A LRM-NPEFF DECOMPOSITION ALGORITHM

Let $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be a set of examples. Let us represent the set of LRM-PEFs via the rank-3 tensor $A \in \mathbb{R}^{n \times c \times m}$, where n is the number of examples, c is the number of classes, and m is the number of parameters. The Fisher information matrix for the i-th example can be expressed as $A_i^T A_i$, where $A_i \in \mathbb{R}^{c \times m}$. While our implementation can handle using a different rank for each PEF (i.e. the number of rows of A_i varying with i), we assume a constant rank here for ease of presentation. The decomposition equation 4 becomes learning matrices $W \in \mathbb{R}^{n \times r}$ and $G \in \mathbb{R}^{r \times m}$ such that

$$\begin{array}{ll} \text{minimize} & \sum_{i=1}^{n} \|F_i - \sum_{j=1}^{r} W_{ij} \mathbf{g}_j \mathbf{g}_j^T\|_F^2 \\ \text{subject to} & W_{ij} \geq 0, \end{array} \tag{6}$$

where $\mathbf{g}_j$ denotes the j-th row of G.

To solve equation 6, we take a coordinate descent approach reminiscent of the multiplicative update algorithm for NMF where we alternate between updating the W and G matrices (Lee & Seung, 1999). Our W-update step is essentially a the W-update step from the multiplicative update NMF algorithm. For the G-update step, we perform a gradient descent step with a fixed learning rate. It is possible to perform both steps without having to explicitly construct and $m \times m$-matrices and instead only perform inner products between m-dimensional vectors. The number of such inner products used in the algorithm is independent of m. We go over how to efficiently perform these steps and split the work amongst multiple GPUs in the following.

A.1 W-UPDATE STEP

Recall that the multiplicative update step in NMF involves computing non-negative numerator and denominator matrices $N, D \in R^{n \times r}$. The matrix W is then updated via the element-wise rule $W_{ij} \mapsto W_{ij} N_{ij} / D_{ij}$.

Computing the numerator starts with computing the rank-3 tensor $B \in \mathbb{R}^{n \times c \times r}$ with elements given by $B_{ijk} = \sum_{\ell=1}^{m} A_{ij\ell} G_{k\ell}$. The numerator is then given element-wise by $N_{ik} = \sum_{j=1}^{c} B_{ijk}^2$. The denominator is then given by $D = W((GG^T) \odot (GG^T))$, where $\odot$ denotes the Hadamard product.

A.2 G-UPDATE STEP

The gradient of the loss with respect to G consists of two terms $T_1, T_2 \in \mathbb{R}^{r \times m}$ that are added together. The first term is given by $T_1 = 4((W^T W) \odot (GG^T))G$. Computation of the second term starts by computing the rank-3 tensor $B \in \mathbb{R}^{n \times c \times r}$ as was done for the W-update step. The second term is then obtained element-wise as $[T_2]_{i\ell} = -4 \sum_{j=1}^{n} \sum_{k=1}^{c} W_{ji} \beta_{jki} A_{jk\ell}$.

A.3 MULTI-GPU IMPLEMENTATION DETAILS

Our strategy for distributing work across multiple GPUs is similar to that of Boureima et al. (2022). We partition the last axis of the A tensor and columns of G across the GPUs. The W matrix is replicated across all GPUs. Two all-reduces are needed per step: when computing B and when computing GG^T. If doing a G-update step immediately after a W-update step, we can cache copies of these matrices during the W-update step and use them for the G-update step.

A.4 OTHER CONSIDERATIONS

We initialized W using the uniform distribution on $[0, 1]$. We initialized G using a normal distribution with zero mean and standard deviation of $\sqrt{2}/\sqrt{rm}$. Since the PEFs were normalized to roughly unit L2 norm, we chose this scaling so that the initial reconstructions would also have roughly unit L2 norms as well.

After initialization, we found it crucial to freeze W and only train G for a bit before commencing joint training. This is because if the G is a poor fit for the W, the W update step will end up setting W to zero. Since the W update is multiplicative, it remains zero throughout the remainder of training if this happens. We suspect that this behavior can be explained due to the nature of the multiplicative update step. It can be shown that the multiplicative update step is equivalent to gradient descent with

a variable element-wise learning rate (Burred, 2014). Unlike traditional gradient descent that uses a small gradient step, the variable learning can become large. This makes it possible for the W to jump directly to zero or some similarly small value. If the loss is greater than the Frobenius norm of the PEFs, then setting W to zero will result in a lower loss. Hence, jumping to zero can decrease the loss in such cases.

A.5 Convergence

While we do not provide a proof of convergence of our LRM-NPEFF decomposition algorithm, we can make a heuristic argument for its convergence. Following the proof of convergence for regular multiplicative-update NMF (Lee & Seung, 1999), we can show that the loss will be non-increasing following the W-update step. For a sufficiently small step size, we can expect the gradient descent step from the G-update to not increase the loss as well. Since the loss is bounded from below by 0, it follows that the loss should eventually converge. When actually running LRM-NPEFF, we found the loss to be non-increasing with the rate of decrease decelerating as the number of steps increased.

