VICE: VARIATIONAL INFERENCE FOR CONCEPT EMBEDDINGS

Anonymous authors
Paper under double-blind review

ABSTRACT

In this paper we introduce Variational Inference for Concept Embeddings (VICE), a novel method for learning object concept embeddings from human behavior in an odd-one-out task. We use variational inference to obtain a sparse, non-negative solution, with uncertainty information about each embedding value. We leverage this information in a statistical procedure for selecting the dimensionality of the model, based on hypothesis-testing over a validation set. VICE performs as well or better than previous methods on a variety of criteria: accuracy of predicting human behavior in an odd-one-out task, calibration to (empirical) human choice probabilities, reproducibility of object representations across different random initializations, and superior performance on small datasets. The latter is particularly important in cognitive science, where data collection is expensive. Finally, VICE yields highly interpretable object representations, allowing humans to describe the characteristics being represented by each latent dimension.

1 INTRODUCTION AND RELATED WORK

Human knowledge about object concepts encompasses many types of information, ranging from function to visual appearance, as well as encyclopedic facts or taxonomic characteristics. This knowledge supports the identification of objects, inferences about what interactions they support, or what the effects of such interactions in the environment will be. Key questions for cognitive scientists modelling human performance in experiments are 1) which of this information is accessible to participants and 2) how is it used across different tasks. Several studies (McRae et al., 2005; Devereux et al., 2013; Buchanan et al., 2019; Hovhannisyan et al., 2021) have asked subjects to list properties for hundreds to thousands of objects, yielding thousands of answers about the types of information above. Properties exist at many levels, ranging from categorization (e.g. "is an animal") to very specific facts (e.g. "is eaten in France"). Objects are implicitly represented as a vector of binary properties. This approach is agnostic to downstream prediction tasks, but does not provide an indication of which properties are more important – other than frequency of listing – and does not allow for graded property values. An alternative approach is for researchers to postulate dimensions of interest, and then ask human subjects to place each object in each dimension. An example is Binder et al. (2016), who collected ratings for hundreds of objects, as well as verbs and adjectives, in 65 dimensions reflecting sensory, motor, spatial, temporal, affective, social, and cognitive experiences.

The overall problem is then one of discovering a representation for objects that is not biased by a particular task, and is interpretable without requiring researchers to postulate the types of information represented. Several researchers have tried to develop interpretable concept representation spaces from text corpora, via word embeddings with positivity and sparsity constraints (Murphy et al., 2012), topic model representations of Wikipedia articles about objects (Pereira et al., 2013), transformations of word embeddings into sparse, positive spaces (Subramanian et al., 2018; Panigrahi et al., 2019) or predictions of properties (Devereux et al., 2013) or dimensions (Utsumi, 2020), or text corpora combined with imaging data (Fyshe et al., 2014) or with object images (Derby et al., 2018). Finally, Derby et al. (2019) introduced a neural network mapping the sparse feature space of a semantic property norm to the dense space of a word embedding, identifying informative combinations of properties or allowing ranking of candidate properties for arbitrary words.

Recently, Zheng et al. (2019) and Hebart et al. (2020) introduced SPoSE, a model of the mental representations of 1,854 objects in a 49-dimensional space. The model was derived from a dataset of

1.5M Amazon Mechanical Turk (AMT) judgments of object similarity, where subjects were asked which of a random triplet of objects was the odd one out. The model embedded each object as a vector in a space where each dimension was constrained to be sparse and positive. Triplet judgments were predicted as a function of the similarity between embedding vectors of the three objects considered. The authors showed that these dimensions were predictable as a *combination* of elementary properties in the Devereux et al. (2013) norm, which often co-occur across many objects. Hebart et al. (2020) further showed that 1) human subjects could coherently label what the dimensions were “about”, ranging from categorical (e.g. is animate, food, drink, building) to functional (e.g. container, tool) or structural (e.g. made of metal or wood, has inner structure). Subjects could also predict what dimension values new objects would have, based on knowing the dimension value for a few other objects. These results suggest that SPoSE captures core object knowledge that subjects use. Navarro & Griffiths (2008) introduced a related method for learning semantic concept embeddings from similarity data, which infers the number of latent dimensions using the Indian Buffet Process (IBP, Griffiths & Ghahramani (2011)), but their approach is not directly applicable to our setting due to reliance on continuous-valued similarity ratings instead of forced-choice behavior. Furthermore, it is known to be challenging to scale the IBP to the number of features and observations considered in our work (Ghahramani, 2013). Roads & Love (2021) introduced a related method for deriving an object embedding from behavior in a 8-rank-2 task. Their method aimed to predict behavior from the embeddings, using active sampling to query subjects with the most informative stimuli. The method was not meant to produce interpretable dimensions, but rather construct object similarity matrix as efficiently as possible.

There is growing interest by cognitive scientists in using SPoSE, as it makes it possible to discover an item representation for *any* kind of item amenable to an odd-one-out comparison in a triplet task. Furthermore, the combination of positivity and sparsity constraints in each dimension of the representation leads to interpretability by human subjects: no item is represented by every dimension, and most dimensions are present for only a few items. That item representation can then be used within other behavioral prediction models, to make predictions about neuroimaging data, etc.

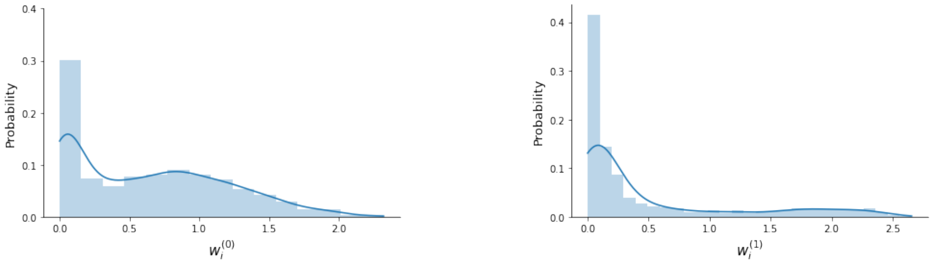


Figure 1: Histograms and PDFs of the first two SPoSE dimensions after training.

For this potential to be realized, however, we believe a number of issues with SPoSE should be addressed. The first is the use of an l_1 sparsity penalty to promote interpretability of dimensions. l_1 achieves sparsity at the cost of unnecessarily shrinking larger values (Belloni & Chernozhukov, 2013). In SPoSE, 6-11 dominant dimensions for an object account for most of the prediction performance; the cost of removing irrelevant dimensions is to potentially make dominant dimensions smaller than they should be, and affect performance. Second, the l_1 penalty is analogous to having a Laplace prior over those values. If we consider the distribution of values across objects for the two most important SPoSE dimensions, in Figure 1, we can see that they have a *bimodal* distribution, with a *spike* around 0 and a much smaller, wide *slab* of probability for non-zero values, which is not Laplace. Overcoming this "wrong" prior requires more data than strictly necessary to learn the representation. SPoSE was developed with a dataset that was orders of magnitude larger than what a typical experiment might collect, but it was never tested on smaller datasets. Finally, SPoSE uses a heuristic, subjective criterion for determining how many dimensions the solution should have.

