Discovery of categorical concepts and their structure with binary matrix factorization

Anonymous Author(s)
Affiliation
Address
email

Abstract

Symbolic, compositional reasoning is thought to be an important part of human cognition, and yet most state of the art neural models, and indeed our own brains, use continuous distributed representations. These representations offer impressive gradient-based learning capabilities, but it is often difficult to know what symbolic algorithm they might implicitly be implementing, if any. Taking to heart the idea that latent symbolic structure is linearly encoded in neural representations, we offer a new approach for inferring latent categorical structure in an unsupervised way via binary matrix factorization. A rich family of discrete structures, like clustering, hierarchy, and linear ordering, can be expressed by constrained binary representations; casting the discrete latent inference problem as a general binary matrix factorization allows us to infer higher-order constraints (such as "hierarchy" or "ordering") from data, or softly encourage them through regularization, rather than strictly enforce them. Numerical experiments demonstrate that our approach can robustly recover the ground truth when it exists. We substantially extend prior algorithms for BMF in terms of speed, scalability, and scope, by using a simple simulated annealing approach with new regularization schemes to handle the very common case of non-identifiability. Finally, we test this approach on real data from machine learning and neuroscience and find it useful at uncovering interpretable compositional structure.

1 Introduction

In biological and artificial learning systems, compositional structure is important to flexible behavior, yet it is difficult to detect at the representational level. Neural representations are rarely factorized into purely-selective concept neurons; when there is a neat conceptual structure it is most often embedded into high-dimensional neural modes [24, 20, 57, 4, 7]. Modern machine learning systems also use distributed continuous representations, which are rarely factorized even when symbolic or compositional structure are explicitly incorporated into the model [1, 55, 40].

We will show that the discovery of latent categories can be formulated as a binary matrix factorization (BMF). Given a representation, $\mathbf{X}$, we will try to find a binary representation, $\mathbf{S}$, which encodes an assignment of items to logical variables in such a way that preserves distances. We refer to these logical variables as 'concepts'. Many structures–including analogies, clustering, hierarchy, ordering, and hybrids of these–can be captured by binary concepts with an appropriate structure. However, BMF is a difficult combinatorial problem, so, in addition to introducing it as a tool for concept discovery, we offer efficient new algorithms.

Identifying discrete latent structure can enable compact descriptions of neural computation. For example, the "mechanistic interpretability" field of AI research has had great success with unsupervised approaches (e.g. sparse autoencoders/NMF) to discover latent variables with causal effects

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

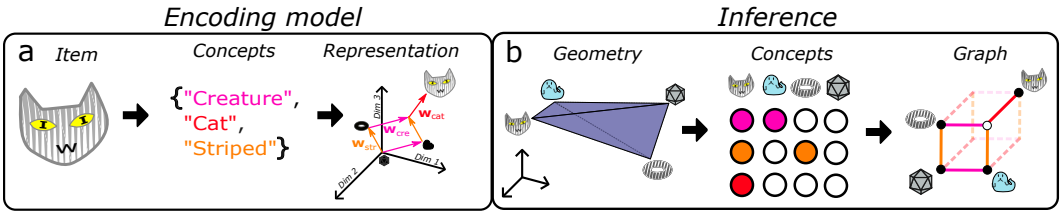


Figure 1: Cartoon of a compositional representation. (**a**) Based on the assumed encoding model, (**b**) in this paper we derive several inference schemes.

on network behavior [23]. While the factors learned by such approaches are continuous, they are often analyzed discretely (is it active or not), and summarized as essentially categorical concepts. But these binarized concepts may no longer be an accurate model of the data. If latent variables are to be interpreted categorically, there might be some advantage in using a model which is categorical by construction.

In addition to interpreting fixed models, there might be some value to logical representations when building a model. An early critique of connectionism was its inability to account for systematic generalization [15]. Despite remarkable and unanticipated advances, modern connectionist systems still struggle with abstract reasoning and out-of-distribution generalization [6, 46, 47, 3]. Being able to efficiently convert learned continuous representations into a symbolic equivalent can help leverage advantages of both gradient-based and symbolic computation [40, 28].

Contributions We provide several new components to the study of concepts and compositionality:

- The study of concept representations in neuroscience and AI relies heavily on foreknowledge or pre-existing hypotheses of relevant concepts, and we introduce an approach for unsupervised discovery. We demonstrate its ability to find meaningful structure in real data.
- Conceptually, we operationalise influential ideas about the representation of concepts [35, 51, 53] and provide a framework for studying the structure of concepts empirically.
- We give efficient and effective heuristics, without restrictive assumptions of other BMF algorithms. We support their utility with extensive numerical simulations. Interestingly, the most performant of our algorithms is based on Hopfield networks, a model of memory.
- On the technical side, the literature on BMF has restricted itself to the case of identifiable factorizations; we substantially extend the scope of that work by connecting binary latents to compositional structure, developing new regularization schemes, and providing an overlooked connection to graph structure as a visualisation tool.

By bringing insights from the literature on BMF to machine learning and neuroscience, we can offer a new perspective on an important and under-studied problem. In general, we hope that the simplicity and effectiveness of our approach can inspire deeper exploration of this problem.

2 Model specification

At a high level, we are trying to infer latent compositional structure which may have generated our continuous data. In practice, the quality of our solutions and ease of finding them will depend substantially on our choice of objective function and algorithm. Since ours is a relatively underexplored problem, we will present a few variants which will turn out to have complementary strengths. For added interpretability, we also describe the relationship between binary embeddings and a certain family of graphs, which will provide both a useful visualization tool and regularization scheme.

Notation Henceforth the 'data' will refer to the matrix, $\mathbf{X} \in \mathbb{R}^{d \times p}$, of shape d (number of dimensions) by p (number of points). We will refer to the matrix of binary latents, $\mathbf{S} \in \{0,1\}^{k \times p}$, as the 'concepts'. There is also a real 'feature vector' associated with each concept, $\mathbf{w}_\alpha$, which is organized in the weight matrix $\mathbf{W} \in \mathbb{R}^{d \times k}$. We will use low case letters to denote vectors; when

indexing a matrix, we use Greek indices for rows/concepts, while Roman indices for columns/items – for example $\mathbf{s}_i$ refers to the vector of concepts for item i, while $\mathbf{s}_\alpha$ refers to the vector of items which are instances of concept α.

When working with binary vectors, we will use the shorthand notation $\tilde{\mathbf{s}}$ to mean the complement of $\mathbf{s}$, i.e. $\mathbf{1} - \mathbf{s}$. Furthermore the empirical average is denoted by $\langle \cdot \rangle$; for example, $\langle \mathbf{s} \rangle \doteq \frac{1}{p}\mathbf{S}\mathbf{1}$.

Model Our model of compositional representations is that each item is the sum of a set of reusable component representations, as illustrated in Fig. 1. We can write this as:

$$\mathbf{x}_i \sim \sum_\alpha s_{\alpha,i}\mathbf{w}_\alpha$$

or equivalently write as a matrix factorization:

$$\mathbf{X} \sim \mathbf{W}\mathbf{S} \tag{1}$$

Similar models There are many well-known matrix factorizations with a similar set of constraints as ours, and other techniques with conceptually similar goals. Our model is clearly a special case of semi[1]-nonnegative matrix factorization (NMF) [11], sparse autoencoders [23], or sparse PCA. These sparse nonnegative methods have proven very useful, but their sparsity penalty often results in spike-and-slab distributions over activations, which can lead to ambiguous category boundaries. Conceptually, our model is also a close neighbor of similarity-based hierarchical [9] or overlapping clustering [61], and of models for structure discovery [25, 32, 26]. See A.1 for more connections to the literature.

Identifiability When is a given set of latent concepts, $\mathbf{S}^*$, uniquely recoverable from $\mathbf{X}$? Unfortunately, a tractable, necessary and sufficient condition does not exist [10], but several sufficient conditions have been derived [31, 60, 61, 65]. When the conditions are met there are exact algorithms available, but this situation is more restrictive than we would like. For example, the case of hierarchically structured concepts is not identifiable.

2.1 Objective functions

Feature MSE The most straightforward objective function to choose would be the mean squared error (MSE) between the data and binary reconstruction:

$$\mathcal{L}_{MSE}(\mathbf{S}, \mathbf{W}) = \frac{1}{p}\|\mathbf{X} - \mathbf{W}\mathbf{S}\|_F^2$$

Theoretical arguments and empirical results in machine learning [50, 59] and neuroscience [37, 4] suggest that subspaces coding for different latent variables should be roughly orthogonal, and it also is very helpful in practice to impose that $\mathbf{W}^T\mathbf{W} = \sigma\mathbb{I}$ with $\sigma > 0$.

For optimization, where we iterate over items or batches of items, it is helpful to re-write the loss in terms of individual data points:

$$\mathcal{L}_{MSE}(\mathbf{S}, \mathbf{W}) = \frac{1}{p}\sum_{i=1}^{p}(\sigma\mathbf{1} - 2\mathbf{W}^T\mathbf{x}_i)^T\mathbf{s}_i \tag{2}$$

where we have made use of both the orthogonality of $\mathbf{W}$ and the fact that $\mathbf{s}^T\mathbf{s} = \mathbf{1}^T\mathbf{s}$ when $\mathbf{s}$ is binary. Being a sum over p linear functions of $\mathbf{s}$ will make available very fast optimization algorithms.

