint arXiv:2212.03827*, 2022.

Chang, T. A., Tu, Z., and Bergen, B. K. The geometry of multilingual language model representations. *arXiv preprint arXiv:2205.10964*, 2022.

Chen, B., Fu, Y., Xu, G., Xie, P., Tan, C., Chen, M., and Jing, L. Probing bert in hyperbolic spaces. *arXiv preprint arXiv:2104.03869*, 2021.

Dasgupta, S. Learning mixtures of gaussians. In *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*, pp. 634–644. IEEE, 1999.

Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., et al. Toy models of superposition. *arXiv preprint arXiv:2209.10652*, 2022.

Engel, J., Hoffman, M., and Roberts, A. Latent constraints: Learning to generate conditionally from unconditional generative models. *arXiv preprint arXiv:1711.05772*, 2017.

Ethayarajh, K., Duvenaud, D., and Hirst, G. Towards understanding linear word analogies. *arXiv preprint arXiv:1810.04882*, 2018.

Falck, F., Zhang, H., Willetts, M., Nicholson, G., Yau, C., and Holmes, C. C. Multi-facet clustering variational autoencoders. *Advances in Neural Information Processing Systems*, 34, 2021.

Frandsen, A. and Ge, R. Understanding composition of word embeddings via tensor decomposition. *arXiv preprint arXiv:1902.00613*, 2019.

Gittens, A., Achlioptas, D., and Mahoney, M. W. Skip-gram - zipf + uniform = vector additivity. In *Annual Meeting of the Association for Computational Linguistics*, 2017a.

Gittens, A., Achlioptas, D., and Mahoney, M. W. Skip-gram- zipf+ uniform= vector additivity. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 69–76, 2017b.

Gladkova, A., Drozd, A., and Matsuoka, S. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the NAACL Student Research Workshop*, pp. 8–15, 2016.

Gurnee, W., Nanda, N., Pauly, M., Harvey, K., Troitskii, D., and Bertsimas, D. Finding neurons in a haystack: Case studies with sparse probing. *arXiv preprint arXiv:2305.01610*, 2023.

Hyvärinen, A., Khemakhem, I., and Monti, R. Identifiability of latent-variable and structural-equation models: from linear to nonlinear. *arXiv preprint arXiv:2302.02672*, 2023.

Jiang, Y. and Aragam, B. Learning latent causal graphs with unknown interventions. In *Advances in Neural Information Processing Systems*, 2023.

Jiang, Y., Aragam, B., and Veitch, V. Uncovering meanings of embeddings via partial orthogonality. *arXiv preprint arXiv:2310.17611*, 2023.

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pp. 2207–2217. PMLR, 2020.

Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pp. 2668–2677. PMLR, 2018.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

Kivva, B., Rajendran, G., Ravikumar, P., and Aragam, B. Learning latent causal graphs via mixture oracles. In Ranzato, M., Beygelzimer, A., Dauphin, Y. N., Liang, P., and Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pp. 18087–18101, 2021.

Kivva, B., Rajendran, G., Ravikumar, P., and Aragam, B. Identifiability of deep generative models without auxiliary information. *Advances in Neural Information Processing Systems*, 35:15687–15701, 2022.

Koller, D. and Friedman, N. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

Kushner, H. and Yin, G. G. *Stochastic approximation and recursive algorithms and applications*, volume 35. Springer Science & Business Media, 2003.

Lachapelle, S., Rodríguez, P., Sharma, Y., Everett, K., Priol, R. L., Lacoste, A., and Lacoste-Julien, S. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ICA. In Schölkopf, B., Uhler, C., and Zhang, K. (eds.), *1st Conference on Causal Learning and Reasoning, CLeaR 2022, Sequoia Conference Center, Eureka, CA, USA, 11-13 April, 2022*, volume 177 of *Proceedings of Machine Learning Research*, pp. 428–484. PMLR, 2022.

Levesque, H., Davis, E., and Morgenstern, L. The winograd schema challenge. In *Thirteenth International Conference on the Principles of Knowledge Representation and Reasoning*. Citeseer, 2012.

Li, B., Zhou, H., He, J., Wang, M., Yang, Y., and Li, L. On the sentence embeddings from pre-trained language models. *arXiv preprint arXiv:2011.05864*, 2020.

Li, K., Patel, O., Viégas, F., Pfister, H., and Wattenberg, M. Inference-time intervention: Eliciting truthful answers from a language model. *arXiv preprint arXiv:2306.03341*, 2023.

Liang, V. W., Zhang, Y., Kwon, Y., Yeung, S., and Zou, J. Y. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35: 17612–17625, 2022.

McGrath, T., Kapishnikov, A., Tomašev, N., Pearce, A., Wattenberg, M., Hassabis, D., Kim, B., Paquet, U., and Kramnik, V. Acquisition of chess knowledge in alphazero. *Proceedings of the National Academy of Sciences*, 119 (47):e2206625119, 2022.

Mikolov, T., Yih, W.-t., and Zweig, G. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pp. 746–751, 2013.

Mimno, D. and Thompson, L. The strange geometry of skip-gram with negative sampling. In *Conference on Empirical Methods in Natural Language Processing*, 2017.

Moschella, L., Maiorca, V., Fumero, M., Norelli, A., Locatello, F., and Rodola, E. Relative representations enable zero-shot latent space communication. *arXiv preprint arXiv:2209.15430*, 2022.

Nanda, N., Lee, A., and Wattenberg, M. Emergent linear representations in world models of self-supervised sequence models. *arXiv preprint arXiv:2309.00941*, 2023.

OpenAI. GPT-4 technical report, 2023.

Park, K., Choe, Y. J., and Veitch, V. The linear representation hypothesis and the geometry of large language models, 2023.

Pearl, J. *Causality*. Cambridge university press, 2009.

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.

Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Raghu, M., Gilmer, J., Yosinski, J., and Sohl-Dickstein, J. Svcca: Singular vector canonical correlation analysis for deep understanding and improvement. *stat*, 1050:19, 2017.

Rajendran, G., Kivva, B., Gao, M., and Aragam, B. Structure learning in polynomial time: Greedy algorithms, bregman information, and exponential families. *Advances in Neural Information Processing Systems*, 34:18660–18672, 2021.

Rajendran, G., Reizinger, P., Brendel, W., and Ravikumar, P. An interventional perspective on identifiability in gaussian lti systems with independent component analysis. *arXiv preprint arXiv:2311.18048*, 2023.

Rajendran, G., Buchholz, S., Aragam, B., Schölkopf, B., and Ravikumar, P. Learning interpretable concepts: Unifying causal representation learning and foundation models. *arXiv preprint*, 2024.

Reif, E., Yuan, A., Wattenberg, M., Viegas, F. B., Coenen, A., Pearce, A., and Kim, B. Visualizing and measuring the geometry of bert. *Advances in Neural Information Processing Systems*, 32, 2019.

Ri, N., Lee, F.-T., and Verma, N. Contrastive loss is all you need to recover analogies as parallel lines. *arXiv preprint arXiv:2306.08221*, 2023.

Rudolph, M. and Blei, D. Dynamic bernoulli embeddings for language evolution. *arXiv preprint arXiv:1703.08052*, 2017.

Rudolph, M., Ruiz, F., Mandt, S., and Blei, D. Exponential family embeddings. *Advances in Neural Information Processing Systems*, 29, 2016.

Schölkopf, B. and von Kügelgen, J. From statistical to causal learning. *arXiv preprint arXiv:2204.00607*, 2022.

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., and Bengio, Y. Toward causal representation learning. *Proceedings of the IEEE*, 109(5): 612–634, 2021. arXiv:2102.11107.

Schut, L., Tomasev, N., McGrath, T., Hassabis, D., Paquet, U., and Kim, B. Bridging the human-ai knowledge gap: Concept discovery and transfer in alphazero. *arXiv preprint arXiv:2310.16410*, 2023.

Soudry, D., Hoffer, E., Nacson, M. S., Gunasekar, S., and Srebro, N. The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878, 2018.

Spirtes, P., Glymour, C. N., and Scheines, R. *Causation, prediction, and search*. MIT press, 2000.

Squires, C. and Uhler, C. Causal structure learning: a combinatorial perspective. *Foundations of Computational Mathematics*, pp. 1–35, 2022.

Tiedemann, J. Parallel data, tools and interfaces in opus. In Chair), N. C. C., Choukri, K., Declerck, T., Dogan, M. U., Maegaard, B., Mariani, J., Odijk, J., and Piperidis, S. (eds.), *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may 2012. European Language Resources Association (ELRA). ISBN 978-2-9517408-7-7.

Tigges, C., Hollinsworth, O. J., Geiger, A., and Nanda, N. Linear representations of sentiment in large language models. *arXiv preprint arXiv:2310.15154*, 2023.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Trager, M., Perera, P., Zancato, L., Achille, A., Bhatia, P., and Soatto, S. Linear spaces of meanings: compositional structures in vision-language models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15395–15404, 2023.

Varici, B., Acarturk, E., Shanmugam, K., Kumar, A., and Tajer, A. Score-based causal representation learning with interventions. *arXiv preprint arXiv:2301.08230*, 2023.

Volpi, R. and Malagò, L. Evaluating natural alpha embeddings on intrinsic and extrinsic tasks. In *Workshop on Representation Learning for NLP*, 2020.

Volpi, R. and Malagò, L. Natural alpha embeddings. *Information Geometry*, 4(1):3–29, 2021.

Wang, Z., Gui, L., Negrea, J., and Veitch, V. Concept algebra for score-based conditional model. In *ICML 2023*

Workshop on Structured Probabilistic Inference {\&} Generative Modeling, 2023.

Wu, J., Zou, D., Braverman, V., and Gu, Q. Direction matters: On the implicit bias of stochastic gradient descent with moderate learning rate. *arXiv preprint arXiv:2011.02538*, 2020.

Zimmermann, R. S., Sharma, Y., Schneider, S., Bethge, M., and Brendel, W. Contrastive learning inverts the data generating process. In *International Conference on Machine Learning*, pp. 12979–12990. PMLR, 2021.

A. Linearity from log-odds: Proof of Theorem 3.2

In this section, we prove Theorem 3.2 restated below for convenience.

Theorem 3.2 (Log-odds implies linearity)**.** *Fix a concept $i \in [m]$. Suppose for any concept vector $c \in \mathcal{C}$ and context $d \in \mathcal{D}$ such that $d_i = \diamond$, we have*

$$\ln \frac{\hat{p}(c_{(i\to 0)}|d)}{\hat{p}(c_{(i\to 1)}|d)} = \ln \frac{p(C_i = 0)}{p(C_i = 1)}$$

Then, the vectors $\overline{\Delta_{c,i}}$ over all $c \in \mathcal{C}$ are parallel.