In this paper we introduce VICE, an approach for variational inference of object concept embeddings in a space with interpretable sparse, positive dimensions, which addresses the SPoSE issues identified above. Specifically, we encourage sparsity and small weights by using a *spike-and-slab* prior. This is more appropriate than a Laplace prior, because *importance* – the value an object takes in a dimension – is different from *relevance* – whether the dimension matters for that object – and they can be controlled separately with a spike-and-slab prior. The prior hyperparameters are meant to be intuitive to a user, and to make it easier to specify hypotheses about dimensional structure. We use variational Bayes both because it is a Bayesian approach, and also because it assumes a unimodal posterior for the loading of each object in each dimension. It also allows a more principled procedure for determining how many dimensions the model should have, by taking into account uncertainty about their values. We compare our model with SPoSE over different subsets of the dataset used to develop it, and verify that it performs as well or better by various criteria: prediction of behavior, calibration of the prediction of decision probabilities, and reproducibility of solutions across seeds. Importantly, it has significantly better performance on smaller datasets ($5 - 10\%$ of the original SPoSE dataset). Our implementation of VICE is available on GitHub[1], and will be de-anonymized upon acceptance.

2 Methods

2.1 Odd-one-out task

The odd-one-out task is motivated by the problem of discovering object embeddings based on similarity judgments involving a set of m different object concepts, which we will denote by $c_1, \ldots, c_m$ (e.g. $c_1 =$ ‘aardvark’,..., $c_{1854} =$ ‘zucchini’). These similarity judgments are collected from human participants, who are given queries which consist of a ‘triplet‘ of three concepts $\{c_{i_1}, c_{i_2}, c_{i_3}\}$, for instance, $\{c_{268}, c_{609}, c_{1581}\} = \{$‘suit’, ‘flamingo’, ‘car’$\}$. Participants are asked to consider the three pairs within the triplet $\{(c_{i_1}, c_{i_2}), (c_{i_1}, c_{i_3}), (c_{i_2}, c_{i_3})\}$, and to decide which item had the smallest similarity to the other two (the "odd-one-out"). This is equivalent to choosing the pair with the greatest similarity. Let (y_1, y_2) denote the indices in this pair, e.g. for ‘suit’ and ‘flamingo’ they would be $(y_1, y_2) = (268, 609)$. A dataset $\mathcal{D}$ is a set of N pairs of concept triplets and one-hot vectors that correspond to the index two most similar concepts, i.e. $(\{c_{i_1}, c_{i_2}, c_{i_3}\}, (y_{i_1}, y_{i_2}))$.

Figure 2: Sample suit-flamingo-car triplet for the odd-one-out task. (Creative Commons; Authors: Janderk1968, Charles J. Sharp, and Wally Gobetz, respectively)

2.2 SPoSE

Sparse Positive object Similarity Embedding (SPoSE) (Zheng et al., 2019) is an approach for finding interpretable item dimensions from an odd-one-out task. It does so by finding an embedding vector $\mathbf{x_i} = (x_{i1}, \ldots, x_{ip})$ for every item c_i. The similarity S_{ij} of two items (e.g. c_i and c_j) is computed by the dot product of the corresponding embeddings (i.e. x_i and x_j), $S_{ij} = \langle \mathbf{x_i}, \mathbf{x_j} \rangle$ From these similarities, the probability of choosing (y_{i_1}, y_{i_2}) as the most similar pair of items given the item triplet $\{c_{i_1}, c_{i_2}, c_{i_3}\}$ and given embedding vectors $\{x_{i_1}, x_{i_2}, x_{i_3}\}$ is computed as:

$$p((y_{i_1}, y_{i_2}) | \{c_{i_1}, c_{i_2}, c_{i_3}\}, \{x_{i_1}, x_{i_2}, x_{i_3}\}) = \frac{\exp(S_{y_{i_1}, y_{i_2}})}{\exp(S_{i_1,i_2}) + \exp(S_{i_1,i_3}) + \exp(S_{i_2,i_3})}. \tag{1}$$

SPoSE uses a maximum a posterori (MAP) estimation to find the most likely embedding given the training data and a prior:

$$\arg\max_X \log p(X | \mathcal{D}_{train}) = \arg\max_X \log p(\mathcal{D}_{train} | X) + \log p(X), \tag{2}$$

[1] Link to anonymous GitHub repository: `https://anonymous.4open.science/r/VICE-59F0`

where X is a matrix containing the embedding vectors for all of the items and $p(X)$ is a prior for the embeddings, and $p(\mathcal{D}_{train,j}|X)$ is defined in (7). To induce sparsity in the embeddings, SPoSE uses a mean-field Laplace prior, leading to this objective:

$$\arg\max_{X} \sum_{j=1}^{n_{train}} \log p(\mathcal{D}_{train,j}|X) + \lambda \sum_{i=1}^{m} ||\mathbf{x_i}||_1 \tag{3}$$

Here, $|| \cdot ||_1$ is the l_1 norm, so $||\mathbf{x}||_1 = \sum_{f=1}^{p} |x_f|$, and $x_f \geq 0$ for $f = 1, \ldots, p$. The regularization parameter, λ, is selected out of a grid of candidate values by choosing the one that achieves the lowest (average) cross-entropy on the validation set (across twenty random seeds). The final dimensionality of the embedding, p, is determined heuristically from the data. If p is set to be larger than the number of dimensions supported by the data, the SPoSE algorithm will shrink entire dimensions towards zero by removing weights with a magnitude less than a given absolute threshold. While a threshold of 0.1 is suggested (Zheng et al., 2019), no justification is given for that particular value, which is problematic given that the number of dimensions removed is quite sensitive to that choice.

2.3 VICE

2.3.1 Variational Bayesian Inference

Given the goal of better approximating $p(X|\mathcal{D}_{train})$, we use variational inference. We approximate $p(X|\mathcal{D}_{train})$ with a variational distribution, $q_\theta(X)$, where q is our chosen family of distributions, and θ is a parameter that is learned in order to optimize the Kullback–Leibler (KL) divergence to the true posterior, $p(X|\mathcal{D}_{train})$. In variational inference, the KL divergence objective function is:

$$\arg\min_{\theta} \mathbb{E}_{q_\theta(X)} \left[\frac{1}{n_{train}} (\log q_\theta(X) - \log p(X)) - \frac{1}{n_{train}} \sum_{i=1}^{n_{train}} \log p(\mathcal{D}_{train,i}|X) \right] \tag{4}$$

In order to use variational inference, a parametric variational distribution must be chosen. For VICE, we use a Gaussian variational distribution with a diagonal covariance matrix $q_\theta(X) = \mathcal{N}(\boldsymbol{\mu}, \text{diag}(\boldsymbol{\sigma}^2))$, where the learnable parameters θ are $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$. This means that each embedding dimension has a mean, the most likely value for that dimension, and a standard deviation, the propensity of the embedding value to be close to the mean.