Kernel MSE Since we are assuming that $\mathbf{W}$ is roughly orthogonal, we can reframe the problem as one of aligning the geometry of $\mathbf{S}$ to that of $\mathbf{X}$. A common way to quantify geometric alignment is the centered kernel alignment (CKA) [8, 30], which is related (but not identical) to the feature MSE with orthogonal $\mathbf{W}$ [18]. If we allow $\mathbf{S}$ to be scaled by a positive factor, $\sqrt{\sigma}$, then maximizing the CKA is equivalent to minimizing the MSE between the centered kernel matrices:

$$\mathcal{L}_{CKA}(\mathbf{S}, \sigma) = \frac{1}{p^2}\|\bar{\mathbf{X}}^T\bar{\mathbf{X}} - \sigma\bar{\mathbf{S}}^T\bar{\mathbf{S}}\|_F^2$$

[1] or standard NMF if we further restrict $\mathbf{W}$ to be non-negative

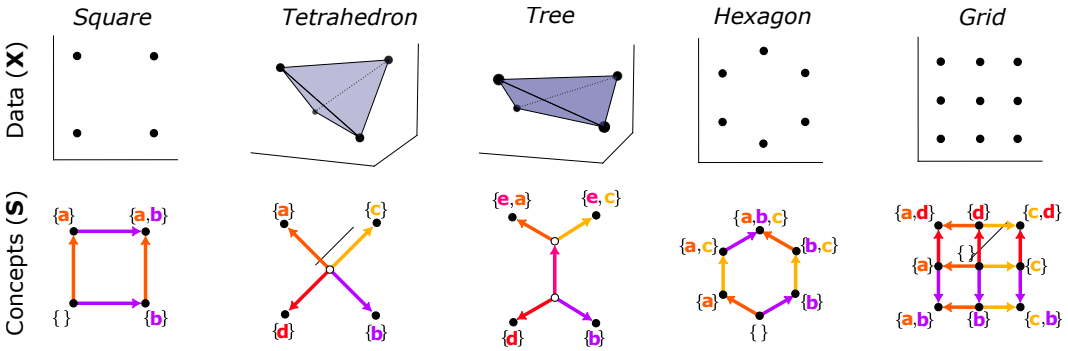


Figure 2: Geometries and associated graphs. We show the manually-computed analograms of the best-fitting concepts for each data geometry. Set labels on each node indicate the nonzero indices of its binary representation. Notice how the labels can be recovered by removing all edges of a certain color and labeling all nodes on one side of the resulting partition with the associated concept.

where $\bar{\mathbf{X}} \doteq \mathbf{X} - \langle \mathbf{x} \rangle \mathbf{1}^T$ and $\bar{\mathbf{S}} \doteq \mathbf{S} - \langle \mathbf{s} \rangle \mathbf{1}^T$. The per-datum loss is:

$$\mathcal{L}_{CKA}(\mathbf{S}, \sigma) = \frac{1}{p} \sum_{i=1}^{p} \sigma^2 \, \bar{\mathbf{s}}_i^T \Sigma^{(\mathbf{S})} \bar{\mathbf{s}}_i - 2\sigma \, \mathbf{x}_i^T \Sigma^{(\mathbf{XS})} \bar{\mathbf{s}}_i \tag{3}$$

where we introduce the empirical covariance, $\Sigma^{(\mathbf{S})} \doteq \frac{1}{p} \bar{\mathbf{S}} \bar{\mathbf{S}}^T$, and cross-covariance, $\Sigma^{(\mathbf{XS})} \doteq \frac{1}{p} \bar{\mathbf{X}} \bar{\mathbf{S}}^T$. When p is large each item has a negligible effect on these matrices, but when p is small some subtle corrections are needed which are described in Appendix A.2.

2.2 A set of concepts has an associated graph

When p is large, it becomes hard to make sense of the concepts purely in terms of their many members. We want a visualization tool in which the global structure of the concepts is clear. For instance, hierarchical clustering can be visualized with a dendrogram, *i.e.* a tree on which observations are leaves and cluster assignments can be recovered by cutting the tree at a certain depth. In a similar way, we will define a graph that encodes the structure of $\mathbf{S}$.

The process is illustrated in Fig. 1b. First, note that the set of all k-dimensional binary vectors forms a hypercube with 2^k nodes, which defines a graph, G, if we connect all pairs of nodes which differ by only one element (colored lines). The columns of $\mathbf{S}$ are nodes in this k-cube graph, but usually a very small subset (solid nodes). We can imagine finding the smallest subgraph which (1) contains all columns of $\mathbf{S}$ and (2) preserves shortest path distance between all columns of $\mathbf{S}$ (solid lines). This is called the "isometric hull" of $\mathbf{S}$ in G, which we will refer to as the *analogram* for short.

The analogram is a very useful description of the global structure of our binary representation, and we show some examples of geometries with their associated analograms in Fig. 2. There is a very intuitive relationship between the geometric structure and the resulting graph structure, and the binary concept labels can be uniquely recovered from the analogram [10]. Unfortunately isometric hulls are NP hard to find [27], so finding the smallest analogram for a given $\mathbf{S}$ is intractable. Nevertheless, we develop a heuristic which works well for certain structures, based on identifying sub/superset relations, and is described in the Appendix A.4.

2.3 Regularization

Outside of some special low-dimensional cases, there will be many possible solutions to our factorization (1), which renders interpretation very difficult. In addition to preventing overfitting, regularization can help "break the tie" and make the model well-specified.

Sparsity To encourage sparse concepts, we can add an L1 penalty to the loss: $\mathcal{L}_{l1}(\mathbf{S}) = \mathbf{1}^T \mathbf{S} \mathbf{1}$.

Hierarchy We find it is much more effective to control the relationships between concepts. partially overlapping concepts, or to penalize the number of unique concepts, we introduce a new regularization scheme we call hierarchical regularization. We define the following penalty: $\mathcal{L}_H(\mathbf{S}) = \sum_{\alpha,\beta} \min\{\langle \mathbf{s}_\alpha \mathbf{s}_\beta \rangle, \langle \tilde{\mathbf{s}}_\alpha \mathbf{s}_\beta \rangle, \langle \mathbf{s}_\alpha \tilde{\mathbf{s}}_\beta \rangle\}$ which penalizes any concept co-occurrences which cannot be absorbed by subset/superset relations. Intuitively, this function will be minimized (= zero) when every pair of concepts is either disjoint, a subset, a superset, or identical. Optimizing this function turns out to involve a simple quadratic function (Appendix A.4).

3 Optimization: alternating least squares with simulated annealing

Being a challenging combinatorial problem, we cannot expect efficient solutions that work in every situation. There are already remarkably effective exact methods for very identifiable data, but they do not always fail gracefully when their assumptions are violated. For general data, one can use alternating least squares, which alternates between optimizing each matrix holding the other fixed. ALS has been found to be very sensitive to initialization in the case of BMF [29, 61] due to its discrete nature, but we have found that simulated annealing (see Appendix A.3) can dramatically increase performance without too much cost in time.

Feature MSE loss When we enforce orthogonal $\mathbf{W}$ and use only sparsity regularization, each step of optimization has a closed form update as shown in Algorithm 1. It is essentially iterative quantization [16], an algorithm developed in the hashing literature, though we find the addition of simulated annealing and regularization are critical.

Kernel MSE loss While the update of the continuous scale parameter, σ, has a closed form, the update of $\mathbf{S}$ is an NP hard optimization problem. Recall that the loss (Eq. 3) is a quadratic function of the binary vectors $\mathbf{s}_i$. Optimizing such functions is a widely-studied problem [54]. But our particular quadratic function, whose quadratic term, $\Sigma^{(\mathbf{S})}$, is the covariance matrix of binary vectors, suggests a very simple heuristic: minimizing such a function is the job of a Hopfield network[2][21, 22].

In particular, if we define the coefficients:

$$\mathbf{J}_{\alpha,\beta} = \begin{cases} \Sigma^{(\mathbf{S})}_{\alpha,\beta} & \text{if } \alpha \neq \beta \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

$$\mathbf{h}_\alpha = \mathbf{x}_i^T \Sigma^{(\mathbf{XS})}_\alpha - \sigma \langle \mathbf{s} \rangle_\alpha \tag{5}$$

then doing greedy minimization of $\mathbf{s}^T \mathbf{J} \mathbf{s} - 2\mathbf{h}^T \mathbf{s}$ results in the dynamics of a Hopfield network (Algorithm 2). Note again that these specific updates are in the case of large p, and small-dataset effects are accounted for in the more careful derivations of Appendix A.2. In either case, the resulting update of $\mathbf{S}$ has the same complexity as a matrix-vector multiplication.

Algorithm 1 Binary PCA

function STEP($\mathbf{S}$, $\mathbf{X}$, $T > 0$)
  $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T = \mathbf{S}\mathbf{X}^T$ ▷ *SVD*
  $\mathbf{W} = \mathbf{U}\mathbf{V}^T$
  $\sigma = \frac{\operatorname{tr} \mathbf{\Sigma}}{\mathbf{1}^T \mathbf{S} \mathbf{1}}$
  $\mathbf{S} \sim \text{Bern}(\frac{1}{T}\left[2\mathbf{W}^T\mathbf{X} - \sigma\right])$
  return $\mathbf{S}$

Algorithm 2 Kernel BMF

function STEP($\mathbf{S}$, $\mathbf{X}$, $T > 0$)
  $\sigma = \langle \bar{\mathbf{S}}^T \bar{\mathbf{S}}, \bar{\mathbf{X}}^T \bar{\mathbf{X}} \rangle_F / \|\bar{\mathbf{S}}^T \bar{\mathbf{S}}\|_F^2$
  for $i = 1, ..., p$; $\alpha = 1, ..., k$ **do**
    $\mathbf{J}, \mathbf{h} = (4), (5)$
    $\mathbf{S}_{\alpha,i} \sim \text{Bern}\left(\frac{1}{T}\left[\mathbf{h}_\alpha - \sigma \sum_\beta \mathbf{J}_{\alpha\beta} \mathbf{S}_{\beta,i}\right]\right)$
  return $\mathbf{S}$

4 Numerical experiments

To assess the effectiveness of our heuristics, we do extensive simulations of synthetic data with different structures and identifiability properties. All experiments were done on a CPU cluster.

[2]The classic Hopfield model lacks the continuous input term, $\mathbf{x}_i$, but our "input weights", $\Sigma^{(\mathbf{XS})}$, are also a (cross-) covariance. So, in the large p case, all the weights can be learned with Hebbian rules.

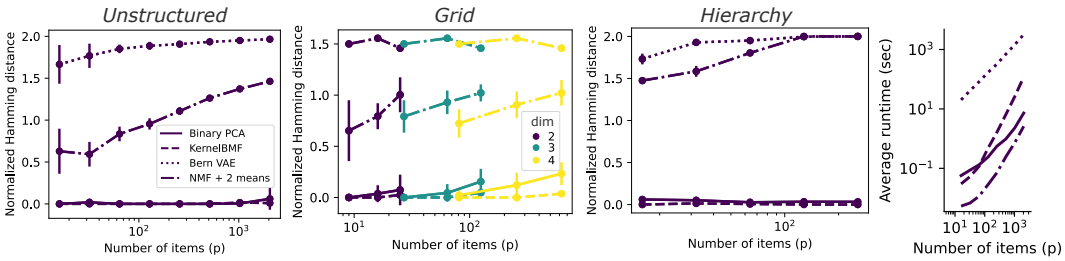


Figure 3: Synthetic data simulations, averaging over 12 random seeds with standard deviation error bars. Data are generated by drawing a true $\mathbf{S}$ according to the specified structure, drawing a random orthonormal $\mathbf{W}$, and adding iid Gaussian noise to $\mathbf{WS}$ given a desired SNR (in this case 10). Average run times are shown on the far right. Note that there is non-polynomial behavior for small p due to suboptimal implementation, but the scaling of each algorithm can be seen into the larger p.

