Hierarchy-based Image Embeddings for Semantic Image Retrieval

— Supplemental Material —

1. $d_{\mathcal{G}}$ applied to trees is a metric

Theorem 1. *Let $\mathcal{G} = (V, E)$ be a directed acyclic graph whose egdes $E \subseteq V \times V$ define a hyponymy relation between the semantic concepts in V. Furthermore, let $\mathcal{G}$ have exactly one unique root node $\mathrm{root}(\mathcal{G})$ with indegree $\deg^-(\mathrm{root}(\mathcal{G})) = 0$. The lowest common subsumer $\mathrm{lcs}(u, v)$ of two concepts $u, v \in V$ hence always exists. Moreover, let $\mathrm{height}(u)$ denote the maximum length of a path from $u \in V$ to a leaf node, and $\mathcal{C} \subseteq V$ be a set of classes of interest.*

Then, the semantic dissimilarity $d_{\mathcal{G}} : \mathcal{C} \times \mathcal{C} \to \mathbb{R}$ between classes given by

$$d_{\mathcal{G}}(u, v) = \frac{\mathrm{height}(\mathrm{lcs}(u, v))}{\max_{w \in V} \mathrm{height}(w)} \tag{1}$$

is a proper metric if

(a) $\mathcal{G}$ is a tree, i.e.*, all nodes $u \in V \setminus \{\mathrm{root}(\mathcal{G})\}$ have indegree $\deg^-(u) = 1$, and*

(b) all classes of interest are leaf nodes of the hierarchy, i.e.*, all $u \in \mathcal{C}$ have outdegree $\deg^+(u) = 0$.*

Proof. For being a proper metric, $d_{\mathcal{G}}$ must possess the following properties:

(i) Non-negativity: $d_{\mathcal{G}}(u, v) \geq 0$.

(ii) Symmetry: $d_{\mathcal{G}}(u, v) = d_{\mathcal{G}}(v, u)$.

(iii) Identity of indiscernibles: $d_{\mathcal{G}}(u, v) = 0 \Leftrightarrow u = v$.

(iv) Triangle inequality: $d_{\mathcal{G}}(u, w) \leq d_{\mathcal{G}}(u, v) + d_{\mathcal{G}}(v, w)$.

The conditions (i) and (ii) are always satisfied since $\mathrm{height} : V \to \mathbb{R}$ is defined as the length of a path, which cannot be negative, and the lowest common subsumer (LCS) of two nodes is independent of the order of arguments.

The proof with respect to the remaining properties (iii) and (iv) can be conducted as follows:

(b)→(iii) Let $u, v \in \mathcal{C}$ be two classes with $d_{\mathcal{G}}(u, v) = 0$. This means that their LCS has height 0 and hence must be a leaf node. Because leaf nodes have, by definition, no further children, $u = \mathrm{lcs}(u, v) = v$. On the other hand, for any class $w \in \mathcal{C}$, $d_{\mathcal{G}}(w, w) = 0$ because $\mathrm{lcs}(w, w) = w$ and w is a leaf node according to (b).

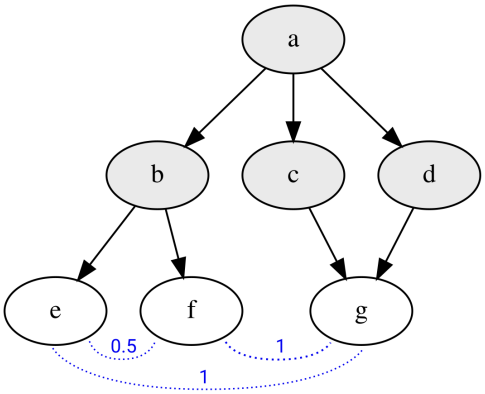
(a)→(iv) Let $u, v, w \in \mathcal{C}$ be three classes. Due to (a), there exists exactly one unique path from the root of the hierarchy to any node. Hence, $\mathrm{lcs}(u, v)$ and $\mathrm{lcs}(v, w)$ both lie on the path from $\mathrm{root}(\mathcal{G})$ to v and they are, thus, either identical or one is an ancestor of the other. Without loss of generality, we assume that $\mathrm{lcs}(u, v)$ is an ancestor of $\mathrm{lcs}(v, w)$ and thus lies on the root-paths to u, v, and w. In particular, $\mathrm{lcs}(u, v)$ is a subsumer of u and w and, therefore, $\mathrm{height}(\mathrm{lcs}(u, w)) \leq \mathrm{height}(\mathrm{lcs}(u, v))$. In general, it follows that $d_{\mathcal{G}}(u, w) \leq \max\{d_{\mathcal{G}}(u, v), d_{\mathcal{G}}(v, w)\} \leq d_{\mathcal{G}}(u, v) + d_{\mathcal{G}}(v, w)$, given the non-negativity of $d_{\mathcal{G}}$.

