VICE: Variational Interpretable Concept Embeddings

Lukas Muttenthaler*
Machine Learning Group
Technische Universität Berlin
BIFOLD†
Berlin, Germany

Charles Y. Zheng
Machine Learning Team, FMRI Facility
National Institute of Mental Health
Bethesda, MD, USA

Patrick McClure‡
Department of Computer Science
Naval Postgraduate School
Monterey, CA, USA

Robert A. Vandermeulen
Machine Learning Group
Technische Universität Berlin
BIFOLD†
Berlin, Germany

Martin N. Hebart
Vision and Computational Cognition Group
MPI for Human Cognitive and Brain Sciences
Leipzig, Germany

Francisco Pereira
Machine Learning Team, FMRI Facility
National Institute of Mental Health
Bethesda, MD, USA

Abstract

A central goal in the cognitive sciences is the development of numerical models for mental representations of object concepts. This paper introduces Variational Interpretable Concept Embeddings (VICE), an approximate Bayesian method for embedding object concepts in a vector space using data collected from humans in an odd-one-out triplet task. VICE uses variational inference to obtain sparse, non-negative representations of object concepts with uncertainty estimates for the embedding values. These estimates are used to automatically select the dimensions that best explain the data. We derive a PAC learning bound for VICE that can be used to estimate generalization performance or determine sufficient sample size in experimental design. VICE rivals or outperforms its predecessor, SPoSE, at predicting human behavior in the odd-one-out triplet task. Furthermore, VICE's object representations are more reproducible and consistent across random initializations, highlighting the unique advantage of using VICE for deriving interpretable embeddings from human behavior.

1 Introduction

Human knowledge about object concepts encompasses many types of information, including function or purpose, visual appearance, encyclopedic facts, or taxonomic characteristics. A central question in cognitive science concerns the representation of this knowledge and its use across different tasks. One approach to this question is inductive and lets subjects list properties for hundreds to thousands of objects [30, 10, 6, 22], yielding large lists of responses about different types of properties. Specifically, objects are represented as vectors of binary properties. While this approach is agnostic to downstream prediction tasks, it may be biased since subjects may leave out important features and/or mention

*Also affiliated with the Max Planck Institute for Human Cognitive and Brain Sciences, Leipzig, Germany.
†Berlin Institute for the Foundations of Learning and Data, Germany.
‡Work was partially done while affiliated with the National Institute of Mental Health, Bethesda, MD, USA.

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

unimportant ones. For example, one may forego a general property (e.g., "is an animal") while providing a highly specific fact (e.g., "is found in Florida"). In an alternative, deductive approach, researchers postulate dimensions of interest and subsequently let subjects place objects in each dimension. Binder et al. [3] employed such an approach and collected ratings for hundreds of objects, verbs, and adjectives. These ratings were gathered over 65 dimensions, reflecting sensory, motor, spatial, temporal, affective, social, and cognitive experiences. Nonetheless, it is desirable to discover object representations that are not biased by the conducted behavioral task and whose dimensions are interpretable without necessitating *a priori* assumptions about their semantic content.

Recently, Zheng et al. [50] and Hebart et al. [21] introduced SPoSE, a model of the mental representations of 1,854 objects in a 49-dimensional space. The model was learned from human judgments about object similarity, where subjects were asked to determine an odd-one-out object in random triplets of objects. SPoSE embedded each object in a vector space so that each dimension is non-negative and sparse (most objects have a close-to-zero entry for a given dimension). The authors showed that the embedding dimensions of objects were interpretable and that subjects could coherently label what the dimensions were "about," ranging from categorical (e.g., animate, food) to functional (e.g., tool), structural (e.g., made of metal or wood), or visual (e.g., coarse pattern). The authors hypothesized that interpretability arose from combining positivity and sparsity constraints so that no object was represented by every dimension, and most dimensions were present for only a few objects. In addition, SPoSE could predict human judgements close to the estimated best attainable performance [50, 21].

Despite its notable performance, SPoSE has several limitations. The first stems from the use of an ℓ_1 sparsity penalty to promote interpretability. In SPoSE, 6 to 11 dominant dimensions for an object account for most of the prediction performance. These dimensions are different between objects. A potential issue with enforcing SPoSE to have even fewer dimensions is that it may cause excessive shrinkage of the dominant values [1]. Second, when inspecting the distributions of values across objects, most SPoSE dimensions do not reflect the exponential prior induced by the ℓ_1 penalty. Overcoming this prior may lead to suboptimal performance and inconsistent solutions, specifically in low data regimes. Third, SPoSE uses an ad-hoc criterion for determining the dimensionality of the solution via an arbitrary threshold on the density of each dimension. Finally, SPoSE has no criterion for determining convergence of its optimization process, nor does it provide any formal guarantees on the sample size needed to learn a model of desired complexity.

To overcome these limitations we introduce VICE, a variational inference (VI) method for embedding object concepts with interpretable, sparse, and non-negative dimensions. We start by discussing related work and a description of the triplet task and SPoSE, followed by our presentation of theory and experimental results.

Contribution 1: VICE solves major limitations of SPoSE First, VICE encourages shrinkage while allowing for small entry values by using a *spike-and-slab* prior [5, 12, 15, 31, 38, 43, 48]. We deem this more appropriate than an exponential prior, because *importance* – the value an object takes in a dimension – is different from *relevance* – whether the dimension is applicable to that object – and both can be controlled separately. Second, we use VI with a unimodal posterior for representing each object in a dimension which yields a mean value and an uncertainty estimate. While unimodality makes it possible to use the mean values as representative object embeddings, the uncertainty estimates allow us to use a statistical procedure to automatically select the dimensions that best explain the data. Third, we use this procedure to introduce a convergence criterion that reliably identifies *representational stability*, i.e. the consistency in the number of selected dimensions.

Contribution 2: A PAC bound on the generalization of SPoSE and VICE models This bound can be used *retrospectively* to provide guarantees about the generalization performance of a converged model. Furthermore, it can be used *prospectively* to determine the sample size required to identify a representation given the number of objects and a maximum possible number of dimensions.

Contribution 3: Extensive evaluation of model performance across multiple datasets We compare VICE with SPoSE over three different datasets. One of these datasets contains concrete objects, another consists of adjectives, and the third is composed of food items. Experimentally, we find that VICE rivals or outperforms the performance of SPoSE in modeling human behavior. Moreover, we find that VICE yields dimensions that are much more reproducible, with a lower variance for the number of (selected) dimensions. Both of these measures are particularly important in the cognitive sciences. Lastly, we compare VICE with SPoSE for reduced amounts of data and show that VICE has significantly better performance on all measures.

A PyTorch implementation of VICE featuring Continuous Integration is publicly available at `https://github.com/LukasMut/VICE`. The GitHub repository additionally includes code to reproduce all of the experiments that are presented in this paper.

2 Related work

Navarro & Griffiths [34] introduced a method for learning semantic concept embeddings from item similarity data. The method infers the number of embedding dimensions using the Indian Buffet Process (IBP) [18]. Their approach relies on continuous-valued similarity ratings rather than discrete forced-choice behavior and is not directly applicable to our setting. It is also challenging to scale the IBP to the number of dimensions and samples in our work [39]. Roads & Love [41] introduced a method for learning object embeddings from behavior in an 8-rank-2 task. Their method predicts behavior from the embeddings by using active sampling [16] to query subjects with the most informative stimuli, yielding an object similarity matrix. The interpretability of embedding dimensions was not considered in this work.

A different approach has been to develop interpretable concept representations from text corpora. Early methods used word embeddings with positivity and sparsity constraints [33]. Later work in this direction used topic model representations of Wikipedia articles about objects [37], transformations of word embeddings into sparse non-negative representations [46, 35], or predictions of properties [10] or dimensions [49]. Others have considered using text corpora in conjunction with imaging data [13, 8]. Finally, Derby et al. [9] introduced a neural network function that maps the sparse feature space of a semantic property norm to the dense space of a word embedding, identifying informative combinations of properties and ranking candidate properties for new words.

3 Triplet task

The *triplet task*, also known as the *odd-one-out* triplet task, is used for discovering object concept embeddings from similarity judgments over a set of m different objects. These judgments are collected from human participants who are given queries that consist of a *triplet* of objects (e.g., {"suit", "flamingo", "car"}). Participants are asked to consider the three pairs in a triplet {{"suit","flamingo"}, {"suit","car"}, {"flamingo","car")}, and to decide which pair is the most similar, leaving the third as the odd-one-out. We assign each object a numerical index, e.g., $1 \leftarrow$ "aardvark", $\ldots, 1854 \leftarrow$ "zucchini". Let $\{y, z\}$ denote the indices in this pair, e.g., $\{y, z\} = \{268, 609\}$ for "suit" and "flamingo." A dataset $\mathcal{D}$ is a set of n ordered pairs of presented triplets and selected pairs. That is, $\mathcal{D} \coloneqq \big(\{i_s, j_s, k_s\}, \{y_s, z_s\}\big)_{s=1}^{n}$, where $\{y_s, z_s\} \subset \{i_s, j_s, k_s\}$. In Appendix A we discuss in detail why the triplet task appears to be a sensible choice for modeling object similarities in humans.

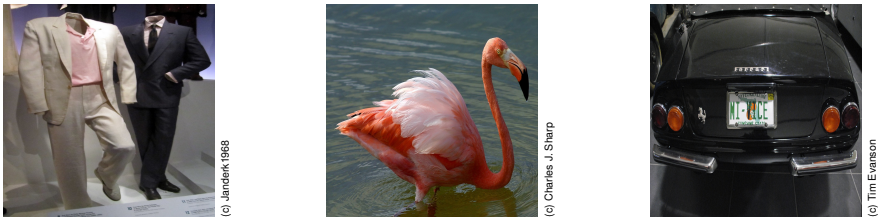

Figure 1: Example triplet containing the objects "suit", "flamingo", and "car" (creative commons images).