Metrics and comparison models As baselines for our two algorithms, we will compare against a Bernoulli VAE [43] and semi-NMF [11] with feature-wise 2-means clustering. We do not include exact methods because they have been extensively evaluated recently [29, 61] on unstructured data, and are unlikely to be applicable on most structured data. To the extent possible, we apply the same regularization to each method and manually pick the best hyper-parameters independently for each method. Reported results are the average of 12 random seeds.

In each of our experiments, we will be comparing the concepts discovered by a model, $\mathbf{S}$, to the ground truth concepts, $\mathbf{S}^*$. Our criterion will be the Hamming distance up to permutation of the concept indices: $d_H^\pi(\mathbf{S}^*, \mathbf{S}) = \min_\pi \sum_\alpha d_H(\mathbf{s}_a^*, \mathbf{s}_{\pi[\alpha]})$ where π is a k-permutation and d_H is the Hamming distance. This can be computed very efficiently as it is a linear assignment problem. We also normalize by the sum of the target concept $\mathbf{s}_\alpha^*$ so that results are more comparable as p grows.

Unstructured concepts The most basic and unrealistic case is that of unstructured concepts drawn from independent Bernoullis: $\mathbf{S}_{\alpha,i} \sim \text{Bern}(p = 0.5)$. As long as the number of concepts, k, is small enough ($k < \sqrt{2p}$) the BMF problem almost certainly has a unique solution [31]. Despite achieving perfect training loss, the VAE does not find the ground truth answer; neither does NMF; both our algorithms score nearly perfectly across all tested instance sizes.

Grid-structured concepts We next consider concepts with a grid structure, like in Fig. 2 but generalized to n values and m dimensions. We see a very similar pattern of performance, but with the Alg. 2 showing some advantage. The models tend to do worse as n grows, conditional on m.

Hierarchically-structured concepts Finally we generate concepts with a hierarchical structure in which top-level categories are randomly sub-divided recursively until they become singletons. The resulting data geometry is high-dimensional and could have a large number of solutions. For our BMF algorithms we set the $\mathcal{L}_H$ regularization coefficient to be large (close to 1). This proves to be the most difficult task, with only the BMF algorithms achieving reasonable recovery.

5 Application: exploratory analysis of data

Here we will examine the utility of our concept discovery approach as an exploratory data analysis tool. To provide ourselves with a sense of 'ground truth', we will focus on three well-studied, very different types of data in which there are prior expectations of the underlying concepts: the Indo-European languages, representations of a large language model (LLM), and the fly connectome.

Indo-European cognates To test the model on data that is likely hierarchical, we turn to a kind of phylogenetic data: cognates. A cognate group is a set of words share a common ancestor, such as English 'water' and German 'wasser'. When two languages have many cognates in common, it is indicative (but not the only factor) that they diverged recently in history. We will see if BMF

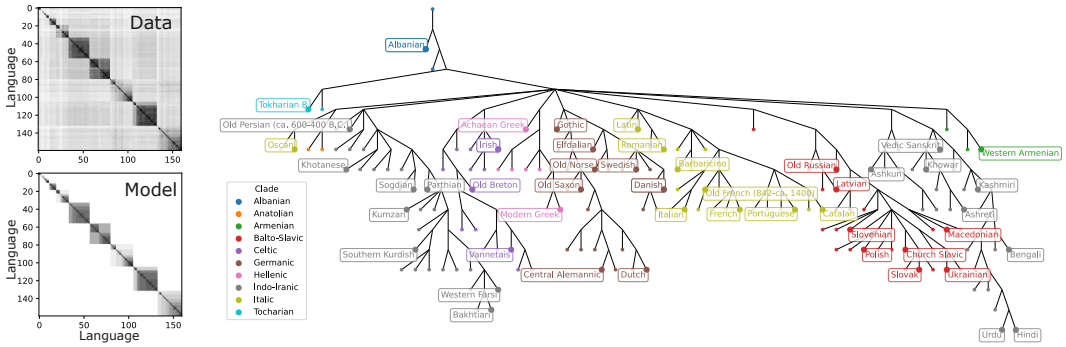


Figure 4: Best model fit to Indo-european cognacy data, with the analogram shown on the right.

applied to Indo-European cognate data finds a structure in accordance with known evolution of these languages.

We use the large manually curated dataset of Indo-European cognates compiled by [19]. In this data, with 160 languages and 5000 cognate groups. $\mathbf{X}_{lc} \in \{0, 1\}$ indicates whether language l has a member in cognate group c. When we run the Kernel BMF algorithm (2) on the data in a highly over-parameterized[3] regime (900 concepts), the $\mathcal{L}_H$ regularization results in only 266 unique concepts, which are themselves hierarchically organized. We achieve a strong CKA of 0.98 between the model fit and the data (4 and a normalized Bures similarity [18] (NBS) of 0.99.

When we plot the analogram of $\mathbf{S}$ we see a tree whose main subtrees correspond to the 'ground truth' clade labels (Fig. 4). There are some mistakes at this level, namely the sundering of the Indo-Iranian languages and the ejection of Oscan and Umbrian from the Italics.

At a finer level, there are some sensible sub-differentiations (e.g. the organization of the Romance languages) as well as some mistakes (e.g. Old Russian is not the ancestor of all the Slavic languages). Discrepancies compared with linguistics knowledge simply reflect the fact that actual reconstructions of the Indo-european family tree use many sources of data like archaeology and written history; this analysis is demonstrating what kind of information can be gleamed from cognacy alone.

Language model representations Word embeddings have long been known to exhibit compositional structure [45, 52, 48, 51, 63], and so we begin by analyzing the representations of the Gemma LLM [44]. We specifically use the 2 billion parameter variant of Gemma-2 from Huggingface[64].

Among the many representations in an LLM, we used the whitened readout weights since they are a context-free representation of each token, are causally related to the network output [51], and have recently been shown to encode hierarchical categories in an orthogonal manner [50]. Specifically, if $\mathbf{U}$ are the $d \times w$ weights from the final layer of Gemma to the output logits, with the mean column subtracted, then the canonical representation introduced by [51] is $\mathbf{X} = (\mathbf{U}\mathbf{U}^T)^{-1/2}\mathbf{U}$.

We analyzed a subset of the large Gemma vocabulary, based on English words taken from WordNet [14]. We do not use the WordNet hierarchy in any way other than to select all the words considered to be subtypes of "person". There are 1794 "people" in WordNet that also appear as whole words Gemma's vocabulary; we average the representations over the capitalized and/or plural forms of each words. The result is a 2304 dimensions $\times$ 1794 words data matrix.

Using the Kernel BMF algorithm (2) with 900 concepts and a mild regularization factor of 0.1 provided the best fit among those we tried, achieving an NBS of 0.77 and CKA of 0.66. Despite the $\mathcal{L}_H$ regularization (2.3), there was no reduction in effective concepts or substantial hierarchy displayed, suggesting that a larger k may be beneficial. Unfortunately, time is prohibitive in data of this size. The discovered concepts were sparse, with a median size of 6 and maximum of 108.

Our approach for exploring the model is to look at a subset of words, interpreting the (relatively few) active concepts, and then seeing how well they extend to the rest of the dataset. We begin with the

[3]We see qualitatively similar results with smaller k, but this empirically seems to help the learning dynamics of the model. The regularization helps reduce the effective number of concepts.

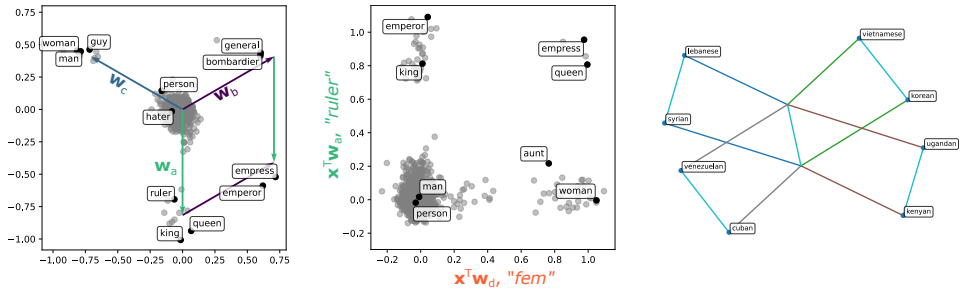


Figure 5: Various projectiosn of Gemma-2-2B embeddings.

famous quadruplet, the words “king”, “queen”, “man”, and “woman”. In previous work they have been analyzed in terms of two concepts (‘class’ and ‘gender’); our model finds multiple concepts which look the same when restricted to these four words, but apply differently to the rest of the data.

The first ‘class’-like concept includes ‘ruler’, ‘empress’, ‘emperor, and generally words associated with ‘ruler’. There is a distinct concept which separates ‘emperor’ and ‘empress’ from these, and it includes martial words like ‘general’ and ‘bombardier’. Another, which groups together ‘man’ and ‘woman’, includes generic referents like ‘guy’, ‘gal’, ‘gent’ and so on. In Fig. 5 we show the full dataset projected onto the $\mathbf{w}_\alpha$ vectors associated with these concepts (rotated for visualization).

We only find one ‘gender’-like concept in the model, and we show the data projection against that of the ‘imperial’ concept in the middle of Fig. 5. The relative sparsity of this concept reflects the fact that most words in the vocabulary are not explicitly gendered feminine.

Finally, we can search for further quadruplets by selecting pairs of words which only differ by one concept. Due to noise, this will be relatively few such pairs, but we do find some. We see that ‘grandchild‘:‘grandson’::‘grandparent’:‘grandfather’, which appears to disentangle masculine inflection from seniority. There is also a set of 8 ethnic identifiers which are organized in a remarkably regular pattern (Fig. 5 right) that corresponds to continental categories and a fifth concept that is more difficult to parse.

Fruit fly optic lobe connectome The full wiring diagram of the Drosophila brain has recently been published[12], and tools for trawling the data are needed. In our context we must ask, what would a “concept” be in a wiring diagram? Biological brains are built from a huge number of cell types[41] which can be based on, for example, morphology, developmental lineage, and combinations of expressed genes [17]. A neuron’s connectivity can be influenced by these genetic factors in a combinatorial way [36], which could potentially lead to compositionality in the cell type-to-cell type connectivity. The “concepts” in this case would correspond to connectivity motifs which appear in different morphological cell types. To see if our method can uncover such structure, we will look at the intrinsic cell types of the optic lobe which have recently been found to exhibit some connectivity-based clustering [42].