A.6 Run Time and Scaling Information

Typically, our LRM-NPEFF decomposition took between 1 to 8 hours depending on the problem size and number of GPUs used. We used a server with 4x A6000 GPUs; however, we often did not use all of the GPUs. Using the same set up as the NLI model in section 3.1, we report times per joint update step (i.e. a W-update followed by a G update) here. For a 512 component decomposition, we had a step time of 29156 ms on 2 GPUs and 15022 ms on 4 GPUs. For a 128 component decomposition, we had a step time of 14388 ms on 1 GPU and 6975 ms on 2 GPUs. For a 32 component decomposition, we had a step time of 3190 ms on 1 GPU and 1530 ms on 2 GPUs. Thus our implementation obtains a near linear speed up with increasing the number of GPUs.

B Computation of LRM-PEF Frobenius Norm

Let us represent the PEF $F \in \mathbb{R}^{m \times m}$ of an example as an LRM-PEF $A^T A$, where $A \in \mathbb{R}^{c \times m}$. Using properties of the Frobenius norm, we see that

$$\|F\|_F = \|A^T A\|_F = \|A A^T\|_F. \tag{7}$$

Since AA^T only has c^2 elements, its Frobenius norm can easily be evaluated.

C D-NPEFF Perturbation Method

We make use of a method based on the Fisher-weighted parameter averaging (FWPA) introduced by Matena & Raffel (2021) to construct a perturbation to selectively disrupt the processing represented by a D-NPEFF component. Let $\theta \in \mathbb{R}^m$ denote the parameters of the original model. Let $\mathbf{f} \in \mathbb{R}^m$ denote the diagonal of the Fisher information matrix of the original model over the entire data set. This is simply the expectation of the diagonal PEFs with respect to the data distribution. Let $\mathbf{h} \in \mathbb{R}^m$ denote the pseudo-Fisher of the component we wish to perturb. Our FWPA-based perturbation method takes in the following hyperparameters: perturbation magnitude $\delta > 0$, merging coefficient $\lambda \in [0, 1]$, and sign-pattern $\mathbf{s} \in \{-1, 1\}^m$ (discussed in the following paragraphs). The parameters $\phi \in \mathbb{R}^m$ of the perturbed model are provided element-wise by

$$\phi_i = \frac{(1-\lambda) f_i \theta_i + \lambda h_i (\theta_i + s_i \delta)}{(1-\lambda) f_i + \lambda h_i}, \tag{8}$$

where we default to having $\phi_i = \theta_i$ when both f_i, h_i are approximately zero. This can be interpreted as the Fisher-weighted merge of the original model with a corrupted version where each parameter has been shifted by a magnitude of δ. Intuitively, we expect the perturbed parameters to be closer to their original values when they are more important to the original model's behavior and farther away when more important for the given component. This has the effect of selectively altering the model's predictions for examples for which it uses the component's corresponding sub-computation.

Sign Pattern The need for the sign pattern hyperparameter arises from the fact that the expression predicting the KL-divergence between perturbed and original predictions depends only on the element-wise square of the perturbation. Different choices of the sign pattern will result in different distributions over classes that all should have the same KL-divergence with the original predictive distribution. The invariance of the KL-divergences to the choice of sign pattern, however, can break down when we consider the finite perturbations used in practice instead of the infinitesimal perturbations of the theory.

We use a heuristic method for choosing the sign pattern. First, we assume that we have some set of examples $\mathcal{D}_{\mathrm{p}} = \{\mathbf{x}_1, \ldots, \mathbf{x}_\ell\}$ whose predictions we wish to selectively perturb. This will typically be the top examples for a component. For each parameter, we want to move in the direction that increases the KL-divergence of the model's predictions on these examples the most. Hence, we use a sign vector given element-wise by

$$s_i = \operatorname{sign} \frac{\partial}{\partial \theta_i} \sum_{j=1}^{\ell} D_{\mathrm{KL}} \left(\mathtt{sg} \left[p_\theta(y|\mathbf{x}_j) \right] \| p_\theta(y|\mathbf{x}_j) \right), \tag{9}$$

where the $\mathtt{sg}$ is the stop-gradient operator.

Other Hyperparameters Once we have chosen a sign pattern, we then find hyperparameters δ, λ such that the average KL-divergence for the chosen examples $\mathcal{D}_{\mathrm{p}}$ fell in a predetermined range. We accomplished this via a randomized search heuristic.

Recall that our goal is to find values $\delta > 0$ and $\lambda \in [0, 1]$ such that the average KL-divergence of the perturbed model's predictions belongs to some range $[\ell_1, \ell_2]$. Our heuristic is based on the assumption that the average KL-divergence increases with increasing δ and λ. This corresponds to the perturbed parameters becoming increasing dissimilar to the originals. We also assume that the user has specified some maximum value for hyperparameter δ, which we denote D. Hence $\delta \in (0, D]$.

We start by selecting initial values of δ and λ at random from their respective ranges. We evaluate the average KL-divergence for the perturbed model with these hyperparameter values. We pick one of δ or λ to change, alternating between iterations of the heuristic. If the KL-divergence is too high, we pick a value halfway between the current value of the hyperparameter and its minimum value. We evaluate using the new hyperparameters. If the KL-divergence is too low, we try again using a value a quarter of the way to the original value of the hyperparameter and its minimum value. We repeat this until we get a KL-divergence value either within the specified range or higher than the range. If it is in the range, we stop. Otherwise, we keep that value of the hyperparameter and repeat with the other hyperparameter. We perform an analogous algorithm when the KL-divergence is too high.