□

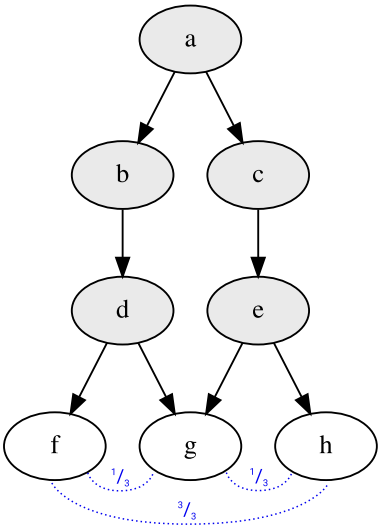
Remark regarding the inversion

If $d_{\mathcal{G}}$ is a metric, all classes $u \in \mathcal{C}$ of interest must necessarily be leaf nodes, since $d_{\mathcal{G}}(u, u) = 0 \Rightarrow \mathrm{height}(\mathrm{lcs}(u, u)) = \mathrm{height}(u) = 0$.

However, (iv)→(a) does not hold in general, since $d_{\mathcal{G}}$ may even be a metric for graphs $\mathcal{G}$ that are not trees. An example is given in Fig. 1a. Nevertheless, most such graphs violate the triangle inequality, like the example shown in Fig. 1b.



(a) A non-tree hierarchy where $d_{\mathcal{G}}$ is a metric.



(b) A non-tree hierarchy where $d_{\mathcal{G}}$ violates the triangle inequality.

Figure 1: Examples for non-tree hierarchies.

2. Further Quantitative Results

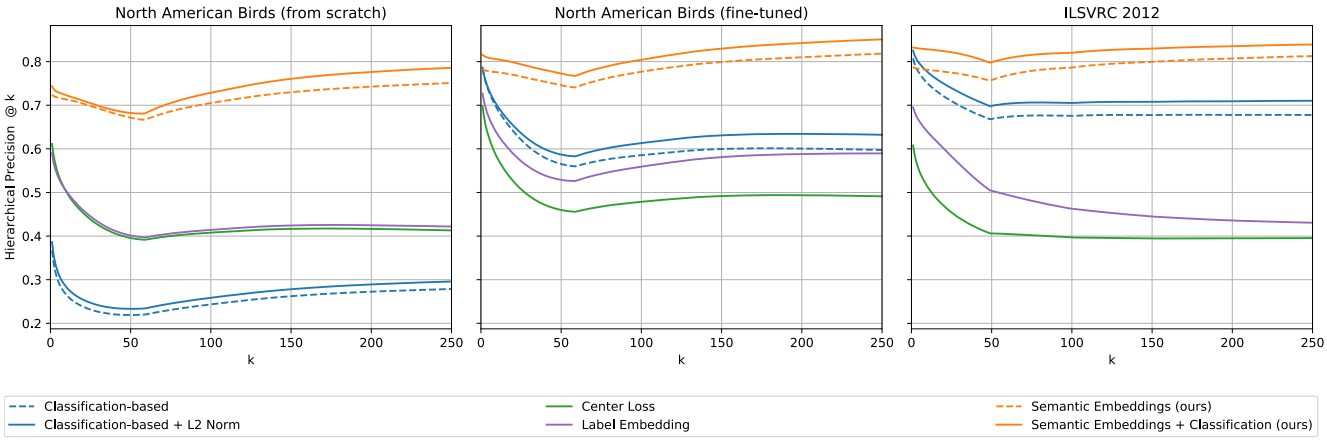


Figure 2: Hierarchical precision on NAB and ILSVRC 2012.

