Identifying Visual Attributes for Object Recognition from Text and Taxonomy

Caglar Tirkaz [a,*]

Jacob Eisenstein [b]
T. Metin Sezgin [c]
Berrin Yanikoglu [a]

[a] *Faculty of Engineering and Natural Sciences, Sabanci University, Istanbul, 34956, Turkey*

[b] *School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA 30308, USA*

[c] *College of Engineering, Koc University, Istanbul, 34450, Turkey*

Abstract

Attributes of objects such as *"square"*, *"metallic"*, *"red"*, *etc*. allow a way for humans to explain or discriminate object categories. These attributes also provide a useful intermediate representation for object recognition, including support for zero-shot learning from textual descriptions of object appearance. However, manual selection of relevant attributes among thousands of potential candidates is labor intensive. Hence, there is increasing interest in mining attributes for object recognition. In this paper, we introduce two novel techniques for nominating attributes and a method for assessing the suitability of candidate attributes for object recognition. The first technique for attribute nomination estimates attribute qualities based on their ability to discriminate objects at multiple levels of the taxonomy. The second technique leverages the linguistic concept of *distributional similarity* to further refine the estimated qualities. Attribute nomination is followed by our attribute assessment procedure, which assesses the quality of the candidate attributes based on their performance in object recognition. Our evaluations demonstrate that both taxonomy and distributional similarity serve as useful sources of information for attribute nomination, and our methods can effectively exploit them. We use the mined attributes in supervised and zero-shot learning settings to show the utility of the selected attributes in object recognition. Our experimental results show that in the supervised case we can improve on a state of the art classifier while in the zero-shot scenario we make accurate predictions outperforming previous automated techniques.

Key words: Object recognition, Zero-shot learning, Attribute mining, Attribute-based classification

1 Introduction

While much research in object recognition has focused on distinguishing categories, recent work has begun to focus on *attributes* that generalize across many categories [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15]. Attributes such as *"pointy"* and *"legged"* are semantically meaningful, interpretable by humans, and serve as an intermediate layer between the top-level object categories and the low-level image features. Moreover, attributes are generalizable and allow a way to create compact representations for object categories. This enables a number of useful new capabilities: *zero-shot learning* where unseen categories are recognized [3,4,7], generation of textual descriptions and part localization [2,5,10], prediction of color or texture types [1], and improving performance of fine-grained recognition tasks (*i.e.* butterfly and bird species or face recognition) [3,6,12,13,15] where categories are closely related.

However, using attributes for object recognition requires answering a number of challenging technical questions — most crucially, specifying the set of attributes and the category-attribute associations. Most prior work uses a predefined list of attributes specified either by domain experts [3,4,12] or researchers [2,6,11,13], but such lists may be time-consuming to generate for a new task, and the attributes in the generated list may not correspond to the optimal set of attributes for the task at hand. A natural alternative is to identify attributes automatically, for example, from textual descriptions of categories. However, this is challenging because the number of potential attributes is large, and evaluating the quality of each potential attribute is expensive.

In this paper, we present a system that automatically discovers a list of attributes for object recognition. As we approach the problem from a computer vision perspective, we are mainly concerned with "visual" attributes that directly relate to the appearance of objects, such as "*red*" or "*metallic*". However, an attribute that relates with visual qualities in general may not be selected by our system if it does not help the recognition task, *e.g.* "*metallic*" is not a useful attribute if the recognition task is to classify car brands. In contrast, the word "*fragrant*" does not refer to a visual quality, however due to its indirect correlation to visual features (*e.g.* its link to flowers), it may be selected as a useful attribute for object recognition by the proposed method. In the remainder of this paper we use the term "visual attribute" to refer to any word that may help object recognition from images.

Our main contributions are as follows. Firstly, we introduce two methods to se-

* Corresponding author
Email addresses: caglart@sabanciuniv.edu (Caglar Tirkaz), jacob.eisenstein@cc.gatech.edu (Jacob Eisenstein), mtsezgin@ku.edu.tr (T. Metin Sezgin), berrin@sabanciuniv.edu (Berrin Yanikoglu).

lect words in a text corpus that are likely to refer to visual attributes. One of the methods we propose uses a taxonomy defined over categories and promotes words whose occurrence in textual descriptions of categories is coherent with the given taxonomy. The other method builds upon the previous one and integrates distributional similarity of words into the attribute selection process. Secondly, we propose a way to assess the quality of a candidate word as an attribute for object recognition from images. In the experiments, we provide evaluations of the proposed attribute selection strategies for effectively identifying attributes, in the plant and animal domains, and present the efficacy of the proposed techniques at selecting visual attributes. Furthermore, we analyze the mined attributes semantically and then use them for plant and animal identification tasks.

We use three input sources in the proposed methods: textual descriptions and image samples of categories and a taxonomic organization of the object domain. In the plant identification task, the goal is to identify a plant species from the visual appearance of its foliage. We use the plant foliage image dataset provided in ImageCLEF'2012 [1] plant identication task [16]. This dataset contains $9,356$ foliage images of 122 species of which $6,689$ are for training and $2,667$ are for testing. For this dataset, we mine a set of text documents containing encyclopedic information on categories from the web, using Wikipedia [2], Encyclopedia of Life [3] and the Uconn Plant Database [4]. For the animal identification task, we use the animals-with-attributes (AwA) database provided by [4] to evaluate our approach. This is a popular database to test attribute-based recognition and zero-shot learning approaches. This dataset contains $30,475$ images of 50 animals where $24,295$ images of the selected 40 animals are used for training and $6,180$ images of the remaining 10 classes are reserved for testing in the zero-shot learning setting. Similar to the plant identification, we mine textual descriptions for each of the 50 animals in that set using Wikipedia and A-Z Animals [5]. In both of the recognition tasks, the challenge is to find the words referring to visual attributes in the mined documents. We test the effectiveness of the automatically selected attributes for recognition in both zero-shot learning and in traditional supervised learning settings. [6]

Our approach consists of two main components. After reviewing related work in 2, we describe a method for assessing the visual quality of a proposed attribute for object recognition in Section 3. The assessment procedure involves training a binary attribute classifier, where the quality of a candidate word depends on the success of

[1] http://www.imageclef.org/2012

[2] http://en.wikipedia.org/wiki/Main_Page

[3] http://eol.org/

[4] http://www.hort.uconn.edu/plants/

[5] http://a-z-animals.com/

[6] All the collected textual descriptions are available online at: https://drive.google.com/file/d/0Bx-64dmWqUHIT09JRGZDOGxPNkk/view?usp=sharing

the attribute classifier. Classification-based attribute selection is effective but computationally expensive; consequently, in Section 4, we propose a set of techniques for *nominating* candidate attributes that are likely to be of high visual quality: we leverage multi-level discriminability across a category taxonomy, and distributional similarity of the words in the text corpus. Our nomination process takes feedback from the visual quality assessment of candidate attributes, making increasingly accurate predictions as it learns more about the types of words that are found to be of high quality. Once the set of attributes is determined, in Section 5, we illustrate how the selected attributes can be used for object recognition in two different settings. Finally, in the experiments section (Section 6) we present experiments to compare attribute selection strategies. Then, the selected attributes are used for classification of categories in two challenging recognition tasks.

2 Related Work

Although most of the literature on attribute-centric recognition focuses on working with a predetermined list of attributes, a few alternatives propose methods to select attributes interactively [14,15] or automatically [8,9]. The interactive methods first identify local image patches that are important for recognition and then use human supervision to check whether or not these patches refer to attributes. Unlike these methods, we would like to take advantage of a text corpus and select attributes automatically. Berg *et al.* [8] also use a text corpus, but instead of learning which attributes are valuable for recognition from text using textual features, they iteratively test the most frequent words in the corpus to find attributes. We show that an intelligently guided search identifies effective attributes much more rapidly.

There is also previous work in the literature to identify words referring to visual characteristics [17,18,19]. In [17] Barnard and Yanai fit a Gaussian mixture model to image regions and determine the “visualness” of a word based on the entropy of the distribution. Boiy *et al.* [18] mine words having visual information using corpus-based association techniques where words appearing more in the texts of visual corpus rather than non-visual corpus are selected. In [19] authors use several strategies to mine visual text from large text corpora. First, they generate a graph between adjectives based on distributional similarity and apply bootstrapping to select visual nouns and adjectives. Second they construct a bipartite graph between visually descriptive words and their arguments and use label propagation to extend the list of visual words. Finally, they integrate visual features to improve performance. In comparison, we propose methods that utilize a taxonomy over categories and distributional similarity of words to automatically discover attributes of categories that are likely to refer to visual characteristics.

Another related work [20] involves finding discriminative codes for individual images rather than for categories. In their work Rastegari *et al.* create a system to en-code each image with a binary code to balance discrimination between categories and learnability of the individual attribute classifiers. Although there is no direct semantic mapping of the discovered codes, they achieve state-of-the-art recognition results on Caltech256 [21] and ImageNet [22] databases. In contrast to their work, we define attributes on the category level and rely on a text corpus to automatically mine semantically meaningful and discriminative attributes.