D Experimental Details

D.1 QQP

We fine-tuned the `bert-base-uncased` checkpoint from the Hugging Face repository (Wolf et al., 2019) on a data set derived from the train split of QQP. This data set was simply the train split with 50k examples held out. We trained the model using Adam (Kingma & Ba, 2014) with a learning rate of 1e-5 and batch size of 32 for 40k steps.

The set of 50k examples held out from the train set were used to compute the LRM-PEFs used to learn the LRM-NPEFF decomposition. When creating the sparse approximations, we kept the 65,536 entries with the largest magnitudes for each LRM-PEF. We used the same sparsity when computing LRM-PEFs on the validation set.

We performed LRM-NPEFF on the PEFs from the held out train set examples with 256 components. We pruned entries corresponding to parameters with fewer than 8 non-zero entries across all PEFs. After initialization, we trained only G for 100 steps before performing alternating updates between W and G. The latter stage updated each factor 1500 times. We used a learning rate of 3e-5 for the G-only stage of training and a learning rate of 3e-4 for the joint stage.

D.2 NLI

We used the `connectivity/feather_berts_0` checkpoint from the Hugging Face repository as the model for our NLI experiments. This checkpoint was released as part of McCoy et al. (2019). It was fine-tuned on MNLI from the `bert-base-uncased` pretrained model.

We used two disjoint sets of 50k examples each from the SNLI train split to compute LRM-PEFS. When creating the sparse approximations to these PEFs, we kept the 65,536 entries with the largest magnitudes for each LRM-PEF.

We performed LRM-NPEFF on the PEFs from one of these sets using 512 components. We pruned entries corresponding to parameters with fewer than 14 non-zero entries across all PEFs. After initialization, we trained only G for 100 steps before performing alternating updates between W and G. The latter stage updated each factor 1500 times. We used a learning rate of 1e-4 for the G-only stage of training and a learning rate of 3e-4 for the joint stage.

D.3 VISION

We used a ResNet-50 (He et al., 2016) trained on the ImageNet classification task (Russakovsky et al., 2015), namely the `imagenet` pretrained weights of the `ResNet50` class in TensorFlow (Abadi et al., 2016).

We computed D-PEFs using 20k examples from the ImageNet train split and 30k examples from the ImageNet validation split. We only included terms with a probability of greater than 3e-3 when performing the expectation over classes when computing the D-PEFs. When creating the sparse approximations, we kept the 65,536 entries with the largest magnitudes for each D-PEF.

We performed D-NPEFF on the PEFs from the train split using 512 components. We pruned entries corresponding to parameters with fewer than 6 non-zero entries across all PEFs. We ran NMF for 2500 steps on these D-PEFs to create the D-NPEFF decomposition.

E VISION MODEL D-NPEFF PERTURBATION DETAILS

We selected the top 128 examples for each component to compute the sign pattern. We selected the δ, λ hyperparameters such that their average KL-divergence of the top examples was between 0.25 and 0.35. The δ, λ selection process was repeated 6 times per component. For each run, we computed the ratio of the average KL-divergence for the top 128 examples of the component to the average KL-divergence across the entire set of 30k validation set examples for which we computed PEFs. We used the geometric mean of the KL-divergence ratios across these runs to get an average KL-divergence for each component.

F COMPONENTS SET EXPANSION EXPERIMENT DETAILS

Given a set of PEFs used to compute an NPEFF decomposition, we created another set of PEFs consisting of only the examples on which the model made an incorrect prediction. Expanding the set of NPEFF components on these examples is similar to computing an NPEFF decomposition, but some parameters are initialized non-randomly and some parameters are not updated during the learning process.

Divide the set of components to a group consisting of the original components and another group consisting of the new components that will be learned. The pseudo-Fishers of the former group are initialized using their values from from original NPEFF decomposition and not updated during training. Their corresponding coefficients are initialized using their values from the original decomposition as well. We found that initializing these coefficients this way significantly increased the chance of the expansion learning meaningful components. The coefficients of the expanded components are initialized using the uniform distribution on $[0, 1]$. The rows of G corresponding to the expanded components are initialized using a normal distribution with zero mean and standard deviation of $\sqrt{2}/\sqrt{rm}$, where r is the sum of the number of original components and the number of components in the expansion.

Computation of the expansion proceeds in two stages with an optional third stage at the end. First, all of the coefficients are frozen while the rows of G corresponding to the new components are updated. Then perform alternate updates on the columns of W corresponding to the new components and their corresponding rows of G. The optional final stage involves performing alternating updates on all of the columns of W and updating only the rows of G corresponding to the new components.

For our experiment on the NLI model, we used a learning rate of 1e-3 during the first stage where we were only updating the columns of G corresponding to the new components. We performed a total of 250 updates during this state. In the other stages, we used a learning rate of 3e-3. We ran the second stage for 1000 updates of both NPEFF factors. We did not run the third stage.