Method	CIFAR-100			NAB		ILSVRC
	Plain-11	ResNet-110w	PyramidNet	from scratch	fine-tuned	
Classification-based	0.2078	0.4870	0.3643	0.0283	0.2771	0.2184
Classification-based + L2 Norm	0.2666	0.5305	0.4621	0.0363	0.3194	0.2900
DeViSE	0.2879	0.5016	0.4131	—	—	—
Center Loss	0.4180	0.4153	0.3029	0.1591	0.1802	0.1285
Label Embedding	0.2747	**0.6202**	0.5920	0.1271	0.2417	0.2683
Semantic Embeddings ($\mathcal{L}_{\text{CORR}}$) [ours]	0.5660	0.5900	0.6642	0.4249	0.5246	0.3037
Semantic Embeddings ($\mathcal{L}_{\text{CORR+CLS}}$) [ours]	**0.5886**	0.6107	**0.6808**	**0.4316**	**0.5768**	**0.4508**

Table 1: Classical mean average precision (mAP) on all datasets. The best value per column is set in bold font. Obviously, optimizing for a classification criterion only leads to sub-optimal features for image retrieval.

3. Qualitative Results on ILSVRC 2012

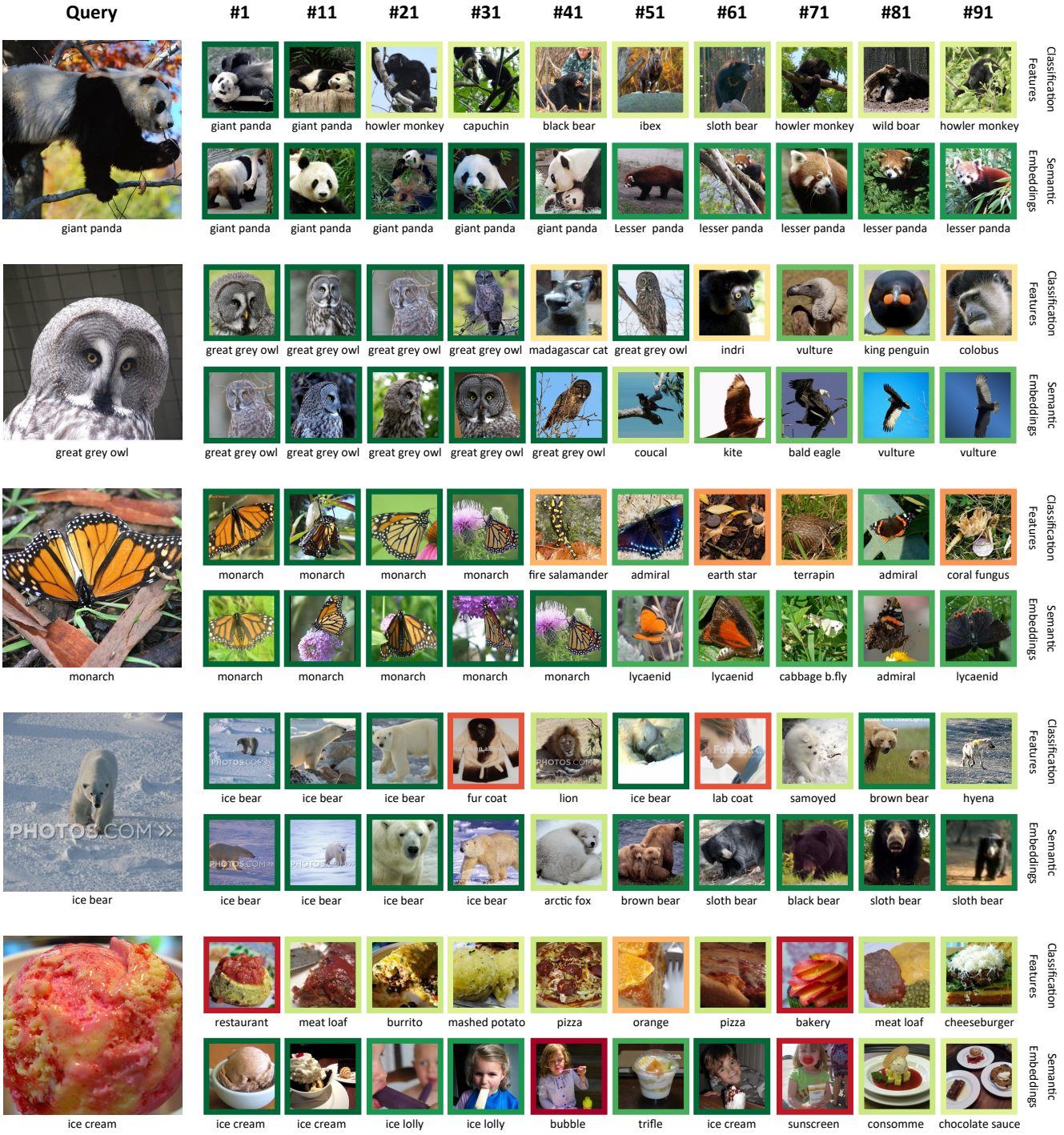


Figure 3: Comparison of a subset of the top 100 retrieval results using L2-normalized classification-based and our hierarchy-based semantic features for 3 exemplary queries on ILSVRC 2012. Image captions specify the ground-truth classes of the images and the border color encodes the semantic similarity of that class to the class of the query image, with dark green being most similar and dark red being most dissimilar.

Image	Classification-based		Semantic Embeddings (ours)	
	1. Giant Panda	(1.00)	1. Giant Panda	(1.00)
	2. American Black Bear	(0.63)	2. Lesser Panda	(0.89)
	3. Ice Bear	(0.63)	3. Colobus	(0.58)
	4. Gorilla	(0.58)	4. American Black Bear	(0.63)
	5. Sloth Bear	(0.63)	5. Guenon	(0.58)
