Evaluating Knowledge Transfer and Zero-Shot Learning in a Large-Scale Setting

Marcus Rohrbach Michael Stark Bernt Schiele
MPI Informatics, Saarbrücken

Abstract

While knowledge transfer (KT) between object classes has been accepted as a promising route towards scalable recognition, most experimental KT studies are surprisingly limited in the number of object classes considered. To support claims of KT w.r.t. scalability we thus advocate to evaluate KT in a large-scale setting. To this end, we provide an extensive evaluation of three popular approaches to KT on a recently proposed large-scale data set, the ImageNet Large Scale Visual Recognition Competition 2010 data set. In a first setting they are directly compared to one-vs-all classification often neglected in KT papers and in a second setting we evaluate their ability to enable zero-shot learning. While none of the KT methods can improve over one-vs-all classification they prove valuable for zero-shot learning, especially hierarchical and direct similarity based KT. We also propose and describe several extensions of the evaluated approaches that are necessary for this large-scale study.

1. Introduction

Inspired by the success of recent object class recognition on individual classes, the simultaneous recognition of many classes has become an active research area. Scaling recognition to larger numbers of classes poses challenges with respect to the expressiveness and learnability of object models as well as the need for increasing amounts of training data. Knowledge transfer between object classes has been advertised as a promising route towards scalable recognition, by efficiently re-using acquired knowledge in the context of newly posed, but related recognition tasks. While experimental studies connected to knowledge transfer have shown promising results they are often limited w.r.t. the size of employed data sets.

As a consequence, it remains unclear whether the benefits demonstrated in small-scale experiments considering only a few classes really take effect in large-scale settings. In fact, Deng et al. [6] found that the relative performance of different recognition methods can change when increasing test database size by an order of magnitude. The major contribution of this paper is therefore to revisit three recently proposed knowledge transfer approaches and to evaluate them in a truly large-scale setting, effectively starting where previous evaluations have left off. We evaluate knowledge transfer on the recently proposed ImageNet data set [7], specifically, on the associated ImageNet Large Scale Visual Recognition Competition 2010 (ILSVRC10) subset [2]. It consists of over 1.2 million images of 1000 object classes, providing a currently unparalleled test bed for vision algorithms in terms of both scale and diversity. Being based on WordNet [18] synonym sets, ImageNet offers the additional advantage of providing a hierarchical organization of object classes according to hypernym/hyponym relations, lending itself to knowledge transfer using object class hierarchies.

Our experimental study follows three prominent directions in knowledge transfer, which have proven effective for comparatively small numbers of object classes. The first direction imposes a hierarchical structure on the space of object classes, according to the general-to-specific ordering defined by the data set [12, 17, 32]. The second direction is based on representing object classes relative to an inventory of generic visual attributes [8, 14, 22], where classes are characterized by distinct patterns of attribute activations. The third direction is based on direct similarities to related classes effectively using the classifiers of most similar classes [1, 10, 22]. For all three directions we go far beyond previous studies in terms of data set size, and evaluate knowledge transfer in the context of both traditional multiclass classification and zero-shot recognition.

Our paper makes the following contributions: First, to the best of our knowledge, we are the first to provide an in-depth study of knowledge transfer in a truly large-scale setting. Second, we compare three different approaches to knowledge transfer: one based on an object class hierarchy, one based on attributes, and one based on direct similarity. Third, we contrast knowledge transfer with the traditional approach of one-versus-all classification [21], which is often neglected in previous knowledge transfer work. Fourth, we challenge fully unsupervised transfer in a zero-shot recognition task aiming to recognize 200 unseen test classes. Fifth, we propose technical modifications to several approaches making them applicable to large-scale data[1].

Sec. 1 discusses related work. Sec. 2 introduces the different knowledge transfer approaches. Sec. 3 motivates our setup for the experiments in Sec. 4 and 5.

To appear in the Proc. of the *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 1641–1648, Colorado Springs, Colorado, June 2011.

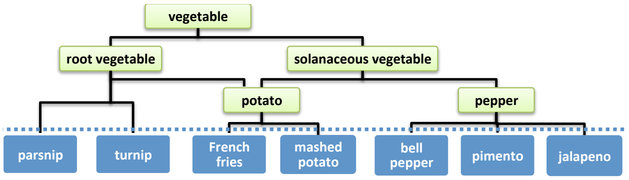


Figure 1: ISVLRC10 subgraph. Leaf (blue), inner nodes (green).

Related work Knowledge transfer for object class recognition comes in different flavors, such as joint learning of multiple classes [28] or transferring object class priors [9]. Recently, three lines of research have gained particular popularity due to their potential scalability.

A first line of research exploits the hierarchical structure of the object class space imposed by a general-to-specific ordering, either based on an existing hierarchy [17, 32] or learned from visual features [12]. Scalability is achieved by associating classifiers to each hierarchy node, allowing for classification in a divide-and-conquer fashion. Our hierarchical classification is closest to [7], combining classifier scores of distinct subgraphs to yield final classification scores. [6] follows a different route by forming a weighted average of all classifiers in a hierarchy for classification. While the latter two approaches report multiclass classification results on (subsets of) the ImageNet data set, our study additionally considers zero-shot recognition.