Proof. Fix any $i \leq m$ and consider the vector $\overline{\Delta_{c,i}} = \Pi_i(g(c_{(i\to 0)}) - g(c_{(i\to 1)}))$ for any $c \in \mathcal{C}$. Let $\widehat{\mathcal{D}} = \{d \in \{\diamond, 0, 1\}^m | d_i = \diamond\}$ be the contexts that do not condition on C_i and let $u_1, \ldots, u_k$ be an orthonormal basis of the space $\text{span}\{f(d)|d \in \widehat{\mathcal{D}}\}$. Also, since $f(d)$ over $d \in \widehat{\mathcal{D}}$ span this space, let

$$u_j = \sum_{d \in \widehat{\mathcal{D}}} \alpha_{j,d} f(d)$$

for all $j = 1, 2, \ldots, k$ and scalars $\alpha_{j,d}$. Note that the $\alpha_{j,d}$ do not depend on c. Now, for any $j \leq k$,

$$\begin{aligned}
\langle \overline{\Delta_{c,i}}, u_j \rangle &= \langle \Pi_i(g(c_{(i\to 0)}) - g(c_{(i\to 1)})), u_j \rangle \\
&= \langle g(c_{(i\to 0)}) - g(c_{(i\to 1)}), \Pi_i u_j \rangle \\
&= \langle g(c_{(i\to 0)}) - g(c_{(i\to 1)}), u_j \rangle \\
&= \langle g(c_{(i\to 0)}) - g(c_{(i\to 1)}), \sum_{d \in \mathcal{D}} \alpha_{j,d} f(d) \rangle \\
&= \sum_{d \in \widehat{\mathcal{D}}} \alpha_{j,d} \langle g(c_{(i\to 0)}) - g(c_{(i\to 1)}), f(d) \rangle
\end{aligned}$$

To this end, we will compute the inner product $\langle g(c_{(i\to 0)}) - g(c_{(i\to 1)}), f(d) \rangle$ for a fixed $d \in \widehat{\mathcal{D}}$. First,

$$\ln \frac{\hat{p}(c_{(i\to 0)}|d)}{\hat{p}(c_{(i\to 1)}|d)} = \ln \frac{p(C_i = 0)}{p(C_i = 1)}$$

But we have

$$\hat{p}(c_{(i\to 0)}|d) = \frac{\exp(f(d)^T g(c_{(i\to 0)}))}{\sum_{c'} \exp(f(d)^T g(c'))}, \qquad \hat{p}(c_{(i\to 1)}|d) = \frac{\exp(f(d)^T g(c_{(i\to 1)}))}{\sum_{c'} \exp(f(d)^T g(c'))}$$

Since the denominator does not depend on c, rearranging implies

$$\begin{aligned}
\langle g(c_{(i\to 0)}) - g(c_{(i\to 1)}), f(d) \rangle &= f(d)^T (g(c_{(i\to 1)}) - g(c_{(i\to 0)})) \\
&= \ln \frac{\hat{p}(c_{(i\to 0)}|d)}{\hat{p}(c_{(i\to 1)}|d)} \\
&= \ln \frac{p(C_i = 0)}{p(C_i = 1)}
\end{aligned}$$

which only depends on i. Call this expression $\alpha^{(i)}$ to get

$$\begin{aligned}
\langle \overline{\Delta_{c,i}}, u_j \rangle &= \sum_{d \in \widehat{\mathcal{D}}} \alpha_{j,d} \langle g(c_{(i\to 0)}) - g(c_{(i\to 1)}), f(d) \rangle \\
&= \sum_{d \in \widehat{\mathcal{D}}} \alpha_{j,d} \alpha^{(i)}
\end{aligned}$$

Therefore,

$$\begin{aligned}\overline{\Delta_{c,i}} &= \sum_{j\leq k}\langle\overline{\Delta_{c,i}}, u_j\rangle u_j \\ &= \sum_{j\leq k}\left(\sum_{d\in\widehat{\mathcal{D}}}\alpha_{j,d}\alpha^{(i)}\right)u_j \\ &= \alpha^{(i)}\sum_{j\leq k, d\in\widehat{\mathcal{D}}}\alpha_{j,d}u_j\end{aligned}$$

Note that regardless of c, the final expression is always parallel to the vector $v_i = \sum_{j\leq k, d\in\widehat{\mathcal{D}}}\alpha_{j,d}u_j$ which does not depend on c. This completes the proof. □

B. Linearity from log-odds for general MRFs

In this section, we generalize the ideas from Appendix A to concepts from a general Markov random field, which is more general than the independent case. Since most of the technical ideas and motivations are in Appendix A, we will go over them lightly here. The goal is to study the structure of the steering vector $\Delta_{c,i} = g(c_{(i\to 1)}) - g(c_{(i\to 0)})$. As we will see, instead of them all lying in a space of dimension 1 as in the independent case (that's what being parallel means), they will now live in a subspace of low dimension. For example, such a phenomenon was experimentally observed by (Li et al., 2023) for the concept of *truthfulness*.

Here, concepts $C_1, \ldots, C_m$ come from a Markov random field with undirected graph $G_C = (V_C, E_C)$ with neighborhood set given by $\text{ne}(i)$. Therefore, they satisfy the property that $p(C_i|C_{[m]\setminus i}) = p(C_i|C_{\text{ne}(i)})$ for all $i \leq m$. Accordingly, we state the log-odds assumption to capture this general conditional independence.

Assumption B.1. For any concept $i \leq m$, any concept vector $c \in \mathcal{C}$ and context $d \in \{\diamond, 0, 1\}^m$ such that $d_i = \diamond$ and $d_j \neq \diamond$ for all $j \in \text{ne}(i)$, we have

$$\ln\frac{\hat{p}(c_{(i\to 0)}|d)}{\hat{p}(c_{(i\to 1)}|d)} = \ln\frac{p(C_i = 0|C_{\text{ne}(i)} = d_{\text{ne}(i)})}{p(C_i = 1|C_{\text{ne}(i)} = d_{\text{ne}(i)})}$$

As before, we assume above that the log odds condition holds when i is not conditioned on, but we weaken this further to say that it only needs to hold specifically when every one of its neighbors j has been conditioned on.

Analogously, we project out the space that could possibly contribute to a 0 conditional probability. This is the space where either d_i has been conditioned on or d_j for some $j \in \text{ne}(i)$ has been conditioned on. Therefore, define $\overline{\Delta_{c,i}} = \Pi_i\Delta_{c,i}$ where Π_i is the projection into the space $\text{span}\{f(d)|d_i = \diamond, d_j \neq \diamond \forall j \in \text{ne}(i)\}$ We now state our main theorem.

Theorem B.2. *Under Assumption B.1, for any fixed $i \leq m$, the vectors $\overline{\Delta_{c,i}}$ for all $c \in \mathcal{C}$ live in a subspace $\mathcal{S}$ of dimension at most $2^{|ne(i)|}$.*

This theorem says that if we assume that the concepts come from a general Markov random field, then the steering vectors live in a space of low dimension. Note that when the concepts are independent, we have $|\text{ne}(i)| = 0 \implies 2^{|\text{ne}(i)|} = 1$ which means all the vectors $\overline{\Delta_{c,i}}$ are parallel. Therefore this theorem is more general than Theorem 3.2.

Proof. We will proceed similarly to the proof of Theorem 3.2. Fix any $i \leq m, c \in \mathcal{C}$ and repeat the computation until we get that for any $j \leq k$,

$$\langle\overline{\Delta_{c,i}}, u_j\rangle = \sum_{d\in\widehat{\mathcal{D}}}\alpha_{j,d}\langle g(c_{(i\to 0)}) - g(c_{(i\to 1)}), f(d)\rangle$$

where $\widehat{\mathcal{D}} = \{d \in \{\diamond, 0, 1\}^m | d_i = \diamond, d_j \neq \diamond \forall j \in \text{ne}(i)\}$. Now, let's recompute $\langle g(c_{(i\to 0)}) - g(c_{(i\to 1)}), f(d)\rangle$. By Assumption B.1, we have

$$\ln\frac{\hat{p}(c_{(i\to 0)}|d)}{\hat{p}(c_{(i\to 1)}|d)} = \ln\frac{p(C_i = 0|C_{\text{ne}(i)} = d_{\text{ne}(i)})}{p(C_i = 1|C_{\text{ne}(i)} = d_{\text{ne}(i)})}$$

which gives

$$\langle g(c_{(i\to 0)}) - g(c_{(i\to 1)}), f(d)\rangle = \ln \frac{p(C_i = 0|C_{\text{ne}(i)} = d_{\text{ne}(i)})}{p(C_i = 1|C_{\text{ne}(i)} = d_{\text{ne}(i)})}$$

which depends both on i and $d_{\text{ne}(i)}$. For all $\sigma \in \{0,1\}^{|\text{ne}(i)|}$, denote $\alpha^{(i),\sigma}$ to be the expression

$$\alpha^{(i),\sigma} = \ln \frac{p(C_i = 0|C_{\text{ne}(i)} = \sigma)}{p(C_i = 1|C_{\text{ne}(i)} = \sigma)}$$

Therefore,

$$\begin{aligned}\overline{\Delta_{c,i}} &= \sum_{j\le k}\left(\sum_{d\in\widehat{\mathcal{D}}} \alpha_{j,d}\alpha^{(i),d_{\text{ne}(i)}}\right) u_j \\ &= \sum_{\sigma\in\{0,1\}^{|\text{ne}(i)|}} \alpha^{(i),\sigma}\left(\sum_{j\le k, d\in\widehat{\mathcal{D}}, d_{\text{ne}(i)}=\sigma} \alpha_{j,d}u_j\right)\end{aligned}$$

which lives in the span of the vectors $v^{(i),\sigma} = \sum_{j\le k, d\in\widehat{\mathcal{D}}, d_{\text{ne}(i)}=\sigma} \alpha_{j,d}u_j$ regardless of c. The number of such vectors is $|\{0,1\}^{|\text{ne}(i)|}| = 2^{|\text{ne}(i)|}$. □

C. Linearity from the implicit bias of gradient descent

In this section, we will prove Theorem 3.3 and Theorem 3.4 and additional auxiliary theorems.

Theorem 3.3 (Gradient descent with fixed embeddings)**.** *Fix $i \in [m]$. Let $\widehat{\mathcal{D}} = \{d^{\diamond}_{(i\to 1)}, d^{\diamond}_{(i\to 0)}\}$ where $d^{\diamond} = [\diamond, ..., \diamond]$ and $\Delta_{c,i} = g(c_{(i\to 1)}) - g(c_{(i\to 0)})$. Suppose the loss function is the following:*

$$\begin{aligned}&L(\{\Delta_{c,i}\}_c, f(d^{\diamond}_{(i\to 1)}), f(d^{\diamond}_{(i\to 0)})) = \\ &\sum_{c\in\overline{\mathcal{C}}}\left(\exp(-\Delta_{c,i}^T f(d^{\diamond}_{(i\to 1)})) + \exp(\Delta_{c,i}^T f(d^{\diamond}_{(i\to 0)}))\right)\end{aligned}$$

where for all $c \in \overline{\mathcal{C}}$, $c_i = 1$ and $f(d^{\diamond}_{(i\to 1)})) \neq f(d^{\diamond}_{(i\to 0)})$. Then fixing f and training g using gradient descent with the appropriate step size, we have

$$\lim_{t\to\infty} \cos(\Delta^t_{c^1,i}, \Delta^t_{c^2,i}) = 1$$

for any $c^1, c^2 \in \overline{\mathcal{C}}$ where the superscript t is meant to represent vectors after t number of iterations.

Proof. First note that we can break the whole optimization problem into smaller subproblems where each subproblem only depends on one counterfactual pair. By Theorem 3 of (Soudry et al., 2018), all $\Delta_{c,i}$ converges to have the same direction as the hard margin SVM solution. □

Proposition C.1. *Suppose $p(c) > 0$ for any $c \in \widehat{\mathcal{C}} = \mathcal{C}$ and $|\hat{p}(c|d) - p(c|d)| < \epsilon$ for all $c \in \widehat{\mathcal{C}}$ and $d \in \widehat{\mathcal{D}}$ such that $0 < \epsilon < p(c|d)$ for all c, d where $p(c|d) > 0$. Then for any latent variable C_i, we have that*

$$\exp(-(g(c_{(i\to 1)}) - g(c_{(i\to 0)}))^T(f(d_{(i\to 0)}) - f(d_{(i\to 1)}))) < \frac{\epsilon^2}{\big(p(c_{c_i=0}|d_{c_i=0}) - \epsilon\big)\big(p(c_{c_i=1}|d_{c_i=1}) - \epsilon\big)}$$

for any $c \in \widehat{\mathcal{C}}$ and any $d \in \widehat{\mathcal{D}}$.