4 Formal setting

Sparse Positive object Similarity Embedding (SPoSE) [50] is an approach for finding interpretable embedding dimensions from the triplet task described above. It does so by finding an embedding vector $\mathbf{x}_i = [x_{i1}, \ldots, x_{id}]$ for every object i. Let X denote the $m \times d$ matrix $(\mathbf{x}_1, \ldots, \mathbf{x}_m)$ and $S \coloneqq XX^T$ be the similarity matrix, where S_{ij} denote its entry at i, j. Each triplet $\{i_s, j_s, k_s\}$ is assumed to be chosen uniformly at random from the collection of all possible sets of object triplets $\mathcal{T}$. That is, $\{i_1, j_1, k_1\}, \ldots, \{i_n, j_n, k_n\} \overset{\text{i.i.d.}}{\sim} \mathcal{U}(\mathcal{T})$. Hence, the probability for choosing a triplet,

$p(\{i_s, j_s, k_s\})$, is the same for all $\mathcal{D}_s \in \{\mathcal{D}_1, \ldots, \mathcal{D}_n\}$ and can therefore be treated as a coefficient for computing $p(\mathcal{D}|X)$. For our probability model the log-likelihood of the data given the embedding matrix, $\log p(\mathcal{D}|X)$, can then be defined as

$$\begin{aligned}
\log p(\mathcal{D}|X) &= \log \prod_{s=1}^{n} p(\mathcal{D}_s|X) \\
&= \log \prod_{s=1}^{n} p(\{y_s, z_s\}, \{i_s, j_s, k_s\}|X) \\
&= \log \prod_{s=1}^{n} p(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X) p(\{i_s, j_s, k_s\}|X) \qquad (1) \\
&= \log \prod_{s=1}^{n} p(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X) p(\{i_s, j_s, k_s\}).
\end{aligned}$$

Recall that $p(\{i_s, j_s, k_s\})$ is a coefficient that is the same for all $s \in \{1, \ldots, n\}$. We let $C \coloneqq p(\{i_s, j_s, k_s\})$ and thus we can further rewrite,

$$\begin{aligned}
\log \prod_{s=1}^{n} p(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X) p(\{i_s, j_s, k_s\}) &= \log \left[C^n \prod_{s=1}^{n} p(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X) \right] \\
&= \log C^n + \log \prod_{s=1}^{n} p(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X) \\
&= n \log C + \sum_{s=1}^{n} \log p(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X), \qquad (2)
\end{aligned}$$

where (1) follows from the chain rule of (conditional) probability. Note that the constant $n \log C$ can be ignored in the minimization of the objective function. SPoSE uses maximum a posteriori (MAP) estimation with a Laplace prior under a non-negativity constraint (equivalent to an exponential prior) to find the most likely embedding X for the training data $\mathcal{D}$. This leads to the training objective

$$\underset{X \geq 0}{\arg\min} \; -\sum_{s=1}^{n} \log p(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X) + \lambda \sum_{i=1}^{m} \sum_{j=1}^{d} |X_{ij}|,$$

where λ is determined using cross-validation. The probability of choosing $\{y_s, z_s\}$ as the most similar pair of objects, given an object triplet $\{i_s, j_s, k_s\}$ and the embedding matrix X, is modeled as

$$p(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X) \coloneqq \frac{\exp(S_{y_s, z_s})}{\exp(S_{i_s, j_s}) + \exp(S_{i_s, k_s}) + \exp(S_{j_s, k_s})}. \qquad (3)$$

5 VICE

In contrast to SPoSE, we use mean-field VI [4] instead of a MAP estimate for approximating the posterior probability $p(X|\mathcal{D})$. In VICE we impose additional constraints on the embedding matrix X by using a prior that encourages shrinkage while allowing entries in X to be close to zero.

5.1 Variational Inference

For VICE we consider approximating $p(X|\mathcal{D})$ with a variational distribution, $q_\theta(X)$, where $q_\theta \in \mathcal{Q}$, and θ is optimized to minimize the KL divergence to the true posterior, $p(X|\mathcal{D})$. In VICE the KL divergence objective function (derived in Appendix B.1) is

$$\underset{\theta}{\arg\min} \; \mathbb{E}_{q_\theta(X)} \left[\frac{1}{n} \left(\log q_\theta(X) - \log p(X) \right) - \frac{1}{n} \sum_{s=1}^{n} \log p\left(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X \right) \right]. \qquad (4)$$

Variational distribution VI requires a choice of a parametric variational distribution $q \in \mathcal{Q}$. For VICE we use a Gaussian distribution with a diagonal covariance matrix $q_\theta(X) = \mathcal{N}(\mu, \text{diag}(\sigma^2))$

where $\theta = \{\mu, \sigma\}$. Therefore, each embedding dimension has a *mean* and a *standard deviation*. We deem a Gaussian variational distribution appropriate for a variety of reasons. First, under certain conditions, the posterior is Gaussian in the infinite-data limit [26]. Second, a unimodal variational distribution makes it possible to use μ as representative object embeddings. A fixed representation is useful for downstream use cases that rely on a single embedding vector for each object. Use cases range from using embeddings as targets in a regression task over unsupervised clustering of embeddings to interpreting embeddings. More complex variational families are not as practical for this. Third, a Gaussian posterior is a computationally convenient choice.

Similarly to Titsias & Lázaro-Gredilla [47], we use a Monte Carlo approximation of Equation 4 by sampling a limited number of Xs from $q_\theta(X)$ during training. We generate X with the reparameterization trick [25, 40], $X_{\theta,\epsilon} = \mu + \sigma \odot \epsilon$, where $\epsilon \in \mathbb{R}^{m \times d}$ is entrywise $\mathcal{N}(0, 1)$, and $\odot$ denotes the Hadamard (element-wise) product. This leads to the objective

$$\arg\min_{\theta} \; \frac{1}{R} \sum_{r=1}^{R} \frac{1}{n} (\log q_\theta(X_{\theta,\epsilon^{(r)}}) - \log p(X_{\theta,\epsilon^{(r)}})) - \frac{1}{n} \sum_{s=1}^{n} \log p(\{y_s, z_s\} | \{i_s, j_s, k_s\}, [X_{\theta,\epsilon^{(r)}}]_+). \tag{5}$$

We apply a ReLU function, denoted by $[\cdot]_+$, to the sampled $X_{\theta,\epsilon}$ values to guarantee that $X_{\theta,\epsilon} \in \mathbb{R}_+^{m \times d}$. As commonly done in the Dropout and Bayesian Neural Network literature [45, 5, 14, 29], we set $R = 1$ for computational efficiency during the optimization process.

Posterior probability estimation Computational efficiency at inference time is not as critical as it is during training. Therefore, we can get a better estimate of the posterior probability distribution over the three possible odd one-one-out choices by letting $R \gg 1$. Using the optimized variational posterior, $q_{\hat{\theta}(X)}$, we approximate the probability distribution with a Monte Carlo estimate [17, 5, 29, 4] from R samples $X^{(r)} = X_{\hat{\theta},\epsilon^{(r)}}$ for $r = 1, \ldots, R$, yielding

$$\hat{p}\left(\{y, z\} | \{i, j, k\}\right) \coloneqq \frac{1}{R} \sum_{r=1}^{R} p(\{y, z\} | \{i, j, k\}, X^{(r)}). \tag{6}$$

Spike-and-slab prior As discussed above, SPoSE induces sparsity through an ℓ_1 penalty which, along with the non-negativity constraint, is equivalent to using an exponential prior. Through examination of the publicly available histograms of weight values in the two most important SPoSE dimensions (see Figure 5 in Appendix C.3), we observed that the dimensions did not resemble an exponential distribution. Instead, they contained a *spike* of probability at zero and a wide *slab* of probability for the non-zero values. To model this, we use a spike-and-slab Gaussian mixture prior [5, 12, 15, 23, 28],

$$p(X) = \prod_{i=1}^{m} \prod_{j=1}^{d} (\pi_{\text{spike}} \mathcal{N}(X_{ij}; 0, \sigma_{\text{spike}}^2) + (1 - \pi_{\text{spike}}) \mathcal{N}(X_{ij}; 0, \sigma_{\text{slab}}^2)), \tag{7}$$

which encourages shrinkage. This prior has three parameters, σ_{spike}, σ_{slab}, and π_{spike}. π_{spike} is the probability that an embedding dimension is drawn from the *spike* Gaussian. Since spike and slab distributions are mathematically interchangeable, by convention we require that $\sigma_{\text{spike}} \ll \sigma_{\text{slab}}$.

5.2 Dimensionality reduction and convergence

For interpretability purposes it is desirable for the object embedding dimensionality, d, to be small. In contrast to SPoSE, which employs a user-defined threshold to prune dimensions, VICE exploits the uncertainty estimates for embedding values to select a subset of informative dimensions.

The pruning procedure works by assigning importance scores to each of the d dimensions, which reflect the number of objects that we can confidently say have a non-zero weight in a dimension. To compute the score, we use the variational embedding for each object i and dimension j – location μ_{ij} and scale σ_{ij} parameters – to compute the posterior probability that the weight is truncated to zero according to the left tail of a Gaussian distribution with that location and scale (see §5.1). This gives us a posterior probability of the weight being zero for each object within a dimension [17]. To calculate the overall importance of a dimension, we estimate the number of objects that have non-zero weights, while controlling the False Discovery Rate [2] with $\alpha = 0.05$. We define the importance of each dimension j to be the number of objects for which $P(X_{ij} > 0) \geq .95$ holds. After convergence,

we prune the model by removing dimensions with 5 or fewer statistically significant objects. This is a commonly used reliability threshold in semantic property norms (e.g., [30, 10]).

In gradient-based optimization, the gradient of an objective function with respect to the parameters of a model, $\nabla\mathcal{L}(\theta)$, is used to iteratively find parameters $\hat{\theta}$ that minimize that function. We use a *representational stability* criterion to determine convergence. That is, the optimization process halts when the number of identified dimensions - as described above - has not changed by a *single* dimension over the past L epochs (e.g., $L = 500$). Given that our goal is to find stable estimates of the number of dimensions, we considered this to be more appropriate than other convergence criteria such as evaluating the cross-entropy error on a validation set or evidence-based criteria [27, 11]. For further details on convergence and the optimization process see Appendix C.1.

6 Sample complexity bound

We use statistical learning theory to obtain estimates of the sample size needed to appropriately constrain VICE (and SPoSE) models. These estimates can be used *retrospectively* to obtain probabilistic guarantees on the generalization of a trained model to unseen data, or *prospectively* to decide how much data to collect for a study. For VICE, and in this analysis only, we extract μ and use it as a fixed embedding to predict the most likely odd-one-out for a query triplet rather than sampling embeddings from the variational distribution.

To obtain a useful bound we make two assumptions. First, we assume that there exists an upper bound M on the largest value in the embedding. Second, we assume that the embeddings obtained by either SPoSE or VICE can be quantized coarsely with marginal losses in predictive performance. We can choose a discretization scale Δ, e.g., $\Delta = 0.5$, and round embedding values to a non-negative integer multiple of Δ. While we employ this quantization primarily to use learning theory bounds for finite hypothesis classes, it could have benefits for interpretation as well (e.g., a dimension could consist of labels such as zero (0), very low (0.5), low (1), medium (1.5), high (2)). In all datasets we have used, SPoSE and VICE embeddings with reasonable priors had values below 2.7 (see Appendix F.2). Given an upper bound M and a discretization scale Δ, one can consider all possible ways of populating the matrix X with weights limited to the set $S = \{0, \Delta, \ldots, (k-1)\Delta, k\Delta\}$. The following bound tells us that the estimated error rate and true error rate for such discretized X matrices are close to each other with high probability.