The data from [42] consist of cell type connectivity fractions, i.e. $\mathbf{C}_{ij}$ gives the fraction of synapses from cell type i that are directed to cell type j for 229 cell types. The matrix we model is $\mathbf{X} = [\mathbf{C}, \mathbf{C}^T]$. We found better performance in this case when, rather than the linear kernel, $\mathbf{X}^T\mathbf{X}$, we use a non-linear kernel, the weighted Jaccard index. By some trial and error on the hyper-parameters, we achieve an NBS of 0.82 and a CKA of 0.74 to the original data, using mild regularization of 0.1 and 140 unique/nontrivial concepts (Fig. 6a).

While the resulting structure is not fully hierarchical, we can see that there is some structure in the subset and superset relations. In Fig. 6b we show the ‘hypernymy’ graph of the concepts, i.e. the transitive reduction of the subset relation graph. There are several isolated concepts with no descendants, and two large clusters of inter-related concepts. We focus in particular on the set of descendants of a top-level concept, 139 (highlighted in red and Fig. 6d).

To see what these concepts correspond to in terms of connectivity, we can inspect the weights of a sparse non-negative regression of $\mathbf{S}$ onto $\mathbf{X}$ (Fig. 6c, $R^2 \approx 0.81$). The top-level concept 139 seems more or less defined as ‘projects onto TmY14’, a visual projection neuron [62]. We can look

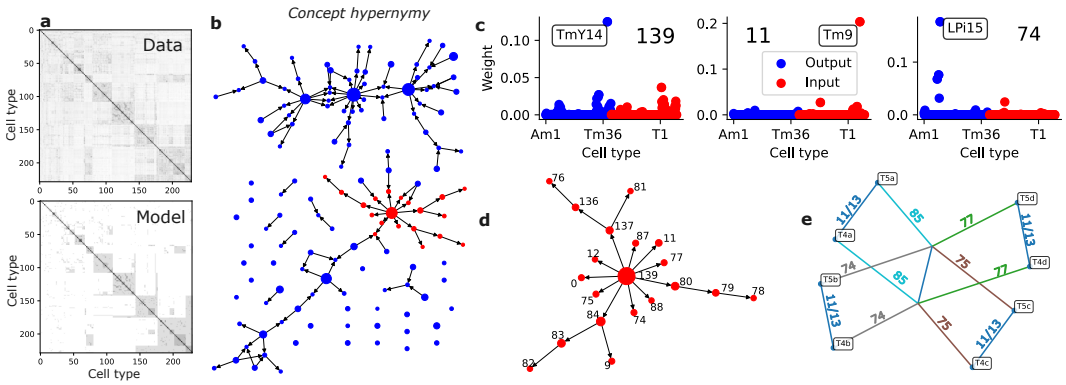


Figure 6: (**a**) Comparison of the data kernel (top) and the model kernel (bottom) sorted according to top-level concepts. (**b**) The 'concept hypernymy' graph, showing direct subset/superset relations between discovered concepts. Note that, among the 140 distinct concepts, there are only 64 top-level (without superset) concepts and 33 connected components. We highlight one particular top-level concept, 139. (**c**) The regression coefficients of a sparse non-negative regression, $\mathbf{X} \sim \mathbf{WS}$. Each cell type is represented as two dots, one for output synapses (blue) and input (red). (**d**) The subgraph for concept 139. (**e**) Analogram of the 8 cell types which participate in the 5 concepts of interest.

further at two sub-concepts; concept 11 is 'defined' as receiving inputs from Tm9 cells (part of the visual motion pathway [5]); concept 74 by sending outputs to LPi15 cells (a newly identified type of interneuron [42]). Looking at the cell types belonging to these categories reveals that they contain the T5a-d cells and T4b/T5b cells respectively. We find counterparts to these categories (the T4a-d cells) and realize that they are part of a documented visual motion subsystem [5]. In accordance with the literature [39], our model suggests that these 8 cell types are organized according to a conjunction of 6 factors; being in the ON pathway (T4) or OFF (T5), and tuning to cardinal directions (forward, a; backward, b; up, c; and down, d). This is summarized in the discovered analogram (Fig. 6e).

6 Discussion

Here we studied the problem of turning a continuous representation into a logical one. We provided two simple algorithms with complementary benefits and demonstrate their efficacy. In the process, we develop tools for detecting and visualizing higher-order structure in the data. When applied to three well-studied but realistic datasets, we find that interpretable structure is readily forthcoming.

While the baselines we examined were not very effective in our synthetic data experiments, some version of them might be. In particular, the case of non-negative data should be investigated more thoroughly – when we generate data with non-negative $\mathbf{W}$, then using standard NMF works remarkably well (but only in the case of iid Bernoulli latents). So, while semi-NMF was not so accurate, it might be possible to augment the standard NMF algorithms to work better as a BMF algorithm as well.

Compact visualizations are essential for an exploratory analysis tool, and this is an aspect of our method which needs some improvement. Analograms are difficult to find for general $\mathbf{S}$ matrices, but we are sometimes able to find other ways to summarize the structure (like the drosophila 'hypernymy' graph). Having more robust and well-though-out methods will be essential moving forward.

Finally, speculatively, it is important to note that, insofar as this is a computational model of concept learning, it is unlikely to capture essential features of human concept learning [33, 34, 53]. In particular, there are limitations to what can be inferring in a purely unsupervised way [38] and our concepts are not task-adapted or causal. One important extension would be to incorporate our method into a supervised model, as a kind of concept bottleneck model [28] but with data-driven rather than hand-coded concepts.

References

[1] Altabaa, A., Webb, T. W., Cohen, J. D., and Lafferty, J. (2024). Abstractors and relational cross-attention: An inductive bias for explicit relational reasoning in transformers. In *The Twelfth International Conference on Learning Representations*.

[2] Andoni, A. and Indyk, P. (2006). Near-optimal hashing algorithms for approximate nearest neighbor in high dimensions. In *2006 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS'06)*, pages 459–468.

[3] Anil, C., Wu, Y., Andreassen, A., Lewkowycz, A., Misra, V., Ramasesh, V., Slone, A., Gur-Ari, G., Dyer, E., and Neyshabur, B. (2022). Exploring length generalization in large language models. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35. Curran Associates, Inc.

[4] Bernardi, S., Benna, M. K., Rigotti, M., Munuera, J., Fusi, S., and Salzman, C. D. (2020). The geometry of abstraction in the hippocampus and prefrontal cortex. *Cell*.

[5] Borst, A. and Groschner, L. N. (2023). How flies see motion. *Annual Review of Neuroscience*, 46(1):17–37.

[6] Chollet, F. (2019). On the measure of intelligence.

[7] Courellis, H. S., Minxha, J., Cardenas, A. R., Kimmel, D. L., Reed, C. M., Valiante, T. A., Salzman, C. D., Mamelak, A. N., Fusi, S., and Rutishauser, U. (2024). Abstract representations emerge in human hippocampal neurons during inference. *Nature*, (8026).

[8] Cristianini, N., Shawe-Taylor, J., Elisseeff, A., and Kandola, J. (2001). On kernel-target alignment. In Dietterich, T., Becker, S., and Ghahramani, Z., editors, *Advances in Neural Information Processing Systems*, volume 14. MIT Press.

[9] Dasgupta, S. (2016). A cost function for similarity-based hierarchical clustering. In *Proceedings of the Forty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '16. Association for Computing Machinery.

[10] Deza, M. M. and Laurent, M. (1997). *Geometry of cuts and metrics*. Springer.

[11] Ding, C. H., Li, T., and Jordan, M. I. (2010). Convex and semi-nonnegative matrix factorizations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(1):45–55.

[12] Dorkenwald, S., Matsliah, A., Sterling, A. R., Schlegel, P., Yu, S.-c., McKellar, C. E., Lin, A., Costa, M., Eichler, K., Yin, Y., Silversmith, W., Schneider-Mizell, C., Jordan, C. S., Brittain, D., Halageri, A., Kuehner, K., Ogedengbe, O., Morey, R., Gager, J., Kruk, K., Perlman, E., Yang, R., Deutsch, D., Bland, D., Sorek, M., Lu, R., Macrina, T., Lee, K., Bae, J. A., Mu, S., Nehoran, B., Mitchell, E., Popovych, S., Wu, J., Jia, Z., Castro, M. A., Kemnitz, N., Ih, D., Bates, A. S., Eckstein, N., Funke, J., Collman, F., Bock, D. D., Jefferis, G. S. X. E., Seung, H. S., Murthy, M., Lenizo, Z., Burke, A. T., Willie, K. P., Serafetinidis, N., Hadjerol, N., Willie, R., Silverman, B., Ocho, J. A., Bañez, J., Candilada, R. A., Kristiansen, A., Panes, N., Yadav, A., Tancontian, R., Serona, S., Dolorosa, J. I., Vinson, K. J., Garner, D., Salem, R., Dagohoy, A., Skelton, J., Lopez, M., Capdevila, L. S., Badalamente, G., Stocks, T., Pandey, A., Akiatan, D. J., Hebditch, J., David, C., Sapkal, D., Monungolh, S. M., Sane, V., Pielago, M. L., Albero, M., Laude, J., dos Santos, M., Vohra, Z., Wang, K., Gogo, A. M., Kind, E., Mandahay, A. J., Martinez, C., Asis, J. D., Nair, C., Patel, D., Manaytay, M., Tamimi, I. F. M., Lim, C. A., Ampo, P. L., Pantujan, M. D., Javier, A., Bautista, D., Rana, R., Seguido, J., Parmar, B., Saguimpa, J. C., Moore, M., Pleijzier, M. W., Larson, M., Hsu, J., Joshi, I., Kakadiya, D., Braun, A., Pilapil, C., Gkantia, M., Parmar, K., Vanderbeck, Q., Salgarella, I., Dunne, C., Munnelly, E., Kang, C. H., Lörsch, L., Lee, J., Kmecova, L., Sancer, G., Baker, C., Joroff, J., Calle, S., Patel, Y., Sato, O., Fang, S., Salocot, J., Salman, F., Molina-Obando, S., Brooks, P., Bui, M., Lichtenberger, M., Tamboboy, E., Molloy, K., Santana-Cruz, A. E., Hernandez, A., Yu, S., Diwan, A., Patel, M., Aiken, T. R., Morejohn, S., Koskela, S., Yang, T., Lehmann, D., Chojetzki, J., Sisodiya, S., Koolman, S., Shiu, P. K., Cho, S., Bast, A., Reicher, B., Blanquart, M., Houghton, L., Choi, H., Ioannidou, M., Collie, M., Eckhardt, J., Gorko, B., Guo, L., Zheng, Z., Poh, A., Lin, M., Taisz, I., Murfin, W., Díez, S., Reinhard, N., Gibb, P., Patel, N., Kumar, S., Yun, M., Wang, M., Jones, D., Encarnacion-Rivera, L., Oswald, A., Jadia, A., Erginkaya, M., Drummond, N., Walter, L., Tastekin, I., Zhong, X., Mabuchi, Y., Figueroa Santiago, F. J., Verma, U., Byrne, N., Kunze, E., Crahan, T., Margossian, R., Kim, H., Georgiev, I., Szorenyi, F., Adachi, A., Bargeron, B., Stürner, T., Demarest, D., Gür, B., Becker, A. N., Turnbull, R., Morren, A., Sandoval, A., Moreno-Sanchez, A., Pacheco, D. A., Samara, E.,