	1. Great Grey Owl	(1.00)	1. Great Grey Owl	(1.00)
	2. Sweatshirt	(0.16)	2. Kite	(0.79)
	3. Bonnet	(0.16)	3. Bald Eagle	(0.79)
	4. Guenon	(0.42)	4. Vulture	(0.79)
	5. African Grey	(0.63)	5. Ruffed Grouse	(0.63)
	1. Monarch	(1.00)	1. Monarch	(1.00)
	2. Earthstar	(0.26)	2. Cabbage Butterfly	(0.84)
	3. Coral Fungus	(0.26)	3. Admiral	(0.84)
	4. Stinkhorn	(0.26)	4. Sulphur Butterfly	(0.84)
	5. Admiral	(0.84)	5. Lycaenid	(0.84)
	1. Ice Bear	(1.00)	1. Ice Bear	(1.00)
	2. Arctic Fox	(0.63)	2. Brown Bear	(0.95)
	3. White Wolf	(0.63)	3. Sloth Bear	(0.95)
	4. Samoyed	(0.63)	4. Arctic Fox	(0.63)
	5. Great Pyrenees	(0.63)	5. American Black Bear	(0.95)
	1. Ice Cream	(1.00)	1. Ice Cream	(1.00)
	2. Meat Loaf	(0.63)	2. Ice Lolly	(0.84)
	3. Bakery	(0.05)	3. Trifle	(0.89)
	4. Strawberry	(0.32)	4. Chocolate Sauce	(0.58)
	5. Fig	(0.32)	5. Plate	(0.79)
	1. Cocker Spaniel	(1.00)	1. Cocker Spaniel	(1.00)
	2. Irish Setter	(0.84)	2. Sussex Spaniel	(0.89)
	3. Sussex Spaniel	(0.89)	3. Irish Setter	(0.84)
	4. Australien Terrier	(0.79)	4. Welsh Springer Spaniel	(0.89)
	5. Clumber	(0.89)	5. Golden Retriever	(0.84)

Figure 4: Top 5 classes predicted for several example images by a ResNet-50 trained purely for classification and by our network trained with $\mathcal{L}_{\mathrm{CORR+CLS}}$ incorporating semantic information. The correct label for each image is underlined and the numbers in parentheses specify the semantic similarity of the predicted class and the correct class. It can be seen that class predictions made based on our hierarchy-based semantic embeddings are much more relevant and consistent.