Similarly to Blundell et al. (2015), we use a Monte Carlo (MC) approximation of the above objective function by sampling a limited number of Xs from $q_{\boldsymbol{\mu},\boldsymbol{\sigma}}(X)$ during training. We generate X by means of the reparameterization trick (Kingma & Welling, 2013), $X_{\theta,\boldsymbol{\epsilon}} = \boldsymbol{\mu} + \boldsymbol{\sigma} \cdot \boldsymbol{\epsilon}$ where $\boldsymbol{\epsilon}$ is an $N \times p$ matrix of standard normal variates, leading to the objective:

$$\arg\min_{\theta} \frac{1}{m} \sum_{j=1}^{m} \left[\frac{1}{n_{train}} (\log q_\theta(X_{\theta,\boldsymbol{\epsilon}^j}) - \log p(X_{\theta,\boldsymbol{\epsilon}^j})) - \frac{1}{n_{train}} \sum_{i=1}^{n_{train}} \log p(\mathcal{D}_{train,i}|[X_{\theta,\boldsymbol{\epsilon}^j}]_+ \right] \tag{5}$$

where $\boldsymbol{\epsilon}^j \in \mathbb{R}^{N \times p}$ is entrywise $\mathcal{N}(0,1)$ and where $[]_+$ is the ReLU function. As commonly done in the dropout and Bayesian neural network literature (Srivastava et al., 2014; Blundell et al., 2015; Gal & Ghahramani, 2016; McClure & Kriegeskorte, 2016), we set m to 1 for computational efficiency.

In Equation 5, the expected log-likelihood of the entire training data is computed. However, using the entire training data set to compute the gradient update often works poorly for non-convex objective functions. This is due to the expensive computational cost of each update and to the convergence to poorly generalizing solutions (Smith et al., 2020). As a result, we stochastically approximate (Robbins & Monro, 1951) the training log-likelihood using random subsets (i.e. mini-batches) of the training dataset, with each mini-batch consisting of b triplets. This leads to the final objective

$$\arg\min_{\theta} \frac{1}{n_{train}} (\log q_\theta(X_{\theta,\boldsymbol{\epsilon}}) - \log p(X_{\theta,\boldsymbol{\epsilon}})) - \frac{1}{b} \sum_{i=1}^{b} \log p(\mathcal{D}_{train,i}|[X_{\theta,\boldsymbol{\epsilon}}]_+)] \tag{6}$$

recalling that

$$p(\mathcal{D}_{train,i}|X) = \frac{\exp(\mathbf{x_{y_{1,i}}}^T \mathbf{x_{y_{2,i}}})}{\exp(\mathbf{x_{i_{1,i}}}^T \mathbf{x_{i_{2,i}}}) + \exp(\mathbf{x_{i_{1,i}}}^T \mathbf{x_{i_{3,i}}}) + \exp(\mathbf{x_{i_{2,j}}}^T \mathbf{x_{i_{3,i}}})}. \tag{7}$$

2.3.2 Spike-and-Slab Prior

A key feature of SPoSE is sparsity. As discussed above, SPoSE induced sparsity using a zero-mean Laplace prior. We can empirically examine whether the Laplace prior is a realistic assumption, given the distribution of values in SPoSE dimensions. As Figure 1 depicts, these histograms do not resemble a Laplace distribution. Instead, it looks like there is a "spike" of probability at zero and a much smaller, but wide, "slab" of probability for the non-zero values of a SPoSE dimension. To model this, we use a *spike-and-slab* Gaussian mixture prior, as introduced in Blundell et al. (2015):

$$p(X) = \prod_{i=1}^{N} \prod_{f=1}^{p} (\pi \mathcal{N}(x_{if}; 0, \sigma^2_{spike}) + (1-\pi)\mathcal{N}(x_{if}; 0, \sigma^2_{slab})) \tag{8}$$

This prior has three parameters. π is the probability that an embedding dimension will be drawn from the spike Gaussian instead of the slab Gaussian. The standard deviations σ_{spike} and σ_{slab} control the likelihood of of an embedding value being set to 0 in the spike or slab distributions, respectively. x_{if} is the embedding weight for the ith item in the fth dimension. Since spike and slab distributions are mathematically interchangeable, by convention we require that, $\sigma_{spike} << \sigma_{slab}$. In our experiments, these are chosen with grid search on one half of the validation set, the “tuning set”. (The other half of the validation set, the “pruning set”, is used for dimensionality reduction, as we describe in §2.3.4.)

2.3.3 Predicting the Odd-one-out using VICE

In this work, we consider two different prediction problems given a new triplet: (1) predicting the choice and (2) predicting the distribution of the choice. In either case, we start with computing the posterior probability distribution over the three triplet choices. If predicting a choice, then we output the choice with the maximum posterior probability. (For details on how we handle ties, see §3.3.1.) If the goal is to predict the distribution, then we return the predicted distribution.

The predicted probability distribution is computed from the variational posterior, $q_\theta(X)$. When making predictions, we want to compute the probability of an odd-one-out for a given triplet. We approximate this probability by using an MC estimate from m samples $X^j = X_{\theta,\boldsymbol{\epsilon}^j}$ for $j = 1, \ldots, m$ (Graves, 2011; Blundell et al., 2015; Kingma & Welling, 2014; McClure & Kriegeskorte, 2016; Blei et al., 2017). Mathematically, this means that we compute the predicted distribution as

$$\hat{p}((y_{i_1}, y_{i_2})|\{c_{i_1}, c_{i_2}, c_{i_3}\}) \approx \frac{1}{m}\sum_{j=1}^{m} p((y_{i_1}, y_{i_2})|\{c_{i_1}, c_{i_2}, c_{i_3}\}, \{[x^j_{i_1}]_+, [x^j_{i_2}]_+, [x^j_{i_3}]_+\}). \tag{9}$$

2.3.4 Dimensionality Reduction for VICE

For interpretability purposes, it is crucial that the model does not use more dimensions than necessary. Zheng et al. (2019) accomplished this through the sparsity-inducing penalty that causes dimensions to shrink towards zero if they do not contribute to explaining the data. (see Section 2.2). Note, however, that these uninformative weights do not totally go to zero because of noise in the gradients. Hence, these dimensions were pruned by choosing a threshold for the L1 norm of the dimension based on looking at the “elbow plot” of the sorted L1 norms of the dimensions; this approach is subjective and highly dependent on the specific dataset. In VICE, the KL penalty we use has a similar effect of causing uninformative weights to shrink. Rather than using a user-defined threshold to prune dimensions, VICE exploits the uncertainty information obtained in training the model to select a set of informative dimensions. The pruning procedure consists of three steps: (1) assigning an importance score to each dimension; (2) clustering dimensions by importance; and (3) choosing the subset of clusters that best explains the validation set. We describe each of these three steps in detail below.

Assign an importance score to each dimension Intuitively, the importance score reflects the number of objects that we can confidently say have non-zero weight in a dimension. To compute the score, we start by using the variational embedding for each item i – location μ_{ij} and scale σ_{ij} parameters, to compute the posterior probability that the weight will be truncated to zero according to the left tail of a Gaussian distribution with that location and scale (as described in §2.3.1). This gives us a posterior probability of the weight taking the value zero for each item within a dimension (Graves, 2011). To calculate the overall importance of a dimension, we estimate the number of items that plausibly have non-zero weights given a user-specified False Discovery Rate target (FDR) (Benjamini & Hochberg, 1995). FDR provides a method for inferring the number of hypotheses which are non-null, based on an array of p-values, with statistical guarantees on the expected proportion of false rejections. We define dimension importance as the number of rejections given by the BH(q) algorithm, with the FDR tolerance q specified by the user, using the posterior zero-probabilities as the p-values.