A second line of research uses an intermediate layer of descriptive attributes to represent object classes [8, 14, 22], encoding high-level visual properties that can be shared among object classes, hence promoting scalability. Our attribute-based object class model is inspired by [14], and uses linguistic knowledge bases to determine both an attribute inventory and the associations between object classes and attributes fully automatically [22].

A third line of research uses direct similarities between object classes. [1] encodes instances of previously unknown classes as collections of "familiar" classifier responses, i.e., similarities to known classes, and applying a nearest-neighbor scheme for classification. While most work based on similarity between classes [1, 10] require a few training samples for new classes, we employ our unsupervised approach [22] where class similarities are mined automatically using semantic relatedness measures with linguistic knowledge bases like Wikipedia or web search.

2. Knowledge transfer approaches

In this paper we explore two distinct settings for knowledge transfer. In a first experiment (Sec. 4) we assume that training data is available for all classes. In this setting knowledge can be transferred (or shared) among all classes and thus may lead to better classification performance. This setting is called *knowledge sharing* in the following. In the second experiment we assume that training data is available for a subset of known classes and that no training data is available for the remaining unseen classes. This setting is called *zero-shot recognition* and described in Sec. 5. We have chosen these two distinct settings as they represent two extreme cases for knowledge transfer.

The following gives an overview of the different knowledge transfer approaches explored in our study. Sec. 2.4 then describes how semantic relatedness is used to enable unsupervised attribute- and direct similarity-based knowledge transfer.

2.1. Hierarchy-based knowledge transfer

We exploit the hierarchical structure of the ILSVRC10 to train two types of classifiers (see for a small sample subgraph Fig. 1). We train classifiers for leaf nodes z_l by using training images of that node as positive samples and all other images as negative samples. Additionally we train classifiers for inner nodes y_i using all images associated to hyponyms of y_i as positive and all images outside the subtree rooted at y_i as negative examples. Fig. 1 shows an example, where a classifier for *solanaceous vegetable* uses *French fries, mashed potato, bell pepper, pimento,* and *jalapeno* images as positives as well as *parsnip* and *turnip* images as negative examples. We exclude the root and any trivial nodes (with only a single hyponym), as they do not provide additional information, resulting in a total of 370 inner node classifiers.

We distinguish three approaches. First, for scoring image x according to a leaf class z_l, we average over all classifier scores $s(y_i|x)$ of hypernyms H_{z_l} of z_l (for a *bell pepper* classifier we thus use the *pepper* and *solanaceous vegetable* classifiers), which we denote the **inner WordNet nodes** model:

$$s^{inn}(z_l|x) = \frac{\sum_{y_i \in H_{z_l}} s(y_i|x)}{|H_{z_l}|} \tag{1}$$

Second, since this model is not capable to distinguish among leaf classes z_l that share the same hypernyms, such as *French fries* and *mashed potato*, we also include leaf node classifiers in the **all WordNet nodes** model:

$$s^{all}(z_l|x) = \frac{s(z_l|x) + \sum_{y_i \in H_{z_l}} s(y_i|x)}{1 + |H_{z_l}|} \tag{2}$$

The third approach is based on the hierarchical cost sensitive classifier proposed by [6]. This formulation ties to optimize for the hierarchical error, defined in Sec. 3.2. To estimate the score of a certain class z_l we use cost-weighted classifier probabilities of all **leaf nodes** Z_l**, cost sensitive** to the cost $c^{z_l}_{z_i}$ between nodes z_i and z_l which is equivalent to the hierarchical error:

$$s^{cost}(z_l|x) = -\sum_{z_i \in Z_l} c^{z_l}_{z_i} p(z_i|x) \tag{3}$$

The hierarchy-based model allows for a flexible combination of leaf and inner node classifiers. In the *knowledge*

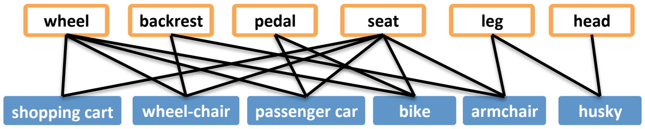


Figure 2: Example part attributes (orange), object classes (blue).

sharing case the inner and leaf node classifiers are trained on training data from all classes. In the *zero-shot* case only those leaf node classifiers can be trained for which training data is available and the inner node classifiers are trained on the known classes only. Fig. 4 gives an example for transferring knowledge using leaf, inner, and all WordNet nodes models accordingly for the *zero-shot* case.