Proposition 6.1. *Given $\delta > 0$, $\epsilon > 0$, and $n \geq (md\log(k+1) + \log(1/\delta)) / (2\epsilon^2)$ training samples $(\{y_s, z_s\}, \{i_s, j_s, k_{1s}\})_{s=1}^{n}$, then*

$$P\left(\sup_{X \in S^{m\times d}} \hat{R}(X) - R(X) < \epsilon\right) \geq 1 - \delta,$$

where

$$\hat{R}(X) := \frac{1}{n}\sum_{s=1}^{n} \mathbb{1}\left(\{y_s, z_s\} \neq \underset{\{y,z\}}{\arg\max}\, p\left(\{y, z\} | \{i_s, j_s, k_s\}, X\right)\right)$$

and

$$R(X) := P_{\{y',z'\}|\{i',j',k'\}}\left(\{y', z'\} \neq \underset{\{y,z\}}{\arg\max}\, p\left(\{y, z\} | \{i', j', k'\}, X\right)\right).$$

Proof sketch. This bound follows from applying Hoeffding's inequality to the $(k+1)^{m\times d}$ elements of $S^{m\times d}$ combined with a union bound. This proof is virtually identical to that of Theorem 2.13 in Mohri et al. [32]. A complete proof can be found in Appendix D. □

To use the bound we first decide on the number of quantization steps $k = \lceil M/\Delta \rceil$. The probability of violating the bound δ is analogous to the Type I error control in hypothesis testing, often 0.05. As the bound depends very weakly on δ, it is convenient to use $\delta = 0.001$. The error tolerance ϵ does have a major effect on the sample size estimate provided by the bound. For the bound to be practically useful, ϵ has to be smaller than the difference between the training error – average zero-one loss over the training set examples – and random guessing error. For the datasets we use in

this paper we observed a difference of $0.2 - 0.3$. A conservative choice would halve the accuracy gap of 0.2, giving $\epsilon = 0.1$. Together, these values result in a prospective sample size estimate of $n \approx 50 \cdot (md \log(5) + \log(1000))$. To use the bound retrospectively, we fix n, δ while varying Δ to get a guarantee on ϵ, which in turn yields a probabilistic upper bound on the generalization error for an embedding, $R(X) \leq \hat{R}(X) + \epsilon$. For more details, see Algorithm 1 in Appendix F.1.

7 Experiments

Data We performed experiments for three triplet datasets: THINGS (used to develop SPoSE [50, 21]), ADJECTIVES[4], and FOOD [7] [5]. THINGS and ADJECTIVES contain random samples from all possible triplet combinations for 1,854 objects and 2,372 adjectives, respectively. THINGS and ADJECTIVES are each comprised of a large training dataset which contains no "repeats," i.e., there is only one human response for each triplet (~ 1.5M triplets for THINGS and ~ 800K triplets for ADJECTIVES). For each of these training datasets, 10% of the triplets are assigned to a predefined validation set. The test sets for both contain the results for 1,000 random triplets that are not contained in the training data, with 25 repeats for each triplet. FOOD contains data for every possible triplet combination for 36 objects and repeats for some triplets. We partitioned this dataset into train (45%), validation (5%), and test (50%) sets with disjoint triplets. Appendix G provides details about stimuli, data collection, and quality control.

Experimental setup We implemented both SPoSE and VICE in PyTorch [36] using Adam [24] with $\eta = 0.001$. To find the best VICE hyperparameter combination, we performed a grid search over $\sigma_{\text{spike}}, \sigma_{\text{slab}}, \pi_{\text{spike}}$, optimizing Equation 5 and evaluating each model on the validation set. All experiments were performed over 20 different random initializations. For VICE we observed that the hyperparameter values with the lowest average cross-entropy error on the validation set were highly similar across the different datasets (see Table 2 in Appendix E). For more information about the experimental setup, including details about training, weight initialization, hyperparameter grid, and optimal combination, see Appendix E.

Evaluation measures VICE estimates $p(\{y, z\}|\{i, j, k\})$, the model's softmax probability distribution over a given triplet (see Equation 6). The first evaluation measure we consider is the prediction *accuracy* with respect to the correct human choice $\{y, z\}$, where the predicted pair is $\arg\max_{\{y,z\}} \hat{p}\left(\{y, z\}|\{i, j, k\}\right)$. We estimate an upper bound on the prediction accuracy by using the repeats in the test set (see Appendix E). A Bayes-optimal classifier would predict the majority outcome for any triplet, whereas random guessing gives an accuracy of $1/3$. Note that the triplet task is subjective and thus there is not necessarily a definitive correct answer for a given triplet. If a test set contains multiple responses for a triplet, it provides information about the relative similarities of the three object pairs. This lets us calculate probability distributions over answers for each triplet. Approximating these distributions closely is important in cognitive science applications. Therefore, we additionally evaluate the models by computing the KL divergences between the models' posterior probability estimates $\hat{p}(\{y, z\}|\{i, j, k\})$ (see Equation 6) and the human probability distributions inferred from the test set across all test triplets and report the mean.

7.1 Results on THINGS

Full dataset For this experiment we compared a representative model from VICE and SPoSE. The representative models were chosen to be those with the median cross-entropy error on the validation set over the 20 seeds. For VICE, we set the number of Monte Carlo samples to $R = 50$. On the test set, VICE and SPoSE achieved similar prediction accuracies of 0.638 and 0.637 respectively (estimated upper bound for accuracy was 0.673). Likewise, VICE and SPoSE achieved similar average KL divergences of 0.100 and 0.103 respectively (random guessing KL divergence was 0.366). The differences between the median model accuracies and KL divergences were not statistically significant according to a two-sided paired *t*-test over individual test triplets. Hence, VICE and SPoSE predicted triplets approximately equally well when trained on the full dataset. This is not surprising since Bayesian methods based on Monte Carlo sampling become more like deterministic maximum likelihood estimation in the infinite data limit. If $n \to \infty$, then the left-hand side in the expectation

[4] ADJECTIVES is not yet published, but was shared with us by Shruti Japee.

[5] FOOD was shared with us by Jason Avery.

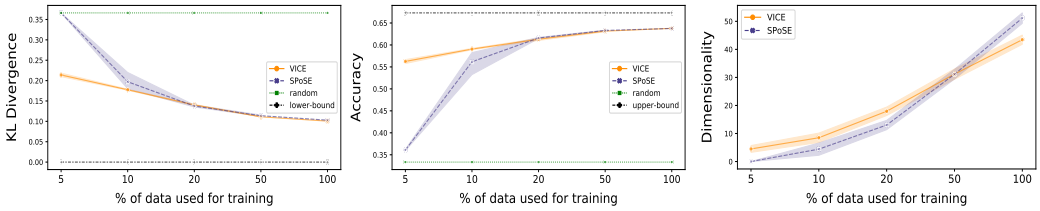

Figure 2: VICE vs. SPoSE. **Left**: KL divergences between model output and human probability distributions. **Center**: Triplet task prediction accuracies. **Right**: Number of identified embedding dimensions. VICE and SPoSE were each trained on differently sized subsets of the THINGS training data. Error bands visualize 95% CIs across the different random initializations and partitions of the data.

of Equation 4, which accounts for the contribution of the prior, goes to zero. Conversely, the effects of the prior are more prominent when n is small, as we show in the following section.

Data efficiency experiments Performance on small datasets is important in cognitive science, since behavioral experiments often have small sample sizes (e.g., tens to hundreds of volunteer in-lab subjects) compared to THINGS, which is particularly large, or they can be costly to collect. To test whether VICE can model the data better than SPoSE in small sample regimes, we performed experiments training on smaller subsets of the training dataset. We did this for subsets with sizes equal to 5%, 10%, 20%, and 50% of the training dataset. For each size, we divided the training set into equal partitions and performed an experiment for each partition, thereby giving multiple results for each subset size. Validation and held-out test sets remained unchanged. In Figure 2 we show the average KL divergence (left) and prediction accuracy (middle) across data partitions and 20 random seeds (used to initialize model parameters) for various dataset sizes. The average performance across partitions was used to compute the confidence intervals (CIs). The differences in prediction accuracies and KL divergences between VICE and SPoSE were notable for the 5% and 10% data subsets. In the former, SPoSE predicted only slightly better than random guessing; in the latter, SPoSE showed a large variability between random seeds and data partitions, as can be seen in the 95% CIs. In comparison, VICE showed small variation in the two performance metrics across random seeds and performed much better than random guessing. The differences between VICE and SPoSE for the 5% and 10% subset scenarios were statistically significant according to a two-sided paired *t*-test comparing individual triplet predictions between median models ($p < 0.001$). For the full training dataset, VICE used significantly fewer dimensions than SPoSE to achieve the same performance.

7.2 Other results

Other datasets On the ADJECTIVES test set, VICE and SPoSE had accuracies of 0.559 and 0.562, respectively (estimated upper bound was 0.607), and KL divergences of 0.083 and 0.088, respectively (random guessing was 0.366). On the FOOD test set, VICE and SPoSE median models had accuracies of 0.693 and 0.698, respectively. The differences in accuracy (and KL divergence for ADJECTIVES) between VICE and SPoSE for both datasets were not statistically significant according to two-sided paired *t*-tests on the median model predictions across triplets on the test set. However, for FOOD, VICE used significantly fewer dimensions than SPoSE to achieve similar performance (see Table 1).

Hyperparameters We observed that VICE's performance is fairly insensitive to hyperparameter selection. Using default hyperparameters, $\sigma_{\text{spike}} = 0.25$, $\sigma_{\text{slab}} = 1.0$, $\pi_{\text{sigma}} = 0.5$, yields an accuracy score within 0.015 of the best cross-validated model in the full dataset setting, on all three datasets.

7.3 Reproducibility and Representational stability

Reproducibility As mentioned in the introduction, a key criterion for learning concept representations beyond predictive performance is *reproducibility*, i.e., learning similar representations for different random initializations on the same training data. To evaluate this, we compared the learned embeddings from 20 differently initialized VICE and SPoSE models. The first aspect of reproducibility we investigate is whether the models yield a consistent number of embedding dimensions across random seeds.

As reported in Table 1, VICE yielded fewer dimensions than SPoSE with less variance in the number of dimensions for all three datasets. The embedding dimensionality was significantly smaller according to an independent t-test for THINGS ($p < 0.001$) and FOOD ($p < 0.001$), but not for ADJECTIVES ($p = 0.108$). The difference in the standard deviations for ADJECTIVES was statistically significant according to a two-sided F-test ($p = 0.002$), but was not statistically significant for THINGS ($p = 0.283$) or FOOD ($p = 0.378$). In addition, we observed that the identified dimensionality is consistent as long as d is chosen to be sufficiently large (see Appendix E).

Table 1: Reproducibility of VICE and SPoSE. Reported are the means and standard deviations with respect to selected dimensions and the average reproducibility score of dimensions (in %) across random seeds. Bold means VICE performed statistically significantly better with $\alpha = 0.05$.