G Quality of Sparse Approximation

When producing a sparse approximation to an LRM-PEF of the NLI model, we kept only 65,536 entries out of a total of about 330 million. Hence only around 0.02% of the sparse LRM-PEF's entries are non-zero. To determine the quality of these sparse representations, we computed the Frobenius distance between a dense PEF matrix and its sparse approximation over a set of 500 examples. These distances were then normalized by dividing them by the Frobenius norm of their corresponding dense PEF matrix. We then subtract the resultant value from 1 to get a score between 0 and 1 for each example. A score of 0 indicates that the sparse approximation captures no information about the dense PEF while a score of 1 indicates a perfect match. A histogram of these scores across examples can be found at fig. 4. When keeping 65,536 entries, there is a large peak at round 0.15 in the distribution. Higher scores become less likely with only a few greater than 0.4. Increasing the number of kept entries to 262,144 shifts the peak of the score distribution to between 0.2 and 0.25. These overall results indicate that although our sparse approximations are not particular close approximations, they capture a significant amount of information given their sparsity.

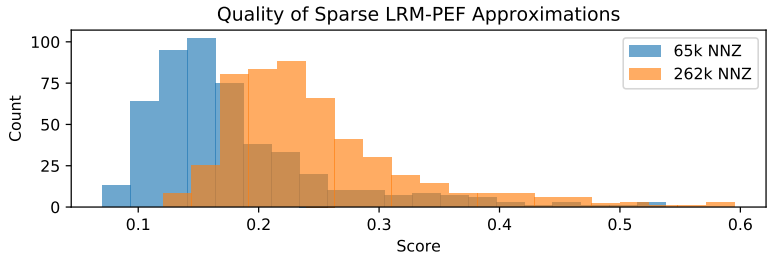


Figure 4: Histogram of per-example values indicating the closeness of the sparse approximation to SNLI LRM-PEFs. A score of 0 indicates that the approximation is identically zero while a score of 1 indicates a perfect match.

H Synthetic Task and Model Details

H.1 Task Details

The synthetic task can be thought of as an extremely simplified NLI task. An example looks like `[BOS] Q1 S1 O1 Q2 S2 O2`, which is a concatenation of a premise and hypothesis. The `Q1,Q2` tokens correspond to `All` or `Some`, and a `Q S O` segment can be thought of as the sentence `[All/Some] [subject] [verb] [object]`. The verb is assumed fixed and shared between the premise and hypothesis, so it is not explicitly represented. Subjects and objects are chosen from the same set of options containing a hierarchical structure so that we can have a `is_a_subtype_of` relation between options. Each example is given a label of `entails` or `neutral`. A couple of examples in "readable" form are:

- `All cows eat grass. Some bovines eat plants.` $\Longrightarrow$ `entails`
- `Some seals eat fish. All mammals eat animals.` $\Longrightarrow$ `neutral`

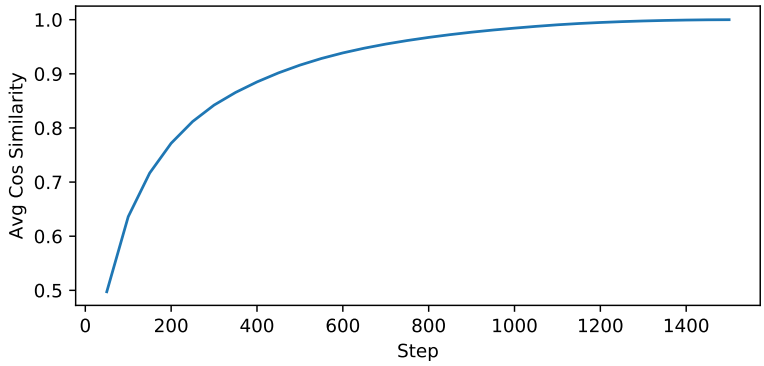


Figure 5: Average cosine similarities of component coefficients with their final values as the NPEFF decomposition progresses.

The set of options for the subjects and objects was obtained from subtree of ImageNet classes rooted at the node with id `n02913152`. This node corresponded to the class "building, edifice". We did this purely to obtain a "natural-looking" hierarchy of items; the exact names of the items in their original context are not used by this task. Overall, this produced a set of 23 options.

H.2 RASP Program

RASP pseudo-code for solving the synthetic is presented in appendix H.2. The output at the last sequence position is the predicted label for the example. It solves the task with 100% accuracy.

H.3 Programmatic Detection of Tunings

Recall that each example will look like `[BOS] Q1 S1 O1 Q2 S2 O2`. For each example, we can create a set of boolean annotation depending on whether its tokens satisfy certain properties. These properties reflect the values of intermediate variables in the RASP program appendix H.2 that is implemented by the model. We then say a component is tuned to a particular property if its top 128 examples all possess that property. A breakdown of the number of components in each decomposition tuned for various properties can be found in table 1.