Cluster dimensions by importance using a Gaussian mixture model Given the importance scores in the previous step, a reasonable approach would be to sort dimensions by importance, and then use the left-out half of the validation set, or "pruning set", to determine the k most important dimensions to include. However, we found that this approach led to high variance, due to the existence of groups of dimensions with very similar importance scores. We hypothesized that these groups of dimension corresponded to different feature types, as observed in Zheng et al. (2019). As McRae et al. (2005) discusses, these features can be grouped into different feature types, such as categorical, functional, encyclopedic, visual-perceptual and non-visual-perceptual. Therefore, the second step in our pruning method creates clusters of dimensions that have similar importance. We fit GMMs with varied number of components k (e.g. $k \in \{1, 2, ..., 6\}$) to the importance scores for each dimension, and find the number of components/modes that show the lowest Bayesian Information Criterion (BIC). Here, we limit the number of possible clusters to 6, as a conservative estimate on the number of distinct feature types (e.g. categorical, functional, perceptual) with possibly differing sparsity ranges–i.e., categorical features may apply to a large subset of items, while specific visual features might apply only to a handful. We cluster dimensions into k modes.

Choosing the subset of dimension clusters that best explains the validation set We find the best non-empty subset of clusters of dimensions, in terms of cross-entropy on the validation "pruning" set, and prune all clusters of dimensions outside of this subset. (If a given feature is uninformative, then features with similar importance scores are likely to be similarly uninformative.)

3 Experiments

3.1 Data

We used two datasets from Zheng et al. (2019), selected after quality control. The first contained judgments on 1,450,119 randomly selected triplets. We used a random subsample of 90% of these triplets for the training set, and the remaining 10% for the validation set (tuning and pruning). The second was an independent test set of 19,968 triplets with 25 repeats for each of 1,000 randomly selected triplets; none of these were present in the training set. Having this many repeats allows us to be confident of the response probability for each triplet. Furthermore, it allows us to establish a model-free estimate of the Bayes accuracy, the best possible accuracy achievable by any model.

3.2 Experimental Setup

Training We implemented both SPoSE and VICE in PyTorch (Paszke et al., 2019) using Adam (Kingma & Ba, 2015) with $\alpha = 0.001$. To guarantee a fair comparison between VICE and SPoSE, each model configuration was trained using 20 different random seeds, for a fixed number of 1000 epochs. Each model was initialized with a weight matrix, $W \in \mathbb{R}^{D \times N}$, where D was set to 100 and N refers to the number of unique items in the dataset (i.e., 1854). In preliminary experiments, we observed that, after pruning, no model was left with a latent space of more than 100 dimensions, which is why we did not consider models with higher initial dimensionality.

Other details Please see section §A.1 for weight initialization and hyperparameter tuning.

3.3 Prediction Experiments

3.3.1 Evaluation measures

Prediction accuracy Since human triplet choices are represented as three-dimensional one-hot-vectors, where 1 represents the odd-one-out choice for a particular triplet, it is simple to compare them with model choices. The choice of a model is computed as $\arg\max p(\hat{y}|\theta)$, where $p(\hat{y}|\theta)$ refers to a model's softmax probability distribution over a triplet given the model parameters (see Equation 9). If there is a tie in the softmax output, we regard this as an incorrect choice. A model can either be correct or incorrect, and no partial credit is given, guaranteeing a conservative measure of a model's prediction behavior. The reported prediction accuracy is the fraction of trials where the model predicted the correct odd-one-out item. We can get an estimated upper bound on the Bayes accuracy, i.e., the best possible accuracy of any model, by using the repeats in the independent test set. As the optimal model predicts the repeat majority outcome for any triplet, this accuracy ceiling – 0.673 – is the average probability of the majority outcome over the set of all triplets.

Predicting Human Uncertainty The triplet task is subjective: there is no correct answer to any given triplet, and often subjects give all three. The independent test set gives us the probability distribution over answers for each triplet, graded information about the relative similarities of the three item pairs. Predicting this distribution precisely is a more stringent test of model quality than prediction accuracy, and of even more relevance in cognitive science applications. We quantify this through the KL divergence between the softmax probabilities of a model (see Section 2.3.3) and the empirical human probability distributions, obtained by computing discrete probability distributions for triplet repeats on the independent test set (see Section 3.1). We use the KL divergence because it is a commonly used measure for assessing the similarity between two probability distributions.

3.3.2 Experiment results

Full dataset We compared pruned median models of VICE and SPoSE, where the median model was identified by the median cross-entropy error on the *tuning* set. For VICE, we set the number of MC samples to $m = 50$ (see Equation 9) . On the independent test set, VICE and SPoSE achieved a similarly high prediction accuracy of 0.6380 and 0.6378, respectively, versus a chance-level accuracy of $0.333(3)$. Likewise, VICE and SPoSE achieved similar KL-divergences of 0.103 and 0.105, respectively, versus a chance-level KL-divergence of 0.366. The differences between the median model predictions across individual triplets in the test set were not statistically significant, under the null hypothesis according to a two-sided paired *t*-test, for either accuracy or KL-divergence. Hence, VICE and SPoSE predicted triplets equally well when they were both trained on the full dataset. This is not surprising, as Bayesian methods based on MC sampling become more like deterministic Maximum Likelihood Estimation (MLE) the more training data is available, per the Bernstein-von Mises theorem (Doob, 1949). As a result, the effects of the prior are most prominent when models are trained on datasets where n_{train} is not particularly large, as we will see in the next section.

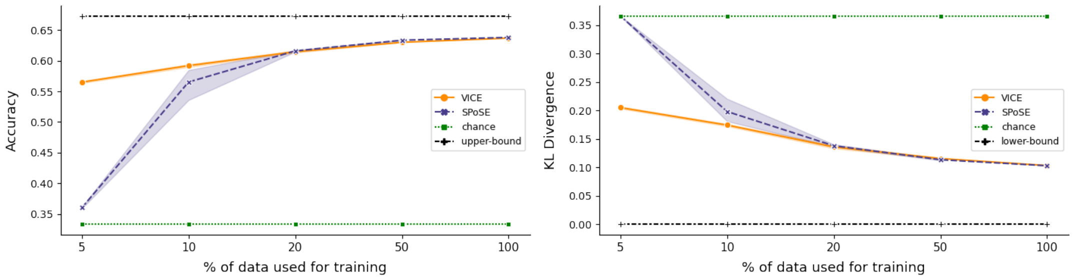


Figure 3: KL divergences to empirical human probability distributions (*left*) and odd-one-out prediction accuracies (*right*) of VICE and SPoSE, each trained on differently sized subsamples of the training data. Error bands depict 95% CIs across the different splits of the data subsets.

Efficiency on smaller datasets Performance on small datasets is especially important in cognitive science, where behavioral experiments often have low sample sizes (e.g. tens to hundreds of volunteer in-lab subjects) or can be costly to scale in AMT. To test whether VICE can model the data better than SPoSE when data are scarce, we created non-overlapping subsets of the training dataset. Specifically, we did this for subsets with sizes equal to 5%, 10%, 20%, and 50% of the dataset, yielding 20, 10, 5, and 2 subsets, respectively. Validation and test sets were unchanged. In Figure 3, we show the average prediction accuracy and KL divergence across random seeds, for models trained on every dataset size, including the full training set. Averages were computed across both random seeds and training subsets, where the average over random seeds was identified first to get a per-subset estimate; performance across subsets was then used to compute the confidence intervals (CIs). Figure 3 shows that the difference in prediction accuracy and KL divergence between VICE and SPoSE became more pronounced the fewer triplet samples were used for training. The difference was striking for the 5% and 10% data subsets, with $\approx 67,500$ and $\approx 135,000$ triplets, respectively. In the former, SPoSE predicted at chance-level; in the latter, it showed a large variation between random seeds and data splits, as can be seen in the 95% CIs in Figure 3. In both low-resource scenarios, VICE showed a compellingly small variation in the two performance metrics across random seeds, and predicted much better than chance-level. The differences between VICE and SPoSE for the 5% and 10% subsample scenarios were statistically significant according to a two-sided paired *t*-test ($p < 0.001$), comparing individual triplet predictions between the pruned median models.