	VICE		SPoSE	
DATA\ METRIC	Selected Dims.	Reproducibility	Selected Dims.	Reproducibility
THINGS	**44** (1.59)	**87.01**%	52 (1.82)	81.30%
ADJECTIVES	21 (**0.77**)	76.76%	22 (1.53)	71.64%
FOOD	**5** (0.95)	**87.38**%	16 (1.02)	62.88%

The second aspect to reproducibility we examine is the degree to which the identified dimensions are similar across random initializations up to a permutation. To calculate the number of reproducible dimensions, we associated each embedding dimension of a model with the most similar embedding dimension across the other models. We quantify reproducibility of a dimension as the average Pearson correlation between one dimension and its best match across the 19 remaining models. In Table 1 we report the average relative number of dimensions with a correlation > 0.8 across models. The embedding dimensions in VICE were more reproducible than those in SPoSE. The difference in average reproducibility scores was statistically significant according to a two-sided, independent t-test for FOOD ($p = 0.030$) and THINGS ($p = 0.008$), but not for ADJECTIVES ($p = 0.466$).

Representational stability For all three datasets, VICE found reproducible and stable dimensions across all 20 random initializations and converged when the embedding dimensionality had not changed over the past $L = 500$ epochs. The median number of epochs to achieve representational stability was comparable for the similarly-sized THINGS and ADJECTIVES datasets, but occurred later for FOOD (likely due to the smaller number of gradient updates per epoch). In Appendix C.1 we show plots that demonstrate that the convergence criterion as defined in §5.2 worked reliably for all datasets.

7.4 Interpretability

Figure 3: Four example VICE dimensions showing the top six objects for THINGS (*top*) and FOOD (*middle*), and wordclouds for ADJECTIVES (*bottom*).

One of the benefits of SPoSE is the interpretability of the dimensions of its concept embeddings which was evidenced through experiments with humans [21]. VICE dimensions are equally interpretable

due to similar constraints on its embedding space. Therefore, it is easy to sort objects within a dimension of the VICE mean embedding matrix μ in descending order, to obtain human judgments of what an embedding dimension represents. To illustrate the interpretability of VICE we show in Figure 3 four example dimensions of a representative model for THINGS, FOOD, and ADJECTIVES. For THINGS the dimensions appear to represent *categorical*, *functional*, *structural*, and *color-related* information. For FOOD the dimensions appear to be a combination of *fried*, *vegetable*, *fruit*, and *sweet* food items. For ADJECTIVES the dimensions seem to reflect *size/magnitude*, *visual appearance*, *negative valence*, and *sensory* adjectives. In Appendix H we show VICE dimensions for THINGS only, since manuscripts about the other datasets are still in preparation, and, hence, cannot be shown publicly.

8 Limitations

The goal of our method is to identify general mental representations of objects. The dimensions identified by a model reflect semantic characteristics that explain task performance for many subjects in the experimental subject population (here, Amazon Mechanical Turk subjects in the United States). As such, it is possible that they reflect biases widespread in that population. Furthermore, the choice of population may affect the identified dimensions. That is, a chéf may classify food items differently from a lay subject, and a linguist would likely have a more complex representation of an adjective. The effects of expertise or developmental stage in mental representations are of obvious interest to cognitive scientists. Therefore, we envision further research in those areas, which may additionally provide some indication of how widely representations can vary.

9 Conclusion

One of the central goals in the cognitive sciences is the development of computational models of mental representations of object concepts. Such models may serve as a component for other behavioral prediction models or as a basis for identifying the nature of concept representations in the human brain. In this paper we introduced VICE, a novel VI approach for learning interpretable object concept embeddings by modeling human behavior in an odd-one-out triplet task. We showed that VICE predicts human behavior close to the estimated best attainable performance across three datasets and that VICE outperforms a competing method, SPoSE, in low data regimes. In addition, VICE has several characteristics that are desirable for scientific use. It has an automated procedure for determining the number of dimensions sufficient to explain the data, which further enables the detection of convergence during training. This leads to better model reproducibility across different random initializations and hyperparameter settings. As a result, VICE can be used out-of-the-box without requiring to perform an extensive search over random seeds or tuning model hyperparameters. Finally, we introduced a PAC learning bound on the generalization performance for a VICE model. Although VICE assumes a shared mental representation across participants - akin to much of the literature -, we believe that the VI framework can be leveraged to model inter-individual differences, which we plan to do in future work.

Acknowledgments

LM and RV acknowledge support by the Federal Ministry of Education and Research (BMBF) for the Berlin Institute for the Foundations of Learning and Data (BIFOLD) (01IS18037A). LM and MNH acknowledge support by a Max Planck Research Group grant awarded to MNH by the Max Planck Society. FP, CZ, and PM acknowledge the support of the National Institute of Mental Health Intramural Research Program (ZIC-MH002968). PM acknowledges the support of the Naval Postgraduate School's Research Initiation Program. This study utilized the high-performance computational capabilities of the Biowulf Linux cluster at the National Institutes of Health, Bethesda, MD (http://biowulf.nih.gov) and the Raven and Cobra Linux clusters at the Max Planck Computing & Data Facility, Garching, Germany (https://www.mpcdf.mpg.de/services/supercomputing/). The authors would like to thank Chris Baker for useful initial discussions, Shruti Japee, Jason Avery, and Alex Martin for sharing data, and Erik Daxberger, Lorenz Linhardt, Adrian Hill, Niklas Schmitz, and Marco Morik for valuable feedback on earlier versions of the paper.

References

[1] Alexandre Belloni and Victor Chernozhukov. Least squares after model selection in high-dimensional sparse models. *Bernoulli*, 19(2):521–547, 2013. doi: 10.3150/11-BEJ410.

[2] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: Series B (Methodological)*, 57(1):289–300, 1995. doi: https://doi.org/10.1111/j.2517-6161.1995.tb02031.

[3] Jeffrey R Binder, Lisa L Conant, Colin J Humphries, Leonardo Fernandino, Stephen B Simons, Mario Aguilar, and Rutvik H Desai. Toward a brain-based componential semantic representation. *Cognitive Neuropsychology*, 33(3-4):130–174, 2016.

[4] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017. doi: 10.1080/01621459.2017.1285773.

[5] Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. In Francis R. Bach and David M. Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pp. 1613–1622. JMLR.org, 2015.

[6] Erin M. Buchanan, K. D. Valentine, and Nicholas P. Maxwell. English semantic feature production norms: An extended database of 4436 concepts. *Behavior Research Methods*, 51(4): 1849–1863, May 2019. doi: 10.3758/s13428-019-01243-z.

[7] Madeline Carrington, Alexander G. Liu, Charles Y. Zheng, Lukas Muttenthaler, Francisco Pereira, Jason Avery, and Alex Martin. Naturalistic food categories are driven by subjective estimates rather than objective measures of food qualities. In *Society for Neuroscience Annual Meeting, Washington DC.*, 2021.

[8] Steven Derby, Paul Miller, Brian Murphy, and Barry Devereux. Using sparse semantic embeddings learned from multimodal text and image data to model human conceptual knowledge. In Anna Korhonen and Ivan Titov (eds.), *Proceedings of the 22nd Conference on Computational Natural Language Learning, CoNLL 2018, Brussels, Belgium, October 31 - November 1, 2018*, pp. 260–270. Association for Computational Linguistics, 2018. doi: 10.18653/v1/k18-1026.

[9] Steven Derby, Paul Miller, and Barry Devereux. Feature2vec: Distributional semantic modelling of human property knowledge. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pp. 5852–5858. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1595.

[10] Barry J. Devereux, Lorraine K. Tyler, Jeroen Geertzen, and Billi Randall. The centre for speech, language and the brain (CSLB) concept property norms. *Behavior Research Methods*, 46(4): 1119–1127, December 2013. doi: 10.3758/s13428-013-0420-4.

[11] David Duvenaud, Dougal Maclaurin, and Ryan Adams. Early stopping as nonparametric variational inference. In Arthur Gretton and Christian C. Robert (eds.), *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pp. 1070–1077, Cadiz, Spain, 09–11 May 2016. PMLR.

[12] Ludwig Fahrmeir, Thomas Kneib, and Susanne Konrath. Bayesian regularisation in structured additive regression: A unifying perspective on shrinkage, smoothing and predictor selection. *Statistics and Computing*, 20(2):203–219, 2010. doi: 10.1007/s11222-009-9158-3.

[13] Alona Fyshe, Partha Pratim Talukdar, Brian Murphy, and Tom M. Mitchell. Interpretable semantic vectors from a joint model of brain- and text-based meaning. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pp. 489–499. The Association for Computer Linguistics, 2014. doi: 10.3115/v1/p14-1046.

[14] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Maria-Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pp. 1050–1059. JMLR.org, 2016.

[15] Edward I. George and Robert E. McCulloch. Variable Selection via Gibbs Sampling. *Journal of the American Statistical Association*, 88(423):881–889, 1993. doi: 10.1080/01621459.1993.10476353.

[16] Jacqueline Gottlieb and Pierre-Yves Oudeyer. Towards a neuroscience of active sampling and curiosity. *Nature Reviews Neuroscience*, 19(12):758–770, November 2018. doi: 10.1038/s41583-018-0078-0.

[17] Alex Graves. Practical variational inference for neural networks. In John Shawe-Taylor, Richard S. Zemel, Peter L. Bartlett, Fernando C. N. Pereira, and Kilian Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a meeting held 12-14 December 2011, Granada, Spain*, pp. 2348–2356, 2011.

[18] Thomas L. Griffiths and Zoubin Ghahramani. The Indian Buffet Process: An Introduction and Review. *Journal of Machine Learning Research*, 12:1185–1224, 2011.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pp. 1026–1034. IEEE Computer Society, 2015. doi: 10.1109/ICCV.2015.123.

[20] Martin N Hebart, Adam H Dickter, Alexis Kidder, Wan Y Kwok, Anna Corriveau, Caitlin Van Wicklin, and Chris I Baker. Things: A database of 1,854 object concepts and more than 26,000 naturalistic object images. *PloS one*, 14(10):e0223792, 2019.

[21] Martin N. Hebart, Charles Y. Zheng, Francisco Pereira, and Chris I. Baker. Revealing the multi-dimensional mental representations of natural objects underlying human similarity judgements. *Nature Human Behaviour*, 4(11):1173–1185, October 2020. doi: 10.1038/s41562-020-00951-3.

[22] Mariam Hovhannisyan, Alex Clarke, Benjamin R. Geib, Rosalie Cicchinelli, Zachary Monge, Tory Worth, Amanda Szymanski, Roberto Cabeza, and Simon W. Davis. The visual and semantic features that predict object memory: Concept property norms for 1, 000 object images. *Memory & Cognition*, 49(4):712–731, January 2021. doi: 10.3758/s13421-020-01130-5.

[23] Hemant Ishwaran and J. Sunil Rao. Spike and slab variable selection: frequentist and bayesian strategies. *The Annals of Statistics*, 33(2):730–773, 2005.

[24] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[25] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In Yoshua Bengio and Yann LeCun (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[26] Bas JK Kleijn and Aad W van der Vaart. The Bernstein-von-Mises theorem under misspecification. *Electronic Journal of Statistics*, 6:354–381, 2012.