2.2. Attribute-based knowledge transfer

We adopt the probabilistic direct attribute prediction model (DAP) introduced by Lampert *et al.* [14]. The DAP represents object classes z_l relative to an inventory of descriptive attributes a_m, realized as probabilistic attribute classifiers $p(a_m|x)$. In the *knowledge sharing* case these are trained on all classes whereas in the *zero-shot* case these are trained on known classes only. Once trained, the attribute classifiers can be flexibly combined to recognize previously unseen classes in the *zero-shot* setting or to recognize known classes in the *knowledge sharing* case. The association between object classes z_l and attributes a_m (see Fig. 2 for an example) is controlled by a matrix of indicator variables $a_m^{z_l}$. Assuming mutual independence of attributes and uniform priors $p(a_m) = 0.5$ yields the following probability estimate of class z_l being present in image x [22]:

$$p^{attr}(z_l|x) \propto \prod_{m=1}^{M} (2 * p(a_m|x))^{a_m^{z_l}} \tag{4}$$

For efficiency reasons, we propose the following non-probabilistic sum formulation, which replaces calibrated attribute probabilities $p(a_m|x)$ by zero-boundary attribute decision scores $s(a_m|x)$:

$$s^{attr}(z_l|x) = \frac{\sum_{m=1}^{M} s(a_m|x)^{a_m^{z_l}}}{\sum_{m=1}^{M} a_m^{z_l}}, \tag{5}$$

Although this formulation does not require calibrated probabilities, it does require normalized scores. We found empirically that a simple z-score is sufficient.

In order to validate the sum formulation, we compare its performance to the probabilistic formulation in Tab. 1 for both error measures (see Sec. 3.2 for details). The important observation is that the sum formulation outperforms the probabilistic formulation.We thus use the sum formulation in the following.

Dataset & Approach	Error	Product	Sum
ILSVRC 10, inner nodes	Top 1	93.5	**90.9**
ILSVRC 10, inner nodes	Top 5	80.1	**71.6**

Table 1: Evaluation of the probabilistic product model suggested by [14] vs. our sum model, see Sec. 2.2. Error in %.

2.3. Direct similarity-based knowledge transfer

Motivated by its superior classification performance [22], we also include a direct similarity based approach. This can be defined as a modification of the attribute-based model that represents object classes relative to a set of K semantically related reference classes z_k, implemented by classifiers $s(z_k|x)$:

$$s^{dir}(z_l|x) = \frac{\sum_{k=1}^{K} s(z_k|x)}{K}, \tag{6}$$

Direct similarity is used only in zero-shot experiments as the most related known class in the knowledge sharing setting is always the class itself.

2.4. Semantic relatedness for attribute- and direct similarity-based approaches

The attribute–based approach relies on an association matrix between a set of attributes and the object classes. The ILSVRC10, however, is neither provided with a set of attributes nor with manual class-attribute associations. Therefore we rely on part attributes mined from WordNet to generate an inventory of attributes for all classes [22]. In total we mine 811 part attributes. An alternative to mine attributes would be to use WordNet's synset definitions [24].

For these mined attributes we use semantic relatedness measures in connection with linguistic knowledge bases to automatically determine associations between the attributes and object classes. While in [22] each class and attribute is associated with one term, the classes and attributes in this work refer to WordNet concepts, called *synsets*, which are represented by several terms. As the semantic relatedness measures are based on terms rather than semantic concepts we take the median over all possible term combinations for a specific association.

For mining class-attribute associations we choose the best performing measures [22, 23] which are applicable to large scale: (1) the explicit semantic analysis based on *Wikipedia* [26]; (2) *Yahoo Holynyms* which is based on hitcounts and uses specific part queries such as "the wheel of the car"; (3) *Yahoo Image* which is based on image-search hitcounts; and (4) *Yahoo Snippets* which is based on web page summaries returned by the search engine. For the direct similarity based approach we replace Yahoo Holonyms with simple *Yahoo Web* queries as it is not applicable for direct similarity. For improved robustness of the attributes we also compute a class level fusion over *all attributes*.

		Learning	Total	Err. top	
Model	Descriptor	method	dim.	5	1
BoW [2]	Sift	LibLinear	1,000	80	91
BoW	Sift	MeanSGD	1,000	72	86
BoW + SPM	rgSift	MeanSGD	8,000	59	76
LLC + SPM	rgSift	MeanSGD	21,000	50	69
Fisher vector	rgSift	MeanSGD	32,768	43	61
LLC+SPM, Fisher	**rgSift**	**MeanSGD**	**53,768**	**38**	**57**
Fisher+SPM [25]	Sift, Color	SGD	262,144	34	–
LLC,SVC+SPM [16]	Hog, Lbp	ASGD	1,179,648	28	47

Table 2: One-vs-all performance of different methods on ILSVRC10. BoW: bag of visual words, SPM: spatial pyramid matching [15], LLC: locality-constrained linear coding [30], Fisher vector [19], SVC: Super-Vector Coding [31], Lbp: local binary patterns, SGD: stochastic gradient decent [3], ASGD: averaging SGD [16].