3.4 Reproducibility Experiments

Metric \ Model	VICE	SPoSE
Number of selected dimensions	$47\ (1.64)$	$52\ (2.30)$
$\%$ of dimensions with $r > 0.8$	$79.00\ (6.60)$	$79.59\ (5.60)$

Table 1: Reproducibility of VICE and SPoSE trained on the full dataset across 20 different random initializations. Reported are the means and standard deviations with respect to selected dimensions.

Beyond predictive performance, a key criterion for learning concept representations is *reproducibility*, i.e., learning similar representations when using different random initializations on the same training data. To assess this, we compare 20 differently initialized VICE and SPoSE models.

The first aspect of reproducibility is finding similar numbers of dimensions, quantified as the standard deviation of that number across all 20 models. As shown in Table 1, VICE identified fewer dimensions than SPoSE, and this had a lower standard deviation across models (1.64 vs 2.30) The difference in standard deviation is, however, not statistically significant according to a two-sided F-test ($F = 0.516, df = 19, p = 0.918$). The second aspect is the extent to which the dimensions identified are similar across initializations. Since the embedding is not an *ordered set* of dimensions, we will deem a dimension learned in one VICE model reproducible if it is present in another independently trained instance of VICE, perhaps in a different column or with some small perturbation to the weights. To evaluate the number of highly reproducible dimensions we correspond each embedding dimension of a given initialization (after the pruning step) with the most similar embedding dimension (in terms of Pearson correlation) of a second initialization. Given 20 differently initialized 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 number of dimensions with a Pearson correlation > 0.8 across the 20 initializations. Selected dimensions are similarly reproducible between VICE and SPoSE (see Table 1). Finally, we investigated whether our uncertainty-based pruning procedure selects reproducible dimensions. We compared the average reproducibility of selected dimensions with the average reproducibility of pruned dimensions, which are discarded by the procedure The average reproducibility of the best subset, i.e., % of dimensions with Pearson's $r > 0.8$, is 79.00%, whereas that of the dimensions that were pruned was 0.00%, i.e. our procedure is highly accurate at identifying those dimensions that reproduce reliably.

3.5 Interpretability

One of the benefits of SPoSE is the interpretability of the dimensions of its concept embeddings, induced by sparsity and positivity constraints, and empirically tested through experiments in Hebart et al. (2020). VICE constrains the embeddings to be sparse through the spike-and-slab prior, and imposes a non-negativity constraint through applying a rectifier on its latent representations. This

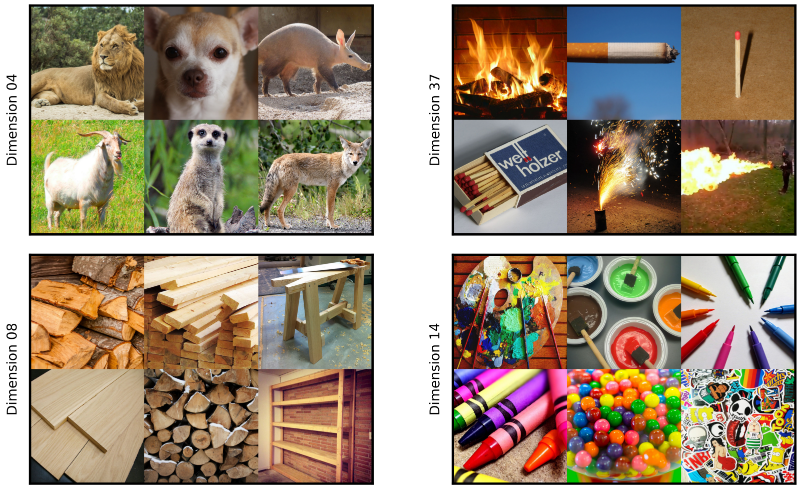


Figure 4: Top six images in four sample VICE dimensions ("animal", *categorical*, "fire; smoke", *functional*, "wood; made of wood", *structural*, and "colorful", *visual*).

means that, just as in SPoSE, it is easy to sort objects within a VICE dimension by their absolute weight values in descending order, to obtain human judgments of what a dimension represents. In Figure 4, we show the top six objects for four example VICE dimensions of the pruned median model, representing *categorical*, *functional*, *structural*, and *visual* information. In Appendix A.2 we show the top ten objects for every VICE dimension. Redoing the SPoSE dimension labeling experiments is beyond the scope of this paper, but we provide dimension labels from a small survey in the Appendix.

4 Discussion

In this paper, we introduced VICE, a novel approach for embedding concepts in a non-negative, sparse space, and using those embeddings to predict human behavior in an odd-one-out task. We solve the same problem as an existing method, SPoSE, but using variational inference and a spike-and-slab prior, which is more appropriate for this modeling situation. VICE yields uncertainty information about the solution, enabling a statistical procedure to automatically determine the number of embedding dimensions, as opposed to the data-dependent heuristics that were used in SPoSE. VICE performs as well as SPoSE in terms of accurately predicting human decisions in an odd-one-out task and modeling the probability distribution over those decisions, but using fewer dimensions. However, this is the case only for the large dataset that was originally used to develop SPoSE. VICE performs substantially better than SPoSE on smaller datasets. Moreover, VICE is more stable than SPoSE, as the dimensionality of the embeddings varies less across random initializations. We believe these improvements stem from the combination of the prior and the dimension selection procedure.

We developed VICE with the goal of making it easier to build interpretable embedding spaces to model any type of item, by using an odd-one-out task. We require fewer participants than SPoSE due to higher data efficiency, which makes behavioral experiments more feasible. Our procedure for determining the number of dimensions aims at removing subjectivity, as the scientific motivation often is to discover the minimum number of latent factors required to describe observations. Note that while the user does need to choose the FDR tolerance q, this can be done before looking at any data, based on their degree of conservatism with regards to controlling false discoveries. Hence, it does not cause the problems associated with data-dependent tuning parameters such as regularization parameters or absolute thresholds. Finally, the spike-and-slab prior we use in VICE has intrinsically meaningful hyperparameters, which makes it easier for researchers to specify competing hypotheses about the representation space being studied.

REFERENCES

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.

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.x.

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.

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.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1613–1622, Lille, France, 07–09 Jul 2015. PMLR.

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.

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.

Steven Derby, Paul Miller, and Barry Devereux. Feature2vec: Distributional semantic modelling of human property knowledge. *arXiv preprint arXiv:1908.11439*, 2019.

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.

Joseph L Doob. Application of the theory of martingales. *Le calcul des probabilites et ses applications*, pp. 23–27, 1949.

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.

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.

Zoubin Ghahramani. Scaling the indian buffet process via submodular maximization. In *International Conference on Machine Learning*, pp. 1013–1021. PMLR, 2013.

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.