Proof. For any $c \in \widehat{\mathcal{C}}$ and $d \in \widehat{\mathcal{D}}$, we have that

$$\frac{p(c_{c_i=1}|d_{c_i=0})}{p(c_{c_i=0}|d_{c_i=0})} = 0, \qquad \frac{p(c_{c_i=0}|d_{c_i=1})}{p(c_{c_i=1}|d_{c_i=1})} = 0$$

These ratios are well-defined because $p(c) > 0$ for any c. Therefore,

$$\frac{\hat{p}(c_{c_i=1}|d_{c_i=0})}{\hat{p}(c_{c_i=0}|d_{c_i=0})} < \frac{\epsilon}{p(c_{c_i=0}|d_{c_i=0}) - \epsilon}, \qquad \frac{\hat{p}(c_{c_i=0}|d_{c_i=1})}{\hat{p}(c_{c_i=1}|d_{c_i=1})} < \frac{\epsilon}{p(c_{c_i=1}|d_{c_i=1}) - \epsilon}$$

Thus,

$$\begin{aligned}
&\exp(-(g(c_{(c_i=1)}) - g(c_{(c_i=0)}))^T (f(d_{(c_i=0)}) - f(d_{(c_i=1)}))) \\
&\qquad = \frac{\hat{p}(c_{c_i=1}|d_{c_i=0})}{\hat{p}(c_{c_i=0}|d_{c_i=0})} \frac{\hat{p}(c_{c_i=0}|d_{c_i=1})}{\hat{p}(c_{c_i=1}|d_{c_i=1})} < \frac{\epsilon^2}{\big(p(c_{c_i=0}|d_{c_i=0}) - \epsilon\big)\big(p(c_{c_i=1}|d_{c_i=1}) - \epsilon\big)}
\end{aligned}$$

□

Theorem C.2. *Given loss function $L(u, v) = \exp(-u^T v)$, any starting point u_0, v_0 where $u_0 \neq -\alpha v_0$ for some $\alpha > 0$, and any step size $\eta < \frac{1}{L(u_0,v_0)}$, the gradient descent iterates will have the following properties:*

(a) $\lim_{t\to\infty} L(t) = \lim_{t\to\infty} L(u_t, v_t) = 0$

(b) $\lim_{t\to\infty} ||u_t|| \to \infty$ *and* $\lim_{t\to\infty} ||v_t|| \to \infty$

(c) $\cos(u_t, v_t)$ *increases monotonically with* t

(d) $\lim_{t\to\infty} \cos(u_t, v_t) = 1$

Proof. Before proving the theorem, let's write out a few equations. By the gradient descent algorithm, we have the following equations:

$$\begin{aligned}
u_{t+1} &= u_t + \eta L(t) v_t \\
v_{t+1} &= v_t + \eta L(t) u_t
\end{aligned}$$

Thus,

$$\begin{aligned}
||u_{t+1}||^2 &= ||u_t||^2 + 2\eta L(t)\langle u_t, v_t\rangle + \eta^2 L(t)^2 ||v_t||^2 \\
||v_{t+1}||^2 &= ||v_t||^2 + 2\eta L(t)\langle u_t, v_t\rangle + \eta^2 L(t)^2 ||u_t||^2 \\
\langle u_{t+1}, v_{t+1}\rangle &= (1 + \eta^2 L(t)^2)\langle u_t, v_t\rangle + \eta L(t)(||u_t||^2 + ||v_t||^2)
\end{aligned}$$

Now let's prove each claim one by one.

First of all, we know that

$$\langle u_{t+1}, v_{t+1}\rangle - \langle u_t, v_t\rangle = \eta^2 L(t)^2 \langle u_t, v_t\rangle + \eta L(t)(||u_t||^2 + ||v_t||^2)$$

The difference is positive if $\langle u_t, v_t\rangle > 0$. To deal with the case of a negative inner product, we will use induction to prove that for any t, $\langle u_{t+1}, v_{t+1}\rangle - \langle u_t, v_t\rangle$ is positive and thus $L(t)$ decreases monotonically.

Base case ($t = 0$): The difference is positive if $\langle u_0, v_0\rangle > 0$. Let's consider the case when $\langle u_0, v_0\rangle \leq 0$

$$\begin{aligned}
\langle u_1, v_1\rangle - \langle u_0, v_0\rangle &= \eta^2 L(0)^2 \langle u_0, v_0\rangle + \eta L(0)(||u_0||^2 + ||v_0||^2) \\
&= (\eta^2 L(0)^2 - 2\eta L(0))\langle u_0, v_0\rangle + \eta L(0)(u_0 + v_0)^2 \\
&= \eta L(0)(\eta L(0) - 2)\langle u_0, v_0\rangle + \eta L(0)(u_0 + v_0)^2 > 0
\end{aligned}$$

The last inequality is due to $\eta < \frac{1}{L(u_0,v_0)}$.

Inductive step : Again, the difference is positive if $\langle u_t, v_t\rangle > 0$. Let's consider the case when $\langle u_t, v_t\rangle \leq 0$

$$\begin{aligned}
\langle u_{t+1}, v_{t+1}\rangle - \langle u_t, v_t\rangle &= \eta^2 L(t)^2 \langle u_t, v_t\rangle + \eta L(t)(||u_t||^2 + ||v_t||^2) \\
&= (\eta^2 L(t)^2 - 2\eta L(t))\langle u_t, v_t\rangle + \eta L(t)(u_t + v_t)^2 \\
&= \eta L(t)(\eta L(t) - 2)\langle u_t, v_t\rangle + \eta L(t)(u_t + v_t)^2 > 0
\end{aligned}$$

The last inequality is due to $\eta < \frac{1}{L(u_0,v_0)}$ and the inductive hypothesis that $L(t)$ decreases monotonically.

It is worth noting because $\langle u_t, v_t \rangle$ is monotonically increasing, there must exist a time t_p such that $\langle u_{t_p}, v_{t_p} \rangle > 0$. If the opposite is true, then $\langle u_t, v_t \rangle$ must converge to a nonpositive number which is not possible because the difference between consecutive numbers in the sequence is strictly positive unless both u_t and v_t converges to zero. In other words, if $\langle u_t, v_t \rangle$ does not diverge, it must be a Cauchy sequence which needs both u_t and v_t to reach 0. Suppose $u_t \to 0$ and $v_t \to 0$. We know that

$$u_{t+1} + v_{t+1} = (1 + \eta L(t))(u_t + v_t) \tag{3}$$

which means the only possible scenario that $u_t \to 0$ and $v_t \to 0$ is when $u_0 + v_0 = 0$ which is excluded by the assumption on initial u_0 and v_0.

Because $L(t) > 0$, we know that $\lim_{t\to\infty} L(t)$ has a limit. Suppose the limit is some constant $c_1 \neq 0$. Then, we have

$$\begin{aligned}
\langle u_T, v_T \rangle - \langle u_{t_p}, v_{t_p} \rangle &= \sum_{t=t_p}^{T-1} \big(\langle u_{t+1}, v_{t+1} \rangle - \langle u_t, v_t \rangle \big) \\
&= \sum_{t=t_p}^{T-1} \eta^2 L(t)^2 \langle u_t, v_t \rangle + \sum_{t=t_p}^{T-1} \eta L(t)(||u_t||^2 + ||v_t||^2) \\
&\geq \sum_{t=t_p}^{T-1} \eta^2 c_1^2 \langle u_{t_p}, v_{t_p} \rangle \\
&> \sum_{t=t_p}^{T-1} C
\end{aligned}$$

for some constant C. This would imply $\langle u_T, v_T \rangle \to \infty$ which contradicts that $\lim_{t\to\infty} L(t) > 0$. Therefore, $\lim_{t\to\infty} L(t) = 0$.

For the second property, we already know at least one of $||u_t||$ and $||v_t||$ will converge to infinity for the loss to converge to zero. Suppose one of them converges to a constant. Without loss of generality, let's assume $\lim_{t\to\infty} ||u_t|| \to \infty$ and for all t, $||v_t|| \leq C_v$ for some constant C_v. This implies that $\lim_{t\to\infty} \frac{||v_t||}{||u_t||} \to 0$. On the other hand, let's consider the following equation of $q_t = \frac{||v_t||}{||u_t||}$ for $t \geq t_p$:

$$q_{t+1}^2 = \frac{||v_{t+1}||^2}{||u_{t+1}||^2} = \frac{||v_t||^2 + 2\eta L(t)\langle u_t, v_t \rangle + \eta^2 L(t)^2 ||u_t||^2}{||u_t||^2 + 2\eta L(t)\langle u_t, v_t \rangle + \eta^2 L(t)^2 ||v_t||^2}$$

If $q_t^2 < 1$, then

$$\frac{2\eta L(t)\langle u_t, v_t \rangle + \eta^2 L(t)^2 ||u_t||^2}{2\eta L(t)\langle u_t, v_t \rangle + \eta^2 L(t)^2 ||v_t||^2} > 1$$

Therefore, if $q_t^2 < 1$, then $q_{t+1}^2 > q_t^2$. Similarly, if $q_t^2 > 1$, then $q_{t+1}^2 < q_t^2$. As a result, q_t will not converge to zero, which is a contradiction.

For the third property, let's consider this equation.

$$\begin{aligned}
\cos(u_{t+1}, v_{t+1})^2 &= \frac{(\langle u_{t+1}, v_{t+1} \rangle)^2}{||u_{t+1}||^2 ||v_{t+1}||^2} \\
&= \frac{(\langle u_{t+1}, v_{t+1} \rangle)^2}{(||u_t||^2 + 2\eta L(t)\langle u_t, v_t \rangle + \eta^2 L(t)^2 ||v_t||^2)(||v_t||^2 + 2\eta L(t)\langle u_t, v_t \rangle + \eta^2 L(t)^2 ||u_t||^2)}
\end{aligned}$$

where

$$\begin{aligned}
(\langle u_{t+1}, v_{t+1} \rangle)^2 = (\langle u_t, v_t \rangle)^2 &+ (2\eta^2 L(t)^2 + \eta^4 L(t)^4)(\langle u_t, v_t \rangle)^2 + \eta^2 L(t)^2 (||u_t||^4 + ||v_t||^4) \\
&+ 2(1 + \eta^2 L(t)^2)(\eta L(t))\langle u_t, v_t \rangle (||u_t||^2 + ||v_t||^2) + 2\eta^2 L(t)^2 ||u_t||^2 ||v_t||^2
\end{aligned}$$

Therefore,

$$\cos(u_{t+1}, v_{t+1})^2 = \frac{(\langle u_t, v_t\rangle)^2 + \Delta X}{(||u_t||^2 + \Delta Y)(||v_t||^2 + \Delta Z)}$$

where

$$\begin{aligned}
\Delta X &= (2\eta^2 L(t)^2 + \eta^4 L(t)^4)(\langle u_t, v_t\rangle)^2 + \eta^2 L(t)^2(||u_t||^4 + ||v_t||^4) \\
&\quad + 2(1 + \eta^2 L(t)^2)(\eta L(t))\langle u_t, v_t\rangle(||u_t||^2 + ||v_t||^2) + 2\eta^2 L(t)^2 ||u_t||^2 ||v_t||^2 \\
\Delta Y &= 2\eta L(t)\langle u_t, v_t\rangle + \eta^2 L(t)^2 ||v_t||^2 \\
\Delta Z &= 2\eta L(t)\langle u_t, v_t\rangle + \eta^2 L(t)^2 ||u_t||^2
\end{aligned}$$

then,

$$\begin{aligned}
\Delta X - (\Delta Y)||v_t||^2 - (\Delta Z)||u_t||^2 - (\Delta Y)(\Delta Z) \\
&= (\eta^4 L(t)^4 - 2\eta^2 L(t)^2)(\langle u_t, v_t\rangle)^2 - ||u_t||^2||v_t||^2) \\
&= (2\eta^2 L(t)^2 - \eta^4 L(t)^4)||u_t||^2||v_t||^2(1 - \cos(u_t, v_t)^2)
\end{aligned}$$

This is zero if and only if $\cos(u_t, v_t)^2 = 1$. Otherwise, it is strictly positive. Therefore, if $t \geq t_p$, by Lemma C.3, $\cos(u_t, v_t)$ increases monotonically.