In [23], Yu *et al*. design a category-attribute matrix on the *known* categories where they balance the separation of the categories while also considering the learnability of the attribute classifiers. In order to perform zero-shot learning, they use human supervision to create a similarity matrix between the novel and the known categories while using the trained attribute classifiers. While we also design a category-attribute matrix, we use no human-supervision in the process other than supplying the readily-available taxonomy on the categories. Moreover, since we use a text corpus to mine the attributes, the attributes we discover can be directly mapped to semantically meaningful units.

The most similar prior work to ours is [9], where Rohrbach *et al*. use state-of-the-art natural language processing techniques and provide experiments for several linguistic knowledge bases for mining attributes. However, there are key differences. Rohrbach *et al*. consider PART-OF relations encoded in WordNet [24]. In contrast, we mine attributes using a taxonomy defined on the categories and consider the whole text so our method can discover attributes referring to color or context that cannot be explained with PART-OF relations. Furthermore, expanding the set of candidate attributes to all words requires accurate nomination heuristics; in our approach, we provide this by leveraging the object category taxonomy and distributional similarity.

Object classification using a taxonomy has also been studied before. In [25] Griffin and Perona describe a way to automatically learn hierarchical relationships between images of categories and use this taxonomy in the recognition task. Deng *et al*. [26] show that there is a correlation between the structure of the semantic hierarchy of the WordNet and visual confusion between the categories. They present a cost function based on the WordNet hierarchy for classification of 10000 categories and show that it produces more semantically meaningful classification results. By defining a taxonomy over categories, Binder *et al*. [27] train an ensemble of local SVMs on various levels of the taxonomy and use trained classifiers in the recognition task. In contrast to previous lines of work that utilize a taxonomy to improve speed and recognition accuracy from images, we rely on the readily available taxonomy of life to discover attributes of categories in a text corpus that help object recognition.

Finally, in [28], a unimodal topic model that integrates textual and image features is built for the tasks of computing word association and similarity. More recently, in [29] the authors combine visual attribute classifiers with text-based distributional

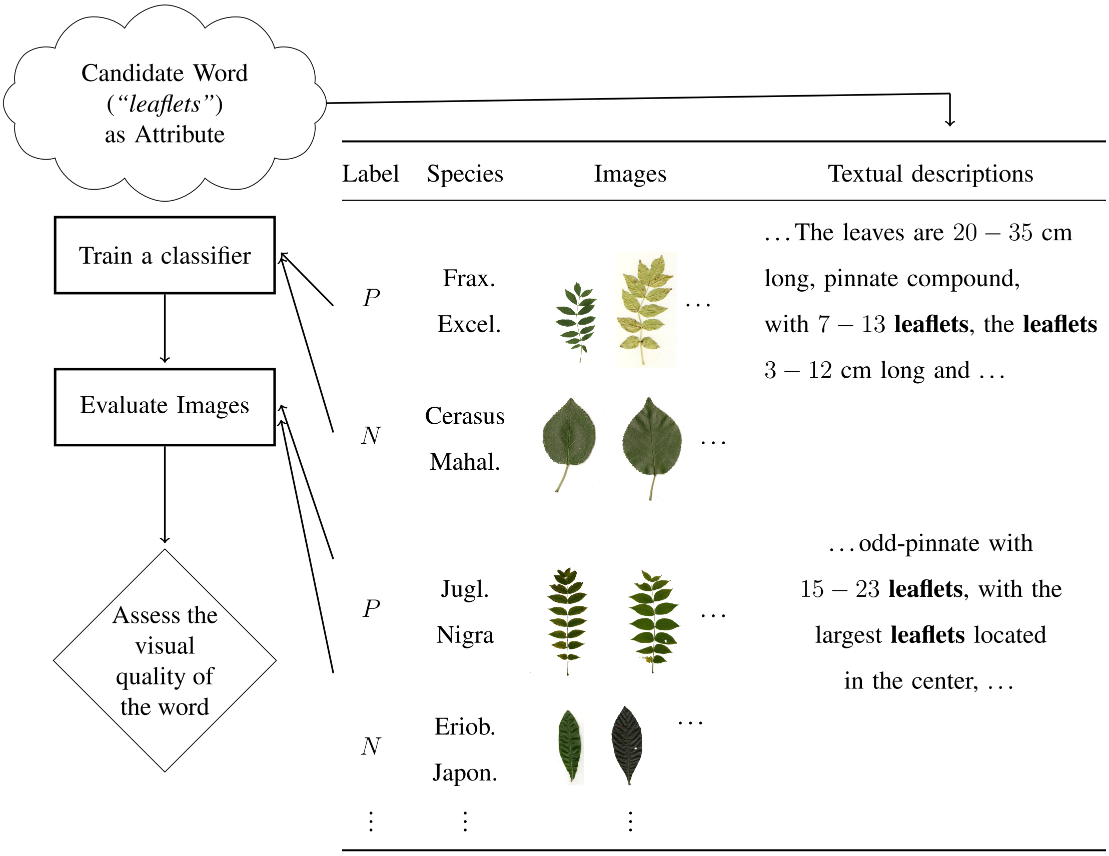
Candidate Word
(*"leaflets"*)
as Attribute
Train a classifier
Evaluate Images
Assess the visual quality of the word
Label
Species
Images
Textual descriptions
P
Frax. Excel.
. . . The leaves are 20 – 35 cm long, pinnate compound, with 7 – 13 **leaflets**, the **leaflets** 3 – 12 cm long and . . .
N
Cerasus Mahal.
P
Jugl. Nigra
. . . odd-pinnate with 15 – 23 **leaflets**, with the largest **leaflets** located in the center, . . .
N
Eriob. Japon.

Fig. 1. The flow for how the visual quality of a candidate word is assessed. Each category has a set of images and textual descriptions. Given a candidate word, each category is associated with a positive (P) or negative (N) label for this candidate, using its textual descriptions (This is unlike previous works [20,30,8] where instances are associated with labels). Images of half of the P and N categories are used to train an attribute classifier and the classifier is evaluated on the remaining images. The candidate word is assessed based on the classifier responses on the evaluation images.

models for finding word associations. In both of these papers, improved results are obtained when textual and image features are used together. However, these lines of work aim to ground natural language semantics in visual features, rather than using language to improve object recognition from images.

3 Assessing the Visual Quality of Attributes

In the textual description of a category such as a plant or animal, only a very small fraction of words refer to attributes — such as *"legged"*, *"green"*, *"big"*, *"ugly"* or *"sharp"*. Moreover, some of these words refer to very high level qualities (*e.g. "ugly"*) that may not be robustly detectable using automatic methods. Also, attributes that are beneficial might change depending on the recognition task. For example, the word *"green"* can be used as an attribute for various tasks, but it is not a useful attribute for plant identification using images of foliage, as most plant foliage is green.

In this section, we present a method to test whether a given candidate word refers to an attribute that can be recognized using visual features. The method is based on the assumption that a word denoting an attribute of an object category will appear frequently in its description. Furthermore, we require from an attribute classifier trained with a proportion of object categories having the attribute versus others, to do a good job in separating *novel* categories having the attribute from others. Figure 1 summarizes our process for assessing whether a candidate word is accepted as an attribute; each step is described in detail in the following subsections.

3.1 Constructing the Training Set

Given a candidate word as an attribute, all object categories in the image collection are *automatically* labeled as either having or not having the attribute, based on their textual descriptions. The categories having the attribute are associated with the label positive (P) while the remaining categories are associated with the negative (N) label. This is unlike previous works [20,30,8] that use instance-attribute rather than category-attribute association.

The category-attribute association is based on the occurrence frequency of the candidate word in the description of that category in the text corpus. Specifically, we compute the mean and standard deviation of the word frequency across all categories, and then associate a category with P if the frequency of the word in descriptions of that category is at least one standard deviation above the mean frequency. All other categories are associated with the label N. We have tried various other methods for finding the category-attribute associations but saw that this method works quite well in practice.

3.2 Training the Attribute Classifier

The P and N categories identified in Section 3.1 are split into training and evaluation sets, using a $50/50$ split. If category is placed in the training set, then all image instances for that category are used for training, and vice versa for the evaluation set. In this way, we avoid attributes specific to a single category that can not be shared across categories. To be selected, an attribute must be recognizable in novel categories that are unseen in the training data.

Using the P and N image instances reserved for training, we train a binary attribute classifier that learns to differentiate between them. In other words, given a candidate attribute (*e.g. "striped"*), the classifier is trained to separate the set of images labeled as having this attribute (zebras, tiger, . . .) or not (lion, panther, . . .) based on their textual description. The trained attribute classifier is used to detect the attribute a in a given image x, with its output interpreted as $p(a|x)$.

Our experiments focus on plant and animal identification tasks. In the plant identification experiments, we use a binary attribute classifier that operates on a set of features extracted from images; specifically histogram of gradients [31], shape context [32] and local curvature on the object boundary. Each attribute classifier is trained using these feature descriptors of the images in the P and N sets reserved for training.