Thomas L Griffiths and Zoubin Ghahramani. The indian buffet process: An introduction and review. *Journal of Machine Learning Research*, 12(4), 2011.

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.

Martin N. Hebart, Charles Y. Zheng, Francisco Pereira, and Chris I. Baker. Revealing the multidimensional 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.

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.

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.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

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.

Patrick McClure and Nikolaus Kriegeskorte. Robustly representing uncertainty in deep neural networks through sampling. *CoRR*, abs/1611.01639, 2016.

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.

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.

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

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.

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.

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.

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.

Herbert Robbins and Sutton Monro. A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, 22(3):400 – 407, 1951. doi: 10.1214/aoms/1177729586. URL `https://doi.org/10.1214/aoms/1177729586`.

Samuel Smith, Erich Elsen, and Soham De. On the generalization benefit of noise in stochastic gradient descent. In *International Conference on Machine Learning*, pp. 9058–9067. PMLR, 2020.

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.

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.

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

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.

A Appendix

A.1 Experimental Setup

Weight initialization We initialized the weights of the encoder for the means of the distributions, W_μ, following a Kaiming He initialization (He et al., 2015). The weights of the encoder for the logarithm of the scales of the distributions, $W_{\log(\sigma)}$, were initialized with $\epsilon = -\frac{1}{s_{W_\mu^0}}$, such that $W^0_{\log(\sigma)} = \epsilon \mathbf{1}$. This initialization allowed us to avoid bias terms within the linear transformations of the encoders, and additionally ensured, through computing $\sigma = \exp(\log(\sigma))$, that σ is a small continuous number in $\mathbb{R}^+$ at the beginning of training.

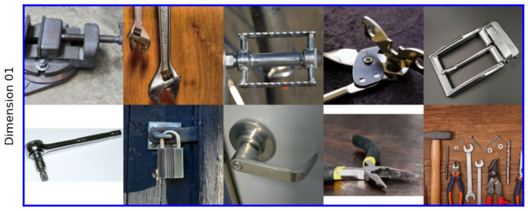
Hyperparameter grid To find the optimal VICE hyperparameter combination, we performed a grid search over $\pi, \sigma_{spike}, \sigma_{slab}$ (see Equation 8). The final grid was the Cartesian product of these parameter sets: $\pi = \{0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$, $\sigma_{spike} = \{0.125, 0.25, 0.5, 1.0, 2.0\}$, $\sigma_{slab} = \{0.25, 0.5, 1.0, 2.0, 4.0, 8.0\}$, subject to the constraint $\sigma_{spike} << \sigma_{slab}$, where combinations that did not satisfy the constraint were discarded. We observed that setting $\sigma_{slab} > 8.0$ led to numerical overflow issues during optimization, which is why 2^3 was the upper-bound for σ_{slab}. For SPoSE, we use the same range as Zheng et al. (2019), with a finer grid of 64 values.

Optimal hyperparameters We found the optimal VICE hyperparameter combination through a two step procedure, for which the validation set was split into equally sized *pruning* and *tuning* sets. First, among the final 180 combinations (see Cartesian product above), we applied our pruning method (see Section 2.3.4) to each model and kept the subsets of dimensions that led to the lowest cross-entropy error on one half of the validation set, which we call *pruning* set. Second, we evaluated each model with pruned parameters on the other half of the validation set, which we refer to as *tuning* set. We defined the optimal hyperparameter combination as that with the lowest average cross-entropy error on the *tuning* set across twenty different random initializations. The optimal hyperparameter combinations for VICE were $\sigma_{spike} = 0.125, \sigma_{slab} = 1.0, \pi = 0.4$; for SPoSE, it was $\lambda = 5.75$.

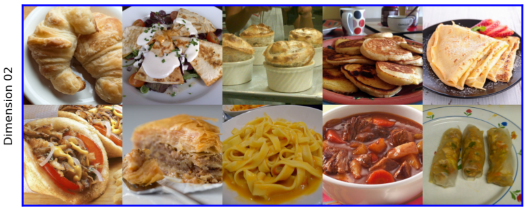
A.2 Object dimensions

Here, we display the top 10 objects for each of the 47 VICE dimensions, according to their absolute weight value. As we have done for every other experiment, we used the *pruned* median model to guarantee the extraction of a representative sample of object dimensions without being over-optimistic with respect to their interpretability (see Section 3.3.1 for how the median model was identified). For each dimension we collected human responses in a small survey with a sample of convenience ($n = 9$). The labels that are shown below each object dimension represent the most common answer across human responses, when they were asked to name the respective dimension. More than one label is displayed, whenever there was a tie in the most common response. Labels were edited for coherence across similar answers (e.g. "metallic" and "made of metal" were deemed to be the same answer).

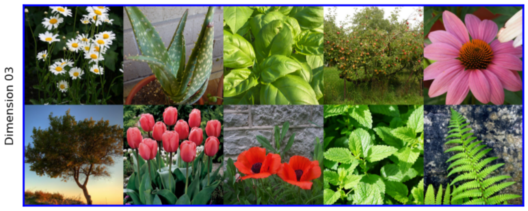
While the illustrations for each dimension display the top 10 items, for our survey, in order to avoid biasing our results, we actually show a continuum of items selected from bins centered around the 25, 50, and 75 percentiles in addition to top items, and a random set of items with close to zero weight in the dimension.