Before studying the case of $t < t_p$, let's first consider this difference.

$$\begin{aligned}
\Delta_t &= ||u_t||^2||v_t||^2 - (\langle u_v, v_t\rangle)^2 \\
\Delta_{t+1} - \Delta_t &= (\eta^4 L(t)^4 - 2\eta^2 L(t)^2)||u_t||^2||v_t||^2(1 - \cos(u_t, v_t)^2) < 0
\end{aligned}$$

Therefore, when $t < t_p$, because $\langle u_v, v_t\rangle$ increases but $\langle u_v, v_t\rangle < 0$, $(\langle u_v, v_t\rangle)^2$ decreases. Since Δ_t decreases, $||u_t||^2||v_t||^2$ need to decrease as well. Therefore, $cos(u_t, v_t)$ increases when $t < t_p$.

In fact, because $\Delta_t \geq 0$, Δ_t must have a limit. In particular, this would also imply that Δ_t has an upper bound.

Finally, we can prove the last property. Because $\cos(u_t, v_t)$ increases monotonically and $\cos(u_t, v_t) \leq 1$, it must have a limit. Note that the limit does not have to be 1 for the loss to converge to 0. The key observation is that by (3), the direction of $u_t + v_t$ is always the same. In fact, it is reasonable to guess that both u_t and v_t will converge in that direction. Let's project u_{t+1} onto the orthogonal complement of span$(u_t + v_t)$. And we want to show that the projected vector is bounded. Therefore, we only need to consider $t > t_p$.

$$||u_{t+1}^{\perp}||^2 = ||u_{t+1}||^2 - ||\frac{\langle u_{t+1}, u_t + v_t\rangle}{||u_t + v_t||}||^2$$

then,

$$||u_{t+1}^{\perp}||^2 = \frac{(1 - \eta L(t))^2(||u_t||^2||v_t||^2 - (\langle u_v, v_t\rangle)^2)}{||u_t + v_t||^2} \leq \frac{(1 - \eta L(t))^2(||u_t||^2||v_t||^2 - (\langle u_v, v_t\rangle)^2)}{||u_{t_p} + v_{t_p}||^2}$$

We already know that the numerator is bounded. Therefore, $||u_{t+1}^{\perp}||$ is also bounded. Similarly, $||v_{t+1}^{\perp}||$ is also bounded.

We know that both $||u_t||$ and $||v_t||$ diverge to infinity, thus $\lim_{t\to\infty} \cos(u_t, v_t + u_t) = 1$ and $\lim_{t\to\infty} \cos(v_t, v_t + u_t) = 1$. Because of the following well-known inequality,

$$\begin{aligned}
\cos(u_t, v_t + u_t)\cos(v_t, v_t + u_t) - \sqrt{1 - \cos(u_t, v_t + u_t)^2}\sqrt{1 - \cos(v_t, v_t + u_t)^2} \\
&\leq \cos(u_t, v_t) \\
&\leq \cos(u_t, v_t + u_t)\cos(v_t, v_t + u_t) + \sqrt{1 - \cos(u_t, v_t + u_t)^2}\sqrt{1 - \cos(v_t, v_t + u_t)^2}
\end{aligned}$$

we have $\lim_{t\to\infty} \cos(u_t, v_t) = 1$ □

Lemma C.3. *Let $X, \Delta X, Y, \Delta Y, Z, \Delta Z > 0$ where $YZ \neq 0$ and $(Y + \Delta Y)(Z + \Delta Z) \neq 0$, if*

$$\Delta X > (\Delta Y)Z + (\Delta Z)Y + (\Delta Y)(\Delta Z), \quad \frac{X}{YZ} < 1$$

then,

$$\frac{X+\Delta X}{(Y+\Delta Y)(Z+\Delta Z)} > \frac{X}{YZ}$$

Proof. The proof is straightforward. □

Finally, we will prove our main Theorem 3.4, which is equivalent to the following theorem with simplified notation.

Theorem C.4. *Given loss function*

$$L(u^{(1)}, u^{(2)}, \{v^{(i)}\}_{i=1}^K) = \sum_{i=1}^K \big(\exp(-\langle u^{(1)}, v^{(i)}\rangle) + \exp(\langle u^{(2)}, v^{(i)}\rangle)\big)$$

Suppose at starting time, $\langle u_0^{(1)}, u_0^{(2)}\rangle = 0$, $\langle u_0^{(1)}, v_0^{(i)}\rangle = 0$, $\langle u_0^{(2)}, v_0^{(i)}\rangle = 0$ *and* $\langle v_0^{(i)}, v_0^{(j)}\rangle = 0$, $||u_0^{(1)}|| = ||u_0^{(2)}|| = C_u$, $||v_0^{(i)}|| = C_v$ *for some positive constants* C_v, C_u *and all* $i, j \in [K]$, *then the gradient descent iterates will have the following properties:*

(a) $\lim_{t\to\infty} \cos(v_t^{(i)}, v_t^{(j)}) = 1$ *for all* $i, j \in [K]$

(b) $\lim_{t\to\infty} \cos(u_t^{(1)} - u_t^{(2)}, v_t^{(i)}) = 1$ *for all* $i \in [K]$

Proof. Before proving the theorem, let's write out a few equations. For simplicity, let's denote $\ell_t^{1,i} = \exp(-\langle u_t^{(1)}, v_t^{(i)}\rangle)$, $\ell_t^{2,i} = \exp(\langle u_t^{(1)}, v_t^{(i)}\rangle)$, $\Delta_t = u_t^{(1)} - u_t^{(2)}$ and $\Delta_t^i = \ell_t^{1,i} u_t^{(1)} - \ell_t^{2,i} u_t^{(2)}$.

By the gradient descent algorithm with learning rate η,

$$\begin{aligned}
v_{t+1}^{(i)} &= v_t^{(i)} + \eta \Delta_t^i \\
u_{t+1}^{(1)} &= u_t^{(1)} + \eta \sum_{j=1}^K \ell_t^{1,j} v_t^{(j)} \\
u_{t+1}^{(2)} &= u_t^{(2)} - \eta \sum_{j=1}^K \ell_t^{2,j} v_t^{(j)}
\end{aligned}$$

Furthermore,

$$\begin{aligned}
\Delta_{t+1} &= u_{t+1}^{(1)} - u_{t+1}^{(2)} = u_t^{(1)} - u_t^{(2)} + \eta \sum_{j=1}^K \ell_t^{1,j} v_t^{(j)} + \eta \sum_{j=1}^K \ell_t^{2,j} v_t^{(j)} \\
&= \Delta_t + \eta \sum_{j=1}^K (\ell_t^{1,j} + \ell_t^{2,j}) v_t^{(j)} \\
\Delta_{t+1}^i &= \ell_{t+1}^{1,i} u_{t+1}^{(1)} - \ell_{t+1}^{2,i} u_{t+1}^{(2)} = \ell_{t+1}^{1,j} u_t^{(1)} + \eta \ell_{t+1}^{1,j} \sum_{j=1}^K \ell_t^{1,j} v_t^{(j)} - \ell_{t+1}^{2,i} u_t^{(2)} + \eta \ell_{t+1}^{2,j} \sum_{j=1}^K \ell_t^{2,j} v_t^{(j)} \\
&= \ell_{t+1}^{1,i} u_t^{(1)} - \ell_{t+1}^{2,i} u_t^{(2)} + \eta\big(\ell_{t+1}^{1,i} \sum_{j=1}^K \ell_t^{1,j} v_t^{(j)} + \ell_{t+1}^{2,i} \sum_{j=1}^K \ell_t^{2,j} v_t^{(j)}\big)
\end{aligned}$$

With our initial condition, by symmetry, one can show that for any $t \geq 1$.

(a) $\ell_t^{1,i} = \ell_t^{1,j} = \ell_t^{2,i} = \ell_t^{2,j}$ for any $i, j \in [K]$ and the loss decreases monotonically.

(b) $\Delta_t^i = \Delta_t^j$.

(c) $\langle v_t^{(i)}, \sum_{j=1}^K v_t^{(j)} \rangle = C_t^1$ for some positive constant C_t^1 that does not depend on index i. And C_t^1 increases monotonically with t.

(d) $\langle v_t^{(i)}, \Delta_t^i \rangle = C_t^2$ for some positive constant C_t^2 that does not depend on index i.

(e) $\langle u_t^{(1)}, \Delta_t^i \rangle = -\langle u_t^{(2)}, \Delta_t^i \rangle = C_t^3$ for some positive constant C_t^3 that does not depend on index i.

(f) $\langle u_t^{(1)}, v_t^i \rangle = -\langle u_t^{(2)}, v_t^i \rangle = C_t^4$ for some positive constant C_t^4 that does not depend on index i.

We'll prove this by induction.

Base step ($t = 1$) First of all, by the initial condition, $\ell_t^{1,i} = \ell_t^{1,j} = \ell_t^{2,i} = \ell_t^{2,j} = 1$.

$$\begin{aligned}
\ell_1^{1,i} &= \exp(-\langle u_1^{(1)}, v_1^{(i)} \rangle) = \exp\big(- \langle u_0^{(1)} + \eta \sum_{j=1}^K \ell_0^{1,i} v_0^{(j)}, v_0^{(i)} + \eta \Delta_0^i \rangle \big) \\
&= \exp\big(- \langle u_0^{(1)}, v_0^{(i)} \rangle - \eta \langle u_0^{(1)}, \Delta_0^i \rangle - \eta \langle v_0^{(i)}, \sum_{j=1}^K \ell_0^{1,j} v_0^{(j)} \rangle - \eta^2 \langle \Delta_0^i, \sum_{j=1}^K \ell_0^{1,i} v_0^{(i)} \rangle \big) \\
&= \exp\big(- \eta \langle u_0^{(1)}, \Delta_0^i \rangle - \eta \ell_0^{1,i} ||v_0^{(i)}||^2 \big) = \exp\big(- \eta \ell_0^{1,i} ||u_0^{(1)}||^2 - \eta \ell_0^{1,i} ||v_0^{(i)}||^2 \big) \\
&= \exp\big(- \eta \ell_0^{1,i} C_u^2 - \eta \ell_0^{1,i} C_v^2 \big) = \exp\big(- \eta C_u^2 - \eta C_v^2 \big) \\
&= \ell_1^{1,j}
\end{aligned}$$

Similarly

$$\begin{aligned}
\ell_1^{2,i} &= \exp(\langle u_1^{(2)}, v_1^{(i)} \rangle) = \exp\big(\langle u_0^{(2)} - \eta \sum_{j=1}^K \ell_0^{2,i} v_0^{(j)}, v_0^{(i)} + \eta \Delta_0^i \rangle \big) \\
&= \exp\big(\langle u_0^{(2)}, \Delta_0^i \rangle - \eta \ell_0^{1,i} ||v_0^{(i)}||^2 \big) = \exp\big(- \eta C_u^2 - \eta C_v^2 \big) \\
&= \ell_1^{2,j} = \ell_1^{1,i} = \ell_1^{1,j}
\end{aligned}$$

For simplicity, let use $\ell_1 = \ell_1^{1,i} = \ell_1^{2,i}$. This also implies that $\Delta_1^i = \Delta_1^j = \ell_1 \Delta_1$.