[27] Maren Mahsereci, Lukas Balles, Christoph Lassner, and Philipp Hennig. Early stopping without a validation set. *arXiv e-prints*, art. arXiv:1703.09580, March 2017.

[28] Gertraud Malsiner-Walli and Helga Wagner. Comparing spike and slab priors for bayesian variable selection. *arXiv e-prints*, art. arXiv:1812.07259, December 2018.

[29] Patrick McClure and Nikolaus Kriegeskorte. Robustly representing uncertainty in deep neural networks through sampling. *arXiv e-prints*, art. arXiv:1611.01639, November 2016.

[30] Ken McRae, George S. Cree, Mark S. Seidenberg, and Chris Mcnorgan. Semantic feature production norms for a large set of living and nonliving things. *Behavior Research Methods*, 37 (4):547–559, November 2005. doi: 10.3758/bf03192726.

[31] T. J. Mitchell and J. J. Beauchamp. Bayesian variable selection in linear regression. *Journal of the American Statistical Association*, 83(404):1023–1032, 1988. doi: 10.1080/01621459.1988.10478694.

[32] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. Adaptive Computation and Machine Learning Series. The MIT Press, 2018.

[33] Brian Murphy, Partha Talukdar, and Tom Mitchell. Learning effective and interpretable semantic models using non-negative sparse embedding. In *Proceedings of COLING 2012*, pp. 1933–1950, Mumbai, India, December 2012. The COLING 2012 Organizing Committee.

[34] Daniel J Navarro and Thomas L Griffiths. Latent features in similarity judgments: A nonparametric Bayesian approach. *Neural Computation*, 20(11):2597–2628, 2008.

[35] Abhishek Panigrahi, Harsha Vardhan Simhadri, and Chiranjib Bhattacharyya. Word2sense: Sparse interpretable word embeddings. In Anna Korhonen, David R. Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pp. 5692–5705. Association for Computational Linguistics, 2019. doi: 10.18653/v1/p19-1570.

[36] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 8024–8035, 2019.

[37] Francisco Pereira, Matthew Botvinick, and Greg Detre. Using Wikipedia to learn semantic feature representations of concrete concepts in neuroimaging experiments. *Artificial Intelligence*, 194:240–252, 2013. ISSN 0004-3702. doi: https://doi.org/10.1016/j.artint.2012.06.005. Artificial Intelligence, Wikipedia and Semi-Structured Resources.

[38] Kolyan Ray, Botond Szabó, and Gabriel Clara. Spike and slab variational Bayes for high dimensional logistic regression. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[39] Colorado Reed and Zoubin Ghahramani. Scaling the Indian Buffet Process via Submodular Maximization. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 1013–1021. JMLR.org, 2013.

[40] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In Eric P. Xing and Tony Jebara (eds.), *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 1278–1286, Bejing, China, 22–24 Jun 2014. PMLR.

[41] Brett D Roads and Bradley C Love. Enriching ImageNet with human similarity judgments and psychological embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3547–3557, 2021.

[42] Herbert Robbins and Sutton Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, September 1951. doi: 10.1214/aoms/1177729586.

[43] Veronika Rocková and Nicholas Polson. Posterior concentration for sparse deep learning. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 938–949, 2018.

[44] Samuel L. Smith, Erich Elsen, and Soham De. On the generalization benefit of noise in stochastic gradient descent. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 9058–9067. PMLR, 2020.

[45] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[46] Anant Subramanian, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard H. Hovy. SPINE: sparse interpretable neural embeddings. In Sheila A. McIlraith and Kilian Q. Weinberger (eds.), *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pp. 4921–4928. AAAI Press, 2018.

[47] Michalis Titsias and Miguel Lázaro-Gredilla. Doubly stochastic variational bayes for non-conjugate inference. In Eric P. Xing and Tony Jebara (eds.), *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pp. 1971–1979, Bejing, China, 22–24 Jun 2014. PMLR.

[48] Michalis K. Titsias and Miguel Lázaro-Gredilla. Spike and Slab Variational Inference for Multi-Task and Multiple Kernel Learning. In John Shawe-Taylor, Richard S. Zemel, Peter L. Bartlett, Fernando C. N. Pereira, and Kilian Q. Weinberger (eds.), *Advances in Neural Information Processing Systems 24: Annual Conference on Neural Information Processing Systems 2011, NeurIPS 2011, December 12-14, 2011, Granada, Spain*, pp. 2339–2347, 2011.

[49] Akira Utsumi. Exploring what is encoded in distributional word vectors: A neurobiologically motivated analysis. *Cognitive Science*, 44(6), 2020. doi: 10.1111/cogs.12844.

[50] Charles Y. Zheng, Francisco Pereira, Chris I. Baker, and Martin N. Hebart. Revealing interpretable object representations from human behavior. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See § 9.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See `https://github.com/LukasMut/VICE` for our codebase. See `https://osf.io/jum2f/` and `https://things-initiative.org/` for data.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See § 7, § 7 and E.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Figure 2 and Table 1.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See E
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] See `https://github.com/LukasMut/VICE`.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] See G.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

   We re-used crowdsourced data collected by various collaborators, and attribute it to them in the main paper. Even though we did not collect these data ourselves, we provide a summary of the information asked for in 5. in G, as well as a short description of the task and stimuli in Section 3.

A Triplet task

The triplet task compares favorably to other possible alternatives for eliciting human pairwise object similarity judgments. Asking for similarity ratings directly, on a numeric or qualitative scale, introduces the difficulty of differing inter-individual calibration of ratings. Having a choice task reduces this inter-individual variability by reducing the number of possible actions the user can make, e.g., three in the triplet task. Within choice tasks, there are also k-way forced choice tasks which specify a reference object and query the user for the most similar object to the reference among a list of k candidate objects. It seems likely that the choice in the forced choice task is not strictly driven by pairwise item similarities between the choices and the reference, but that the reference may influence the selection of features used to assess item similarities. For instance, the subject may choose the most prominent feature associated with the reference, and then make a uni-dimensional assessment of which candidate scores highest on that feature. This is less likely to be an issue in the triplet task because with no reference object, the subject has to consider all three pairs of items to evaluate which pair has greater similarity than the other two pairs. In particular, it would be less feasible to try to find a single feature that would be relevant for all three pairs of objects. Hence, we speculate that the triplet task may promote more multidimensional similarity assessments than the k-way forced-choice task.

Many studies find it convenient to pool triplet tasks from multiple participants. This could be justified under an assumption where subjects within a population all share the same common mental representations of items, which in turn determine the pairwise similarities that govern the choice of the odd-one-out selection. Alternatively, it could be the case that the representation of each item has random variation, both within a subject and across subjects. In such a scenario, it could be useful to use a model which assumes that a *probabilistic* (rather than a *deterministic*) embedding governs the triplet choices. As we discussed theoretically in §5 and showed empirically in §7, variational bayesian inference is one way that appears appropriate to model such a probabilistic embedding.

B Objective function

B.1 KL divergence

In VI, one minimizes the KL divergence between $q_\theta(X)$, the variational posterior, and, $p(X|\mathcal{D})$, the true posterior,

$$\arg\min_\theta D_{\mathrm{KL}}(q_\theta(X)\|p(X|\mathcal{D})),$$

where

$$\begin{aligned}
D_{\mathrm{KL}}(q_\theta(X)\|p(X|\mathcal{D})) &= \mathbb{E}_{q_\theta(X)}\left[\log \frac{q_\theta(X)}{p(X|\mathcal{D})}\right] \\
&= \mathbb{E}_{q_\theta(X)}\big[\log q_\theta(X) - \log p(X|\mathcal{D})\big] \\
&= \mathbb{E}_{q_\theta(X)}\left[\log q_\theta(X) - \log \frac{p(\mathcal{D}|X)p(X)}{p(\mathcal{D})}\right] \\
&= \mathbb{E}_{q_\theta(X)}\big[\log q_\theta(X) - \log p(X) - \log p(\mathcal{D}|X)\big] + \log p(\mathcal{D}) \\
&= \mathbb{E}_{q_\theta(X)}\left[\log q_\theta(X) - \log p(X) - n\log C - \sum_{s=1}^{n} \log p(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X)\right] \\
&\quad + \log p(\mathcal{D}) \\
&= \mathbb{E}_{q_\theta(X)}\left[\log q_\theta(X) - \log p(X) - \sum_{s=1}^{n} \log p(\{y_s, z_s\}|\{i_s, j_s, k_s\}, X)\right] \\
&\quad - n\log C + \log p(\mathcal{D}).
\end{aligned}$$

B.2 VICE objective function

Because $n \log C$ and $\log p(\mathcal{D})$ are constants and not functions of the variational parameters, we can ignore both terms in the minimization and get the following VICE objective

$$\arg\min_{\theta} \; \mathbb{E}_{q_\theta(X)} \left[\log q_\theta(X) - \log p(X) - \sum_{s=1}^{n} \log p(\{y_s, z_s\} | \{i_s, j_s, k_s\}, X) \right].$$

Multiplying this by $(1/n)$, where n is the number of training examples, does not change the minimum of the objective function and results in

$$\arg\min_{\theta} \; \mathbb{E}_{q_\theta(X)} \left[\frac{1}{n}(\log q_\theta(X) - \log p(X)) - \frac{1}{n} \sum_{s=1}^{n} \log p(\{y_s, z_s\} | \{i_s, j_s, k_s\}, X) \right].$$

C Optimization, convergence and prior

C.1 Gradient-based optimization

In gradient-based optimization, the gradient of an objective function with respect to the parameters of a model, $\nabla \mathcal{L}(\theta)$, is used to iteratively find parameters $\hat{\theta}$ that minimize that function. Equation 5 computes the expected log-likelihood of the entire training data. However, using every training data point to compute a gradient update is computationally expensive for large datasets and often generalizes poorly for non-convex objective functions [44]. In VICE, we stochastically approximate the training log-likelihood using random subsets (i.e., mini-batches) of the training data, where each mini-batch consists of B triplets [42]. This leads to an objective function that is a doubly stochastic approximation of Equation 5 [47], due to sampling weights from the variational distribution, $q_\theta \in \mathcal{Q}$, and sampling a random mini-batch of training examples during each gradient step (i.e., performing mini-batch gradient descent),

$$\mathcal{L}_{batch} = \frac{1}{n}(\log q_\theta(X_{\theta,\epsilon}) - \log p(X_{\theta,\epsilon})) - \frac{1}{B} \sum_{b=1}^{B} \log p(\{y_b, z_b\} | \{i_b, j_b, k_b\}, [X_{\theta,\epsilon}]_+)],$$

where $p(\{y, z\} | \{i, j, k\}, X)$ for a single sample is defined in Equation 3. To find parameters, θ, that optimize Equation 5, we iteratively update both the means, μ, and the standard deviations, σ, of each VICE dimension, by

$$\mu_{t+1} \coloneqq \mu_t - \alpha \nabla_{\mu_t} \mathcal{L}_{batch}$$

and

$$\sigma_{t+1} \coloneqq \sigma_t - \alpha \nabla_{\sigma_t} \mathcal{L}_{batch},$$

where α is the learning rate for θ.