I TCAV

TCAV provides a means to test whether a model possesses conceptual sensitivity to a concept represented by a group of examples (Kim et al., 2018). Given activations corresponding the conceptual group and activations correspondingly to examples not from the group, TCAV learns a binary linear classifier to distinguish between the two groups. This classifier can be expressed as a vector with the same dimension as the activations that is the normal vector of the oriented hyperplane forming the decision boundary of the classifier. This is called the group's concept activation vector (CAV). Then derivative of the log probability[1] for each class with respect to the activations in the direction of the CAV is computed over a set of examples. The fraction of examples with a positive directional derivative is recorded for each class to create a score. This process can be repeated multiple times using a different set of random examples to represent the examples not containing the concept. A two sided t-test can then be performed for these scores to compute a p-value for the null hypothesis of a TCAV score of 0.5.

For the TCAV experiments in this paper, we used the top 32 component/cluster examples to represent the conceptual groups. We used a random set of 128 examples to form the baseline group. We computed the TCAV for each run using a set of 5k examples. We performed 500 runs for each component/cluster.

[1]Note that Kim et al. (2018) uses logits while we use log probabilities.

```
def is_subtype_of(sop_a, sop_b):
    token_equals_b = {t: t == sop_b for t in all_token_ids}
    return sop_a == sop_b | reduce_or(
        t == sop_a & reduce_or(t == token_equals_b[s] for s in supertypes
    (t))
        for t in item_tokens
    )

def label_example(tokens):
    # Align the hypothesis and premise.
    hypo_aligned = tokens
    prem_aligned = shift_by(SENTENCE_LEN, tokens)

    is_prem_subtype_of_hypo = is_subtype_of(prem_aligned, hypo_aligned)
    is_hypo_subtype_of_prem = is_subtype_of(hypo_aligned, prem_aligned)

    # Use the last sequence position for a computation space.
    is_s1_subtype_of_s2 = shift_by(1, is_prem_subtype_of_hypo)
    is_s2_subtype_of_s1 = shift_by(1, is_hypo_subtype_of_prem)
    is_o1_subtype_of_o2 = is_prem_subtype_of_hypo

    is_all = tokens == ALL_TOKEN_ID
    is_some = tokens == SOME_TOKEN_ID
    is_all_q1 = shift_by(2 * SENTENCE_LEN - Q_PREM_INDEX - 1, is_all)
    is_some_q1 = shift_by(2 * SENTENCE_LEN - Q_PREM_INDEX - 1, is_some)
    is_all_q2 = shift_by(2 * SENTENCE_LEN - Q_HYPO_INDEX - 1, is_all)
    is_some_q2 = shift_by(2 * SENTENCE_LEN - Q_HYPO_INDEX - 1, is_some)

    # The output at the last sequence position will be the label for the
    example.
    label = reduce_or(
        ((is_all_q1 & is_all_q2) & (is_s2_subset_of_s1 &
    is_o1_subset_of_o2)),
        ((is_some_q1 & is_some_q2) & (is_s1_subset_of_s2 &
    is_o1_subset_of_o2)),
        ((is_all_q1 & is_some_q2) & (is_s2_subset_of_s1 &
    is_o1_subset_of_o2)),
    )
    return label

```

Figure 6: RASP pseudo-code solving the synthetic NLI-like task.

Table 1: Number of components tuned to a particular concept from the TRACR model NPEFF runs. The concepts ending with `_item` mean that the top 128 examples for that component satisfy the corresponding relation for some fixed item. The specific item can differ from component to component. Note that many components were tuned to combinations of the more elementary concepts that we searched for.

Property	32 Comps Run	128 Comps Run
o1_equals_item	20	30
o1_equals_o2	0	8
o1_strict_subtype_of_o2	0	4
o1_subtype_of_item	2	1
o1_subtype_of_o2	1	0
o1_supertype_of_item	0	1
o2_equals_item	5	34
o2_subtype_of_item	0	2
o2_supertype_of_item	0	2
q1_is_all	0	7
q1_is_all_and_q2_is_all	1	3
q1_is_all_and_q2_is_some	1	8
q1_is_some	0	9
q1_is_some_and_q2_is_all	1	2
q1_is_some_and_q2_is_some	1	2
q2_is_all	0	8
q2_is_some	0	1
s1_equals_s2	0	1

For the LRM-PEF components and k-means clusters of the NLI model, every component had a p-value of less than 1e-23 for at least one class. Even with a Bonferroni correction of 1500, which is the number of runs times the number of classes, every component had a statistically significant TCAV scores using any reasonable threshold.

J Additional Component Tunings

J.1 NLI

Each of the sub-sections here corresponds to a single component. The top 6 examples per component are listed in descending order of component coefficient.

J.1.1 Component 0

```
[LABEL] entailment     [PRED] entailment     [COEFF] 36.0252
[P] a brown dog is running though a river.
[H] an animal running

[LABEL] entailment     [PRED] entailment     [COEFF] 33.5824
[P] the brown dog is laying down on a blue sheet.
[H] an animal is laying down.

[LABEL] entailment     [PRED] entailment     [COEFF] 33.4161
[P] a small brown and black dog playing with a toy
[H] an animal is playing.

[LABEL] entailment     [PRED] entailment     [COEFF] 32.9659
[P] two dogs appear to be kissing with one another on flat ground.
[H] animals are close together.
```

[LABEL] **entailment** [PRED] **entailment** [COEFF] 29.5676
[P] a little brown dog is running in the snow.
[H] an animal in snow.