For the experiments we perform on the Animals with Attributes (AwA) dataset, we train the attribute classifiers using the pre-computed descriptors (color, local self similarity, oriented gradients, rgSIFT, SIFT and SURF histograms) supplied by the authors of the AwA database [4] as well an additional feature descriptor extracted using a convolutional neural network. The feature descriptors of 40 training categories specified in [4] are used in training of the attribute classifiers. During assessment of a candidate word, we train a binary linear SVM as the attribute classifier for each feature descriptor.

3.3 Assessing the visual quality

The trained attribute classifier is used to produce $p(a|x)$ for each image instance in the evaluation set. We then analyze the probability distributions obtained by the positive and negative samples in the evaluation set, to determine whether the candidate word is accepted as an attribute. The candidate word is accepted as an attribute if the distribution for the instances of the P categories is significantly greater than the distribution of the instances of the N categories. In order to compare the distributions we perform a t-test at $p < .01$. While precision of the attribute classifier at separating positive and negative instances has been used to assess visual quality of a candidate before [30,8], we preferred to use a t-test which allows us to a detect statistical difference between the distributions of positive and negative instances.

Figure 2 presents the histograms of the probability distributions of the positive and negative evaluation instances for two words that are candidate attributes: *"deciduous"* and *"leaflets"*. The candidate word *"deciduous"* fails the assessment, as the classifier assigns essentially the same distributions to the instances of P and N categories in the evaluation set. In contrast, the distribution for the instances of the P categories for the word *"leaflets"* is skewed to the right compared to that of the instances of N categories. We take this to mean that the word *"leaflets"* corresponds to some visual characteristics.

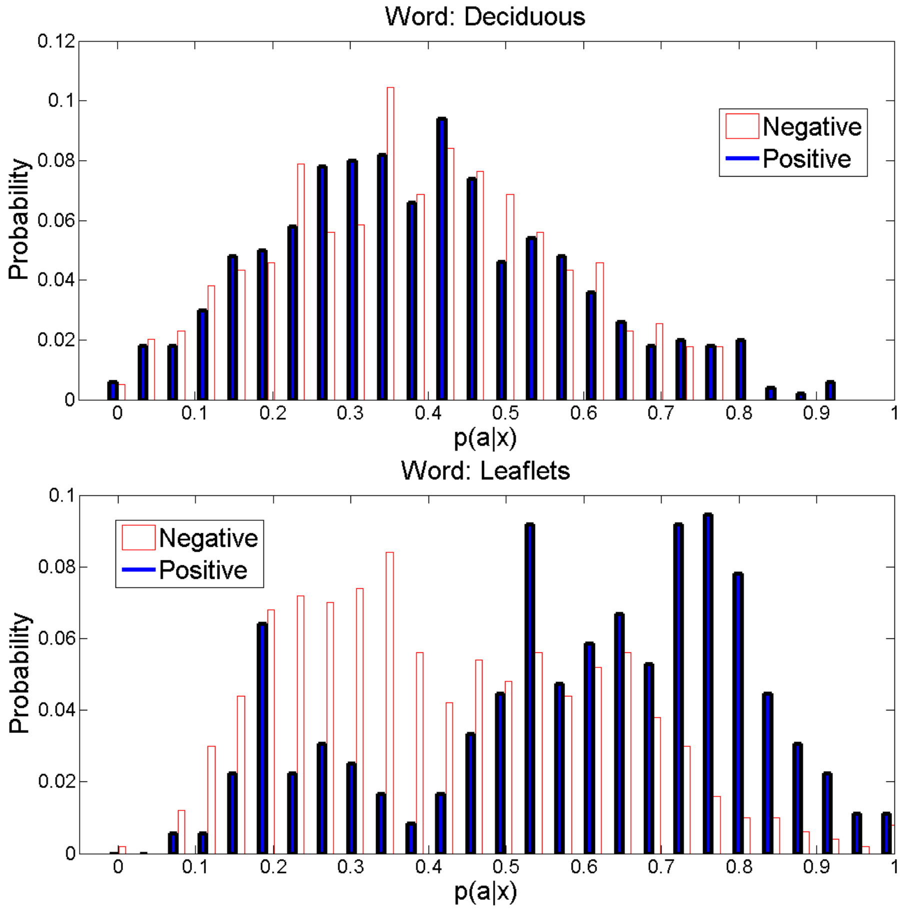

Fig. 2. Two sample words (*"deciduous"* and *"leaflets"*) are assessed for visual quality and the histograms for the attribute classifier predictions are presented. *"Deciduous"* is not accepted because the the classifier predictions are similar for the instances having (positives) and without (negatives) the attribute. On the contrary, *"leaflets"* is accepted because the attribute classifier produces higher probabilities for the instances having the attribute.

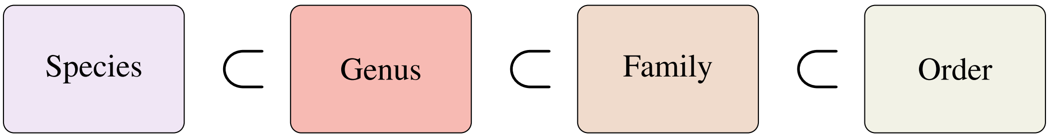

Fig. 3. The lowest four ranks in the hierarchy of biological classification. Each species is also a member of a genus, family, order, *etc*.

4 Attribute Candidate Ranking

The previous section describes an effective procedure for determining if a word is an attribute, but it requires training a binary classification system. Doing so for several thousand candidate attributes can be very time consuming. Hence, we propose techniques to rank the candidates so that the most promising ones are considered first, allowing us to obtain a good set of attributes without iterating through all

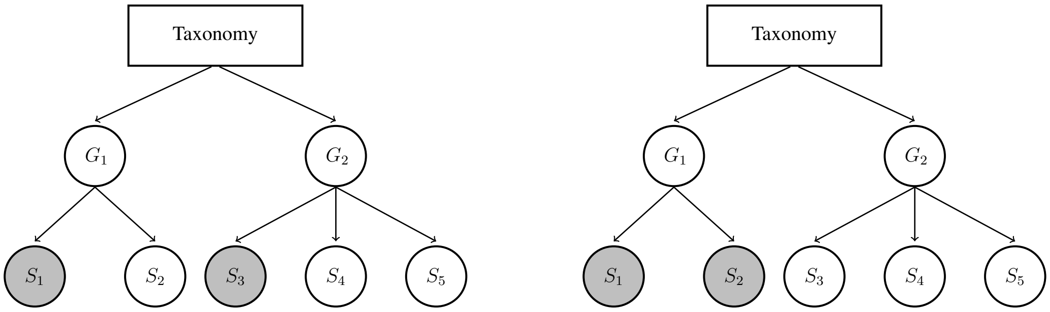

Fig. 4. A toy example where a taxonomy over 5 distinct species and 2 genera is illustrated. We present the division of the species as positive and negative for two candidate words where the nodes that are gray are positive and the others are negative. On the given taxonomy, picking the word on the left splits the species that are in the same genus while the word on the right respects the taxonomy.

possible candidates.

The first technique we propose measures each word's effectiveness as an attribute based on its ability to discriminate not only individual categories, but also higher-order sets of categories as defined by a taxonomy on categories. For example, a good attribute should distinguish not just cars and trucks, but higher-order categories, like vehicles.

The second method we propose organizes all candidate words into a hierarchy, using *distributional similarity*, so that words with similar distributional properties are in similar parts of the hierarchy. As we assess candidate words, we obtain firm evidence about the visual quality of individual words. This information is then propagated to the neighbors of the assessed candidate word in the hierarchy: *i.e.* if a word is found to be a good attribute, then distributionally-similar words will be ranked higher as well, and vice versa. We now discuss these ideas in detail.

4.1 Use of object taxonomy

The living organisms have a taxonomy where organisms are categorized based on similarities or common characteristics. The lowest four ranks of this taxonomy are displayed in Figure 3: each species is associated with a genus, family, and order. Two species in the same genus are therefore more similar to each other than to other species in different genera. Likewise, two species in the same family are more similar to each other compared to other species in different families.

In this work we are working on classification of plants and animals from images and their biological taxonomy is readily available. [7] Assuming this conceptual taxonomy has some correspondence in the visual appearance of each object category, we would like to choose attributes that match it. Such attributes should help us discriminate between highly disparate object classes.

We motivate the use of taxonomy in finding words that are likely to denote visual characteristics with a toy example. Suppose we have a taxonomy defined over 5 species (the leaf nodes) and 2 genera (the middle nodes), as shown in Figure 4 and we have a dictionary of 2 words. We would like to pick one of the words as a candidate attribute and the task is deciding which one to pick. As described in Section 3 each category (species) is associated with either a positive (gray nodes) or negative (white nodes) label depending on the occurrence frequency of the word in textual descriptions of the category. Now, consider the first word that creates the labeling on the left. This word creates a positive set using categories S_1 and S_3, but these categories belong to different genera. In contrast, the second word that generates the tree on the right side of the figure, illustrates a candidate that respects the taxonomic organization of categories. We hypothesize that words that conform to the taxonomy, such as the second word, will be more likely to have meaningful visual properties, and the results in Section 6 bear this intuition out. We now describe a ranking procedure that will favor such words.