Croke, H., Thomson, A., Laughland, C., Dutta, S. B., de Antón, P. G. A., Huang, B., Pujols, P., Haber, I., González-Segarra, A., Choe, D. T., Lukyanova, V., Mancini, N., Liu, Z., Okubo, T., Flynn, M. A., Vitelli, G., Laturney, M., Li, F., Cao, S., Manyari-Diaz, C., Yim, H., Duc Le, A., Maier, K., Yu, S., Nam, Y., Bąba, D., Abusaif, A., Francis, A., Gayk, J., Huntress, S. S., Barajas, R., Kim, M., Cui, X., Sterne, G. R., Li, A., Park, K., Dempsey, G., Mathew, A., Kim, J., Kim, T., Wu, G.-t., Dhawan, S., Brotas, M., Zhang, C.-h., Bailey, S., Del Toro, A., Yang, R., Gerhard, S., Champion, A., Anderson, D. J., Behnia, R., Bidaye, S. S., Borst, A., Chiappe, E., Colodner, K. J., Dacks, A., Dickson, B., Garcia, D., Hampel, S., Hartenstein, V., Hassan, B., Helfrich-Forster, C., Huetteroth, W., Kim, J., Kim, S. S., Kim, Y.-J., Kwon, J. Y., Lee, W.-C., Linneweber, G. A., Maimon, G., Mann, R., Noselli, S., Pankratz, M., Prieto-Godino, L., Read, J., Reiser, M., von Reyn, K., Ribeiro, C., Scott, K., Seeds, A. M., Selcho, M., Silies, M., Simpson, J., Waddell, S., Wernet, M. F., Wilson, R. I., Wolf, F. W., Yao, Z., Yapici, N., and Zandawala, M. (2024). Neuronal wiring diagram of an adult brain. *Nature*, 634(8032).

[13] Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., Grosse, R., McCandlish, S., Kaplan, J., Amodei, D., Wattenberg, M., and Olah, C. (2022). Toy models of superposition.

[14] Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. Bradford Books.

[15] Fodor, J. A. and Pylyshyn, Z. W. (1988). Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1):3–71.

[16] Gong, Y., Lazebnik, S., Gordo, A., and Perronnin, F. (2013). Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

[17] Harris, K. D., Hochgerner, H., Skene, N. G., Magno, L., Katona, L., Bengtsson Gonzales, C., Somogyi, P., Kessaris, N., Linnarsson, S., and Hjerling-Leffler, J. (2018). Classes and continua of hippocampal ca1 inhibitory neurons revealed by single-cell transcriptomics. *PLOS Biology*, 16(6):e2006387.

[18] Harvey, S. E., Larsen, B. W., and Williams, A. H. (2023). Duality of bures and shape distances with implications for comparing neural representations.

[19] Heggarty, P., Anderson, C., Scarborough, M., King, B., Bouckaert, R., Jocz, L., Kümmel, M. J., Jügel, T., Irslinger, B., Pooth, R., Liljegren, H., Strand, R. F., Haig, G., Macák, M., Kim, R. I., Anonby, E., Pronk, T., Belyaev, O., Dewey-Findell, T. K., Boutilier, M., Freiberg, C., Tegethoff, R., Serangeli, M., Liosis, N., Stroński, K., Schulte, K., Gupta, G. K., Haak, W., Krause, J., Atkinson, Q. D., Greenhill, S. J., Kühnert, D., and Gray, R. D. (2023). Language trees with sampled ancestors support a hybrid model for the origin of indo-european languages. *Science*, 381(6656).

[20] Higgins, I., Chang, L., Langston, V., Hassabis, D., Summerfield, C., Tsao, D., and Botvinick, M. (2021). Unsupervised deep learning identifies semantic disentanglement in single inferotemporal face patch neurons. *Nature communications*, 12(1):6456.

[21] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*.

[22] Hopfield, J. J. and Tank, D. W. (1985). "neural" computation of decisions in optimization problems. *Biological Cybernetics*.

[23] Huben, R., Cunningham, H., Smith, L. R., Ewart, A., and Sharkey, L. (2024). Sparse autoencoders find highly interpretable features in language models. In *The Twelfth International Conference on Learning Representations*.

[24] Kaufman, M. T., Benna, M. K., Rigotti, M., Stefanini, F., Fusi, S., and Churchland, A. K. (2022). The implications of categorical and category-free mixed selectivity on representational geometries. *Current Opinion in Neurobiology*.

[25] Kemp, C. and Tenenbaum, J. B. (2008). The discovery of structural form. *Proceedings of the National Academy of Sciences*.

[26] Kemp, C. and Tenenbaum, J. B. (2009). Structured statistical models of inductive reasoning. *Psychological Review*, 116(1):20–58.

[27] Knauer, K. and Nisse, N. (2019). Computing metric hulls in graphs. *Discrete Mathematics and Theoretical Computer Science*.

[28] Koh, P. W., Nguyen, T., Tang, Y. S., Mussmann, S., Pierson, E., Kim, B., and Liang, P. (2020). Concept bottleneck models.

[29] Kolomvakis, C. and Gillis, N. (2023). Robust binary component decompositions. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

[30] Kornblith, S., Norouzi, M., Lee, H., and Hinton, G. (2019). Similarity of neural network representations revisited.

[31] Kueng, R. and Tropp, J. A. (2021). Binary component decomposition part i: The positive-semidefinite case. *SIAM Journal on Mathematics of Data Science*.

[32] Lake, B. M., Lawrence, N. D., and Tenenbaum, J. B. (2018). The emergence of organizing structure in conceptual representation. *Cognitive Science*, 42(S3):809–832.

[33] Lake, B. M., Salakhutdinov, R., and Tenenbaum, J. B. (2015). Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338.

[34] Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40:e253.

[35] Levy, O. and Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 27.

[36] Li, J., Han, S., Li, H., Udeshi, N. D., Svinkina, T., Mani, D., Xu, C., Guajardo, R., Xie, Q., Li, T., et al. (2020). Cell-surface proteomic profiling in the fly brain uncovers wiring regulators. *Cell*, 180(2):373–386.

[37] Lindsey, J. W. and Issa, E. B. (2024). Factorized visual representations in the primate visual system and deep neural networks. *Elife*, 13:RP91685.

[38] Locatello, F., Bauer, S., Lucic, M., Rätsch, G., Gelly, S., Schölkopf, B., and Bachem, O. (2019). Challenging common assumptions in the unsupervised learning of disentangled representations.

[39] Maisak, M. S., Haag, J., Ammer, G., Serbe, E., Meier, M., Leonhardt, A., Schilling, T., Bahl, A., Rubin, G. M., Nern, A., Dickson, B. J., Reiff, D. F., Hopp, E., and Borst, A. (2013). A directional tuning map of drosophila elementary motion detectors. *Nature*, 500(7461):212–216.

[40] Mao, J., Gan, C., Kohli, P., Tenenbaum, J. B., and Wu, J. (2019). The neuro-symbolic concept learner: Interpreting scenes, words, and sentences from natural supervision. In *International Conference on Learning Representations*.

[41] Masland, R. H. (2004). Neuronal cell types. *Current Biology*, 14(13):R497–R500.

[42] Matsliah, A., Yu, S.-c., Kruk, K., Bland, D., Burke, A. T., Gager, J., Hebditch, J., Silverman, B., Willie, K. P., Willie, R., Sorek, M., Sterling, A. R., Kind, E., Garner, D., Sancer, G., Wernet, M. F., Kim, S. S., Murthy, M., Seung, H. S., David, C., Joroff, J., Kristiansen, A., Stocks, T., Braun, A., Silies, M., Skelton, J., Aiken, T. R., Ioannidou, M., Collie, M., Linneweber, G. A., Molina-Obando, S., Dorkenwald, S., Panes, N., Gogo, A. M., Rastgarmoghaddam, D., Pilapil, C., Candilada, R. A., Serafetinidis, N., Lee, W.-C., Borst, A., Wilson, R. I., Schlegel, P., and Jefferis, G. S. X. E. (2024). Neuronal parts list and wiring diagram for a visual system. *Nature*, 634(8032):166–180.

[43] Mena, F. and Ñanculef, R. (2019). A binary variational autoencoder for hashing. In Nyström, I., Hernández Heredia, Y., and Milián Núñez, V., editors, *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*.