Robust associations for large scale. In contrast to prior work we have a significantly larger amount of potential classes associated to each attribute. To learn precise attribute classifiers we use only the most likely classes as positives and least likely as negatives, leaving out the potentially noisy middle part. For the attribute *backrest* in Fig. 2 we would thus use *wheelchair* and *armchair* as positives, *bike* and *husky* as negatives, and not use the classes *shopping cart* and *passenger car* which are uncertain in respect to the attribute *backrest*.

Parameter selection. For attribute- and direct similarity-based knowledge transfer, continuous semantic relatedness measures have to be discretized to yield binary associations between attributes and object classes and in between object classes, respectively, by thresholding. Since we found large performance differences depending on thresholding [23], we determine threshold values on the validation set, and fix them for the rest of the experiments. In particular, for attribute-based knowledge transfer, we set the threshold such that, on average, 3% of all attributes are active for a given object class. For the direct similarity based approach, we set the threshold such that the $K = 5$ most related object class models are considered.

3. Experimental setup

Evaluating and comparing the different knowledge transfer approaches of Sec. 2 in a large scale setting requires careful design of the experimental setup. The following details and argues for our choices concerning data set, image representation, and learning methodology.

3.1. Dataset

The number of available datasets containing more than a few hundred object classes with sufficiently many images per class is still limited. Caltech256 [11] is frequently used, however, it consists only of 256 classes and 30k images. NUS-WIDE [5] is significantly larger with 270k images and over 5k unique tags but contains ground truth for only 81 categories. The tiny image data set [27] (80 million images, loosely labeled with 75,062 WordNet nouns) provides a significantly larger number of images but is mostly restricted to 32x32 pixel images.

Recently, Deng *et al.* proposed ImageNet [7] (3.2 million images of 5247 WordNet synonym sets) as a resource for truly large-scale experimentation. Based on this dataset the ImageNet Large Scale Visual Recognition Challenge 2010 (ILSVRC10, [2]) has been introduced. We have chosen this subset for large-scale experiments as it is a well-defined subset of 1,000 object classes (1.2 million images, divided into distinct portions for training, validation, and test) for classification experiments, suggesting this benchmark to be the de-facto choice for large-scale experiments in the near future.

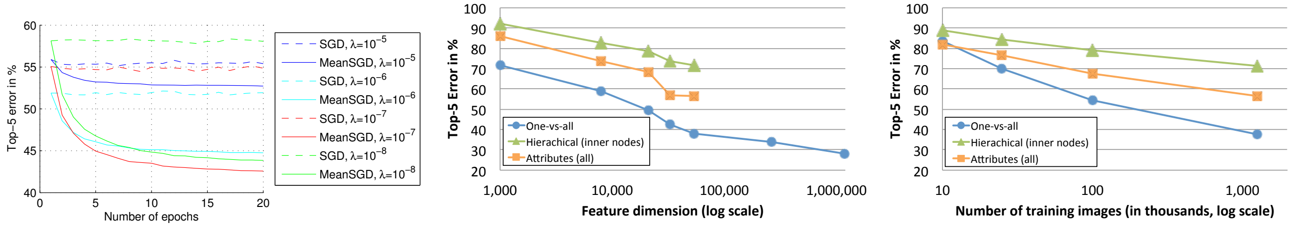
3.2. Performance measures

ILSVRC10 [2] introduced and defined the following performance measures used throughout the paper. Performance is measured as the top-n error rate (the n most confident classification hypotheses are considered as potentially correct) and distinguishes two error measures. The first is a *flat* measure which equals 0 if the test class is predicted correctly within the n most confident hypotheses, and 1 otherwise. The second is a *hierarchical* measure, which equals the minimum height of the lowest common ancestors between true and hypothesized classes. As suggested in [2] we report top-n errors for $n = 5$ and $n = 1$, which corresponds to 1-accuracy. In order to avoid fitting the test data, we use the provided validation set for preliminary experimentation and parameter selection (Fig. 3a and 3b, Tab. 1 and 2). The final results (Sec. 4 and 5, Fig. 3c, Tab. 3 and 4) are obtained on the test set.

3.3. Image representation

In order to allow for a sufficient range of experiments on the ILSVRC10 dataset, we require an image representation that is both powerful enough to achieve good performance and reasonably sized to support efficient learning. We thus base our choice on the outcome of the ILSVRC10 competition, which we recapitulate in part in Tab. 2, and seek to find a compromise between performance and manageable runtimes.

We observe that the performance ranges from 80% top-5 error rate for a BoW Sift baseline (Tab. 2, first row) to an impressive performance of as low as 34% and 28% top-5 error of the best performing approaches (Tab. 2, last two rows). In an attempt to regulate the performance-runtime tradeoff, we explore different combinations of techniques used by the best performing approaches [25, 16] such as spatial pyramid matching (SPM [15]), locality-constrained



(a) Convergence of SGD and MeanSGD for different step sizes λ on ILSVRC10 (one-vs-all, Fisher vector, rgSift).