On the other hand,

$$\begin{aligned}
\langle v_1^{(i)}, \sum_{j=1}^K v_1^{(j)} \rangle &= \langle v_0^{(i)} + \eta \Delta_0^i, \sum_{j=1}^K (v_0^{(j)} + \eta \Delta_0^i) \rangle \\
&= ||v_0^{(i)}||^2 + \eta^2 \langle \Delta_0^i, \sum_{j=1}^K \Delta_0^j \rangle \\
&= ||v_0^{(i)}||^2 + \eta^2 K ||\Delta_0^i||^2 = C_v^2 + 2\eta^2 K C_u^2 \\
&> \langle v_0^{(i)}, \sum_{j=1}^K v_0^{(j)} \rangle = ||v_0^{(i)}||^2 = C_v^2
\end{aligned}$$

Notice that the constant does not depend on i.

$$\begin{aligned}
\langle v_1^{(i)}, \Delta_1^i \rangle &= \ell_1 \langle v_0^{(i)} + \eta \Delta_0^i, \Delta_1 \rangle = \ell_1 \langle v_1^{(i)}, \Delta_0 + \eta \sum_{j=1}^K (\ell_0^{1,j} + \ell_0^{2,j}) v_0^{(j)} \rangle \\
&= \ell_1 \langle v_0^{(i)} + \eta \Delta_0^i, \Delta_0 + 2\eta \sum_{j=1}^K v_0^{(j)} \rangle = 2\ell_1 \eta ||v_0^{(i)}||^2 + \ell_1 \eta (||u_0^{(1)}||^2 + ||u_0^{(2)}||^2) \\
&= 2\ell_1 \eta C_v^2 + 2\eta \ell_1 ||C_u||^2
\end{aligned}$$

Notice that the constant does not depend on i.

$$\begin{aligned}
\langle u_1^{(1)}, \Delta_1^i \rangle &= \ell_1 \langle u_1^{(1)}, \Delta_1 \rangle = \ell_1 \langle u_0^{(1)} + \eta \sum_{j=1}^{K} \ell_0^{1,i} v_0^{(j)}, \Delta_0 + \eta \sum_{j=1}^{K} (\ell_0^{1,j} + \ell_0^{2,j}) v_0^{(j)} \rangle \\
&= \ell_1 \langle u_0^{(1)} + \eta \sum_{j=1}^{K} v_0^{(j)}, \Delta_0 + 2\eta \sum_{j=1}^{K} v_0^{(j)} \rangle = \ell_1 ||u_0^{(1)}||^2 + 2\ell_1 \eta^2 \sum_{j=1}^{K} ||v_0^{(j)}||^2 \\
&= \ell_1 C_u^2 + 2\ell_1 \eta^2 K ||C_v||^2 \\
&= -\langle u_1^{(2)}, \Delta_1^i \rangle
\end{aligned}$$

Again, the constant does not depend on i.

Finally,

$$\begin{aligned}
\langle u_1^{(1)}, v_1^i \rangle &= \langle u_0^{(1)} + \eta \sum_{i=1}^{K} \ell_0^{1,i} v_0^{(i)}, v_0^{(i)} + \eta \Delta_0^i \rangle \\
&= \langle u_0^{(1)} + \eta \sum_{i=1}^{K} v_0^{(i)}, v_0^{(i)} + \eta \Delta_0^i \rangle \\
&= \eta ||u_0^{(1)}||^2 + \eta ||v_0^{(i)}||^2 = \eta C_u^2 + \eta C_v^2 \\
&= -\langle u_1^{(2)}, v_1^i \rangle
\end{aligned}$$

Inductive step By the inductive hypothesis, let $\ell_t = \ell_t^{1,i} = \ell_t^{2,i}$.

$$\begin{aligned}
\ell_{t+1}^{1,i} &= \exp(-\langle u_{t+1}^{(1)}, v_{t+1}^{(i)} \rangle) = \exp\big(- \langle u_t^{(1)} + \eta \sum_{j=1}^{K} \ell_t^{1,i} v_t^{(j)}, v_t^{(i)} + \eta \Delta_t^i \rangle \big) \\
&= \exp\big(- \langle u_t^{(1)}, v_t^{(i)} \rangle - \eta \langle u_t^{(1)}, \Delta_t^i \rangle - \eta \langle v_t^{(i)}, \sum_{j=1}^{K} \ell_t^{1,j} v_t^{(j)} \rangle - \eta^2 \langle \Delta_t^i, \sum_{j=1}^{K} \ell_t^{1,i} v_t^{(i)} \rangle \big) \\
&= \exp\big(- C_t^4 - \eta C_t^3 - \eta \ell_t C_t^1 - \eta^2 \ell_t K C_t^2 \big) \\
&= \ell_{t+1}^{1,j} = \ell_{t+1}^{2,i} = \ell_{t+1}^{2,j}
\end{aligned}$$

$$\begin{aligned}
\langle v_{t+1}^{(i)}, \sum_{j=1}^{K} v_{t+1}^{(j)} \rangle &= \langle v_t^{(i)} + \eta \Delta_t^i, \sum_{j=1}^{K} (v_t^{(j)} + \eta \Delta_t^i) \rangle \\
&= \langle v_t^{(i)}, \sum_{j=1}^{K} v_t^{(j)} \rangle + \eta \langle v_t^{(i)}, \sum_{j=1}^{K} \Delta_t^i \rangle + \eta \langle \Delta_t^i, \sum_{j=1}^{K} v_t^{(j)} \rangle + \eta^2 \langle \Delta_t^i, \sum_{j=1}^{K} \Delta_t^i \rangle \\
&= C_t^1 + 2\eta K C_2^t + \eta^2 \ell_t \langle u_t^{(1)} - u_t^{(2)}, \sum_{j=1}^{K} \Delta_t^i \rangle \\
&= C_t^1 + 2\eta K C_2^t + \eta^2 \ell_t \langle u_t^{(1)} - u_t^{(2)}, \sum_{j=1}^{K} \Delta_t^i \rangle \\
&= C_t^1 + 2\eta K C_2^t + 2\eta^2 \ell_t K C_t^4 > \langle v_t^{(i)}, \sum_{j=1}^{K} v_t^{(j)} \rangle > 0
\end{aligned}$$

By same logic, one show the inductive step for $\langle v_t^{(i)}, \Delta_t^i \rangle$, $\langle u_t^{(1)}, \Delta_t^i \rangle$, $\langle u_t^{(2)}, \Delta_t^i \rangle$, $\langle u_t^{(1)}, v_t^i \rangle$ and $\langle u_t^{(2)}, v_t^i \rangle$.

Therefore, we can simplify the notation. The gradient descent iterates can be rewritten as:

$$v_{t+1}^{(i)} = v_t^{(i)} + \eta \ell_t \Delta_t \quad u_{t+1}^{(1)} = u_t^{(1)} + \eta \ell_t \sum_{i=1}^{K} v_t^{(i)}$$
$$u_{t+1}^{(2)} = u_t^{(2)} - \eta \ell_t \sum_{i=1}^{K} v_t^{(i)} \quad \Delta_{t+1} = \Delta_t + 2\eta \ell_t \sum_{j=1}^{K} v_t^{(j)}$$

Furthermore, let $V_t = \sum_{j=1}^{K} v_t^{(j)}$

$$V_{t+1} = V_t + K\eta \ell_t \Delta_t$$
$$\Delta_{t+1} = \Delta_t + 2\eta \ell_t V_t$$

In essence, the rest of the proof follows from the proof of Theorem C.2 and symmetry.

Loss converging to zero One first notice that by the induction argument above, the loss must be monotonically decreasing. In fact, $\lim_{t\to\infty} \ell_t = 0$. To see this, notice that

$$\langle u_{t+1}^{(1)}, v_{t+1}^{(i)} \rangle - \langle u_t^{(1)}, v_t^{(i)} \rangle \geq \eta \langle v_t^{(j)}, \ell_t \sum_{j=1}^{K} v_t^{(j)} \rangle \geq \ell_t \eta \langle v_1^{(j)}, \sum_{j=1}^{K} v_t^{(j)} \rangle = \ell_t \eta C_1^1$$

Suppose ℓ_t is not converging to zero. Then ℓ_t has an lower bound, which means $\langle u_{t+1}^{(1)}, v_{t+1}^{(i)} \rangle$ will increase to infinity. This is a contradiction. Therefore, $\lim_{t\to\infty} \ell_t = 0$.

In fact, one can also show that, $\lim_{t\to\infty} ||V_t|| \to \infty$ and $\lim_{t\to\infty} ||\Delta_t|| \to \infty$. To see this, one first notice that,

$$\lim_{t\to\infty} \exp(-\langle \Delta_t, V_t \rangle) = \lim_{t\to\infty} \ell_t^{2K} \to 0$$

Therefore, at least one of $||V_t||$ or $||\Delta_t||$ needs to go to infinity. Suppose only $||\Delta_t||$ reaches infinity, then $\lim_{t\to\infty} \frac{||V_t||}{||\Delta_t||} \to 0$. On the other hand,

$$\frac{||V_{t+1}||^2}{||\Delta_{t+1}||^2} = \frac{||V_t||^2 + 2K\eta \ell_t \langle V_t, \Delta_t \rangle + K^2 \eta^2 \ell^2 ||\Delta_t||^2}{||\Delta_t||^2 + 4\eta \ell_t \langle V_t, \Delta_t \rangle + 4\eta^2 \ell^2 ||V_t||^2}$$

Suppose $\frac{||V_t||^2}{||\Delta_t||^2} < K/2$, then

$$\frac{2K\eta \ell_t \langle V_t, \Delta_t \rangle + K^2 \eta^2 \ell^2 ||\Delta_t||^2}{4\eta \ell_t \langle V_t, \Delta_t \rangle + 4\eta^2 \ell^2 ||V_t||^2} \geq K/2$$

Therefore, if $\frac{||V_t||^2}{||\Delta_t||^2} < K/2$, $\frac{||V_{t+1}||^2}{||\Delta_{t+1}||^2} > \frac{||V_t||^2}{||\Delta_t||^2}$. Similarly, if $\frac{||V_t||^2}{||\Delta_t||^2} > K/2$, $\frac{||V_{t+1}||^2}{||\Delta_{t+1}||^2} < \frac{||V_t||^2}{||\Delta_t||^2}$. Therefore, $\lim_{t\to\infty} \frac{||V_t||}{||\Delta_t||}$ will not be zero. So both $\lim_{t\to\infty} ||V_t|| \to \infty$ and $\lim_{t\to\infty} ||\Delta_t|| \to \infty$.

Cosine similarity between $v_t^{(i)}, v_t^{(j)}$ converges to one Note for any i, $\lim_{t\to\infty} ||v_t^{(i)}|| \to \infty$. This is because by symmetry,

$$\lim_{t\to\infty} ||v_t^{(i)}|| \geq \lim_{t\to\infty} \frac{||V_t||}{K} \to \infty$$

Then,

$$v_T^{(i)} = v_0^{(i)} + \eta \sum_{t=0}^{T-1} \Delta_t^w$$

Let's denote $D_T = \eta \sum_{t=0}^{T-1} \Delta_t^w$. Then $||v_0^{(i)}|| + ||D_T|| \geq ||v_T^{(i)}||$. Thus, $\lim_{T\to\infty} ||D_T|| = \infty$.

Finally

$$\cos(v_t^{(i)}, v_t^{(i)}) = \frac{\langle v_0^{(i)} + D_T, v_0^{(j)} + D_T\rangle}{||v_t^{(i)}||||v_t^{(j)}||} \geq \frac{\langle v_0^{(i)} + D_T, v_0^{(j)} + D_T\rangle}{(||v_0^{(i)}|| + ||D_t||)(||v_0^{(j)}|| + ||D_t||)}$$
$$= \frac{\langle v_0^{(i)} + v_0^{(j)}, D_T\rangle + ||D_T||^2}{(||v_0^{(i)}|| + ||D_t||)(||v_0^{(j)}|| + ||D_t||)} = \frac{\langle v_0^{(i)} + v_0^{(j)}, D_T\rangle/||D_t||^2 + 1}{(||v_0^{(i)}||/||D_t|| + 1)(||v_0^{(j)}||/||D_t|| + 1)}$$

Thus,

$$\lim_{t\to\infty} \cos(v_t^{(i)}, v_t^{(i)}) = 1$$

Cosine similarity between $v_t^{(i)}, \Delta_t$ converges to one Finally, one can show that $\lim_{t\to\infty} \cos(\Delta_t, V_t) \to 1$ follows the same proof of Theorem C.2. For completeness, we will present the full proof here.