[44] Mesnard, T., Hardin, C., Dadashi, R., Bhupatiraju, S., Pathak, S., Sifre, L., Rivière, M., Kale, M. S., Love, J., Tafti, P., Hussenot, L., Sessa, P. G., Chowdhery, A., Roberts, A., Barua, A., Botev, A., Castro-Ros, A., Slone, A., Héliou, A., Tacchetti, A., Bulanova, A., Paterson, A., Tsai, B., Shahriari, B., Lan, C. L., Choquette-Choo, C. A., Crepy, C., Cer, D., Ippolito, D., Reid, D., Buchatskaya, E., Ni, E., Noland, E., Yan, G., Tucker, G., Muraru, G.-C., Rozhdestvenskiy, G., Michalewski, H., Tenney, I., Grishchenko, I., Austin, J., Keeling, J., Labanowski, J., Lespiau, J.-B., Stanway, J., Brennan, J., Chen, J., Ferret, J., Chiu, J., Mao-Jones, J., Lee, K., Yu, K., Millican, K., Sjoesund, L. L., Lee, L., Dixon, L., Reid, M., Mikuła, M., Wirth, M., Sharman, M., Chinaev, N., Thain, N., Bachem, O., Chang, O., Wahltinez, O., Bailey, P., Michel, P., Yotov, P., Chaabouni, R., Comanescu, R., Jana, R., Anil, R., McIlroy, R., Liu, R., Mullins, R., Smith, S. L., Borgeaud, S., Girgin, S., Douglas, S., Pandya, S., Shakeri, S., De, S., Klimenko, T., Hennigan, T., Feinberg, V., Stokowiec, W., hui Chen, Y., Ahmed, Z., Gong, Z., Warkentin, T., Peran, L., Giang, M., Farabet, C., Vinyals, O., Dean, J., Kavukcuoglu, K., Hassabis, D., Ghahramani, Z., Eck, D., Barral, J.,

Pereira, F., Collins, E., Joulin, A., Fiedel, N., Senter, E., Andreev, A., and Kenealy, K. (2024). Gemma: Open models based on gemini research and technology.

[45] Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space.

[46] Mitchell, M., Palmarini, A. B., and Moskvichev, A. K. (2023). Comparing humans, GPT-4, and GPT-4v on abstraction and reasoning tasks. In *AAAI 2024 Workshop on "Are Large Language Models Simply Causal Parrots?"*.

[47] Moskvichev, A. K., Odouard, V. V., and Mitchell, M. (2023). The conceptARC benchmark: Evaluating understanding and generalization in the ARC domain. *Transactions on Machine Learning Research*.

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

[49] Nikolaev, A. G. and Jacobson, S. H. (2010). *Simulated Annealing*, pages 1–39. Springer US, Boston, MA.

[50] Park, K., Choe, Y. J., Jiang, Y., and Veitch, V. (2024). The geometry of categorical and hierarchical concepts in large language models. In *ICML 2024 Workshop on Mechanistic Interpretability*.

[51] Park, K., Choe, Y. J., and Veitch, V. (2023). The linear representation hypothesis and the geometry of large language models. In *Causal Representation Learning Workshop at NeurIPS 2023*.

[52] Pennington, J., Socher, R., and Manning, C. (2014). GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

[53] Piantadosi, S. T., Muller, D. C., Rule, J. S., Kaushik, K., Gorenstein, M., Leib, E. R., and Sanford, E. (2024). Why concepts are (probably) vectors. *Trends in Cognitive Sciences*, 28(9):844–856.

[54] Punnen, A. P. (2022). *The Quadratic Unconstrained Binary Optimization Problem*. Springer Cham.

[55] Rigotti, M., Miksovic, C., Giurgiu, I., Gschwind, T., and Scotton, P. (2022). Attention-based interpretability with concept transformers. In *International Conference on Learning Representations*.

[56] Salakhutdinov, R. and Hinton, G. (2009). Semantic hashing. *International Journal of Approximate Reasoning*, 50(7):969–978.

[57] She, L., Benna, M. K., Shi, Y., Fusi, S., and Tsao, D. Y. (2021). The neural code for face memory. *BioRxiv*, pages 2021–03.

[58] Smolensky, P., McCoy, R. T., Fernandez, R., Goldrick, M., and Gao, J. (2022). Neurocompositional computing: From the central paradox of cognition to a new generation of ai systems. *AI Magazine*, 43(3):308–322.

[59] Sorscher, B., Ganguli, S., and Sompolinsky, H. (2022). Neural representational geometry underlies few-shot concept learning. *Proceedings of the National Academy of Sciences*.

[60] Sørensen, M., De Lathauwer, L., and Sidiropoulos, N. D. (2021). Bilinear factorizations subject to monomial equality constraints via tensor decompositions. *Linear Algebra and its Applications*.

[61] Sørensen, M., Sidiropoulos, N. D., and Swami, A. (2022). Overlapping community detection via semi-binary matrix factorization: Identifiability and algorithms. *IEEE Transactions on Signal Processing*.

[62] Takemura, S.-y., Bharioke, A., Lu, Z., Nern, A., Vitaladevuni, S., Rivlin, P. K., Katz, W. T., Olbris, D. J., Plaza, S. M., Winston, P., Zhao, T., Horne, J. A., Fetter, R. D., Takemura, S., Blazek, K., Chang, L.-A., Ogundeyi, O., Saunders, M. A., Shapiro, V., Sigmund, C., Rubin, G. M., Scheffer, L. K., Meinertzhagen, I. A., and Chklovskii, D. B. (2013). A visual motion detection circuit suggested by drosophila connectomics. *Nature*, 500(7461):175–181.

[63] Tigges, C., Hollinsworth, O. J., Geiger, A., and Nanda, N. (2024). Language models linearly represent sentiment. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*.

[64] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. (2020). Huggingface's transformers: State-of-the-art natural language processing.

[65] Zhang, Z., Li, T., Ding, C., and Zhang, X. (2007). Binary matrix factorization with applications. In *Seventh IEEE International Conference on Data Mining (ICDM 2007)*.

[66] Zhou, Y., Lake, B. M., and Williams, A. (2024). Compositional learning of functions in humans and machines.

A Appendix

A.1 Prior work

We are in many ways motivated by ongoing interest in symbolic capabilities of continuous representations. There is a long tradition of neuro-symbolic paradigms which do this explicitly; for example, tensor product representations [58], and more recently transformer-inspired architectures [1, 40]. Yet compositional and symbol-like vectors can appear in more standard connectionist architectures [66], and also in biology, where it is hidden by lack of mechanistic understanding and requires bespoke analyses to discover. In such a case it could be very useful to have a systematic way of making explicit underlying compositional structure.

The field of mechanistic interpretability offers many ideas and methods related to continuous representations of categorical structure. Our formalism and method can be seen as operationalising the linear representation hypothesis [50, 51] into a tool. In particular, the theoretical arguments of [50] favor our choice of orthogonal weights and distance-matching objective. A similar point of view has been taken by recent work on the representation of sparse variables in language models [13, 23], and we aim to enable a similar discovery process for categorical variables.

In computer science, what we seek has been called locality-sensitive hashing [2]. [56] proposed a binary latent variable model for similarity-preserving hashing of documents, and [43] tackled the same problem with a variational autoencoder. At the level of the generative model these approaches are very similar to ours, but with different algorithms and goals, as these methods are often non-linear and do not always seek interpretable features.

In the community detection and applied math literature, our specific factorization problem has been studied as (semi) binary matrix factorization (SBMF) or binary component decomposition (BCD). Remarkably, in special cases an algebraic solution is available via tensor decomposition [60], but it is highly sensitive to violations of its assumptions. There are several optimization-based approaches [65, 29, 61] which are generally built around the assumption of very low-rank data, and thus may not be applicable in the general case. Our specific model formulation closely follows that of [31], and we substantially extend the scope of their model by fitting more general structures to to noisy data.

A.2 Derivation of per-datum losses

The feature MSE is straightforward:

$$\begin{aligned}\mathcal{L}(\mathbf{S},\mathbf{W}) &= \frac{1}{p}\sum_{i=1}^{p}\mathbf{s}_i^T\mathbf{W}^T\mathbf{W}\mathbf{s}_i - 2\mathbf{x}_i^T\mathbf{W}\mathbf{s}_i + \text{const}(\mathbf{S},\mathbf{W}) \\ &= \frac{1}{p}\sum_{i=1}^{p}\mathbf{s}_i^T\mathbf{s}_i - 2\mathbf{x}_i^T\mathbf{W}\mathbf{s}_i + \text{const}(\mathbf{S},\mathbf{W}) \\ &= \frac{1}{p}\sum_{i=1}^{p}(\mathbf{1} - 2\mathbf{W}^T\mathbf{x}_i)^T\mathbf{s}_i + \text{const}(\mathbf{S},\mathbf{W})\end{aligned}$$

A.2.1 Kernel MSE

This is a slow derivation of the conversion of the kernel MSE into a series of quadratics. In the case of small p, it is necessary to account for the effect that each item's $\mathbf{s}$ has on the covariance $\mathbf{S}\mathbf{S}^T$. First we will write that out for the uncentered case, then we will reintroduce centering.

Note that **in this section matrices are transposed relative to the main text**, just for historical reasons.

Uncentered loss First, for simplicity, we will illustrate using a simpler uncentered version of the kernel MSE. The centered derivations are very similar but have more annoying terms which we can add back in at the end.

Let us write out the loss:

$$\|\mathbf{X}\mathbf{X}^T - \mathbf{S}\mathbf{S}^T\|_F^2 = \operatorname{tr}(\mathbf{X}\mathbf{X}^T\mathbf{X}\mathbf{X}^T) + \operatorname{tr}(\mathbf{S}\mathbf{S}^T\mathbf{S}\mathbf{S}^T) - 2\operatorname{tr}(\mathbf{X}\mathbf{X}^T\mathbf{S}\mathbf{S}^T)$$