(b) Error vs. number of feature dimensions (for details see Tab. 2)

(c) Error vs. number of training images

Figure 3: Performance vs. (a) number of epochs, (b) feature dimensionality, and (c) number of training images.

linear coding (LCC [30]), and the Fisher vector [19] (we adapted the implementation of [13]), in connection with the color sift variant rgSift [29] (Tab. 2, rows 2 to 6).

As can be seen from Tab. 2 and Figure 3b (blue dots) the performance increases monotonically with descriptor dimensionality. While the last two approaches perform best they use feature vectors of several 100k and over 1Mio dimensions, resulting in prohibitive runtimes for our purposes. For this paper, we opted for the Fisher vector and LLC+SPM representation as a sensible compromise between performance (38% top-5 error rate, Tab. 2, row 6) and runtime. For combining the two representations we simply average their scores. We fix this representation for all remaining experiments.

3.4. Learning method

Motivated by the potential of stochastic gradient-based optimization for rapid convergence, and in line with the two best performing ILSVRC10 approaches, we use linear SVM classifiers, trained using stochastic gradient descent (SGD) [3]. Similar in spirit to averaging SGD (ASGD) [20, 16], we average the SVM's weight and bias. However, in contrast to [16] we do not average after each step, but take the mean of the results after each epoch (one pass over the data). More specifically, we save the weight vector w_i and bias b_i after each epoch i (the data is randomly reordered before each epoch). While the score of the normal SGD after n epochs only depends on the weights and bias after the final epoch

$$f_{SGD}(x) = \langle w_n, x \rangle + b_n, \tag{7}$$

we compute the mean over all epochs in MeanSGD:

$$f_{MeanSGD}(x) = \frac{\sum_{i=1}^{n} \langle w_i, x \rangle + b_i}{n} \tag{8}$$

(where $\langle w, x \rangle$ is the scaler product of w an x).

As can be seen in Figure 3a, using MeanSGD (solid lines) instead of SGD (dashed lines) significantly speeds up convergence and improves performance. We use hinge loss and fix, according to Figure 3a, the step size λ to 10^{-7} and the number of epochs n to 20 epochs.

In order to benefit from modern multi-core hardware, we further implemented a parallelized version of MeanSGD based on Bouttou's SGD [4], exploiting data parallelism. It requires about 20 hours (including file and network I/O) for training all 1,000 one-vs-all classifiers with 20 epochs using the 53,768 dimensional Fisher vector on a 32-core machine. The code including a Matlab wrapper is available on our webpage.

4. Large scale knowledge sharing

As motivated in Sec. 2, in a first set of experiments we consider the *knowledge sharing* case where we assume to have training samples for all classes.

Tab. 3 gives results for classifying all test images of the ILSVRC10 data set into 1000 classes, using the provided training set for training. Performance is measured in terms of the corresponding flat and hierarchical (in brackets) variants of top-5 and top-1 error (see Sec. 3.2). The table compares the performance of standard one-vs-all classification (part 1 of Tab. 3, using leaf node z_l classifiers only), hierarchical models (part 2), and attribute-based models (part 3).

We proceed by examining Tab. 3 from top to bottom. First, we observe that the standard one-vs-all approach (Tab. 3 part 1) achieves a remarkable top-5 error rate of 37.6% with a hierarchical error rate of 2.91.

In contrast, the hierarchical model using only inner nodes (Tab. 3 part 2) performs relatively poorly (top-5 error of 71.3%, hierarchical error 7.31). This drop is understandable, considering the much smaller number of available inner node classifiers (370 compared to 1,000 leaf node classifiers). Adding the leaf nodes boosts the performance of the hierarchical model by more than 20% w.r.t. the flat top-5 error rate (50.4%, hierarchical error rate 5.49). Surprisingly, the resulting performance is still slightly worse than one-vs-all – the effect of adding confusion by adding more uniformly weighted classifiers is apparently more pronounced than added discriminative power. When examining the results more closely we find that the performance of the inner leaf node classifier does not correlate with the level of abstraction in the hierarchy. However, we find that it strongly depends on the semantic grouping, e.g. the category flower which is associated with 87 leaf nodes can be very well separated from other nodes in contrast to the class node described with the synset {fastener, fastening, holdfast, fixing}, which has 10 visually diverse and difficult child nodes such as *button, hair slide, knot,* and *screw*.

The hierarchical approach based on [6] uses one-vs-all leaf nodes, but makes them sensitive to the hierarchical cost (see Sec. 2.1). With 48.6% top-5 error (Tab. 3 part 2) it clearly outperforms the hierarchical approach using only inner WordNet nodes (by 23%) and slightly all WordNet nodes (by 2%). However, compared to plain one-vs-all the flat top-5 error increases by 11% and even the hierarchical error by 1.8. The main reason for this less discriminant hierarchical classifier seems to be that this approach uses all classifiers but the one trained for the specific class to be detected.