[LABEL] **entailment** [PRED] **entailment** [COEFF] 28.6355
[P] a brown dog is standing in the water.
[H] an animal in the water

J.1.2 COMPONENT 3

[LABEL] **neutral** [PRED] **neutral** [COEFF] 21.7448
[P] a woman is taking a picture with her camera.
[H] a woman is taking a picture of a house.

[LABEL] **neutral** [PRED] **neutral** [COEFF] 21.3761
[P] a woman is pushing a red stroller down a sidewalk.
[H] a lady is pushing a stroller with three babies.

[LABEL] **neutral** [PRED] **neutral** [COEFF] 20.1738
[P] 2 women wearing brightly colored clothes are sitting next to a dirt road on a rock having a conversation while they're watching the field.
[H] 2 women sitting next to a dirt road, having a conversation about lunch last week.

[LABEL] **neutral** [PRED] **neutral** [COEFF] 19.6292
[P] a man with a large camera is taking photographs.
[H] man taking photographs of his family.

[LABEL] **neutral** [PRED] **neutral** [COEFF] 19.3897
[P] a black dog runs on the beach.
[H] a dog runs after a ball on the beach.

[LABEL] **neutral** [PRED] **neutral** [COEFF] 19.1972
[P] a man is rock climbing under a large cliff.
[H] a man is rock climbing to reach a rare plant.

J.1.3 COMPONENT 8

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] 24.7734
[P] two climbers attempting to climb a steep, snow - covered peak, nearing the top, where two other climbers await them.
[H] the women are in the club singing

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] 21.3290
[P] man lays prostrate on the ground on his back possibly in exhaustion.

[H] three men are playing basketball.

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] 20.5294
[P] three women are standing in a field with two men in wheelchairs and hats.
[H] a group of men and women are in the office.

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] 19.4801
[P] man riding a mountain bike doing a jump in the air.
[H] three women ride scooters in town.

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] 17.6381
[P] group of kids wearing a blue and gray school uniform while playing.
[H] the men are racing cars.

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] 16.9087
[P] a middle - eastern street vendor sells drinks to some young boys.
[H] young boys are playing soccer.

J.1.4 COMPONENT 17

[LABEL] **entailment** [PRED] **entailment** [COEFF] 7.6072
[P] gross couple kissing outdoors.
[H] a couple is kissing.

[LABEL] **entailment** [PRED] **entailment** [COEFF] 7.3462
[P] two people sitting beside a few small boats.
[H] a couple is sitting.

[LABEL] **entailment** [PRED] **entailment** [COEFF] 7.1038
[P] a bricklayer smoothing out concrete.
[H] someone is smoothing concrete.

[LABEL] **entailment** [PRED] **entailment** [COEFF] 6.6635
[P] laborers baling hay in a field.
[H] people are working with hay.

[LABEL] **entailment** [PRED] **entailment** [COEFF] 6.1639
[P] man in white facing body of water.
[H] a man is near water

[LABEL] **entailment** [PRED] **entailment** [COEFF] 5.9790
[P] a red cone on the side of a street.
[H] an object is near an edge.

J.2 VISION

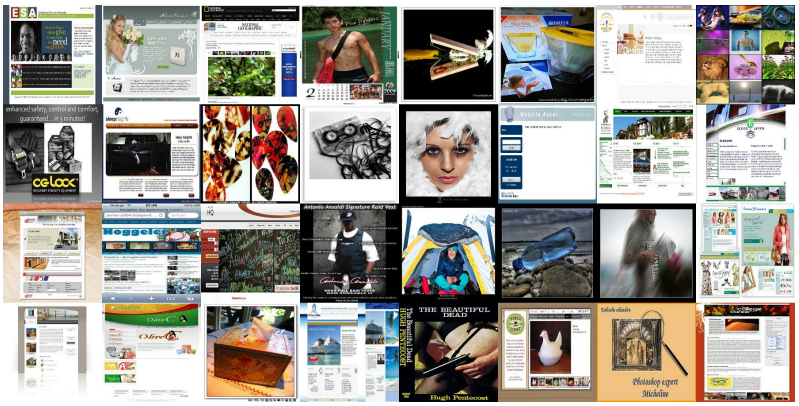
Each of the sub-sections here corresponds to a single component. The top 32 examples per component are listed with component coefficient decreasing in a row-major manner from left-to-right and top-to-bottom.

J.2.1 COMPONENT 29

J.2.2 COMPONENT 35

J.2.3 COMPONENT 171

J.2.4 COMPONENT 276



K INCONSISTENTLY LABELED QQP COMPONENTS

K.0.1 COMPONENT 21

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 82.3713
how do i become a travel writer?
how do you become a travel writer?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 70.3491
how do i become successful in my life?
how can you be successful in your life?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 69.6145
how do i make new friends in a new city?
how do you make friends in a new city?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 61.7586
how do i make my website for free?
how do you make your own website for free?

[LABEL] **duplicate** [PRED] **duplicate** [COEFF] 56.2426
how can i write a blog post on quora?
how do you write a blog on quora?

[LABEL] **duplicate** [PRED] **duplicate** [COEFF] 54.0405
how do i raise my iq?
how can you increase your iq?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 48.8949
what should i do to manage my time?
how do you manage your time?