Remember that the trace terms, like those above, correspond to fourth-order summations in this case:

$$\operatorname{tr}(\mathbf{X}\mathbf{X}^T\mathbf{S}\mathbf{S}^T) = \sum_{i,j=1}^{p}\sum_{k=1}^{d}\sum_{l=1}^{b}\mathbf{X}_{ik}\mathbf{X}_{jk}\mathbf{S}_{il}\mathbf{S}_{jl}$$

and likewise for the other term. If we are minimizing with respect to all elements of $\mathbf{S}$ simultaneously, then we have a quartic (fourth-order) optimization, which is hard in general even for continuous variables. Instead, we can optimize one row at a time keeping all others fixed – a kind of block coordinate descent. By nudging one summation and separating the $i = j$ case we can make things easier. For the first term we have:

$$\begin{aligned}
\operatorname{tr}(\mathbf{S}\mathbf{S}^T\mathbf{S}\mathbf{S}^T) &= \sum_{i=1}^{p}\sum_{k,l=1}^{b}\mathbf{S}_{ik}\mathbf{S}_{il}\sum_{j=1}^{p}\mathbf{S}_{jk}\mathbf{S}_{jl} \\
&= \sum_{i=1}^{p}\sum_{k,l=1}^{b}\mathbf{S}_{ik}\mathbf{S}_{il}\left(\sum_{j\neq i}^{p}\mathbf{S}_{jk}\mathbf{S}_{jl} + \mathbf{S}_{ik}\mathbf{S}_{il}\right) \\
&= \sum_{i=1}^{p}\sum_{k,l=1}^{b}\mathbf{S}_{ik}\mathbf{S}_{il}\sum_{j\neq i}^{p}\mathbf{S}_{jk}\mathbf{S}_{jl} + (\mathbf{S}_{ik}\mathbf{S}_{il})^2 \\
&= \sum_{i=1}^{p}\sum_{k,l=1}^{b}\mathbf{S}_{ik}\mathbf{S}_{il}\sum_{j\neq i}^{p}\mathbf{S}_{jk}\mathbf{S}_{jl} + \mathbf{S}_{ik}\mathbf{S}_{il} \\
&= \sum_{i=1}^{p}\mathbf{s}_i^T(\tilde{\mathbf{S}}^T\tilde{\mathbf{S}})\mathbf{s}_i + (\mathbf{1}^T\mathbf{s}_i)^2
\end{aligned} \tag{6}$$

where we've used the fact the $0^2 = 0$ and $1^2 = 1$. In the last line, we're using $\tilde{\mathbf{S}}$ to indicate all the rows except i. For the second term of the loss it looks like:

$$\begin{aligned}
\operatorname{tr}(\mathbf{X}\mathbf{X}^T\mathbf{S}\mathbf{S}^T) &= \sum_{i=1}^{p}\sum_{k=1}^{d}\sum_{l=1}^{b}\mathbf{X}_{ik}\mathbf{S}_{il}\sum_{j=1}^{p}\mathbf{X}_{jk}\mathbf{S}_{jl}\mathbf{X}_{ik}\mathbf{X}_{jk}\mathbf{S}_{il}\mathbf{S}_{jl} \\
&= \sum_{i=1}^{p}\sum_{k=1}^{d}\sum_{l=1}^{b}\mathbf{X}_{ik}\mathbf{S}_{il}\left(\sum_{j\neq i}^{p}\mathbf{X}_{jk}\mathbf{S}_{jl} + \mathbf{X}_{ik}\mathbf{S}_{il}\right) \\
&= \sum_{i=1}^{p}\sum_{k=1}^{d}\sum_{l=1}^{b}\mathbf{X}_{ik}\mathbf{S}_{il}\sum_{j\neq i}^{p}\mathbf{X}_{jk}\mathbf{S}_{jl} + (\mathbf{X}_{ik}\mathbf{S}_{il})^2 \\
&= \sum_{i=1}^{p}\mathbf{x}_i^T(\tilde{\mathbf{X}}^T\tilde{\mathbf{S}})\mathbf{s}_i + \mathbf{x}_i^T\mathbf{x}_i\mathbf{1}^T\mathbf{s}_i
\end{aligned} \tag{7}$$

which is quite similar as before, but linear in $\mathbf{s}_i$. This is all to show that, even though the MSE between Gram matrices is quartic, the updates for individual rows is quadratic (when one of them is binary).

Centering Here we will give a recursive form of the centered loss function. It is a bit more complicated than necessary, but it allows us to flexibly add any kernel we want so it seems nice to spell out.

Let's say we have only seen p items, so that $\mathbf{X}$ and $\mathbf{S}$ have p rows. We can compute the centered distance for that. We will now show how to update the loss when a new row is added to each matrix. The kernels when we get row $p+1$ are:

$$\mathbf{K}^{(p+1)} = \begin{pmatrix} \mathbf{K}^{(p)} & \mathbf{k} \\ \mathbf{k}^T & k_0 \end{pmatrix}, \quad \mathbf{Q}^{(p+1)} = \begin{pmatrix} \mathbf{Q}^{(p)} & \mathbf{q} \\ \mathbf{q}^T & q_0 \end{pmatrix}$$

Furthermore, let's assume that $\mathbf{K}^{(p)}$ and $\mathbf{Q}^{(p)}$ are already centered. We will say that the row-mean of $\mathbf{S}$ is $\langle \mathbf{s} \rangle = \frac{1}{p}\mathbf{S}^T\mathbf{1}$, so that $\mathbf{Q}^{(p)} = (\mathbf{S} - \mathbf{1}\langle \mathbf{s} \rangle^T)(\mathbf{S}^T - \langle \mathbf{s} \rangle \mathbf{1}^T)$. Likewise for $\mathbf{X}$ and $\mathbf{K}^{(p)}$. The appendages are thus:

$$\begin{aligned} \mathbf{q} &= (\mathbf{S} - \mathbf{1}\langle \mathbf{s} \rangle^T)(\mathbf{s} - \langle \mathbf{s} \rangle) \\ q_0 &= (\mathbf{s} - \langle \mathbf{s} \rangle)^T(\mathbf{s} - \langle \mathbf{s} \rangle) \end{aligned}$$

where $\mathbf{s}$ is the new row of $\mathbf{S}$.

This all makes centering $\mathbf{K}^{(p+1)}$ and $\mathbf{Q}^{(p+1)}$ straightforward. Here it is after some simplification:

$$\bar{\mathbf{Q}}_{ij}^{(p+1)} = \begin{cases} \mathbf{Q}_{ij}^{(p)} - \frac{1}{p+1}\mathbf{q}_i - \frac{1}{p+1}\mathbf{q}_j + \frac{1}{(p+1)^2}q_0 & i,j = 1,..p \\ \frac{p}{p+1}\mathbf{q}_i - \frac{p}{(p+1)^2}q_0 & i = 1,...p, j = p+1 \\ \frac{p^2}{(p+1)^2}q_0 & i = p+1, j = p+1 \end{cases}$$

which just comes from the fact that $\mathbf{Q}\mathbf{1} = 0$ and $\mathbf{1}^T\mathbf{q} = 0$. The same can be done for $\mathbf{K}$ of course.

Without replicating the algebra here, we can use this form to compute the alignment of $\bar{\mathbf{Q}}_{(p+1)}$ and $\bar{\mathbf{K}}^{(p+1)}$:

$$\left\langle \bar{\mathbf{Q}}^{(p+1)}, \bar{\mathbf{K}}^{(p+1)} \right\rangle_F = \left\langle \bar{\mathbf{Q}}^{(p)}, \bar{\mathbf{K}}^{(p)} \right\rangle_F + 2t\mathbf{k}^T\mathbf{q} + t^2k_0q_0 \tag{8}$$

in which we've defined $t = \frac{p}{p+1}$. The same update can be used for the other inner products, to give an update of the loss. Plugging in the form of $\mathbf{q}$ and q_0, we've shown how to write the loss with respect to one row of $\mathbf{S}$ in a way that is quadratic in that row.

A.2.2 Derivation

All that remains is to write the row-wise loss out explicitly. The form of $\mathbf{q}$ that we supplied earlier is not the only one, and in fact we've considered a few different ways to extend the model kernel. For example, in section **??** we consider an 'infinite-dimensional' version, in which $\mathbf{q}$ is updated using variables in $[0, 1]$. In principle, especially if you aren't committed to a quadratic loss, $\mathbf{q}$ and $\mathbf{k}$ could be formed by many functions. Here we will stick with the simple version given above.

We will be plugging in the linear form of $\mathbf{q}$ into the recursive form of the inner products (8) of the loss:

$$\begin{aligned} \left\| \bar{\mathbf{Q}}^{(n+1)} - \bar{\mathbf{K}}^{(n+1)} \right\|_F^2 &= \left\langle \bar{\mathbf{Q}}^{(n+1)}, \bar{\mathbf{Q}}^{(n+1)} \right\rangle_F + \left\langle \bar{\mathbf{K}}^{(n+1)}, \bar{\mathbf{K}}^{(n+1)} \right\rangle_F - 2\left\langle \bar{\mathbf{Q}}^{(n+1)}, \bar{\mathbf{K}}^{(n+1)} \right\rangle_F \\ &= \left\| \bar{\mathbf{Q}}^{(n)} - \bar{\mathbf{K}}^{(n)} \right\|_F^2 + 2t\mathbf{q}^T\mathbf{q} + t^2q_0^2 - 4t\mathbf{k}^T\mathbf{q} - 2t^2k_0q_0 + \text{const}(\mathbf{q}) \end{aligned}$$

That is, we'll be plugging $\mathbf{q} = \bar{\mathbf{S}}(\mathbf{s} - \langle \mathbf{s} \rangle)$ and $q_0 = (\mathbf{s} - \langle \mathbf{s} \rangle)^T(\mathbf{s} - \langle \mathbf{s} \rangle)$ into the equation above. We will end up with something similar to the uncentered forms (6 and 7). After gathering terms, we have:

$$\begin{aligned} \mathcal{L}(\mathbf{s}) &= \mathbf{s}^T\mathbf{J}\mathbf{s} - 2\mathbf{h}^T\mathbf{s} \\ \mathbf{J} &= 2\bar{\mathbf{S}}^T\bar{\mathbf{S}} + t\langle \tilde{\mathbf{s}} \rangle \langle \tilde{\mathbf{s}} \rangle^T \qquad (9) \\ \mathbf{h} &= \mathbf{J}\langle \mathbf{s} \rangle + t\langle 1 - \mathbf{s} \rangle^T \langle \mathbf{s} \rangle \langle \tilde{\mathbf{s}} \rangle - tk_0\langle \tilde{\mathbf{s}} \rangle + 2\mathbf{S}^T\mathbf{k} \qquad (10) \\ \langle \tilde{\mathbf{s}} \rangle &= 2\langle \mathbf{s} \rangle - \mathbf{1} \end{aligned}$$

A.3 Simulated annealing

Our strategy for optimizing $\mathbf{S}$ is to use greedy search with noise. In general, say we have a set of binary parameters $\Theta = \{\theta_0, ..., \theta_N\}$ and some loss function of those parameters $\mathcal{L}(\Theta)$. We can define the marginal loss of each parameter, θ_i, as the difference between the loss if θ_i is set to 0 versus 1:

$$\Delta(\theta_i; \Theta) = \mathcal{L}(\theta_0, ..., 0, ..., \theta_N) - \mathcal{L}(\theta_0, ..., 1, ..., \theta_N)$$

which depends in general on the state of all parameters other than θ_i.