C.2 Convergence

We define *convergence* as the point in time, t^*, where the number of embedding dimensions has not changed by a *single* dimension over the past L epochs. We denote this as the point of *representational stability*. To ensure convergence, we recommend letting L be relatively large (e.g., $L \gg 100$). We found $L = 500$ to work well for our experiments. Figure 4 shows that the convergence criterion defined in §5.2 worked reliably for all datasets.

C.3 (In-)efficient prior choice

As discussed in §5.1, SPoSE imposes a combination of an ℓ_1 penalty and a non-negativity constraint on its embedding values. This is analogous to having an exponential prior on those values. If we consider the distribution of values across objects for the two most important SPoSE dimensions in Figure 5, we can see that they show a distribution with a spike around 0, and a much smaller, wide slab of probability mass for non-zero values. Overcoming the exponential prior is data inefficient. SPoSE was developed using a dataset that was orders of magnitude larger than what a typical psychological experiment might collect, a setting in which this issue would likely not manifest itself. However, SPoSE has not been tested on the more common, smaller datasets, and our results suggest that the implicit SPoSE prior leads to suboptimal results in those regimes, when compared to the spike-and-slab prior used by VICE, as shown in Figure 2.

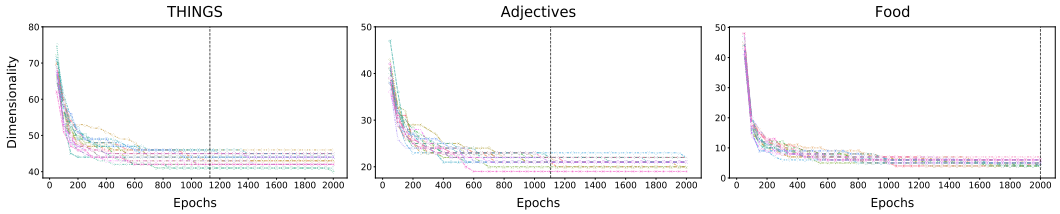

Figure 4: These plots show the number of embedding dimensions over time for THINGS, ADJECTIVES, and FOOD respectively. Each line in a plot corresponds to a single random seed. Vertical dashed lines indicate the median number of epochs (across random seeds) until the convergence criterion with $L = 500$ was met.

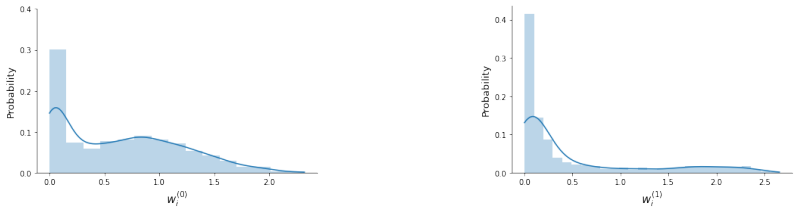

Figure 5: Histograms and PDFs of the first two SPoSE dimensions after training.

D Proof of Proposition 3.1

Proof. Recall that

$$\hat{R}(X) := \sum_{s=1}^{n} \frac{1}{n} \mathbb{1}\left(\{y_s, z_s\} \neq \underset{\{y,z\}}{\arg\max}\, p\left(\{y,z\} | \{i_s, j_s, k_s\}, X\right) \right)$$

and

$$R(X) := P_{\{y',z'\}|\{i',j',k'\}}\left(\{y', z'\} \neq \underset{\{y,z\}}{\arg\max}\, p\left(\{y,z\} | \{i', j', k'\}, X\right) \right).$$

Using the union bound we have that

$$P\left(\sup_{X \in S^{m \times d}} \hat{R}(X) - R(X) \geq \epsilon \right) \leq \sum_{X \in S^{m \times d}} P\left(\hat{R}(X) - R(X) \geq \epsilon \right). \tag{8}$$

Since $\mathbb{E}\left[\hat{R}(X)\right] = R(X)$ and the summands of $\hat{R}(X)$ are independent and satisfy

$$0 \leq \frac{1}{n} \mathbb{1}\left(\{y_s, z_s\} \neq \underset{\{y,z\}}{\arg\max}\, p\left(\{y,z\} | \{i_s, j_s, k_s\}, X\right) \right) \leq \frac{1}{n}$$

for all $X \in S^{m \times d}$ we can apply Hoeffding's Inequality to yield

$$P\left(\hat{R}(X) - R(X) \geq \epsilon \right) \leq \exp\left(-\frac{2\epsilon^2}{\sum_{i=1}^{n} \left(\frac{1}{n} - 0\right)^2} \right) = \exp\left(-2\epsilon^2 n\right) \tag{9}$$

for all $X \in S^{m \times d}$. The set $S^{m \times d}$ contains $(k+1)^{md}$ elements so combining Equation 8 and Equation 9 we get

$$\begin{aligned} P\left(\sup_{X \in S^{m \times d}} \hat{R}(X) - R(X) \geq \epsilon \right) &\leq (k+1)^{md} \exp\left(-2\epsilon^2 n\right) \\ &= \exp\left(md \log(k+1) - 2\epsilon^2 n\right) \\ \iff P\left(\sup_{X \in S^{m \times d}} \hat{R}(X) - R(X) < \epsilon \right) &\geq 1 - \exp\left(md \log(k+1) - 2\epsilon^2 n\right). \end{aligned}$$

To arrive at the proposition statement we need that

$$1 - \delta \geq 1 - \exp\left(md \log(k+1) - 2\epsilon^2 n\right).$$

Solving for n we get

$$\begin{aligned} 1 - \exp\left(md \log(k+1) - 2\epsilon^2 n\right) &\geq 1 - \delta \\ \iff \exp\left(md \log(k+1) - 2\epsilon^2 n\right) &\leq \delta \\ \iff md \log(k+1) - 2\epsilon^2 n &\leq \log(\delta) \\ \iff \left(md \log(k+1) + \log(1/\delta)\right)/(2\epsilon^2) &\leq n. \end{aligned}$$

□

E Experimental details

Training Although we have developed a reliable convergence criterion for VICE (see §5.2), to guarantee a fair comparison between VICE and SPoSE, each model configuration was trained, using 20 different random seeds for 2000 epochs. Each VICE model was initialized with two weight matrices, $\mu \in \mathbb{R}^{m \times d}$ and $\log(\sigma) \in \mathbb{R}^{m \times d}$, where m refers to the number of unique objects in the dataset (THINGS: $m = 1854$; ADJECTIVES: $m = 2372$; FOOD: $m = 36$) and d, the initial dimensionality of the embedding, was set to 100 (the $\log$ ensures that σ is positive). In preliminary experiments, we observed that for sufficiently large d our dimensionality reduction method (see §5.2) prunes to similar representations, regardless of the choice of d. This is why we did not consider models with larger initial embedding dimensionality.

Weight initialization We initialized the means of the variational distributions, μ, following a Kaiming He initialization [19]. The logarithms of the scales of the variational distributions, $\log(\sigma)$, were initialized with $\epsilon = -1/s_\mu$, where s_μ is the standard deviation over the entires of μ, so $\log(\sigma) = \epsilon \mathbf{1}$.

Hyperparameter grid The final grid was the Cartesian product of the following hyperparameter sets: $\pi_{\text{spike}} = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$, $\sigma_{\text{spike}} = \{0.125, 0.25, 0.5, 1.0, 2.0\}$, $\sigma_{\text{slab}} = \{0.25, 0.5, 1.0, 2.0, 4.0, 8.0\}$, subject to the constraint $\sigma_{\text{spike}} \ll \sigma_{\text{slab}}$, where combinations that did not satisfy the constraint were discarded. We observed that setting $\sigma_{\text{slab}} > 8.0$ led to numerical overflow issues during optimization. For SPoSE, we used the same range as was done in Zheng et al. [50], with a finer grid of 64 values.

Optimal hyperparameters We found the optimal VICE hyperparameter combination through a two step procedure. First, among the final 180 combinations (see Cartesian product above), we applied our pruning method (see §5.2) to each model and kept the subsets of dimensions where more than 5 objects had non-zero weight. For SPoSE we used the pruning heuristic proposed in Zheng et al. [50]. We defined the optimal hyperparameter combination as that with the lowest average cross-entropy error on the validation set across twenty different random initializations. The optimal hyperparameter combinations for VICE and SPoSE on the full datasets are reported in Table 2.

Table 2: Optimal hyperparameter combinations for VICE and SPoSE according to the average cross-entropy error on the validation set for the three datasets THINGS, ADJECTIVES, and FOOD.

	VICE			SPoSE
DATA\ HYPERPARAM.	σ_{spike}	σ_{slab}	π	λ
THINGS	0.125	0.5	0.6	5.75
ADJECTIVES	0.25	0.5	0.6	4.96
FOOD	0.25	1.0	0.8	2.90

Computational resources This study utilized the high-performance computational capabilities of the Biowulf Linux cluster at the National Institutes of Health, Bethesda, MD (`http://biowulf.nih.gov`) and the Raven and Cobra Linux clusters at the Max Planck Computing & Data Facility (MPCDF),

Garching, Germany (`https://www.mpcdf.mpg.de/services/supercomputing/`). The total number of CPU hours used were approximately $40,000,000$ (Biowulf) and $5,000$ (MPCDF).

Accuracy upper bound For THINGS [50, 21] and ADJECTIVES, a random subset of triplets was chosen to be presented multiple times to different participants. For a given triplet - repeated over many participants - this provides a way to estimate the distribution of responses over all participants. If for a given triplet the response distribution is $(0.2, 0.3, 0.5)$, then the best predictor for the participants' responses is the third object. This results in an accuracy score of 50%, averaged across repetitions. Alternatively, one may observe a distribution of $(0.1, 0.8, 0.1)$ for a different triplet. The best one could do is to identify the second object as the odd-one-out, and get 80% accuracy. From this, we can see that no classifier can do worse than 33%. Taking the average best prediction accuracy over all of the repeated triplets gives us an estimate for the best possible average prediction score. This is defined to be the upper bound for the prediction performance.

F Generalization error bound

F.1 Algorithm for generalization error upper bound

Algorithm 1 Algorithm for generalization error upper bound via adaptive quantization

Input: μ
$M \leftarrow \max(\mu)$
$\{\Delta_1, \ldots, \Delta_m\} \leftarrow \{0.05, ..., 1.0\}$ ▷ pre-determined set of quantization scales
$\alpha \leftarrow 0.05$ ▷ desired Type I error control rate
$\delta \leftarrow \frac{\alpha}{m}$
for $i \in \{1, \ldots, m\}$ **do**
  $\mu_i^\dagger := \text{quantize}\,(\mu, \Delta_i)$ ▷ quantization with Δ_i
  $\hat{R}_i = \text{loss}\,(\mu_i^\dagger, \mathcal{D})$ ▷ (training) error
  $\bar{R}_i := \hat{R}_i + \sqrt{\frac{md\log(\lceil M/\Delta_i \rceil + 1) + \log(1/\delta)}{2N_{\text{train}}}}$ ▷ generalization upper bound
end for
$i^* := \arg\min\{\bar{R}_i, ..., \bar{R}_m\}$
Output: $(\mu_{i^*}^\dagger, \bar{R}_{i^*})$ ▷ $\bar{R}_{i^*}$ holds with probability $1 - \alpha$

We have the flexibility of choosing the quantization scale post-hoc. As long as we search over a pre-specified set of m quantization scales $\{\Delta_1, \ldots, \Delta_m\}$, using a union bound, the PAC bound holds simultaneously for all quantized embeddings with probability at least $1 - m\delta$. Therefore, we can find the quantization scale that gives us the best probabilistic upper bound on generalization error.