The last line of Tab. 3 part 2 gives the results for a stacking-based combination of inner and leaf node classifiers. We use a SVM (MeanSGD) stacked on top of the scores of all nodes and both features to learn the relative importance of the nodes, i.e. we learn one-vs-all classifiers which use the classifier scores as feature vectors. In contrast to the the previous hierarchical approaches the trained SVM now correctly attenuates the influence of weak (inner) nodes and achieves a top-5 error of 36.8% which is even slightly better than one-vs-all.

Tab. 3 part 3 gives results for attribute-based models using different semantic relatedness measures for determining object class-attribute associations. On average, using single measures (Wikipedia, Yahoo Holonyms, Image, or Snippets) performs in the same order of magnitude as inner WordNet nodes. When combining all attribute-classifiers from the different measures we improve performance by more than 10% to 56.4% top-5 error (15% lower than inner WordNet nodes). However, this cannot compete with the hierarchical approaches including the discriminative leaf nodes.

In the same fashion as for all WordNet nodes we can also stack a SVM on top of the different attribute classifiers to learn an optimal weighting between them. This results in a significant reduction in error by 13% to 43.8% top-5 error, which is, however, still 6% higher than one-vs-all or 7% higher than the stacked hierarchical approach.

Influence of feature representation and amount of training data. In this experiment we further analyze the dependency with respect to the number of feature dimensions and the amount of available training data. In addition to one-vs-all we pick the best approach for both knowledge transfer settings which is not based on one-vs-all leaf nodes: *inner WordNet nodes* for hierarchical setting and *all attributes*.

Approach	Top 5 Error	Top 1 Error
1. One-vs-all		
(=leaf WordNet nodes)	37.6 (2.91)	57.2 (5.77)
2. Hierarchical		
inner WordNet nodes	71.3 (7.31)	90.7 (8.69)
all WordNet nodes	50.4 (5.49)	67.9 (7.54)
leaf nodes, cost sensitive	48.6 (4.71)	60.2 (5.66)
SVM stacking, all nodes	36.8 (2.84)	56.3 (5.59)
3. Attributes		
Wikipedia	63.7 (5.21)	81.5 (8.52)
Yahoo Holonyms	68.7 (5.61)	87.1 (9.24)
Yahoo Image	74.0 (5.80)	90.6 (10.28)
Yahoo Snippets	67.2 (5.33)	84.6 (8.55)
all attributes	56.4 (4.63)	75.9 (7.32)
SVM stacking, all attributes	43.8 (3.38)	63.5 (6.34)

Table 3: Large scale knowledge sharing results. Shown is flat error in % (hierarchical error)

In Figure 3b we plot the error versus the feature dimensionality of the approaches listed in Tab. 2. We observe that for all approaches the performance increases logarithmically with increased feature dimension. From the SIFT representation (1,000 dimensional) to the combined LLC and Fisher vector (53,768 dimensional) the error decreases the most for one-vs-all by 34%, but still strongly by 29% for attributes and 21% for inner WordNet nodes. The relative performance difference between the approaches remains mainly stable across the different features representations which indicates that relative results of the approaches are independent of a specific feature representation.

In Figure 3c we show results for a reduced amount of training data per class to 10, 25, and 100 samples. The first observation is that the hierarchical and the attribute-based knowledge transfer schemes degrade less (17% and 25%, respectively) than the one-vs-all (46%) scheme. However, the relative ordering remains the same for 100 and 25 samples per class. Only for the rather extreme case of only 10 training samples the attribute-based approach slighly outperforms one-vs-all classification by 1.7%.

Summary. We conclude that the benefit of knowledge transfer is in fact limited for this knowledge sharing and standard multiclass classification setting and becomes apparent only in the stacking-based approaches. In case of limited feature representation or reduced training data the absolute performance differences between the approaches decrease, but one-vs-all remains among the best. The hierarchical based approaches only show reasonable performance when leaf nodes are included. As concerns attribute-based approaches, we observe that using all attribute-classifiers based on multiple semantic relatedness measures significantly improves performance.

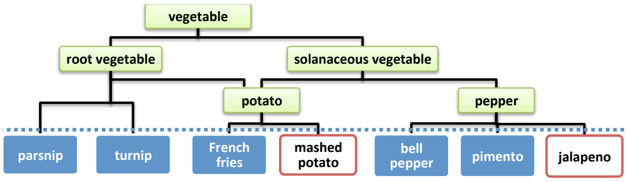


Figure 4: Zero-shot recognition using hierarchies. Unseen object classes (red) *mashed potato* / *jalapeno* can be recognized using neighboring leaf node (*French fries* / *bell pepper*, *pimento*), inner node (*potato* / *pepper*), or all (the respective unions) classifiers.