Let's first do a simple variable change,

$$\sqrt{2}V_{t+1} = \sqrt{2}V_t + \sqrt{2K}\eta\ell_t\sqrt{K}\Delta_t$$
$$\sqrt{K}\Delta_{t+1} = \sqrt{K}\Delta_t + \sqrt{2K}\eta\ell_t\sqrt{2}V_t$$

Let $\tilde{V}_t = \sqrt{2}V_t$, $\tilde{\Delta}_t = \sqrt{K}\Delta_t$, and $\tilde{\eta} = \sqrt{2K}\eta$, then

$$\tilde{V}_{t+1} = \tilde{V}_t + \tilde{\eta}\ell_t\tilde{\Delta}_t$$
$$\tilde{\Delta}_{t+1} = \tilde{\Delta}_t + \tilde{\eta}\ell_t\tilde{V}_t$$

One first notice that, $\tilde{V}_t + \tilde{\Delta}_t$ always has the direction at any t. Therefore, let's consider the $\tilde{V}_{t+1}^{\perp}$ which is the residual after projecting onto the direction of $\tilde{V}_t + \tilde{\Delta}_t$,

$$||\tilde{V}_{t+1}^{\perp}||^2 = ||\tilde{V}_{t+1}||^2 - ||\frac{\langle \tilde{V}_{t+1}, \tilde{V}_t + \tilde{\Delta}_t\rangle}{||\tilde{V}_t + \tilde{\Delta}_t||}||^2$$
$$= \frac{(1-\tilde{\eta}\ell_t)^2\big(||\tilde{V}_t||^2||\tilde{\Delta}_t||^2 - (\langle \tilde{V}_t, \tilde{\Delta}_t\rangle)^2\big)}{||\tilde{V}_t + \tilde{\Delta}_t||^2}$$
$$\leq C_\eta \frac{||\tilde{V}_t||^2||\tilde{\Delta}_t||^2 - (\langle \tilde{V}_t, \tilde{\Delta}_t\rangle)^2}{||\tilde{V}_t + \tilde{\Delta}_t||^2}$$

Note that $\tilde{\eta}\ell_t$ converges to zero. Therefore, there's an upper bound C_η on $(1-\tilde{\eta}\ell_t)^2$.

On the other hand, let $O_t = ||\tilde{V}_t||^2||\tilde{\Delta}_t||^2 - (\langle \tilde{V}_t, \tilde{\Delta}_t\rangle)^2$. Then

$$O_{t+1} - O_t = (\tilde{\eta}^4\ell_t^4 - 2\tilde{\eta}^2\ell_t^2)||\tilde{V}_t||^2||\tilde{\Delta}_t||^2(1 - \cos(\tilde{V}_t, \tilde{\Delta}_t))$$

Once again, because ℓ_t is eventually converging to zero, O_t will decrease at some point. This is because $\tilde{\eta}^4\ell_t^4 - 2\tilde{\eta}^2\ell_t^2 < 0$ if $\ell_t < \frac{\sqrt{2}}{\tilde{\eta}}$ and $||\tilde{V}_t||^2||\tilde{\Delta}_t||^2(1 - \cos(\tilde{V}_t, \tilde{\Delta}_t)) \geq 0$. Because $O_t \geq 0$, it will reach a limit. Therefore, O_t must have an upper bound. Finally, the denominator is diverging and by our inductive statements, it must have a nonzero lower bound.

Therefore $||\tilde{V}_{t+1}^{\perp}||$ is bounded. And as $\lim_{t\to\infty} ||V_t|| \to \infty$ and $\lim_{t\to\infty} ||\Delta_t|| \to \infty$, we have

$$\lim_{t\to\infty} \cos(\tilde{V}_t, \tilde{V}_t + \tilde{\Delta}_t) = 1$$
$$\lim_{t\to\infty} \cos(\tilde{\Delta}_t, \tilde{V}_t + \tilde{\Delta}_t) = 1$$

Thus,

$$\lim_{t\to\infty} \cos(\tilde{\Delta}_t, \tilde{V}_t) = 1$$

This would also imply that $\lim_{t\to\infty} \cos(\Delta_t, v_t^{(i)}) \to 1$ for all i.

□

D. Orthogonality

In this section, we will prove our main theorems on orthogonality.

Theorem 4.1. *Let $\widehat{\mathcal{C}} = \mathcal{C}$ and $\widehat{\mathcal{D}} = \mathcal{D}$. Assuming $p(c) > 0$ for any $c \in \mathcal{C}$ and C_i and C_j are two latent variables separated in G_C. Given any binary vector $c \in \mathcal{C}$, there exists a subset $\mathcal{D}_c \subset \mathcal{D}$ such that $d_i = \diamond$ and $p(c|d) > 0$ for any $d \in \mathcal{D}_c$. If one further assume that $\hat{p}(c|d) = p(c|d)$ for any $d \in \mathcal{D}_c$, then*

$$g(c_{(i\to 1)}) - g(c_{(i\to 0)}) \perp f(d_{(j\to c_j)}) - f(d_{(j\to\diamond)})$$

for any $d \in \mathcal{D}_c$.

Proof. First of all, for any binary vector c, $\mathcal{D}_c$ is non-empty by the positivity assumption because $d^\diamond \in \mathcal{D}_c$ where $d^\diamond = \{\diamond, ..., \diamond\}$.

Consider an arbitrary $c \in \mathcal{C}$ and an arbitrary $d \in \mathcal{D}_c$. Without loss of generality, let $c_i = 1$. Suppose $d_j = \diamond$, then it must be that $d_{(j\to c_j)} \in \mathcal{D}_c$ because $d_{(j\to c_j)}$ agrees with c on the j-th entry. Similarly, if $d_j = c_j$, then $d_{(j\to\diamond)} \in \mathcal{D}_c$.

By the positivity assumption and the fact that $d_i = \diamond$,

$$\begin{aligned} p(c_{(i\to 1)}|d_{(j\to\diamond)}) > 0, \qquad & p(c_{(i\to 0)}|d_{(j\to\diamond)}) > 0 \\ p(c_{(i\to 1)}|d_{(j\to c_j)}) > 0, \qquad & p(c_{(i\to 0)}|d_{(j\to c_j)}) > 0 \end{aligned}$$

Thus,

$$\begin{aligned} \hat{p}(c_{(i\to 1)}|d_{(j\to\diamond)}) = p(c_{(i\to 1)}|d_{(j\to\diamond)}) \quad & \hat{p}(c_{(i\to 0)}|d_{(j\to\diamond)}) = p(c_{(i\to 0)}|d_{(j\to\diamond)}) \\ \hat{p}(c_{(i\to 1)}|d_{(j\to c_j)}) = p(c_{(i\to 1)}|d_{(j\to c_j)}) \quad & \hat{p}(c_{(i\to 0)}|d_{(j\to c_j)}) = p(c_{(i\to 0)}|d_{(j\to c_j)}) \end{aligned}$$

Then by the Hammersley–Clifford theorem, we can factorize the joint distribution over cliques:

$$p(c) \propto \prod_k \Psi_k(c_{I_k})$$

where $\Psi_k(c_{I_k})$ is a function that only depends on the clique of random variables C_{I_k}.

By this factorization, if $p(c_{(i\to 1)}|d) > 0$ and $p(c_{(i\to 0)}|d) > 0$,

$$\ln \frac{p(c_{(i\to 1)}|d)}{p(c_{(i\to 0)}|d)} = \beta(c_{I_i}, d_{I_i})$$

where β is some function that only depends on cliques that involve C_i. In other words, $i \in I_i$ and $i' \in I_i$ if $C_{i'}$ and C_i are in the same clique in G_C.

Thus,

$$\ln \frac{p(c_{(i\to 1)}|d_{(j\to\diamond)})}{p(c_{(i\to 0)}|d_{(j\to\diamond)})} = \ln \frac{p(c_{(i\to 1)}|d_{(j\to c_j)})}{p(c_{(i\to 0)}|d_{(j\to c_j)})}$$

and,

$$\ln \frac{\hat{p}(c_{(i\to 1)}|d_{(j\to\diamond)})}{\hat{p}(c_{(i\to 0)}|d_{(j\to\diamond)})} = \ln \frac{\hat{p}(c_{(i\to 1)}|d_{(j\to c_j)})}{\hat{p}(c_{(i\to 0)}|d_{(j\to c_j)})}$$

Therefore,

$$\begin{aligned} \Big(g(c_{(i\to 1)}) - g(c_{(i\to 0)})\Big)^T f(d_{(j\to\diamond)}) &= \Big(g(c_{(i\to 1)}) - g(c_{(i\to 0)})\Big)^T f(d_{(j\to c_j)}) \\ \Big(g(c_{(i\to 1)}) - g(c_{(i\to 0)})\Big)^T \Big(f(d_{(j\to\diamond)}) - f(d_{(j\to c_j)}\Big) &= 0 \end{aligned}$$

□

Corollary D.1. *Under the conditions of Theorem 4.1, if concept C_i and C_j have matched-representations, then*

$$g(c_{(i\to 1)}) - g(c_{(i\to 0)}) \perp g(c_{(j\to 1)}) - g(c_{(j\to 0)})$$

for any $c \in \mathcal{C}$.

Proof. Consider two binary vectors $c^{(0)}, c^{(1)} \in \mathcal{C}$ where $c_j^{(0)} = 0$ and $c_j^{(1)} = 1$ but they agree on other entries.

By Theorem 4.1,

$$g(c^{(0)}_{(i\to 1)}) - g(c^{(0)}_{(i\to 0)}) \perp f(d^{(0)}_{(j\to c_j)}) - f(d^{(0)}_{(j\to \diamond)})$$

for any $d^{(0)} \in \mathcal{D}_{c^{(0)}}$. Similar statements can be made for $c^{(1)}$ as well.

Note that $\mathcal{D}_{c^{(0)}} \cap \mathcal{D}_{c^{(1)}} \neq \emptyset$ by the positivity assumption. Let $d \in \mathcal{D}_{c^{(0)}} \cap \mathcal{D}_{c^{(1)}}$. Then,

$$g(c^{(0)}_{(i\to 1)}) - g(c^{(0)}_{(i\to 0)}) \perp f(d_{(j\to 0)}) - f(d_{(j\to \diamond)})$$
$$g(c^{(1)}_{(i\to 1)}) - g(c^{(1)}_{(i\to 0)}) \perp f(d_{(j\to 1)}) - f(d_{(j\to \diamond)})$$

By assumption, there exists a unit vector u_i, such that

$$g(c^{(0)}_{(i\to 1)}) - g(c^{(0)}_{(i\to 0)}) = \alpha^{(0)} u_i \quad g(c^{(1)}_{(i\to 1)}) - g(c^{(1)}_{(i\to 0)}) = \alpha^{(1)} u_i$$

for some $\alpha^{(0)}, \alpha^{(1)} > 0$. Therefore,

$$\left\langle u_i, f(d_{(j\to 0)}) - f(d_{(j\to \diamond)}) \right\rangle = \left\langle u_i, f(d_{(j\to 1)}) - f(d_{(j\to \diamond)}) \right\rangle = 0$$

Thus,

$$\left\langle u_i, f(d_{(j\to 0)}) - f(d_{(j\to 1)}) \right\rangle = 0$$

Because latent variable C_i has linaer and matched representations,

$$g(c_{(i\to 1)}) - g(c_{(i\to 0)}) \perp g(c_{(j\to 1)}) - g(c_{(j\to 0)})$$

for any $c \in \mathcal{C}$. □

E. Simulated Experiments

In this section, we will provide additional details on our simulated experiments with the latent conditional model.

Additional details To let models learn conditional distributions, we train them to make predictions using cross-entropy loss. To turn the aforementioned generated binary vectors into a prediction task, we randomly generate binary masks μ for each vector in the batch. And if $\mu_i = 1$, the i-th entry of the vector is left untouched, and if $\mu_i = 0$, then it is set to -1 which is a numerical representation of the token $\diamond$. The model is trained to use masked vectors to predict the original vectors.

These sampled binary vectors and ternary vectors are mapped into one-hot encodings to avoid neural networks exploiting the inherent structures of these vectors. f and g are modeled as a linear function, which are essentially lookup tables. This construction is made without loss of generality.

Adam optimizer Although the theory is presented with gradient descent, the empirical result is actually robust to the choice of optimizers. We repeat the previous experiments on the complete set of conditionals with Adam optimizer (Kingma & Ba, 2015) using a learning rate 0.001 and observe similar linear representation patterns in Table 2.