F.2 Quantization

This section describes a number of empirical findings which support the feasibility of obtaining useful bounds by using our proposed quantization-based PAC bound (see §6) and the algorithm for obtaining retrospective generalization bounds (see Appendix F.1).

Recall that we require two assumptions for our bounding approach to be effective. Specifically, we assume,

1. Sparse embeddings obtained by either SPoSE or VICE can be quantized in a relatively coarse fashion, with only marginal losses in predictive performance.
2. There exists an upper bound M on the largest value in an embedding.

In the following we present empirical results which demonstrate that those assumptions are satisfied for the three datasets, THINGS, ADJECTIVES, and FOOD, which we have used to evaluate VICE.

Recall that it is possible to choose a certain quantization scale, Δ, and round an embedding value to a non-negative integer multiple of Δ. While we employ quantization mainly to use STL bounds for finite hypothesis classes, it could have benefits for interpretation as well (e.g., a dimension would consist of labels such as zero (0), very low (0.5), low (1), medium (1.5), high (2)).

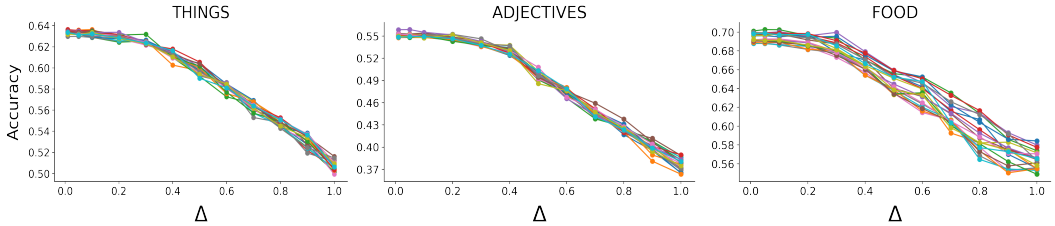

Figure 6: Test performance as a function of the quantization scale, Δ, for twenty random initializations for THINGS, ADJECTIVES, and FOOD.

In Figure 6 we show generalization performances of VICE models as a function of the quantization scale, Δ, for THINGS, ADJECTIVES, and FOOD. To quantize the VICE embeddings, we used the same set of quantization scales for every dataset, $\Delta \in \{0.05, \ldots, 1.0\}$, as defined in Algorithm 1. As Δ increases, the number of bins, to which the embedding values can possibly be assigned, decreases. That is, the space of possible embedding values gets smaller as a function of increasing Δ, and, therefore, maintaining generalization performance becomes more difficult. Note that quantization with $\Delta = 0$ is equivalent to performing inference with the original embeddings.

As a result, the optimal generalization bounds that we get by using Algorithm 1 are typically obtained with $\Delta \leq 0.2$. However, we explore a much larger range since *a priori* we do not know the level of granularity needed to preserve most of the information in an embedding. Theoretically, there exist embeddings, such as highly sparse embeddings with nearly binary elements, which could maintain high accuracy at large quantization scales. Hence, the optimal level of quantization for the bound is an empirical question that may vary across datasets. However, we limit the upper range to 1, because if the largest weight is less than 2.7 (as we will examine below), then $\Delta = 1$ already reduces the number of distinct weight values to 3, which is small enough that we are likely to see diminishing returns by considering even larger discretization scales.

Upper bound The upper bound, M, as defined in §6, for the VICE mean embeddings was approximately 2.7 for all datasets. Empirically, we found the maximum weights for THINGS, ADJECTIVES, and FOOD to be 2.6285, 2.2130 and 2.4307, respectively, across all random initializations. Theoretically, this upper bound is not surprising due to a combination of two factors. First, in most datasets, the cross-entropy error goes to infinity as weights become arbitrarily large, due to the increasing over-confidence of incorrect probability estimates. Hence, the optimal cross-entropy is achieved with bounded weights. Second, almost any kind of regularization, including the ℓ_1 regularization used in SPoSE and the spike-and-slab regularization used for VICE, further encourages the weights to shrink. The spike-and-slab prior, in particular, penalizes large weights much higher than small weights, which is a desirable property in gradient-based optimization.

G Dataset acquisition

All datasets were collected by crowd-sourcing human responses to the triplet task described in §3 on the Amazon Mechanical Turk platform, using workers located in the United States. For ADJECTIVES, all words deemed offensive were removed from the list of adjectives being considered. There are no offensive images in either THINGS or FOOD. All workers provided informed consent, and were compensated financially for their time ($0.5\ c$ per response, and additional $10\ c$ per completed HIT). Workers could participate in blocks of 20 triplet trials and could choose to work on as many such blocks as they liked. The online research was approved by the Office of Human Research Subject Protection at the National Institutes of Health and conducted in accordance with all relevant ethical regulations. Worker ages were not assessed, and no personally identifiable information was collected.

THINGS and ADJECTIVES A total of 5,526 workers participated in collecting the first dataset (5,301 after exclusion; 3,159 female, 2,092 male, 19 other, 31 no response). A total of 336 workers participated in collecting the second dataset (325 after exclusion; 156 female, 103 male, 66 not reported). Workers were excluded if they exhibited overly fast responses in at least 5 sets of 20 trials (the speed cut-off was 25% or more responses < 800ms and 50% or more responses $< 1,100$ms) or

if they carried out at least 200 trials and showed overly deterministic responses ($> 40\%$ of responses in one of the three odd-one-out positions; expected value, 33%).

FOOD A total of 554 subjects participated in collecting the dataset (487 after exclusion). Workers were excluded if they exhibited overly fast responses (reaction time of less than 500ms). They were also excluded if they failed on either of two catch trials in each HIT, where they saw images of '+', '-', '-' or '=', '+', or '=' instead of food pictures; they were instructed on the slide to select the '+'.

H Interpretability

Here, we display the top 6 objects for each of the 45 VICE dimensions THINGS [20]. Objects were sorted in descending order according to their absolute embedding value. As we have done for every other experiment, we used the *pruned* median model to guarantee the extraction of a representative sample of embedding dimensions without being over-optimistic with respect to their interpretability (see §7.1 for how the median model was identified).

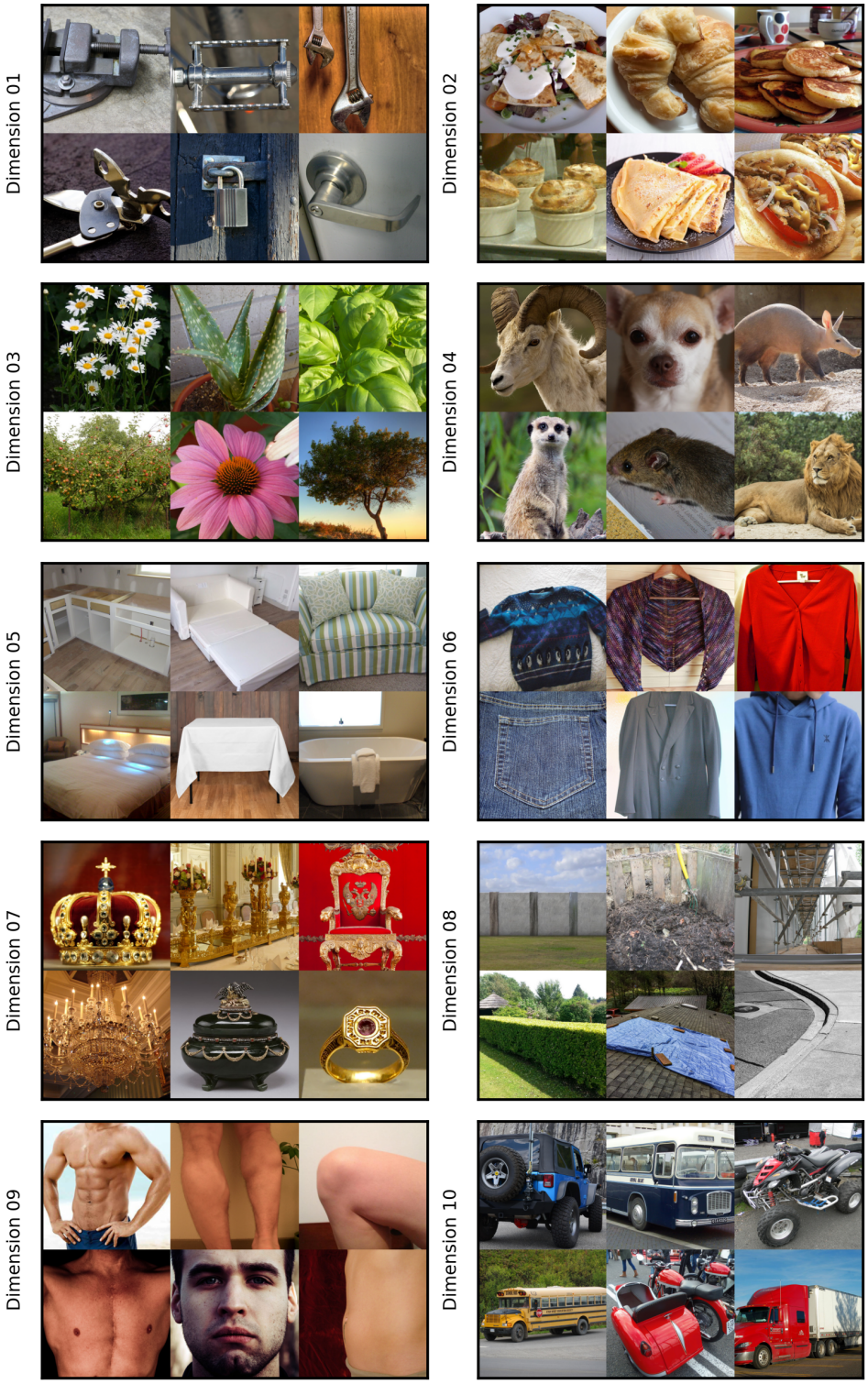

Figure 7: THINGS Dimensions 1-10.