Formally, suppose we are given a set of categories $S = \{S_1, S_2, \ldots, S_M\}$, where each category is represented by a set of text documents $S_i = \left\{t_1^i, t_2^i, \ldots, t_{N_i}^i\right\}$. Assume further that each document has a vector space representation based on tf-idf (term frequency inverse document frequency [33]), where the length of the representation is the same as the dictionary size. Finally, denote by d_{ij} a parametric distance between a pair of text documents t_i and t_j. We will parametrize the distance using a weight vector on the words; words whose discrimination pattern is consistent with the category taxonomy will get higher weights. We pose a constrained optimization problem for this purpose.

For concreteness, we focus on the recognition tasks where the relevant groups are species (S), Genera (G), and Families (F). We pose the following constraints:

$$\begin{aligned}
d_{ij} + 1 &\leq d_{kj} \qquad \forall t_i, t_j \in S_I \text{ and } \forall t_k \notin S_I \\
d_{ij} + 1 &\leq d_{kj} \qquad \forall t_i, t_j \in G_I \text{ and } \forall t_k \notin G_I \\
d_{ij} + 1 &\leq d_{kj} \qquad \forall t_i, t_j \in F_I \text{ and } \forall t_k \notin F_I
\end{aligned} \tag{1}$$

The first constraint states that two documents of the same species should be closer to each other than to documents of the remaining species. Similarly, the second constraint states that two documents belonging to the species of the same genus

[7] The taxonomies of plants and animals we use are available online at: https://drive.google.com/file/d/0Bx-64dmWqUHISHhjbWYwOXUxZFk/view?usp=sharing

should be closer to each other compared to the documents of the remaining genera. The third constraint enforces the same condition at the level of families.

Now suppose $t_i \in S_I$, we define:

$$\begin{aligned}
\boldsymbol{\delta}_{ij} &= |f_{\text{tf-idf}}(t_i) - f_{\text{tf-idf}}(t_j)| \\
d_{ij} &= \langle \boldsymbol{w}_I, \boldsymbol{\delta}_{ij} \rangle
\end{aligned} \tag{2}$$

where $f_{\text{tf-idf}}$ is a function computing the vector of tf-idf values of the dictionary words for its input; $\boldsymbol{\delta}_{ij}$ is the absolute difference vector between the tf-idf vectors of t_i and t_j, $\boldsymbol{w}_I$ is the weight vector for the object category S_I and $\boldsymbol{w}_I$ is the weight vector for the object category S_I and $\langle, \rangle$ denotes dot product.

Our procedure for learning the weights is inspired by the metric learning approach of Frome *et al.* [34]. Suppose $t_i \in S_I$, $t_k \in S_K$; denote by $\boldsymbol{w}_{IK}$ the concatenation of the weight vectors for S_I and S_K; let $\boldsymbol{x}_{ijk} = [-\boldsymbol{\delta}_{ij}\boldsymbol{\delta}_{kj}]$, the concatenation of $\boldsymbol{\delta}_{ij}$ negated and $\boldsymbol{\delta}_{kj}$. Then, the first constraint in Eq. 1 can be re-written as $\langle \boldsymbol{w}_{IK}, \boldsymbol{x}_{ijk} \rangle \geq 1$. We denote the next two constraints that are defined on the genus and family levels similarly, and we define a loss function over all constraints and all triplets:

$$\sum_{Constraints} \sum_{ijk} \lfloor 1 - \langle \boldsymbol{w}_{IK}, \boldsymbol{x}_{ijk} \rangle \rfloor_+ \tag{3}$$

where $\lfloor z \rfloor_+$ denotes the thresholding function $max(0, z)$. After adding a regularization penalty, the objective function becomes:

$$\frac{1}{2} \|\boldsymbol{w}\|^2 + C \sum_{Constraints} \sum_{ijk} \lfloor 1 - \langle \boldsymbol{w}_{IK}, \boldsymbol{x}_{ijk} \rangle \rfloor_+ \tag{4}$$

where $\boldsymbol{w}$ is the concatenation of the weight vectors of all categories and C is the regularization parameter. We select C using cross validation, using the value that minimizes the loss function on held-out data.

Once the regularization parameter is selected, the optimization problem can be solved using a row-action method similar to [34]. The weight vector of each species is updated by iterating over all constraints, and the triplets associated with them as follows:

$$\begin{aligned}
\alpha_{ijk} &\leftarrow \left\lfloor \frac{1 - \langle \boldsymbol{w}_{IK}, \boldsymbol{x}_{ijk} \rangle}{\|\boldsymbol{x}_{ijk}\|^2} + \alpha_{ijk} \right\rfloor_{[0,C]} \\
\boldsymbol{w}_I &\leftarrow \boldsymbol{w}_I - \sum_{ijk} \alpha_{ijk} \boldsymbol{\delta}_{ij} \\
\boldsymbol{w}_K &\leftarrow \boldsymbol{w}_K + \sum_{ijk} \alpha_{ijk} \boldsymbol{\delta}_{kj}
\end{aligned} \tag{5}$$

where $\lfloor z \rfloor_{[0,C]}$ denote the function $min(C, max(z, 0))$. We continue iterating until the change in weights falls below a threshold. Unlike [34], where the authors define constraints on the instances of categories, we have constraints defined over the taxonomy of categories and we do not enforce non-negative weights since words with negative weights can also indicate potential attributes.

Since solving this constrained optimization problem relies on generation of document triplets, let us elaborate on the number of triplets that will be generated. Consider our first constraint in Eq. 1, denote the number of categories by M, and the number of documents per category by N. Then the number of triplets that will be generated is $O(M^2N^3)$. Working with that many triplets might be infeasible, so when generating triplets we select only a subset of all possible triplets. While forming triplets of the form $\langle ijk \rangle$, we select a document k only if it is in the R nearest neighbors of i based on its tf-idf representation, reducing the number of triplets to $O(MN^2R)$. We set $R = 50$ during all experiments. In the experiments, we consider the most frequent $2,000$ non-stoplist words in the text corpus to create tf-idf representations, thus the weight vector of each category is of length $2,000$.

Once the weight vectors for all categories are learned, we assign a weight for each word in our vocabulary by computing the mean absolute weight of the word over the categories. Words that are shared among categories and obey the category taxonomy will get higher weights, and therefore will be considered first as potential attributes.

We have implemented the described optimization using C#. [8] It takes 31 and 276 seconds for the optimization to be completed on the AwA and the ImageClef datasets respectively using a computer having Intel i7 2.00GHz processor.

4.2 Integrating distributional similarity

While taxonomic discriminability is a powerful feature for predicting whether a word will be a useful attribute in general, it ignores the word's meaning and considers only co-occurrence with category-labeled documents. We hypothesize that if a word has high value as an attribute, then words with similar meanings should also have high value, and vice versa. Word meanings — known as *lexical semantics* in linguistics — can be difficult to pin down. This is particularly true in technical domains such as plant/animal biology, where annotated resources such as WordNet may have low coverage. We follow an alternative, data-driven approach to lexical semantics, motivated by the *distributional hypothesis*, which asserts that words with similar meanings tend to appear in similar linguistic contexts [35]. This bears directly on our problem of identifying words that are attributes. For example, if the word *"lobed"* is found to be a visual attribute, then words that appear in similar contexts to *"lobed"* (*e.g.*, *"serrated"*, *"oblong"*) are also likely to be attributes and

[8] Available online at: https://drive.google.com/file/d/0Bx-64dmWqUHIVzJ3djlCUDQxRjQ/view?usp=sharing

should be prioritized for testing. Conversely, if a word such as *"Western"* is found not to be a visual attribute, then words that appear in similar contexts to *"Western"* (*e.g.*, *"Eastern"*, *"Chinese"*) are unlikely to be attributes (despite having high taxonomic discriminability), and can be tested later.

We use a hierarchical clustering of words to capture word similarity. Each word is represented by a vector of frequencies, where the vector of a word represents its co-occurrence with neighboring words [36,37]. In order to construct the co-occurrence vector of a word we use a context window of five words on either side of the target word where the vector dimensions are constituted by the most frequent 2000 non-stoplist words in the text corpus. Finally, we apply a graph clustering algorithm [38] to the word representations, obtaining a word dendrogram (Figure 5) [9] .

To see how word similarity can help, consider the toy example shown in Figure 5. Weights for each word are estimated to be $w_1 = w_2 = 1, w_3 = w_4 = w_5 = 0.8$, using the procedure described in Section 4.1. Initially, we pick W_1, which is tied for the highest weight. However, suppose that W_1 fails the visual quality assessment. The word W_2 is tied for the next highest weight, but it is distributionally similar to W_1. Since W_1 is not selected as an attribute, we downweight our prediction for W_2, and try another word instead. Note that this idea applies to any hierarchical clustering of words, so we could use other word features instead of co-occurrence-based features to cluster the words and use the same idea.

The key advantage of the word dendrogram is two-fold. Firstly, it allows us to integrate distributional similarity into the candidate selection process. Secondly, by propagating information about visual quality assessment between related words, better candidates can be selected as the candidate selection progresses. Initially, the dendrogram helps to smooth the learned weights using taxonomy computed in Section 4.1. As we assess candidate words (Section 3), these initial estimates are replaced with hard evidence. By propagating this evidence through the word dendrogram, we can avoid wasting effort assessing unpromising words whose near neighbors have already failed assessment.