Incomplete set of contexts ($\widehat{\mathcal{D}} \subset \mathcal{D}$) Both the size of $\mathcal{C}$ and $\mathcal{D}$ grow exponentially. To model large language models, not every conditional probability is necessarily trained. In fact, one does not need the complete set of contexts to get linearly encoded representations. To test this, we run experiments on incomplete subsets of contexts by randomly selecting a few masks $\{\mu\}$. Table 3 shows that even with subsets of contexts, one can still get linearly encoded representations.

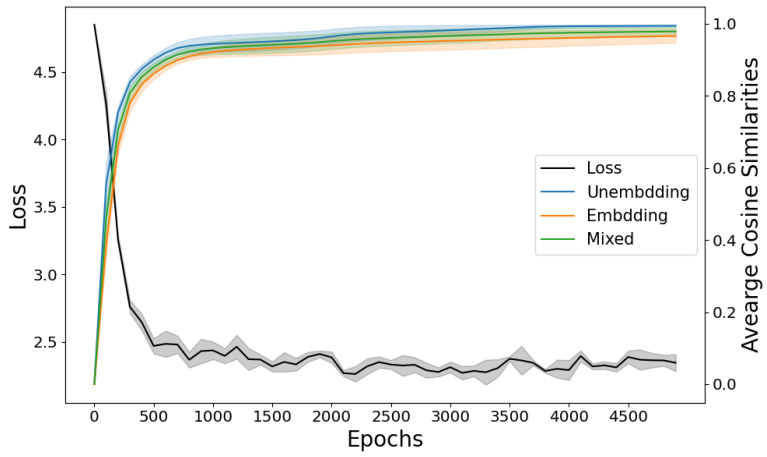


Figure 5. Loss and cosine similarities change as training progresses. The experiments are tested with 7 hidden variables over 10 runs.

Table 2. When the model is trained with Adam optimizer, the m latent variables are represented linearly, and the embedding and unembedding representations are matched. The table shows average cosine similarities among and between steering vectors of unembeddings and embeddings. Standard errors are over 100 runs for 3 variables and 4 variables, 50 runs for 5 variables, 20 runs for 6 variables and 10 runs for 7 variables.

m	UNEMBEDDING	EMBEDDING	UNEMBEDDING AND EMBEDDING
3	0.910±0.015	0.923±0.013	0.926±0.012
4	0.972±0.005	0.959±0.005	0.965±0.005
5	0.985±0.004	0.970±0.004	0.975±0.004
6	0.996±0.001	0.977±0.002	0.983±0.001
7	0.966±0.012	0.918±0.016	0.929±0.015

Incomplete set of concept vectors or violation of positivity ($\widehat{\mathcal{C}} \subset \mathcal{C}$) In real-world language modeling, not every concept vector will be mapped to a token. We test this by allowing some concept vectors c to have zero probabilities (i.e., $p(c) = 0$). For the experiments, we first randomly select a subset $\hat{\mathcal{C}}$ of $\mathcal{C}$ and then use rejection sampling to collect data points. Table 4 shows that one can still get reasonable linearity for unembeddings. The embedding alignments drop, however. One possible explanation is due to a lack of training because the size of the problem grows exponentially. On the other hand, Section 3.2 suggests the connection between linearity and zero conditional probabilities. Because the positivity assumption is violated, the newly introduced zero conditional probabilities might also cause misalignment.

Change of dimensions Previous experiments set the representation dimension to be the same as the number of latent variables. In this set of experiments, we test how decreasing dimensions affects representations by rerunning experiments on 7 variables with the complete set of contexts and binary vectors as well as experiments with 10 variables with incomplete set of contexts and binary vectors. Figure 6 shows that although decreasing dimensions do make representations less aligned, the effect is not significant.

F. Experiments with large language models

In this section, we will provide additional details on our experiments with LLMs.

Examples of counterfactual context and token pairs Example context pairs from OPUS Books (Tiedemann, 2012) that were used to construct the embedding vectors are shown in Table 5. For the unembedding vectors, we reuse the 27 concepts considered in (Park et al., 2023) built atop the work of (Gladkova et al., 2016), and they are listed in Table 6 along with example token pairs.

Table 3. When the model is trained to learn subsets of conditionals for 10 latent variables, the latent variables are still represented linearly and matched. The table shows average cosine similarities. Standard errors are over 10 runs.

MAX NUMBER OF MASKS	UNEMBEDDING	EMBEDDING	UNEMBEDDING AND EMBEDDING
50	0.974±0.009	0.946±0.014	0.959±0.011
100	0.957±0.009	0.915±0.013	0.934±0.011

Table 4. When training with an incomplete set of concept vectors and contexts, unembedding representations are still encoded linearly. The table shows average cosine similarities. Standard errors are over 10 runs.

NUMBER OF HIDDEN VARIBLES	UNEMBEDDING	EMBEDDING	UNEMBEDDING AND EMBEDDING
10	0.951±0.011	0.777±0.010	0.855±0.011
12	0.896±0.011	0.551±0.008	0.696±0.009

Winograd Schema We now present full details about our experiments on the Winograd Schema dataset (Levesque et al., 2012). Recall that we consider context pairs arising from the Winograd Schema, which is a dataset of pairs of sentences that differ in only one or two words and which further contain an ambiguity that can only be resolved with world knowledge and reasoning. Example context pairs from the Winograd Schema (Levesque et al., 2012) are shown in Table 7. Note the final 2 context pairs in the table, which have ambiguous concepts, thus highlighting the nuances of this dataset as well as of natural language. We filter the dataset to have the ambiguous word near the end of the sentence to better align with our theory.

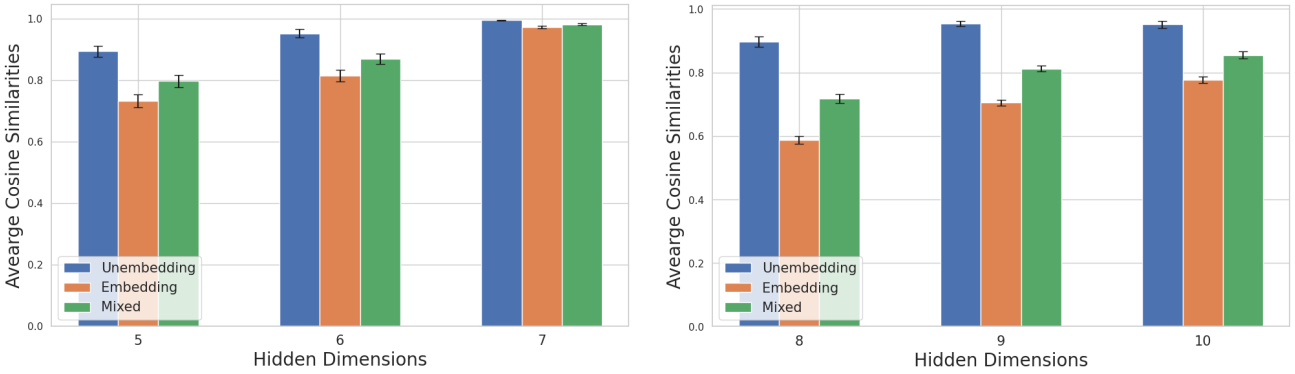
We again compute the unembedding vectors for these context pairs output by LLaMA-2. For the embedding vectors, since there is no predefined set of concepts to consider (for instance consider the ambiguous examples), we therefore take the difference of the embedding vectors for the first token that differs in these corresponding pairs of sentences. We compute all pairwise cosine similarities. We observe that for non-matching contexts, the average similarity is 0.011 with a maximum of 0.081, whereas for matching contexts, the average similarity is 0.042 with a maximum of 0.161. This aligns with our predictions that the embedding vectors align better with the unembedding vectors of the same concept.

As an additional experiment, we compute similarities between these Winograd context pairs and the 27 concepts from (Park et al., 2023) described above. In Table 8, we display the top 3 pairs of contexts and concepts that have the highest similarities. The alignment seems reasonable as baby, woman, male, and female are different attributes of a person; and wide vs narrow and short vs tall can be construed as different manifestations of small vs big.

Table 5. Example language context pairs from OPUS Books

LANGUAGE PAIR	CONTEXT 1	CONTEXT 2
French–Spanish	Quinze ou seize, répliqua l'autre	Quince ó diez y seis, replicó el otro
French–German	Comment est-il mon maître?	Wie ist er mein Herr?
English–French	I hesitated for a moment.	J'hésitai une seconde.
German–Spanish	Ich hasse die Spazierfahrten	No me gusta salir en coche.

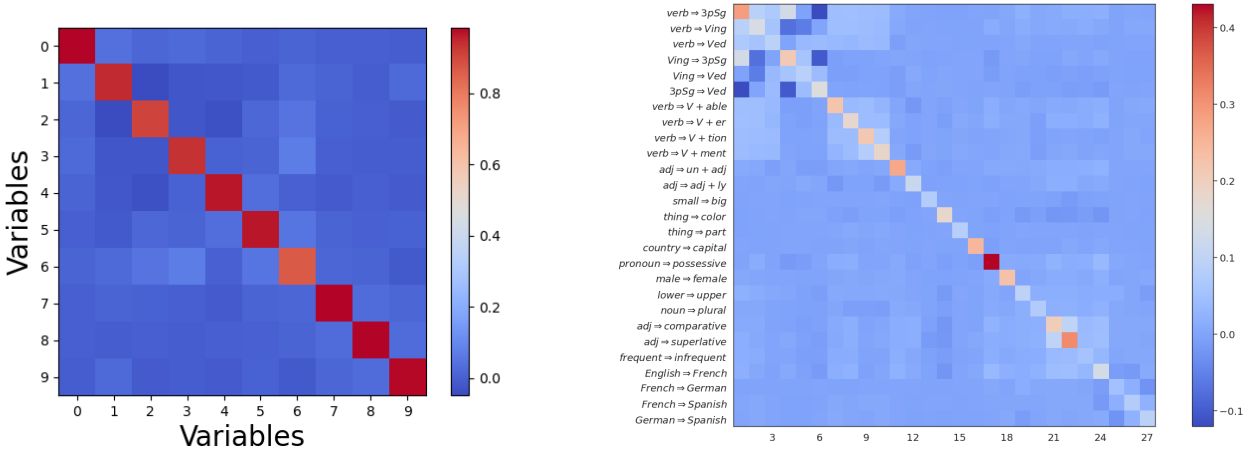
Additional barplots The entire set of similarity barplots for the concepts of French–Spanish, French–German, English–French and German–Spanish are in Figures 8, 9, 10 and 11 respectively. As we see, they satisfy the same behavior as described earlier in Section 5.2, exhibiting relatively high similarity with the matching unembedding vector, close to high similarity with related language concepts and low similarity with unrelated concepts.



(a) 7 Hidden Variables

(b) 10 Hidden Variables

Figure 6. Average cosine similarities under different hidden dimensions show that reducing dimension dose not hurt linearity significantly.



(a) Simulated Experiments

(b) LLaMA-2

Figure 7. The pattern of linear and orthogonal representations matches between simulated experiments and LLaMA-2. Specifically, unembedding steering vectors of the same concept in LLaMA-2 have nontrivial alignment, while steering vectors of different concepts are represented almost orthogonally. The heatmap shows the average cosine similarities between different sets of steering vectors for both simulated experiments and LLaMA-2. For simulated experiments, the cosine similarities are averaged over 10 runs.