5. Large-scale zero-shot recognition

In this section, we apply the knowledge transfer approaches of Sec. 2 to a zero-shot recognition setting, in which the sets of object classes of training and test are *disjoint*. We hence denote training object classes as *known*, and test classes as *unseen*. In order to solve the zero-shot recognition task, knowledge obviously has to be transferred between training and test classes. Lampert *et al*. [14] provided a first benchmark for zero-shot recognition in the form of the Animals-with-Attributes (AwA) data set, consisting of approximately 30,000 images, divided into 40 known animal classes for training and 10 unseen animal classes for testing. In the present experimental study, we lift zero-shot recognition to another level both in terms of data set scale and diversity, by applying it to almost two orders of magnitude more images. In particular, we divide the ILSVRC10 data set randomly into two disjoint sets of object classes, one assumed known (800 classes), and one assumed unseen (200 classes)[1]. In all experiments, we further maintain the original split into training and test data defined by the ILSVRC10 data set, meaning that we train on the known (800 class) fraction of the original training set (1,005,761 images), and test on the unseen (200 class) fraction of the original test set (30,000 images).

Results. Tab. 4 gives results for zero-shot recognition, comparing hierarchical (part 1), attribute-based (part 2), and direct similarity-based (part 3) models. In analogy to Tab. 3, the table further distinguishes among hierarchical models using leaf, inner, and all hierarchy nodes, as well as among different semantic relatedness measures for attribute-based and direct similarity-based models. As the relative ranking of the methods is nearly identical between the different error measures (top-5, top-1, flat and hierarchical error) we use the flat top-5 error as the basis for our discussion.

On average, we observe a significant amount of error across the compared approaches. We stress that this can be expected, since the zero-shot recognition task is of considerable difficulty, and cannot be solved without transferring knowledge between potentially unrelated object classes.

	On 200 unseen classes	
Approach	Top-5 Error	Top-1 Error
1. Hierarchical		
leaf WordNet nodes	72.8 (4.72)	91.3 (11.73)
inner WordNet nodes	66.7 (4.20)	88.7 (11.16)
all WordNet nodes	65.2 (4.10)	88.4 (11.24)
2. Attributes		
Wikipedia	80.9 (5.17)	94.5 (11.69)
Yahoo Holonyms	77.3 (4.91)	94.0 (12.56)
Yahoo Image	81.4 (5.19)	95.5 (12.53)
Yahoo Snippets	76.2 (4.87)	93.3 (11.53)
all attributes	70.3 (4.57)	90.4 (11.62)
3. Direct Similarity		
Wikipedia	75.6 (5.20)	91.8 (11.28)
Yahoo Web	69.3 (4.49)	89.7 (11.10)
Yahoo Image	72.0 (4.60)	90.7 (11.26)
Yahoo Snippets	75.5 (4.89)	91.6 (11.27)
all measures	66.6 (4.41)	88.4 (10.65)

Table 4: Zero-shot recognition. Flat error in % (hierarchical error).

Examining the performance of the hierarchical methods (Tab. 4 part 1) we observe a top-5 error of 72.8% using leaf WordNet nodes only. This is the closest setting examined here to one-vs-all classification. It uses the WordNet hierarchy to identify the most similar known leaf node classes for an unseen test class (see Fig. 4). Using the inner WordNet nodes only, the performance improves to a top-5 error of 66.7%. This is remarkable, since, in comparison to leaf node classifiers, only far fewer and less specific inner node classifiers are used. Furthermore it is in contrast to results in the knowledge sharing experiment (using all classes for training) where performance drops for inner nodes (see Tab. 3): while we benefit from knowledge transfer through the inner nodes for zero-shot recognition, we are loosing precision compared to one-vs-all when sharing knowledge in the inner nodes. The error can slightly be reduced to 65.2% using all WordNet nodes, effectively combining the two previous settings.

Part 2 of Tab. 4 shows the results for attributed-based models using the fully unsupervised mining of both attribute inventory and object class-attribute associations. Overall the obtained error rates for the individual relatedness measures are not competitive to the ones obtained by the hierarchical models. Yahoo Snippets performs best with 76.2% top-5 error. However, when combining all attribute measures we achieve a top-5 error of 70.3% which lies between the performance of leaf and inner WordNet nodes.

On the other hand, the direct similarity-based models reported in part 3 of Tab. 4 obtain as low as 69.3% top-5 error for Yahoo Web and competitive 66.6% when combining the classifiers of all measures, which is only slightly worse than the best performance obtained by a hierarchical method (all WordNet nodes with 65.2%).

The slightly favorable role of direct similarity compared

[1] We provide code, settings, and intermediate results on our web pages to facilitate further research and comparison on large-scale knowledge transfer.

to attribute-based models is consistent with our previous findings [22]. It can be explained by both the limited quality of the automatically mined part attribute inventory and by having one vs. two potential sources of introducing label noise into the system by means of semantic relatedness (mined object class-attribute associations).

The strong performance of hierarchical models can be attributed to the increased amount of supervision given by the hierarchy, while the attribute- and direct similarity-based models are fully unsupervised.