We operationalize this idea in the framework of belief propagation [39,40], treating the word dendrogram as a large graphical model containing binary random variables x_e for both words and word clusters. We treat visual quality assessment result as a latent variable x_e for node e, and perform inference on the distribution $P(\boldsymbol{x}_{1:E}|\boldsymbol{w}_{1:E})$, where $\boldsymbol{w}_{1:E}$ is the local evidence for all nodes $1 : E$. This probability is proportional to $P(\boldsymbol{w}_{1:E}|\boldsymbol{x}_{1:E})P(\boldsymbol{x}_{1:E})$, where the first term indicates the likelihood and the second term indicates the prior. After normalizing the learned weights for words between 0 and 1, the likelihood for each node is set equal to the normalized weight of the respective word. The prior is governed by a compatibility

[9] The word dendrograms for the Awa and the ImageClef datasets are available online at: https://drive.google.com/file/d/0Bx-64dmWqUHIcTN2eEthQnBSMDA/view?usp=sharing

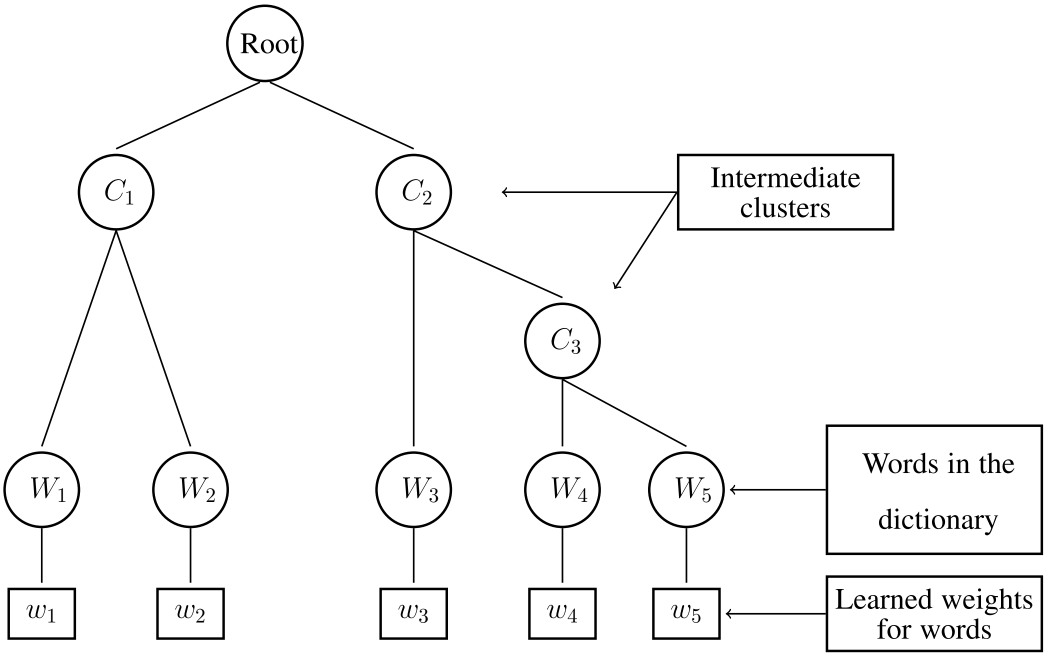

Fig. 5. The graphical model based on the hierarchical clustering of 5 words. The learned weights for words are used to compute likelihoods of nodes being effective visual words and each edge between nodes is governed by a compatibility function favoring the same visual effectiveness for neighbors.

function, and in our experiments for neighboring nodes $x_e^{'}, x_e$ we define:

$$P(x_e^{'}|x_e) = \frac{1}{2}\begin{bmatrix} \alpha & (1-\alpha) \\ (1-\alpha) & \alpha \end{bmatrix}. \tag{6}$$

where α is a constant greater than 0.5. We experimented with various values of α and set it 0.6 in all experiments, obtaining the best speed at selecting visual attributes.. For words whose visual quality has been assessed, we can *clamp* x_e to the true value. By applying belief propagation, information from these clamped nodes is propagated to neighboring nodes, with diminishing influence as we move across the dendrogram.

We use Alg. 1 for nominating and assessing candidate words. With this strategy, we adaptively search the set of possible attributes, focusing our initial efforts on the most promising words while also taking into account the assessed words during later iterations.

Algorithm 1 Procedure for nominating and assessing attributes.

```
function PROPOSE ATTRIBUTES($W$)
    Estimate taxonomy-based discriminability for all words (Section 4.1)
    Build a dendrogram based on distributional similarity (Section 4.2)
    while there are untested words do
        Apply belief propagation to estimate $P(x_e|\boldsymbol{w})$ for all words
        Assess the visual quality of the top-scoring untested word $e$ (Section 3), and clamp $x_e$
    end while
end function
```

5 Attribute Based Classification

After generating a list of attributes as described so far, the discovered attributes can be used for object classification. In order to use the selected attributes in object recognition, we train attribute classifiers (one per attribute), such that each classifier is trained to discriminate between the images of positive and negative classes (see Section 3.1) corresponding to that attribute. We use the attribute classifiers for supervised attribute-based classification of plants and zero-shot learning of animals. During the direct similarity-based classification experiments for zero-shot learning, we utilize the learned weight vectors of categories in Section 4.1. Below, we discuss these approaches.

5.1 Supervised Attribute-based Classification

Using attributes we apply the traditional supervised learning paradigm to recognize test images. In order to extract training features, we apply the attribute classifiers on all the training images of each category. For each image we create a feature vector that is the same length as the the number of attributes containing attribute classifier responses. Next, an SVM classifier is trained [41] using the extracted features. During testing, we extract the feature vector from a test instance (image) by applying the attribute classifiers and concatenating the classifier responses. The extracted feature vector is then classified by the trained SVM classifier. Supervised attribute-based classification offers advantages over traditional classifiers, since the feature vector is compact and the underlying representation is interpretable to humans.

5.2 Zero-Shot Learning

In zero-shot learning, the aim is to learn to recognize novel categories using only their textual descriptions. For instance, having trained attribute classifiers *has-a-torso* and *has-a-tail*, zero-shot learning enables us to label an image of a centaur correctly as having a torso and tail even though the attribute classifiers have never seen an image of a centaur. We use two methods for zero-shot learning: attribute-based recognition and direct-similarity based recognition. These two approaches differ in the way they describe unseen categories. For instance, an unseen category such as *leopard* will be described as living in **Africa** and being a member of the **feline** family by attribute-based recognition whereas direct similarity-based recognition will describe a leopard as being similar to a **lion** and a **bobcat** in appearance.

5.2.1 Attribute-based recognition

For attribute-based recognition, we create a classifier per attribute, without using any example images from the testing categories. In order to label images of a category we rely on the text corpus and use the found category-attribute associations based on the attribute as in Section 3. During testing, we apply the attribute classifiers to a test image and obtain the probability of each attribute existing in the given test image, *i.e.* $p(a_1|x), p(a_2|x), \ldots$. The final classification of a test instance is performed using direct attribute prediction (DAP) proposed by Lampert *et al.* [4].

Using the DAP method, the posterior of a test category given a test image is calculated using:

$$p(z|x) = \sum_{a \in \{0,1\}^M} p(z|a)p(a|x) = \frac{p(z)}{p(a^z)} \prod_{m=1}^{M} p\left(a_m^z|x\right) \tag{7}$$

where z is the test category, a^z is the category-attribute associations for the category and M is the number of attributes. During testing, identical category priors $p(z)$ are assumed for each category. $p(a)$ is computed using a factorial distribution, $p(a) = \prod_{m=1}^{M} p(a_m)$ where the attribute priors are approximated using empirical means over the training categories, $p(a_m) = \frac{1}{K} \sum_{k=1}^{K} a_m^{y_k}$. This leads to the following MAP prediction f, that assigns the best output category from all test categories $z_1, \ldots, z_L$ to a test image x:

$$f(x) = \underset{l=1,\ldots,L}{\arg\max} \prod_{m=1}^{M} \frac{p(a_m^{z_l}|x)}{p(a_m^{z_l})} \tag{8}$$

5.2.2 Direct similarity-based recognition

For direct similarity-based recognition[9], we first train classifiers to separate each training category from others. Next, the semantic similarity between each testing and training category is computed. Using direct similarity, the posterior of a test category given a test image is calculated using:

$$p(z|x) = \prod_{k=1}^{K} \left(\frac{p(y_k|x)}{p(y_k)} \right)^{y_k^z} \tag{9}$$

where $p(y_k|x)$ is the likelihood of a test image belonging to the training category y_k, and y_k^z is the computed semantic similarity between y_k and the testing category z. This is essentially applying the DAP method with $M = K$ (40 in case of the AwA dataset) attributes where each attribute classifier is trained using the instances of a single training category.