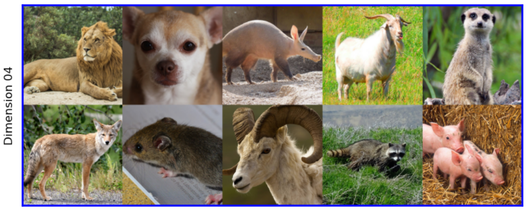
Metallic



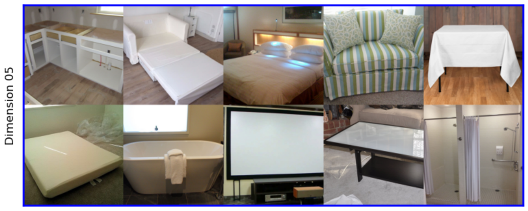
Food



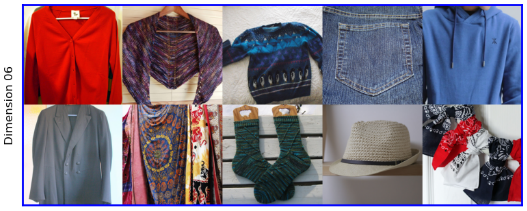
Plants



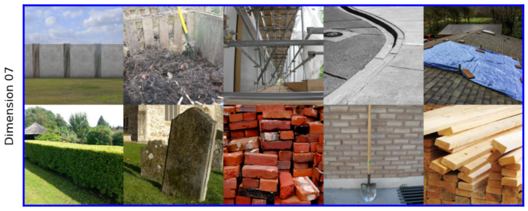
Animal



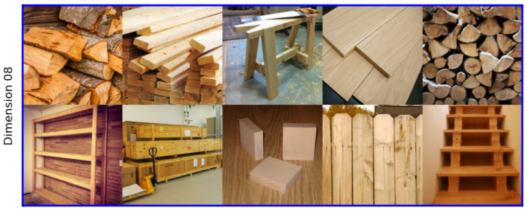
Home



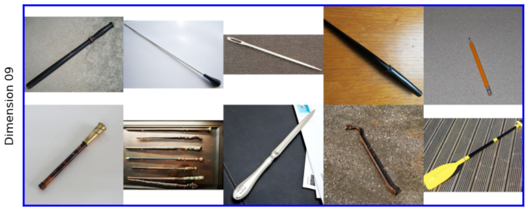
Clothes



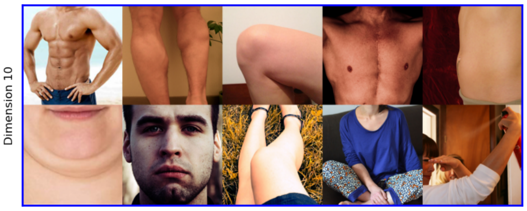
Outdoor



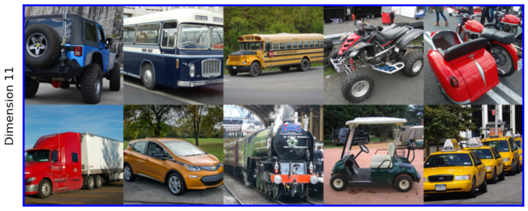
Wood; Made of wood



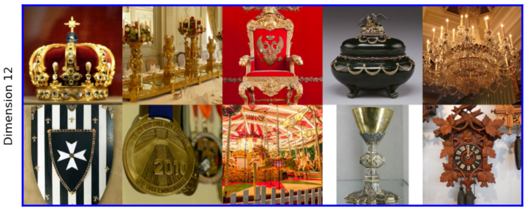
Pointy; Elongated



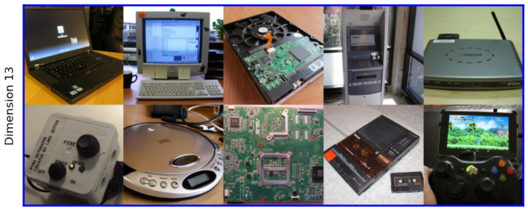
Body parts



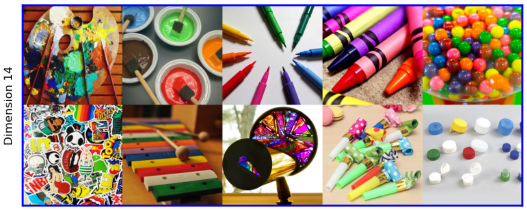
Vehicle; Transportation



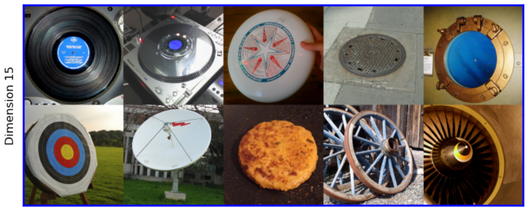
Exquisite; Traditional



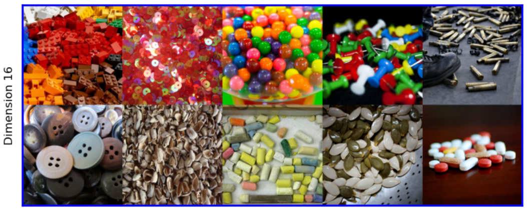
Electronic



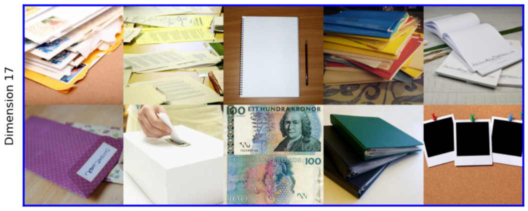
Colorful



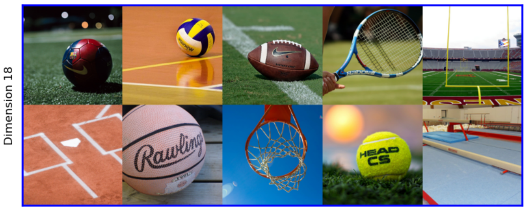
Round; Circular



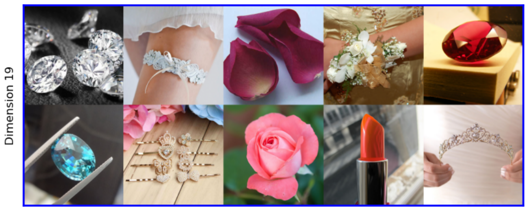
Many objects; Collection



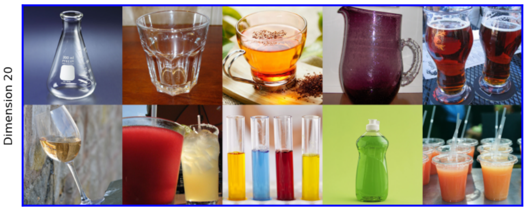
Stationery; Office



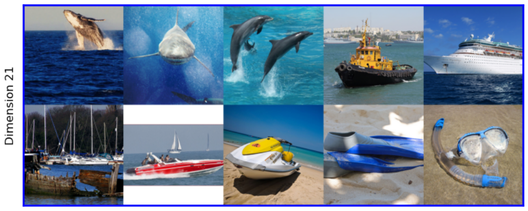
Sports; Games



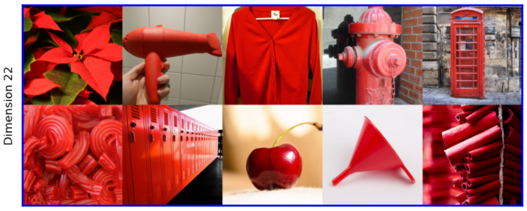
Decorative; Beautiful



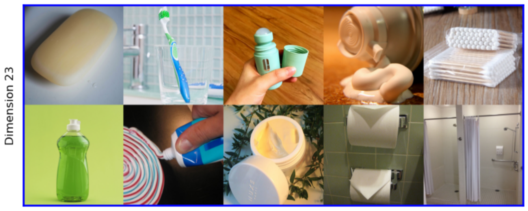
Container; Drinks



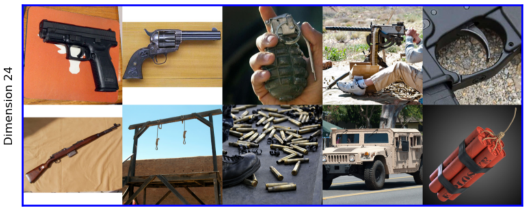
Marine; Water