[LABEL] **duplicate** [PRED] **duplicate** [COEFF] 46.0987
how do i get rid of nightmares?
how can you get rid of nightmares?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 45.2508
how can i get a simple mobile account number?
how do you get a simple mobile account number?

[LABEL] **duplicate** [PRED] **not duplicate** [COEFF] 43.9563
where can i find gold?
where do you find gold?

K.0.2 COMPONENT 31

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 43.9832
what is your review of hindus?
what is your review of hinduism?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 39.8860
how do i convince my parents?
how can i convince my parents?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 32.3885
how do i get job in gulf countries?
how do i get a job in gulf country?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 31.6596
how is friction useful?
how is friction helpful?

[LABEL] **duplicate** [PRED] **duplicate** [COEFF] 29.7447
how do i get into iit bombay?
how to get into iit bombay?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 28.4342
what's the best way to read a technical book?
what is the best way to read technical books?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 27.7442
how do i record whatsapp call?
how do i record a whatsapp video call?

[LABEL] **duplicate** [PRED] **duplicate** [COEFF] 24.3656
how can i found local business directories in australia?
how can i find the local business directories in australia?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 24.0160
how do i get redeem code in google play?
how do i get a redeem code for google play?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 23.8029
what is the process to get a u. s. passport?
what is the process of getting a u. s. passport?

K.0.3 COMPONENT 36

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 178.0185
how should i propose to my girlfriend?
what is the best way to propose to your girlfriend?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 121.0942
how do i switch my it job?
what is the best way of switching my it job?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 109.5911
how do i record my keyboard?
what is the best way to record a keyboard?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 100.9345
how should i kill myself?
what is the easiest way to kill myself?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 98.8751
how do you sell a car?
what is the easiest way to sell a car?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 81.3841
how do i kill spiders?
what is the most effective way to kill a spider?

[LABEL] **duplicate** [PRED] **duplicate** [COEFF] 75.6287
how can i kill myself?
what is the easiest way to kill myself?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 69.2537
how do you make money?
what is the easiest way to make money?

[LABEL] **not duplicate** [PRED] **duplicate** [COEFF] 67.5730
can i build a website on my own?
what is the best way to build your own website?

[LABEL] **duplicate** [PRED] **duplicate** [COEFF] 63.1871
how do i build muscle?
what is the best way to build muscle?

L NLI Components with Faulty Heuristics

L.0.1 Component 18

[LABEL] **neutral** [PRED] **entailment** [COEFF] 16.0689
[P] a young child wearing a yellow striped t - shirt and blue shorts is playing along side the lake and rocks.
[H] a young child is wearing a light yellow striped t - shirt.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 15.6616
[P] young white male child with blond - hair in a red shirt coloring with crayons outside with an adult.
[H] a young white male child with blond - hair has a light red shirt.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 15.4342
[P] a young woman with blond - hair, wearing a short - sleeve gray shirt and blue jean shorts, prepares food for a barbecue.
[H] a young woman has light blond - hair.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 14.5433
[P] a crowd of children in green t - shirts and people holding signs and purple balloons gathers next to a building.
[H] a crowd of children has light green t - shirts.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 12.1538
[P] a female baseball player wearing a blue shirt slides into base, while another player in a white shirt wearing a catcher's mitt jumps.
[H] a female baseball player is wearing a light blue shirt.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 11.7806
[P] in an outdoor location with spectators in the background, a young man in a karate uniform with a blue belt has his arm around the neck of a young woman in a karate uniform with a brown belt while she grips his arm with both hands.
[H] a young man has a karate uniform with a light blue belt.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 8.8071
[P] an older man in a red sweatshirt stands in front of a group of children sitting on benches in front of him.
[H] an older man in a light red sweatshirt stands.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 6.1237
[P] man in yellow behind the wheel of a tractor, hooked to a trailer and under an open sided roof, while others climb on the back.
[H] a man in light yellow is behind the wheel of a tractor.

L.0.2 COMPONENT 34

[LABEL] **neutral** [PRED] **contradiction** [COEFF] 5.5193
[P] a man eating something with a spoon.
[H] a man is eating ice cream.

[LABEL] **neutral** [PRED] **contradiction** [COEFF] 3.9719
[P] a man selling many wicker chairs is pulling a cart of them through the street.
[H] a man is selling blue chairs.

[LABEL] **neutral** [PRED] **contradiction** [COEFF] 3.6611
[P] a man with a green shirt is holding a plant.
[H] a man is delivering flowers.

[LABEL] **neutral** [PRED] **contradiction** [COEFF] 3.6228
[P] a man eating a pink candy bunny.
[H] he eats a gertrude hawk easter bunny.

[LABEL] **neutral** [PRED] **contradiction** [COEFF] 3.5730
[P] two ladies selling their wares in an open market.
[H] two ladies sell their bakes goods at a farmer's market.

[LABEL] **neutral** [PRED] **contradiction** [COEFF] 3.5390
[P] an older gentleman dressed completely in white is eating from a white bowl while sitting in a large overstuffed chair.
[H] an older gentleman is eating cereal.