In order to compute the semantic similarity between the training and testing categories, we use the computed weight vectors of categories in Section 4.1 using:

$$y_k^z = \frac{\langle \boldsymbol{w_k}, \boldsymbol{w_z} \rangle}{\|\boldsymbol{w_k}\| \, \|\boldsymbol{w_z}\|} \tag{10}$$

6 Experiments

We performed experiments to analyze the success of the proposed methods at selecting attributes for plant and animal identification tasks in Section 6.1. Next, we use the attributes selected by the best candidate selection method in object recognition tasks in Section 6.2: We then group the selected attributes with respect to their semantics in Section 6.3. We use the selected attributes for supervised attribute-based classification of plants and we perform zero-shot learning of animals. Below, we explain these experiments in detail.

6.1 Comparison of Methods for Attribute Selection

We compare four different strategies for proposing candidate words as visual attributes:

- *Method-1*: Iteratively selecting most frequently occurring words in the dictionary, as in [8], which is our baseline.
- *Method-2*: Estimating word weights using constraints only on the species (category) level and selecting words with highest weights iteratively. This corresponds to applying the approach from Section 4.1, using only the first constraint in Eq. 1.
- *Method-3*: Estimating word weights using taxonomy constraints and selecting words with highest weights iteratively. This corresponds to applying the approach from Section 4.1, but not applying belief propagation across the word dendrogram.

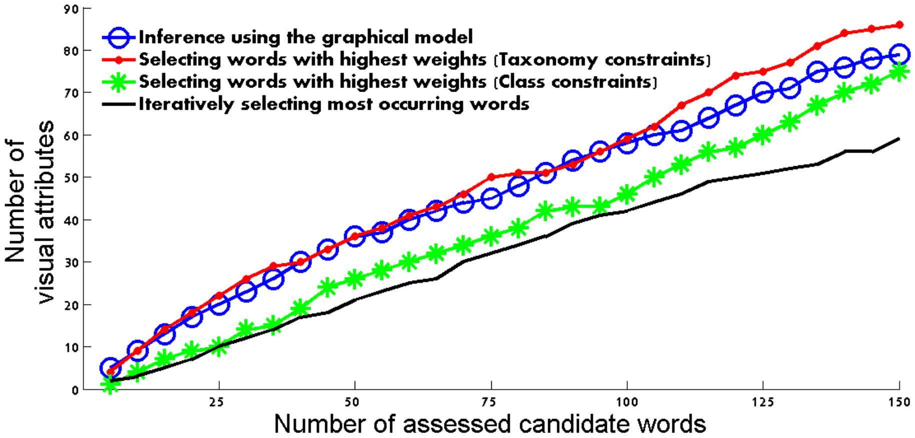

Fig. 6. The comparison of number of visual attributes as a function of the number of candidates using various candidate word selection strategies for the plant identification task. All candidate selection strategies perform better than our baseline (black line) of iteratively selecting the most occurring words.

- *Method-4*: Treating visual quality as a latent variable, and applying belief propagation across the automatically-constructed word dendrogram, as described in Section 4.2.

These methods are evaluated in terms of precision at selecting visual attributes and the resulting recognition accuracy, for the plant and animal identification tasks.

While methods 1 and 2 do not require any information other than textual documents describing categories, *method-3* is applicable when a taxonomy defined over the categories is available and *method-4* requires a word dendrogram to be constructed over the words in the text corpus. In our experiments, we use the weights learned by *method-3* to initialize the belief propagation procedure of *method-4*. However, *method-4* can also be used without a taxonomy, by using *method-2* to initialize the belief propagation. In summary, among the compared word selection strategies, methods 1, 2 and 4 are viable options in the absence of a taxonomy on categories.

6.1.1 Comparison of precisions

We compare the word selection strategies based on the number of candidates they need to assess before acquiring a fixed number of visual attributes. In Figure 6 the performance of each word selection strategy at finding visual attributes is illustrated. The x axis in the figure is the number of candidate attributes that are assessed, and the y axis is the number of the visual attributes among the candidates. Various points on this figure are also presented in Table 1 for comparison. For instance using *method-4*, in order to obtain 60 visual attributes, 104 candidates need to be assessed.

Table 1
Comparison of the number of candidates required to select M attributes (Smaller is better).

	Candidate word selection methods			
	Method-1	*Method-2*	*Method-3*	*Method-4*
$M = 20$	45	41	**32**	35
$M = 40$	80	80	69	**68**
$M = 60$	130	115	**102**	104

Table 2
Comparison of the number of candidates required to select 25 visual attributes on the AwA dataset for each word selection strategy and for each provided feature descriptor (Smaller is better).

Feature	Candidate word selection methods			
descriptor	*Method-1*	*Method-2*	*Method-3*	*Method-4*
Color histogram	45	33	**31**	35
Local self similarity histogram	43	35	**33**	**33**
Histogram of oriented gradients	46	35	**33**	41
rgSIFT	46	35	**33**	36
SIFT	51	45	**33**	35
SURF	49	40	40	**36**

We have used the output of the visual quality assessment as visualness ground-truth similar to [20,30,8]. The reason for this is three-fold: i) Humans/experts do not have a clear agreement as to which attributes are visual, ii) As we are looking for visual attributes that are useful for recognition, the decision becomes even more complicated, and iii) We found that the output of the proposed visual quality assessment correlates with human labels.

We compare the four candidate word selection methods in terms of the number of required candidates in the animal identification task for each feature descriptor. Specifically, we require each method to select 25 visual attributes and compare total number of assessed candidates for each method. The attribute selection results are presented in Table 2. For instance, using the SIFT descriptors and *method-4*, 35 candidates are assessed for selecting 25 visual attributes.

Compared to the baseline method of selecting the most occurring words in the text corpus (*method*-1), using *method*-2 (category-level constraints for learning the weights of words) results in faster, *i.e.* more precise, mining of visual attributes. In our experiments, for both animal and plant identification tasks, *method*-2 is favorable to *method*-1. Thus, we conclude that using category level constraints is useful for automatic attribute selection.

By integrating taxonomy constraints over *method*-2, we observe a significant performance gain using *method*-3. In fact, *method*-3 performs the best in terms of speed at selecting visual attributes in most of our experiments. These results show that having constraints using the taxonomy is crucial to identify candidate words that are likely to be visual attributes.

Adaptive word selection using belief propagation, *method*-4, builds upon *method*-3 and incorporates information about word semantics and visual quality assessment into the word selection procedure, through the dendrogram of word similarity. We observe that, while performing competitively, *method*-4 fails to improve over *method*-3 in terms of speed in most of our experiments. This might be due to the fact that the graphical model needs to explore more words before exploiting the information about word semantics and visual quality assessment results.

Assessing a single candidate attribute (cross validation to find SVM parameters, training the classifier and testing on the evaluation set) on the ImageClef dataset takes around a minute while it takes around nine minutes on the AwA dataset, using a computer having Intel i7 2.00GHz processor. *Method-4* requires assessment of 216/104 candidates before finding 25(*6)/60 attributes for the animal and plant identification tasks, respectively. On the other hand, *method-1* requires assessment of 280/130 candidates for the same task. So, if *method-4* is used to mine visual attributes instead of *method-1*, the total time gained for the plant identification task is 26 minutes whereas it is 576 minutes for the animal identification task. These time gains are significant even for relatively small datasets such as AwA and ImageClef.

6.1.2 Comparison of recognition accuracies

After the attributes are selected, we use the visual attributes in classification experiments. In this section we compare the attribute-based recognition accuracies of word selection strategies and in Section 6.2 we compare a word selection strategy with state of the art methods.

The resulting recognition accuracies for the plant and animal identification tasks are presented in Table 3. The plant and animal identification tasks are supervised attribute-based classification (see Section 5.1) and zero-shot learning (see Section 5.2) tasks respectively. We see that the recognition accuracy for both plant and

Table 3
The recognition accuracies of the evaluated methods for plant and animal identification.

	Candidate word selection methods			
	Method-1	*Method-2*	*Method-3*	*Method-4*
Plant Identification (%)	53.0	52.5	54.2	**54.6**
Animal Identification (%)	26.8	27.1	28.9	**30.4**

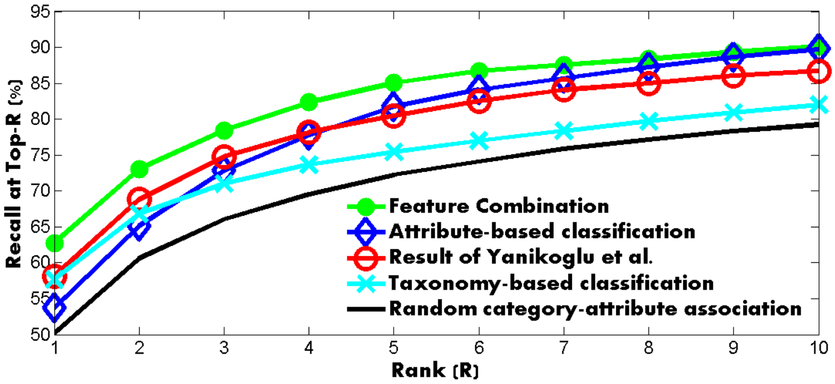

Fig. 7. Comparison of the recognition accuracies of the tested methods on the ImageClef'2012 dataset, for the plant identification task at various ranks.

animal identification is the best when using the attributes selected by *method*-4.