6. Conclusion

This paper explored knowledge transfer (KT) in a truly large-scale setting, going far beyond experimental studies of prior work in KT w.r.t. data set scale, diversity, and range of tested methods. Our evaluation is based on a recently proposed large-scale data set (ILSVRC10, [2]) and includes three prominent approaches to knowledge transfer[1].

For the fully supervised knowledge sharing experiment, the hierarchical approach using the inner or all node classifiers obtained inferior performance to the leaf nodes only, corresponding to the one-vs-all classifiers. Only when learning a stacked one-vs-all SVM on top, the hierarchical approach could slightly surpass performance of the one-vs-all classifiers. In the zero-shot recognition setting however, the hierarchical approaches obtained overall best performance of the explored KT methods.

The attribute based KT methods, in their fully unsupervised incarnation as explored in this paper, consistently produced higher error rates than the hierarchy and direct similarity-based KT methods. As pointed out before this reduced performance can be – at least partly – explained by the limited nature of attributes used here that were restricted to automatically mined part attributes. It remains an open research question how to obtain an inventory of representative and descriptive attributes for this kind of approach.

The direct similarity based KT method performed on a similar level as the hierarchical methods. This is remarkable as this approach is fully unsupervised using semantic relatedness to automatically find the most related known classes. This is in contrast to the hierarchical methods that require additional information given as a hierarchy.

References

[1] E. Bart and S. Ullman. Single-example learning of novel classes using representation by similarity. In *BMVC*, 2005.

[2] A. Berg, J. Deng, and F.-F. Li. Large Scale Visual Recognition Challenge 2010. www.image-net.org/challenges/LSVRC/2010/, 2010.

[3] A. Bordes, L. Bottou, and P. Gallinari. Sgd-qn: Careful quasi-newton stochastic gradient descent. *JMLR*, 2009.

[4] L. Bottou. http://leon.bottou.org/projects/sgd, 2010.

[5] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng. Nus-wide: A real-world web image database from national university of singapore. In *CIVR*, 2009.

[6] J. Deng, A. Berg, K. Li, and L. Fei-Fei. What does classifying more than 10,000 image categories tell us? In *ECCV'10*.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[8] A. Farhadi, I. Endres, and D. Hoiem. Attribute-centric recognition for cross-category generalization. In *CVPR*, 2010.

[9] L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *PAMI*, 2006.

[10] M. Fink. Object classification from a single example utilizing class relevance pseudo-metrics. In *NIPS*, 2004.

[11] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. Technical Report 7694, Caltech, 2007.

[12] G. Griffin and P. Perona. Learning and using taxonomies for fast visual categorization. In *CVPR*, 2008.

[13] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *CVPR*, 2010.

[14] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, 2009.

[15] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.

[16] Y. Lin, F. Lv, S. Zhu, M. Yang, T. Cour, K. Yu, L. Cao, and T. Huang. Large-scale image classification: fast feature extraction and SVM training. In *CVPR*, 2011.

[17] M. Marszalek and C. Schmid. Semantic hierarchies for visual object recognition. In *CVPR*, 2007.

[18] G. A. Miller. Wordnet: a lexical database for english. *CACM'95*.

[19] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV'10*.

[20] B. T. Polyak and A. B. Juditsky. Acceleration of stochastic approximation by averaging. *SICON'92*.

[21] R. Rifkin and A. Klautau. In defense of one-vs-all classication. In *JMLR*, 2004.

[22] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What Helps Where – And Why? Semantic Relatedness for Knowledge Transfer. In *CVPR'10*.

[23] M. Rohrbach, M. Stark, G. Szarvas, and B. Schiele. Combining language sources and robust semantic relatedness for attribute-based knowledge transfer. In *ECCV-PnA'10*.

[24] O. Russakovsky and L. Fei-Fei. Attribute learning in large-scale datasets. In *ECCV-PnA'10*.

[25] J. Sánches, F. Perronnin, and T. Mensink. Improved Fisher Vector for Large Scale Image Classification. www.image-net.org/challenges/LSVRC/2010/ILSVRC2010_XRCE.pdf, 2010.

[26] G. Szarvas, T. Zesch, and I. Gurevych. Combining heterogeneous knowledge resources for improved distributional semantic models. In *PCICLing*, pages 289–303, 2011.

[27] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: a large dataset for non-parametric object and scene recognition. *PAMI*, 2008.

[28] A. Torralba, K. Murphy, and W. Freeman. Sharing visual features for multiclass and multiview object detection. In *CVPR*, 2004.

[29] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek. Evaluating color descriptors for object and scene recognition. *PAMI*, 2010.

[30] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *CVPR*, 2010.

[31] T. Z. Xi Zhou, Kai Yu and T. S. Huang. Image classification using super-vector coding of local image descriptors. In *ECCV*, 2010.

[32] A. Zweig and D. Weinshall. Exploiting object hierarchy: Combining models from different category levels. In *ICCV*, 2007.