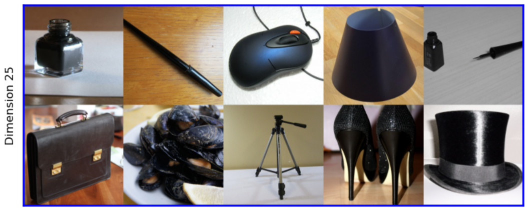
Red



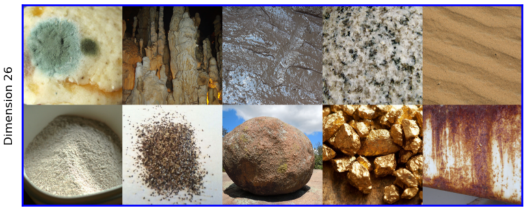
Bathroom; Hygiene



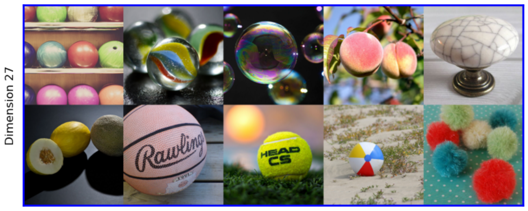
War; Weapon



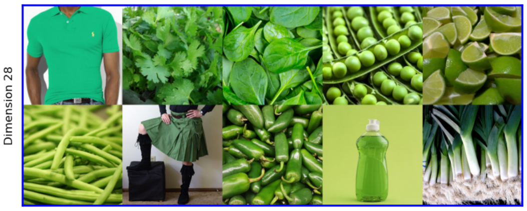
Black



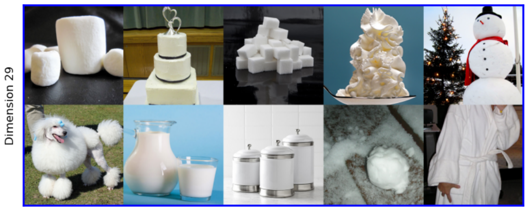
Dust; Grainy texture



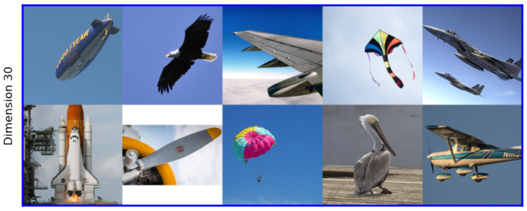
Spherical; Round



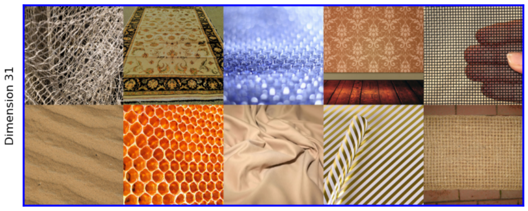
Green



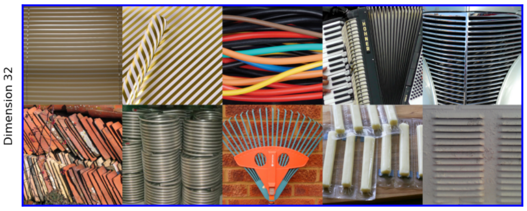
White



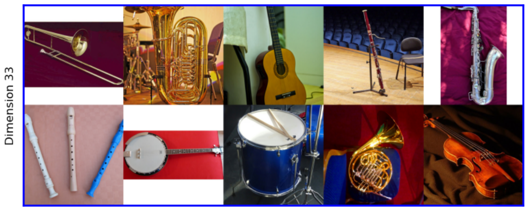
Sky; Flying



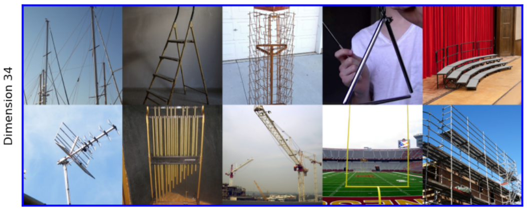
Floor; Pattern



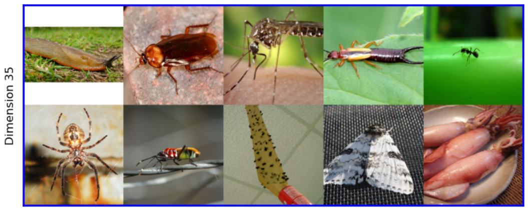
Lines; Grating Pattern



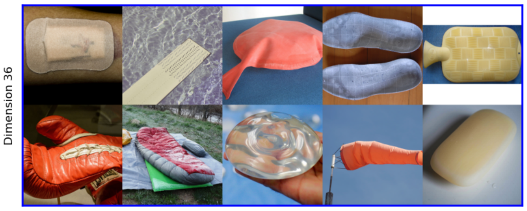
Music; Sound



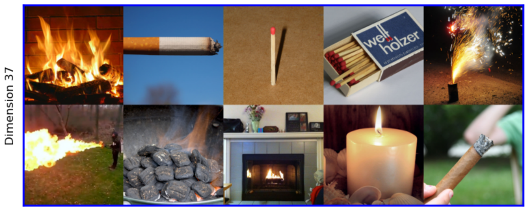
Sky; Tall



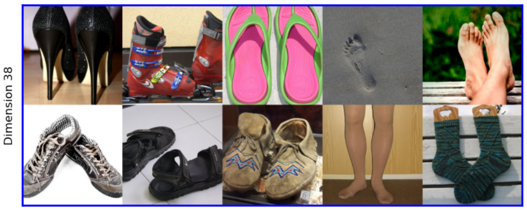
Insects; Pests



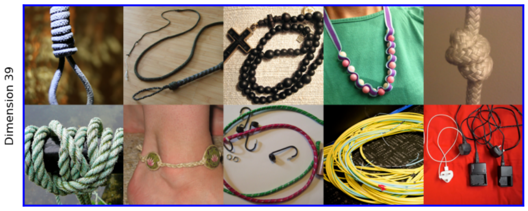
N/A



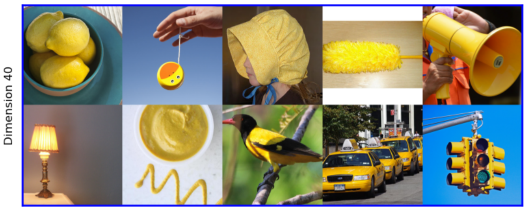
Fire; Smoke



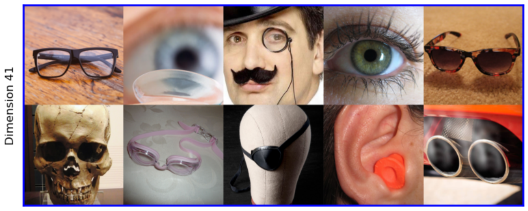
Foot; Footwear



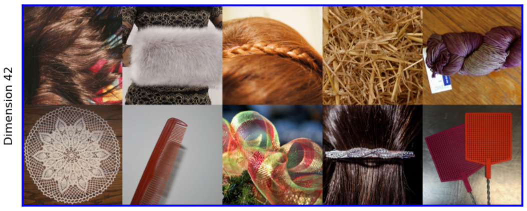
Chain; Rope; Strand



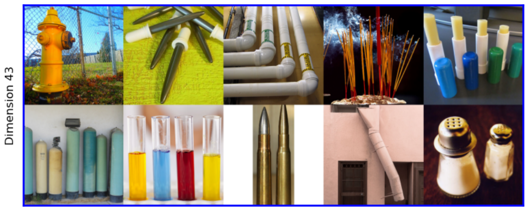
Yellow; Orange



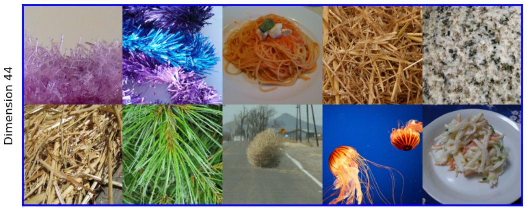
Eyewear; Eyes; Face



Spiky; Hairy



Cylindrical; Elongated



Stringy; Fibrous

Baby; Children

Medical; Healthcare

Ice; Cold