In summary, we conclude that *method*-4 selects the best attributes in terms of recognition accuracy while also performing competitively in terms of precision. As we note note above, we think that the lower precision of *method*-4 with respect to *method*-3 might be due to the initial smoothing of the learned weights and the number of trials&errors required to be able to propagate the evidence acquired from visual quality assessment. However, since *method*-4 also takes into consideration the distributional similarity of words, the visual attributes mined by *method*-4 are of higher quality for attribute-based recognition purposes.

6.2 Classification Experiments

Below, we use the attributes discovered by *method*-4 for attribute-based recognition experiments and the weight vectors leaned by *method*-3 in direct similarity-based recognition experiments. We compare our attribute-based recognition system with the state-of-the-art methods in the two recognition tasks.

6.2.1 Supervised Attribute-based Classification for Plant Identification

We use the selected attributes to perform supervised attribute-based classification of plants. In the plant identification task, retrieval performance of a system is also important, so we present recalls for varying values of rank in Figure 7. For a specific rank, R, a classification decision is accepted as valid if the correct label is in the first Top-R guesses.

We compared five methods in the plant recognition experiment:

(1) Supervised Attribute-based classification described in Section 5.1
(2) The system of Yanikoglu *et al.* [42] which had the best results at the ImageClef'2012 plant identification task
(3) Combination of the first two methods at feature level
(4) Taxonomy-based classification where given a plant taxonomic hierarchy with M nodes train M one-versus-all classifiers
(5) Randomly creating a category-attribute association matrix and using it for attribute-based recognition

The results of the combined system are produced by an SVM classifier that is trained and tested on the concatenation of the attribute-based features (of length 60) with the features used by Yanikoglu *et al.* (of length 142) for each instance. The taxonomy-based classification approach creates classifiers to separate each species, genus and family from others in the taxonomic classification of plants to perform recognition. In total we created 235 classifiers for taxonomy-based classification. In our final experiment we randomly create a category-attribute association matrix with 60 attributes and train classifiers for each attribute. We repeat the same experiment 5 times and present the mean accuracy which is our baseline.

According to the results, randomly creating a category-attribute association matrix to perform attribute-based classification yields the lowest accuracies as expected. Using the taxonomic hierarchy of plants directly without using a text corpus improves over the baseline. However, this method requires the training of more classifiers (235 compared to 60), does not create semantically meaningful attributes and fails to generalize as well the proposed method. Attribute-based classification using the attributes and the corresponding category-attribute associations we mine not only improves over the baseline it also generalizes well to the recognition task. For $R > 4$, the recognition results we obtain using attributes are better than the results Yanikoglu *et al.* obtain. Another observation is that the combined system performs the best for all R suggesting that attribute-based features complement low-level features for the plant identification task.

6.2.2 Zero-Shot Learning on Animals with Attributes (AwA) Dataset

We illustrate the recognition performance of our system for animal identification using the AwA dataset. Lampert *et al.* [4] provide 6 feature descriptors for this database and we computed category-attribute associations for each category and for each descriptor. Next, for each category, we concatenated the category-attribute associations of each descriptor to create an extended representation. Thus, after combining all feature descriptors, each category has a representation of length $150 (= 25 * 6)$ during attribute-based recognition experiments. For direct similarity-based recognition experiments, we use the weight vectors of categories to compute semantic similarity between training and testing categories.

As an extension, in addition to the provided 6 feature descriptors of images, we perform recognition experiments using a new feature descriptor. Specifically, we extract a 4096-dimensional feature vector from each image using the Caffe [43] implementation of the convolutional neural network (CNN) described by Krizhevsky *et al.* [44]. These features are computed by forward propagating a mean-subtracted 227×227 RGB image through five convolutional layers and two fully connected layers. Using these state-of-the art features for attribute-based recognition we discover 50 attributes and perform the recognition experiment. During direct similarity-based recognition experiment, we train the classifiers of each category using the new feature descriptors. [10]

We compare our results with the results of Lampert *et al.* [45], Yu *et al.* [23] and Rohrbach *et al.* [9] as presented in Table 4. While Lampert *et al.* use the manually defined 85 attributes and category associations in their experiments, Yu *et al.* utilize a similarity matrix created with human supervision to design attributes, and Rohrbach *et al.* present experiments with the manually defined attributes and mined category associations, with 74 mined attributes and corresponding category associations , and using direct similarity on several knowledge bases. We use 25 attributes (per feature descriptor) using the provided image descriptors, and 50 attributes using the features extracted by the CNN for attribute-based recognition experiments. In order to perform the direct similarity based recognition experiment with the provided feature descriptors, we concatenate the individual descriptors while training the classifiers of training categories. The comparison includes the knowledge base for the best performing strategy as well the results of Rohrbach *et al.* using Wikipedia as the knowledge base. Comparison with Wikipedia as the knowledge base is important since we also rely on encyclopedic information for attribute selection.

The highest recognition accuracies using the provided feature-set for attribute-based recognition are obtained by using manually defined attributes and category

[10] The extracted CNN features for the Awa dataset are available online at: https://drive.google.com/file/d/0Bx-64dmWqUHIVTdwS1QyTXlMT3M/view?usp=sharing

Table 4
Comparison of zero-shot learning accuracies using attributes and direct similarity.

Author & knowledge base	Attribute selection	Category-attribute association	Accuracy (in %)
1. Attribute-based recognition			
Lampert *et al.* [45]	Manual	Manual	42.2
Rohrbach *et al.* [9] Wikipedia	Manual	Automatic	27.0
Rohrbach *et al.* [9] Wikipedia	Automatic	Automatic	19.7
Rohrbach *et al.* [9] Yahoo Img	Automatic	Automatic	23.6
Our method (Provided features)	Automatic	Automatic	30.4
Our method (CNN features)	Automatic	Automatic	45.7
2. Human supervision to compute similarity matrix			
Yu *et al.* [23] 85 attributes	Automatic	Automatic	42.3
Yu *et al.* [23] 200 attributes	Automatic	Automatic	48.3
3. Direct similarity-based recognition			
Rohrbach *et al.* [9] Wikipedia	Automatic	Automatic	33.2
Rohrbach *et al.* [9] Yahoo Img	Automatic	Automatic	35.7
Our method (Provided features)	Automatic	Automatic	41.8
Our method (CNN features)	Automatic	Automatic	59.4

Table 5
Selected attributes for plant identification.

Semantic category	Visual attributes
Groupings of plants	Sorbus, Acer, nutlet, maple, cernuous, Prunus, hawthorn, samara, mulberry, acorn, sumac, alder, birch, poplar, beech, pod
Context for plants	catkin, drupe, hypanthium, gland, bract, corymb, branchlet, corolla, floret
Visual qualities of plants	lobe, rounded, opposite, yellowgreen, yellow, white, glabrous, purple, serrate, vein, egg-shaped, conic, lanceolate, cordate, ovate, reddishbrown, chapped, pubescent, black, cusp, obovate, entire
False positives	female, root, centimeter, half(a), equal, length, narrowly, minutely, bear, rate, capsule, touch, fragrant

associations followed by our approach. Our method automatically discovers the attributes and the category-attribute associations while the accuracy we obtain surpasses the case where the attributes are defined manually and only the category-attribute associations are mined. This shows the superiority of our approach and the importance of using the taxonomy/distributional similarity information for automatic attribute selection. Furthermore, when using the CNN features in the attribute-based recognition experiment, a relative increase of around 50% in the recognition accuracy is achieved with respect to using the provided descriptors resulting in a zero-shot recognition accuracy of 45.7%.

In the direct similarity-based recognition experiment, using the provided features, our method obtains an accuracy of 41.8% which is higher than the accuracy obtained by Rohrbach *et al.* in their experiments. In this setting, our method gets a comparable performance to the accuracies obtained by Lampert *et al.* and Yu *et al.* with 85 attributes. Yu *et al.* report that the accuracy they obtain can be increased up to 48.3% by using more attributes. Using direct similarity and the CNN features, similar to the case of attribute based-recognition, the recognition accuracy we obtain increases up to 59.4%. To our knowledge, this is the best reported zero-shot recognition accuracy on this dataset.

6.3 The Selected Attributes

We present the 60 attributes discovered by *method*-4 for plant identification task in Table 5. We divide the words into 4 groups, based on their semantics. The table

Table 6
Selected attributes for animal identification.