[LABEL] **neutral** [PRED] **contradiction** [COEFF] 3.5027
[P] a man wearing a red costume stands near others.
[H] the man is wearing a dog costume.

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] 3.4998
[P] a donkey carting greens and two people on a street.
[H] a donkey is carting fruits.

L.0.3 COMPONENT 62

[LABEL] **neutral** [PRED] **entailment** [COEFF] 2.9606
[P] a person on a blue bench under a blue blanket.
[H] a person is resting.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 2.8245
[P] a boy wearing khaki pants and a sports team jersey is jumping down the stairs outside.
[H] a boy is excited.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 2.8070
[P] an athletic man waterskis on a lake.
[H] a fit man is having fun.

[LABEL] **entailment** [PRED] **entailment** [COEFF] 2.7448
[P] 2 girls in metal chairs are laughing together.
[H] two girls are having fun.

[LABEL] **entailment** [PRED] **entailment** [COEFF] 2.3667
[P] man in white shirt and shorts browses an item stand.
[H] a man is browsing for goods.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 2.2851
[P] a gymnast in a competition is looking on with a questioning look on her face.
[H] a gymnast is confused about something.

[LABEL] **entailment** [PRED] **entailment** [COEFF] 2.2077
[P] a little girl with ponytails laughs near a plastic castle play set.
[H] a girl enjoys herself.

[LABEL] **neutral** [PRED] **entailment** [COEFF] 2.1743
[P] a woman standing in a red bikini on a outdoor sand volleyball court.

[H] a woman is playing sports.

M K-MEANS CLUSTERS TOP EXAMPLES

M.1 NLI

M.1.1 COMPONENT 1

[LABEL] **entailment** [PRED] **entailment** [COEFF] -0.0486
[P] Two chinese men looking at papers on a table.
[H] Some people are looking at papers.

[LABEL] **entailment** [PRED] **entailment** [COEFF] -0.0575
[P] Many people riding bikes on a path while another person is walking toward them.
[H] Some people are riding bikes.

[LABEL] **entailment** [PRED] **entailment** [COEFF] -0.0581
[P] A group of young people are in a garage
[H] Some people are in a garage.

[LABEL] **entailment** [PRED] **entailment** [COEFF] -0.0591
[P] A girl in a green shirt and a girl in a yellow shirt standing by water.
[H] Some girls are standing by water.

[LABEL] **entailment** [PRED] **entailment** [COEFF] -0.0614
[P] Three young people planting flowers and covering the area with a tarp.
[H] Some people are planting flowers.

[LABEL] **entailment** [PRED] **entailment** [COEFF] -0.0628
[P] A girl in a blue blouse and a girl in a green shirt sewing.
[H] Some people are sewing.

M.1.2 COMPONENT 4

[LABEL] **neutral** [PRED] **neutral** [COEFF] -0.0800
[P] A person is riding a bike in front of brick buildings.
[H] A person is riding a red bike

[LABEL] **neutral** [PRED] **neutral** [COEFF] -0.0806
[P] A firefighter dressed in gear looking puzzled.
[H] A firefighter dressed in red gear looking puzzled.

[LABEL] **neutral** [PRED] **neutral** [COEFF] -0.0824
[P] A truck - driver is working on his truck.
[H] A truck driver works on his red truck.

[LABEL] **neutral** [PRED] **neutral** [COEFF] -0.0854
[P] A young boy is sitting on a beach filling a water bottle with sand.
[H] A young boy is filling a red water bottle with sand.

[LABEL] **neutral** [PRED] **neutral** [COEFF] -0.0882
[P] Two girls sitting on a chair being sketched.
[H] Two girls sitting on a green chair are being sketched.

[LABEL] **neutral** [PRED] **neutral** [COEFF] -0.0892
[P] Three female dancers are doing dance moves on stage of an auditorium.

[H] Three female dancers, all dressed in blue, are doing dance moves in an auditorium.

M.1.3 COMPONENT 28

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] -0.0535
[P] Two water polo players wrestle for the ball while a goal keeper watches in front of a goal with an angry birds advertisement on the back of the net.
[H] There is no goalkeeper in the waterpolo match.

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] -0.0623
[P] Two people one person dressed in yellow and green with black boots and the other person have on white jacket and brown pants standing on the lake side with birds approaching them
[H] There are no birds at the lake.

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] -0.0668
[P] People wearing glasses and shades are walking, while a man in a black shirt with the word " qualified " on it is facing them.
[H] There are no glasses

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] -0.0671
[P] A family posing with a bride in a white dress at a wedding.
[H] The bride does not have a family.

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] -0.0693
[P] A man plays guitar in the kitchen while a woman adjusts the dials on a dishwasher.
[H] Nobody is using the dishwasher.

[LABEL] **contradiction** [PRED] **contradiction** [COEFF] -0.0700
[P] At a party, an entertainer provides amusement to children in the form of balloon animals.
[H] There are no children at this party.

M.2 VISION

Each of the sub-sections here corresponds to a single cluster. The top 32 examples per cluster are listed with distance to cluster centroid increasing in a row-major manner from left-to-right and top-to-bottom.

M.2.1 COMPONENT 6

M.2.2 COMPONENT 19

M.2.3 COMPONENT 20