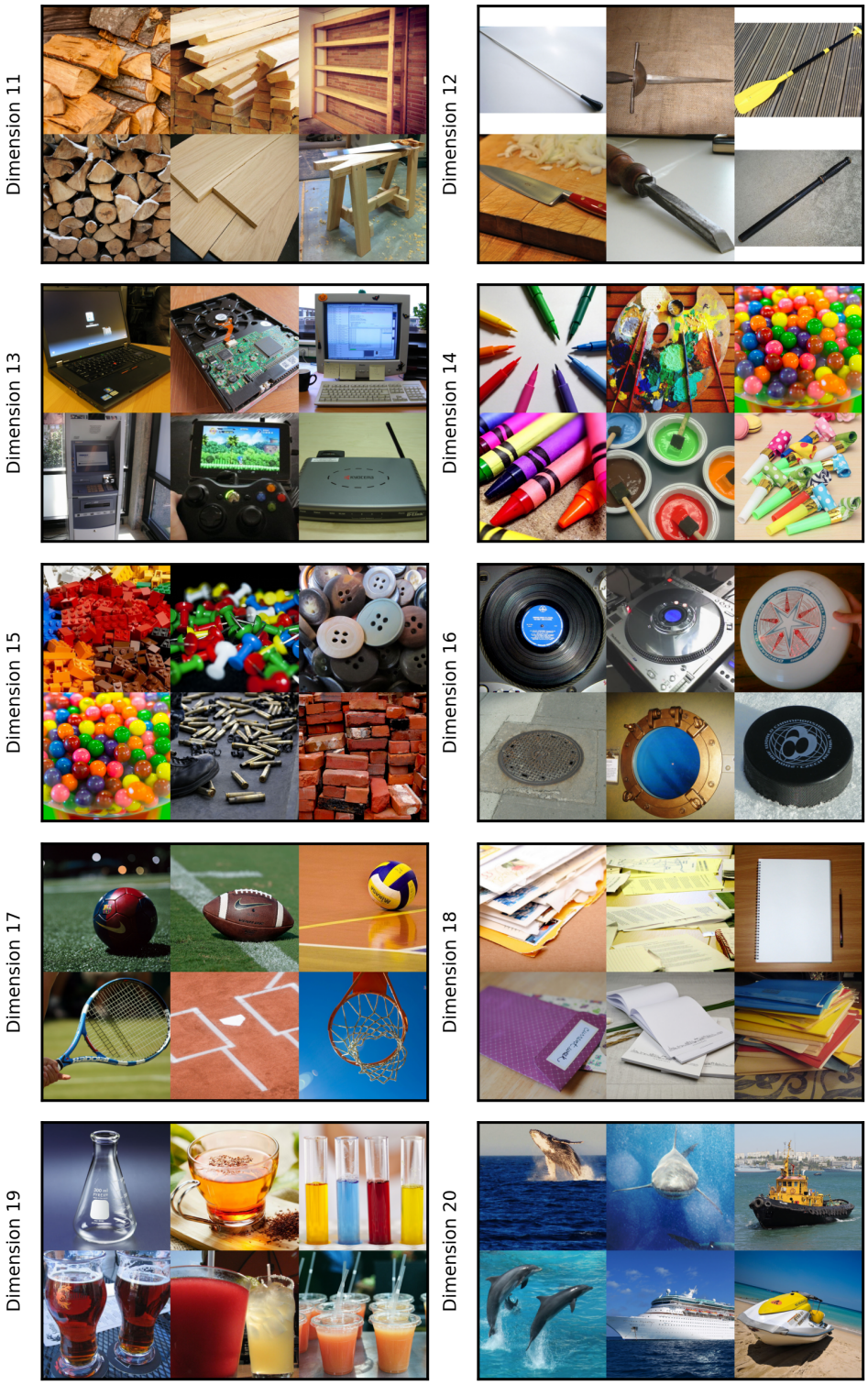

Figure 8: THINGS Dimensions 11-20.

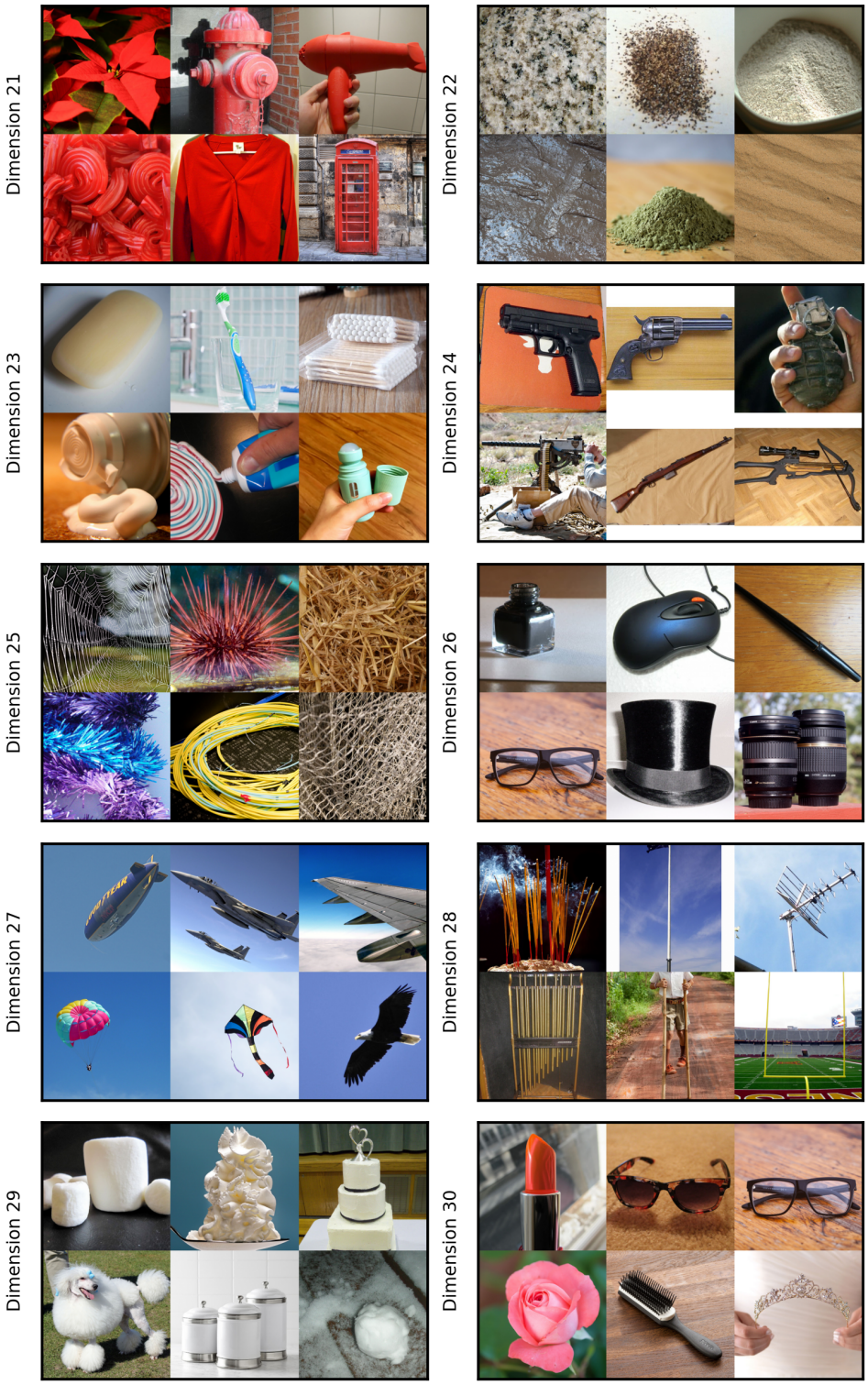

Figure 9: THINGS Dimensions 21-30.

Figure 10: THINGS Dimensions 31-40.

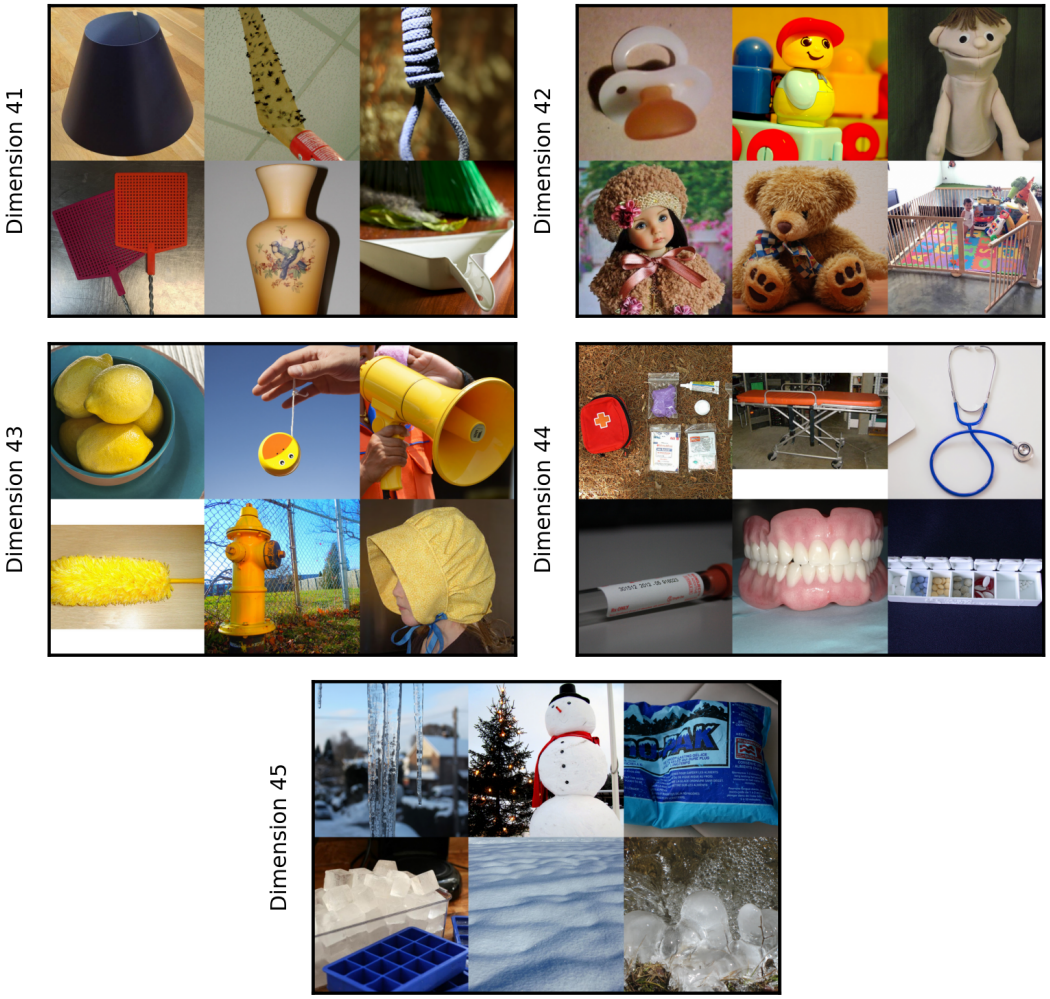

Figure 11: THINGS Dimensions 41-45.