4. Low-dimensional Semantic Embeddings

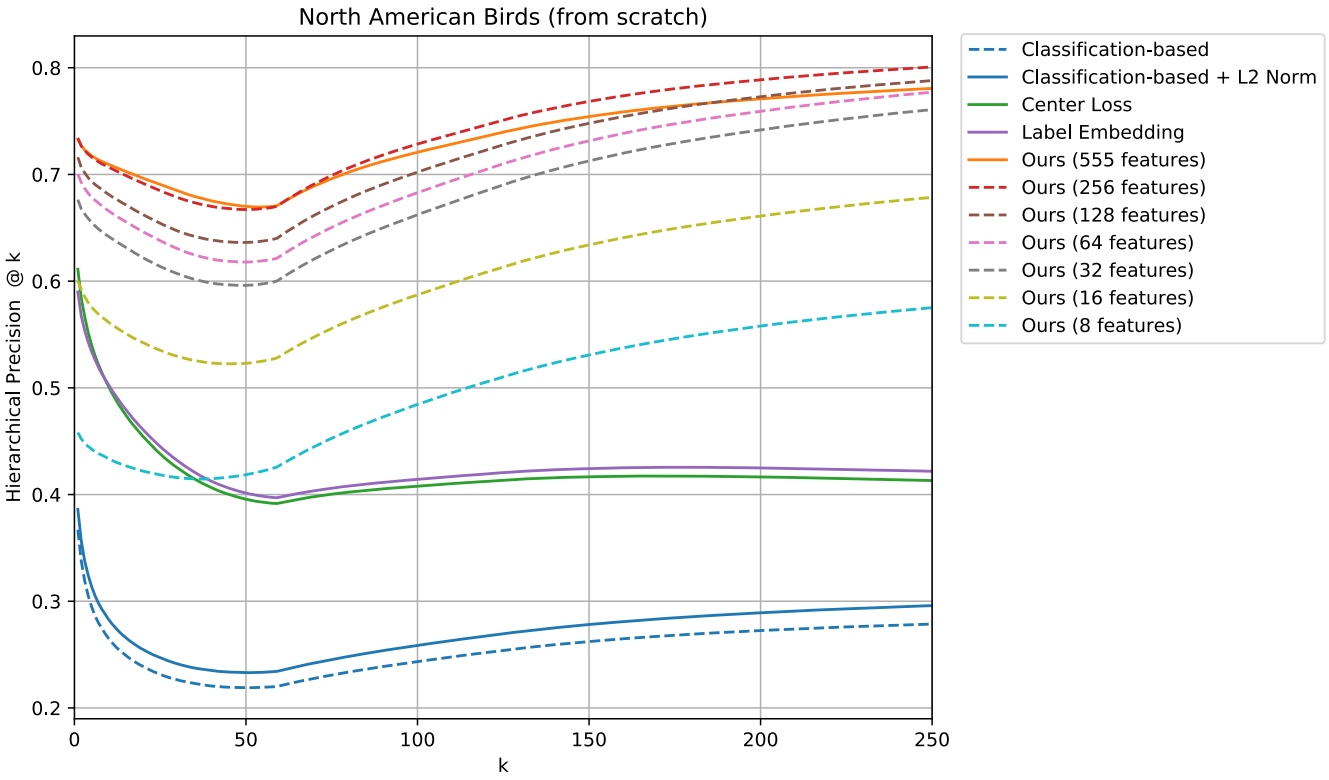


Figure 5: Hierarchical precision of our method for learning image representations based on class embeddings with varying dimensionality, compared with the usual baselines.

As can be seen from the description of our algorithm for computing class embeddings in section 3.2 of the paper, an embedding space with n dimensions is required in general to find an embedding for n classes that reproduces their semantic similarities exactly. This can become problematic in settings with a high number of classes.

For such scenarios, we have proposed a method for computing low-dimensional embeddings of arbitrary dimensionality approximating the actual relationships among classes in section 3.3 of the paper. We experimented with this possibility on the NAB dataset, learned from scratch, to see how reducing the number of features affects our algorithm for learning image representations and the semantic retrieval performance.

The results in Fig. 5 show that obtaining low-dimensional class embeddings through eigendecomposition is a viable option for settings with a high number of classes. Though the performance is worse than with the full amount of required features, our method still performs better than the competitors with as few as 16 features. Our approach hence also allows obtaining very compact image descriptors, which is important when dealing with huge datasets.

Interestingly, the 256-dimensional approximation even gives slightly better results than the full embedding after the first 50 retrieved images. We attribute this to the fact that fewer features leave less room for overfitting, so that slightly lower-dimensional embeddings can generalize better in this scenario.

5. Taxonomy used for CIFAR-100