A fully greedy search sweeps through each parameter and sets it to $\theta_i \leftarrow \mathrm{H}(\Delta(\theta_i; \Theta))$, where H is the Heaviside function. In most cases this will be very prone to local minima, but a well-known remedy is to occasionally make sub-optimal decisions. Instead of deterministically setting θ_i, the value is drawn from a Bernoulli with $\Delta(\theta_i)$ as the natural parameters (i.e. logits):

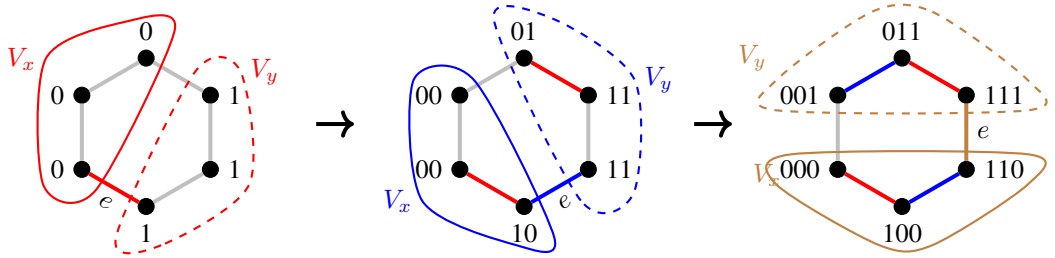
$$\theta_i | \mathbf{\Theta} \sim \mathrm{Bern}\left(\eta = \frac{1}{T}\Delta(\theta_i; \Theta)\right) \tag{11}$$

where the temperature parameter, T, determines the noise level.

During optimization the temperature, T, is slowly lowered from a high initial value towards a smaller value. The process is called "simulated annealing", and it is an active area of research [49] to find good annealing schedules, i.e. functions which determines temperature at each optimization iteration. We use an exponential annealing schedule, $T(k; \gamma, \tau, T_0) = \gamma^{k/\tau} T_0$.

A.4 Analograms and hierarchy regularization

First, here is the sketch of the algorithm for recovering concepts from an analogram: Start with an edge $e = (x, y)$, partition[4] the vertices into those which are closer to x (call them V_x) or closer to y (call them V_y). Each V_x node gets a 0 label, each V_y node gets a 1. Pick another edge, ignoring from now on any edges which cross the partition, and repeat the process. Here is an illustration of the process for a hexagon graph, coloring edged according to the partitions:



This algorithm is described in several places, but we used chapter 19 of [10].

Our heuristic for computing analograms from $\mathbf{S}$ is based on identifying sub/superset pairs of concepts from the empirical covariance $\Sigma^{(\mathbf{S})}$. For simplicity, let us assume that each concept is distinct is active for at most half the data (i.e. $\langle \mathbf{s}_\alpha \rangle \leq 0.5$). Then we can define the following coefficients:

$$\mathbf{J}_{\alpha,\beta} = \begin{cases} 1 & \text{if } \langle \mathbf{s}_\alpha \mathbf{s}_\beta \rangle = \min\{\langle \mathbf{s}_\alpha \rangle, \langle \mathbf{s}_\beta \rangle\} \\ -1 & \text{if } \langle \mathbf{s}_\alpha \mathbf{s}_\beta \rangle = 0 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

$$\mathbf{h}_\alpha = \sum_\beta \mathcal{I}\left[\langle \mathbf{s}_\alpha \rangle = \langle \mathbf{s}_\alpha \mathbf{s}_\beta \rangle\right] \tag{13}$$

where $\mathcal{I}$ is the indicator function. The $\mathbf{J}$ matrix encodes subset/superset and mutual exclusion relations, while the $\mathbf{h}$ vector counts the number of supersets.

[4] This rule partitions the graph because partial cubes are bipartite. In fact, the partitioning is the basis of a certain binary relation, the Djokovic-Winkler relation, which is the theoretical basis of this construction.

Hierarchy regularization Now we provide a scheme for minimizing the hierarchical regularizer (2.3) at a per-item level. We will start by considering the finite data regime, and provide a heuristic extension to the large data regime.

For a concept matrix $\mathbf{S} \in \{0,1\}^{k \times p}$, we define the hierarchical regularizer as

$$\mathcal{H}(\mathbf{S}) = \sum_{\alpha,\beta=1}^{k} \min\{\mathbf{s}_\alpha^T \mathbf{s}_\beta, \mathbf{s}_\alpha^T \tilde{\mathbf{s}}_\beta, \tilde{\mathbf{s}}_\alpha^T \mathbf{s}_\beta, \tilde{\mathbf{s}}_\alpha^T \tilde{\mathbf{s}}_\beta\} \tag{14}$$

which measure the amount of partial overlap between concepts. The function will be zero when all pairs of concepts are either disjoint or sub/supersets of each other.

In order to minimize this function with simulated annealing, need to compute the effect of flipping a single bit of $\mathbf{S}$. In the terms of Appendix A.3, we want to copute $\Delta(\mathbf{S}_{\alpha,i})$.

For shorthand let's give a name to each element of the summand of Eq. 14 after removing the effect of element $\mathbf{S}_{\alpha,i}$:

$$\begin{aligned}
A_{\alpha\beta} &\doteq \mathbf{s}_\alpha^T \mathbf{s}_\beta - \mathbf{S}_{\alpha,i}\mathbf{S}_{\beta,i} \\
B_{\alpha\beta} &\doteq \mathbf{s}_\alpha^T \tilde{\mathbf{s}}_\beta - \mathbf{S}_{\alpha,i}(1 - \mathbf{S}_{\beta,i}) = \mathbf{1}^T \mathbf{s}_\alpha - \mathbf{S}_{\alpha,i} - A_{\alpha\beta} \\
C_{\alpha\beta} &\doteq \tilde{\mathbf{s}}_\alpha^T \mathbf{s}_\beta - (1 - \mathbf{S}_{\alpha,i})\mathbf{S}_{\beta,i} = \mathbf{1}^T \mathbf{s}_\beta - A_{\alpha\beta} \\
D_{\alpha\beta} &\doteq \tilde{\mathbf{s}}_\alpha^T \tilde{\mathbf{s}}_\beta - (1 - \mathbf{S}_{\alpha,i})(1 - \mathbf{S}_{\beta,i}) = p - A_{\alpha\beta} - B_{\alpha\beta} - C_{\alpha\beta}
\end{aligned}$$

from which we can define the values after adding back $\mathbf{S}_{\alpha,i}$ set to either 1 or 0:

$$\begin{aligned}
A^1_{\alpha\beta} &= A_{\alpha\beta} + \mathbf{S}_{\beta,i} & A^0_{\alpha\beta} &= A_{\alpha\beta} \\
B^1_{\alpha\beta} &= B_{\alpha\beta} + (1 - \mathbf{S}_{\beta,i}) & B^0_{\alpha\beta} &= B_{\alpha\beta} \\
C^1_{\alpha\beta} &= C_{\alpha\beta} & C^0_{\alpha\beta} &= C + \mathbf{S}_{\beta,i} \\
D^1_{\alpha\beta} &= D_{\alpha\beta} & D^0_{\alpha\beta} &= D_{\alpha\beta} + (1 - \mathbf{S}_{\beta,i})
\end{aligned}$$

meaning that the Δ term is

$$\Delta(\mathbf{S}_{\alpha,i}) = \sum_\beta \min\{A^0_{\alpha\beta}, B^0_{\alpha\beta}, C^0_{\alpha\beta}, D^0_{\alpha\beta}\} - \min\{A^1_{\alpha\beta}, B^1_{\alpha\beta}, C^1_{\alpha\beta}, D^1_{\alpha\beta}\}$$

which could, in principle, be it. However, we can rephrase this in a way that is computationally simpler and somewhat more intuitive.

This starts from the following fact:

Fact 1 *Let us define $X = (A_{\alpha\beta}, B_{\alpha\beta}, C_{\alpha\beta}, D_{\alpha\beta})$, as well as X^1 and X^0 in the same way. Furthermore, let $k = \arg\min_i X_i$. Then* $\min X^0 - \min X^1 = (X^0 - X^1)_k$

This results from the observation that $\min X \leq \min X^0 \leq \min X + 1$, and likewise for X^1.

Working through the algebra leads us to define a set of coefficients:

$$\mathbf{R}_{\alpha,\beta} = \begin{cases} 1 & \text{if } A_{\alpha\beta} < \min\{B_{\alpha\beta}, C_{\alpha\beta} - 1, D_{\alpha\beta}\} \text{ or } D_{\alpha\beta} \leq \min\{A_{\alpha\beta}, B_{\alpha\beta}, C_{\alpha\beta}\} \\ -1 & \text{if } B_{\alpha\beta} < \min\{A_{\alpha\beta}, C_{\alpha\beta}, D_{\alpha\beta} - 1\} \text{ or } C_{\alpha\beta} \leq \min\{A_{\alpha\beta}, B_{\alpha\beta}, D_{\alpha\beta}\} \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

$$\mathbf{r}_\alpha = \sum_\beta \begin{cases} 1 & \text{if } C_{\alpha\beta} \leq \min\{A_{\alpha\beta}, B_{\alpha\beta}, D_{\alpha\beta}\} \\ -1 & \text{if } D_{\alpha\beta} \leq \min\{A_{\alpha\beta}, B_{\alpha\beta}, C_{\alpha\beta}\} \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

in which case we can write in matrix notation:

$$\Delta(\mathbf{S}_{\alpha,i}) = (\mathbf{R}\mathbf{s}_i)_\alpha - 2\mathbf{r}_\alpha$$

Thus minimizing $\mathcal{H}(\mathbf{S})$ requires a matrix-vector multiplication, just like minimizing $\mathcal{L}_{MSE}$ and $\mathcal{L}_{CKA}$, but with coefficients that must be constructed for each column.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: We believe that our sense of scope is accurate.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: Throughout the paper, but especially in the discussion, we emphasise the scope of our method and provide suggestions for future work.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [NA]

Justification: We do not prove theorems, but derivations are supplied in the appendix.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: All code is available on github, and data were publicly access by us.

   Guidelines:

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

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

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

Answer: [Yes]

Justification:We only used publicly accessible data.

Guidelines:

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

6. **Experimental setting/details**

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

Answer: [Yes]

Justification: There are not many hyperparamters, and we provide the values we used.

Guidelines:

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

7. **Experiment statistical significance**

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

Answer: [Yes]

Justification: We report standard deviations.

Guidelines:

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

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: **[TODO]**

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We believe there are no direct harms of this research.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [NA]

    Justification: This is not a potent enough method.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to

generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

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

    Answer: [NA]

    Justification: No risks.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We cite appropriately.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

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

Answer: [NA]

Justification: Does not release new assets.

Guidelines:

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

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

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

Answer: [NA]

Justification: No human subjects.

Guidelines:

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

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

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

Answer: [NA]

Justification: Not needed.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [NA]

Justification: Don't use them.

Guidelines:

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