Table 6. Concepts and example token pairs, taken from (Park et al., 2023)

CONCEPT	EXAMPLE TOKEN PAIR	CONCEPT	EXAMPLE TOKEN PAIR
verb ⇒ 3pSg	(accept, accepts)	verb ⇒ Ving	(add, adding)
verb ⇒ Ved	(accept, accepted)	Ving ⇒ 3pSg	(adding, adds)
Ving ⇒ Ved	(adding, added)	3pSg ⇒ Ved	(adds, added)
verb ⇒ V + able	(accept, acceptable)	verb ⇒ V + er	(begin, beginner)
verb ⇒ V + tion	(compile, compilation)	verb ⇒ V + ment	(agree, agreement)
adj ⇒ un + adj	(able, unable)	adj ⇒ adj + ly	(according, accordingly)
small ⇒ big	(brief, long)	thing ⇒ color	(ant, black)
thing ⇒ part	(bus, seats)	country ⇒ capital	(Austria, Vienna)
pronoun ⇒ possessive	(he, his)	male ⇒ female	(actor, actress)
lower ⇒ upper	(always, Always)	noun ⇒ plural	(album, albums)
adj ⇒ comparative	(bad, worse)	adj ⇒ superlative	(bad, worst)
frequent ⇒ infrequent	(bad, terrible)	English ⇒ French	(April, avril)
French ⇒ German	(ami, Freund)	French ⇒ Spanish	(année, año)
German ⇒ Spanish	(Arbeit, trabajo)		

Table 7. Example context pairs from Winograd Schema

CONTEXTS
The delivery truck zoomed by the school bus because it was going so fast. The delivery truck zoomed by the school bus because it was going so slow.
The man couldn't lift his son because he was so weak. The man couldn't lift his son because he was so heavy.
Joe's uncle can still beat him at tennis, even though he is 30 years younger. Joe's uncle can still beat him at tennis, even though he is 30 years older.
Paul tried to call George on the phone, but he wasn't successful. Paul tried to call George on the phone, but he wasn't available.
The large ball crashed right through the table because it was made of steel. The large ball crashed right through the table because it was made of styrofoam.

Table 8. Top similarities between Winograd contexts and token concepts

CONTEXTS	MOST SIMILAR CONCEPT	SIMILARITY
Anne gave birth to a daughter last month. She is a very charming woman. Anne gave birth to a daughter last month. She is a very charming baby .	male⇒female	0.311
The table won't fit through the doorway because it is too wide. The table won't fit through the doorway because it is too narrow.	small⇒big	0.309
John couldn't see the stage with Billy in front of him because he is so short. John couldn't see the stage with Billy in front of him because he is so tall.	small⇒big	0.303

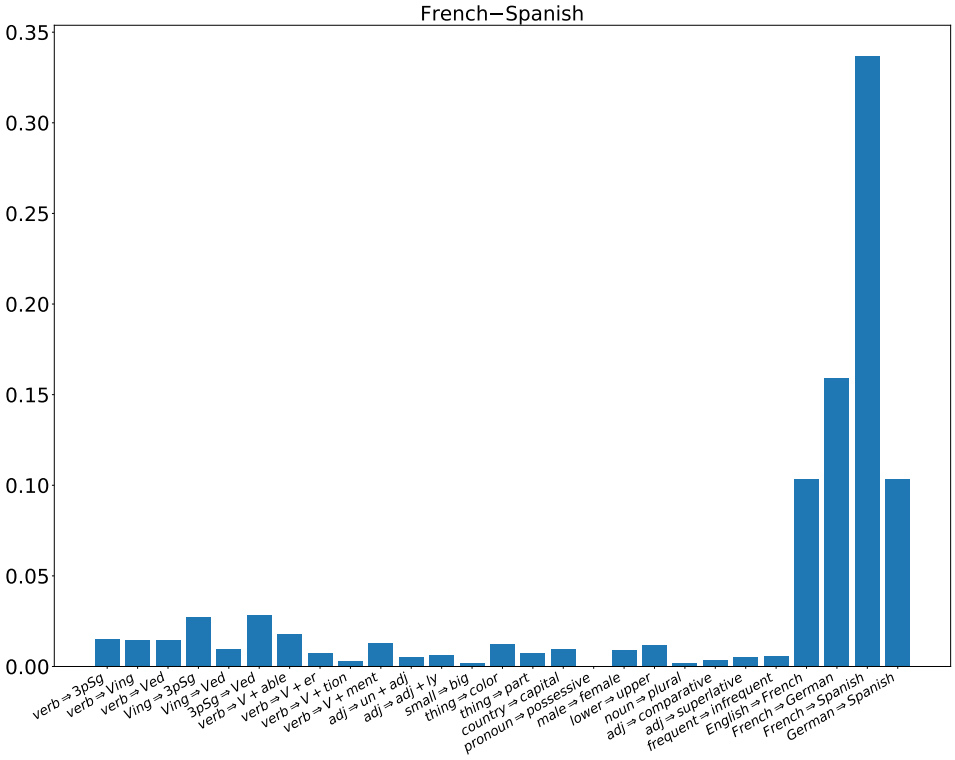


Figure 8. The French–Spanish concept is highly correlated with similar token concepts relative to others. This figure shows all cosine similarities between the French–Spanish concept and token concepts.

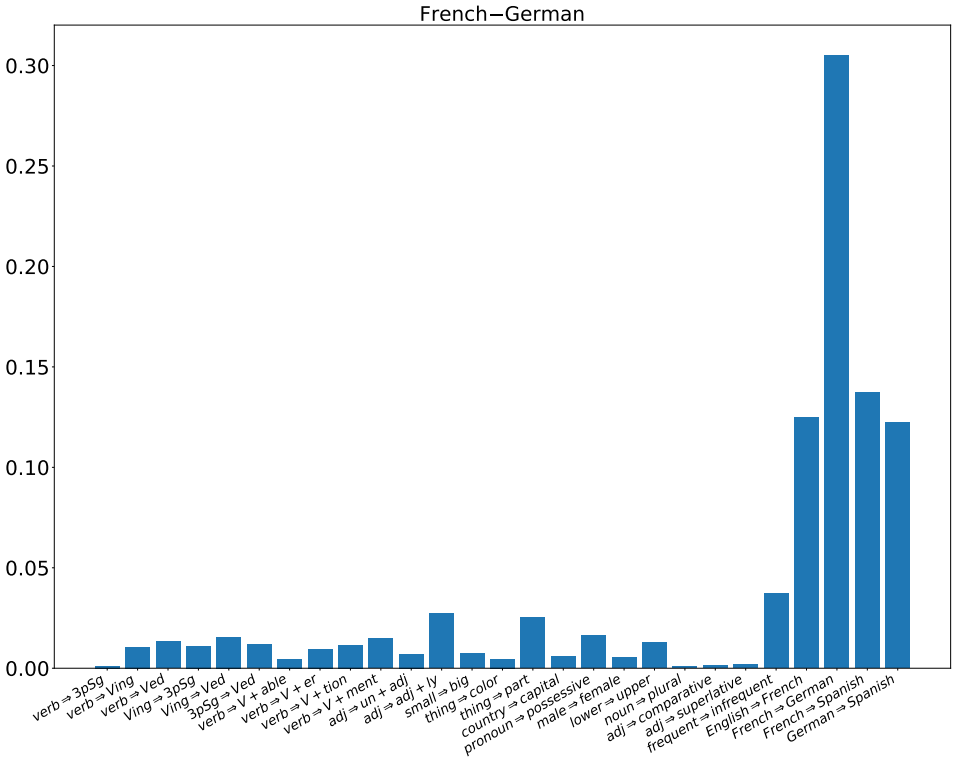


Figure 9. The French–German concept is highly correlated with similar token concepts relative to others. This figure shows all cosine similarities between the French–German concept and token concepts.

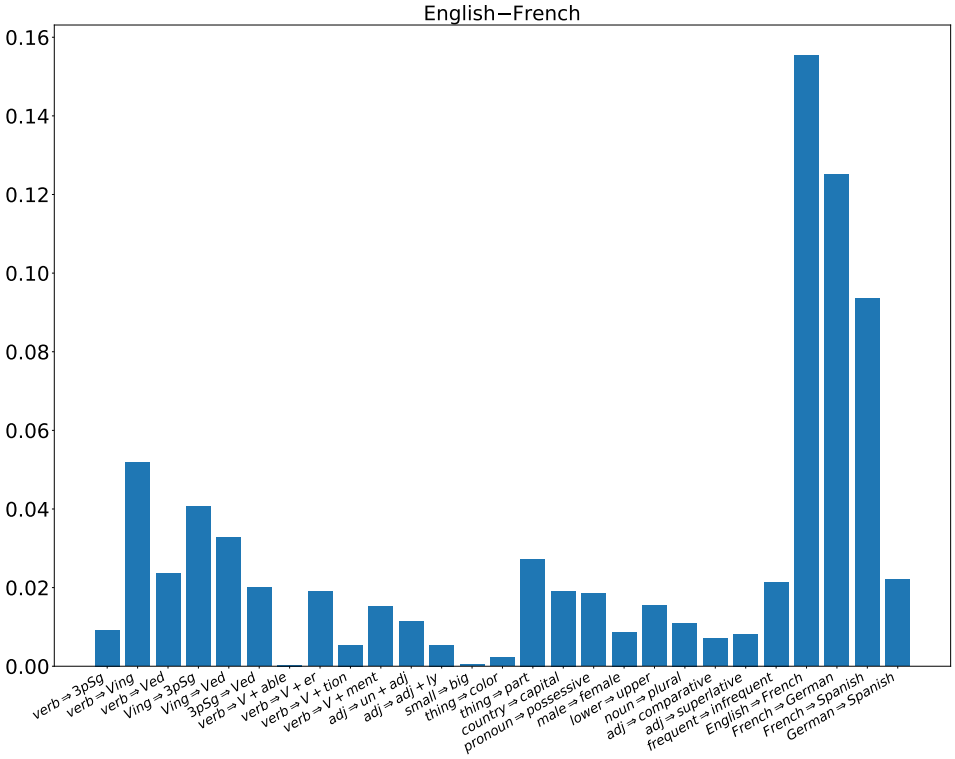


Figure 10. The English–French concept is highly correlated with similar token concepts relative to others. This figure shows all cosine similarities between the English–French concept and token concepts.

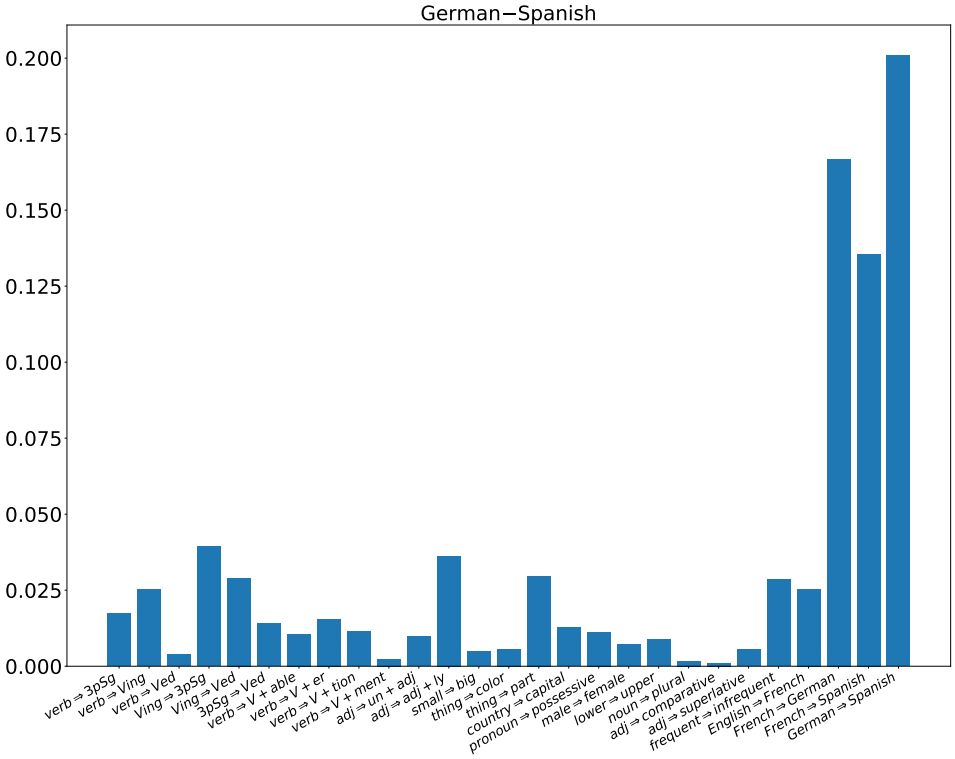


Figure 11. The German–Spanish concept is highly correlated with similar token concepts relative to others. This figure shows all cosine similarities between the German–Spanish concept and token concepts.