Semantic category	Visual attributes
Groupings of animals	ungulate, cetacean, canine, feline, kitten, cub, shark, jackal
Context for animals	Africa, graze, India, hunt, Waters, Indian, Pacific, sea, Atlantic, ocean, marine, Soviet, livestock, agricultural
Visual qualities of animals	hoof, ivory, giant, fancy
False positives	weightkg, jump, subspecies, trophy, disposition, favored, recover, university, imperativeness, estes, company, prey, tip, production, national, genome, standard, breed, intelligence, originally, owner, sizecm, engender, monk

contains words that may be grouped as:

(1) Groupings of plants that are similar to each other such as *"Acer", "Prunus", "Sorbus", etc.*
(2) Context for plants such as *"catkin", "gland", "branchlet", etc.*
(3) Visual qualities of plants such as *"rounded", "yellow", "lobe", etc.*
(4) False positives such as *"female", "root", "centimeter", etc.*

Note that words that are categorized into groupings of plants, parts of plants and visual qualities indeed refer to high level features that are used in object description by humans/experts. As for the words that are labeled as false positives, we included all words that we could not directly relate to visual attributes useful for object recognition, including generic words (*e.g. "length", "minutely", "centimeter"*) and some others that may only be indirectly correlated with visual features (*e.g. "fragrant"*).

The set of 50 attributes that are selected during our experiments using CNN features on the AwA dataset are presented in Table 6. As before, we divided the visual words into 4 semantic categories that we relate to:

(1) Groupings of animals such as *"cetacean", "feline", "shark", etc.*
(2) Context for animals such as *"Africa", "agricultural", "sea", etc.*
(3) Visual qualities of animals such as *"hoof", "ivory", "fancy"*
(4) False positives such as *"intelligence", "favored", "weightkg", etc.*

We remind that we use the word "visual attribute" to refer to any word that may help in object recognition from images. Indeed, we demonstrate the effectiveness of the selected visual attributes in object recognition, even though only a portion of them relate to truly visual qualities.

7 Conclusion and Discussion

In this paper, we tackle the important problem of automatically mining words referring to visual attributes. Our method assists the laborious attribute selection process and allows us to rapidly apply attribute-centric recognition to various recognition tasks. In order to mine attributes, we use the taxonomy of the domain, sample images and textual descriptions of object categories. We show the utility of our approach in two taks: plant identification and zero-shot learning of animals.

The contributions of this paper include two novel methods for identifying plausible candidates that can be used as attributes. While the first method uses the taxonomy defined on the categories and promotes candidates that conform to the taxonomy, the second method combines taxonomy with a distributional similarity measure. By providing a way to assess the visual quality of candidate words, we create an automatic system that generates a set of attributes for plant and animal identification tasks.

During experiments, we illustrate the utility of integrating taxonomy constraints for candidate word selection via two cases. In the first case we use only species (category) level constraints while in the second one we include taxonomy (species, genus and family) constraints. We show that taking advantage of the taxonomy yields substantially better results. The experiments also show that taxonomy-based distributional similarity of words can be used as a cue for selecting candidate words. By performing inference using the graphical model built on word dendrogram, we can further improve the recognition accuracies.

Plant/animal identification are both challenging problems and we demonstrate the usefulness of the mined set of attributes by using the trained attribute classifiers in both of these tasks. In the plant identification task the attribute classifiers are used for feature extraction in a supervised learning setting; while in animal identification, they are used for zero-shot learning of unseen animal categories.

During the plant identification experiments, we show that using attribute-based features bring performance improvements compared to using only lower-level features. The classification results also highlight better performance of attribute-based features with increasing ranks, while providing a compact representation. The zero-shot learning of animals demonstrate the quality of our attribute selection procedure: our system that automatically selects both attributes and category-attribute associations, achieves better recognition results than the state-of-the-art results obtained with manually defined attributes and mined category-attribute associations, in the same dataset. Using direct similarity-based recognition, we further improve our results and obtain recognition accuracies comparable to the case where both attributes and category-attribute associations are selected manually. In the recognition experiments, we also present the performance of our system using state-of-the-art CNN features and, to our knowledge, get the best zero-shot recognition accuracies obtained on the AwA dataset.

References

[1] V. Ferrari, A. Zisserman, Learning visual attributes, in: J. Platt, D. Koller, Y. Singer, S. Roweis (Eds.), Advances in Neural Information Processing Systems 20, MIT Press, Cambridge, MA, 2008, pp. 433–440. 2

[2] A. Farhadi, I. Endres, D. Hoiem, D. Forsyth, Describing objects by their attributes., in: CVPR, IEEE, 2009, pp. 1778–1785. 2

[3] J. Wang, K. Markert, M. Everingham, Learning models for object recognition from natural language descriptions, in: Proceedings of the British Machine Vision Conference, BMVA Press, 2009, pp. 2.1–2.11, doi:10.5244/C.23.2. 2

[4] C. H. Lampert, H. Nickisch, S. Harmeling, Learning to detect unseen object classes by between-class attribute transfer., in: CVPR, IEEE, 2009, pp. 951–958. 2, 3, 8, 17, 24

[5] G. Wang, D. A. Forsyth, Joint learning of visual attributes, object classes and visual saliency., in: ICCV, IEEE, 2009, pp. 537–544. 2

[6] N. Kumar, A. C. Berg, P. N. Belhumeur, S. K. Nayar, Attribute and simile classifiers for face verification., in: ICCV, IEEE, 2009, pp. 365–372. 2

[7] O. Russakovsky, F.-F. Li, Attribute learning in large-scale datasets., in: K. N. Kutulakos (Ed.), ECCV Workshops (1), Vol. 6553 of Lecture Notes in Computer Science, Springer, 2010, pp. 1–14. 2

[8] T. L. Berg, A. C. Berg, J. Shih, Automatic attribute discovery and characterization from noisy web data, in: Proceedings of the 11th European conference on Computer vision: Part I, ECCV'10, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 663–676. 2, 4, 6, 7, 8, 18, 20

[9] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, B. Schiele, What helps where - and why? semantic relatedness for knowledge transfer., in: CVPR, IEEE, 2010, pp. 910–917. 2, 4, 5, 17, 24, 25

[10] A. Farhadi, I. Endres, D. Hoiem, Attribute-centric recognition for cross-category generalization, in: CVPR, 2010, pp. 2352–2359. 2

[11] Y. Wang, G. Mori, A discriminative latent model of object classes and attributes, in: Proceedings of the 11th European conference on Computer vision: Part V, ECCV'10, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 155–168. 2

[12] S. Branson, C. Wah, F. Schroff, B. Babenko, P. Welinder, P. Perona, S. Belongie, Visual recognition with humans in the loop., in: K. Daniilidis, P. Maragos, N. Paragios (Eds.), ECCV (4), Vol. 6314 of Lecture Notes in Computer Science, Springer, 2010, pp. 438–451. 2

[13] N. Kumar, A. C. Berg, P. N. Belhumeur, S. K. Nayar, Describable visual attributes for face verification and image search., IEEE Trans. Pattern Anal. Mach. Intell. 33 (10) (2011) 1962–1977. 2

[14] D. Parikh, K. Grauman, Interactively building a discriminative vocabulary of nameable attributes., in: CVPR, IEEE, 2011, pp. 1681–1688. 2, 4

[15] D. Parikh, Discovering localized attributes for fine-grained recognition, in: Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), CVPR '12, IEEE Computer Society, Washington, DC, USA, 2012, pp. 3474–3481. 2, 4

[16] H. Goëau, P. Bonnet, A. Joly, I. Yahiaoui, D. Barthelemy, N. Boujemaa, J.-F. Molino, The ImageCLEF 2012 plant identification task, CLEF 2012 working notes. 3

[17] K. Barnard, K. Yanai, Mutual information of words and pictures, in: Information Theory and Applications Inaugural Workshop, 2006. 4

[18] E. Boiy, K. Deschacht, M.-F. Moens, Learning visual entities and their visual attributes from text corpora., in: DEXA Workshops, IEEE Computer Society, 2008, pp. 48–53. 4

[19] J. Dodge, A. Goyal, X. Han, A. Mensch, M. Mitchell, K. Stratos, K. Yamaguchi, Y. Choi, H. D. III, A. C. Berg, T. L. Berg, Detecting visual text, in: HLT-NAACL, The Association for Computational Linguistics, 2012, pp. 762–772. 4

[20] M. Rastegari, A. Farhadi, D. Forsyth, Attribute discovery via predictable discriminative binary codes, in: Proceedings of the 12th European Conference on Computer Vision - Volume Part VI, ECCV'12, Springer-Verlag, Berlin, Heidelberg, 2012, pp. 876–889. doi:10.1007/978-3-642-33783-3_63.
URL http://dx.doi.org/10.1007/978-3-642-33783-3_63 4, 6, 7, 20

[21] G. Griffin, A. Holub, P. Perona, Caltech-256 object category dataset, Tech. Rep. 7694, California Institute of Technology (2007).
URL http://authors.library.caltech.edu/7694 5

[22] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: A Large-Scale Hierarchical Image Database, in: CVPR09, 2009. 5

[23] F. Yu, L. Cao, R. Feris, J. Smith, S.-F. Chang, Designing category-level attributes for discriminative visual recognition, in: Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, 2013, pp. 771–778. doi:10.1109/CVPR.2013.105. 5, 24, 25

[24] G. A. Miller, Wordnet: a lexical database for english, Communications of the ACM 38 (11) (1995) 39–41. 5

[25] G. Griffin, P. Perona, Learning and using taxonomies for fast visual categorization, in: Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on, 2008, pp. 1–8. doi:10.1109/CVPR.2008.4587410. 5

[26] J. Deng, A. C. Berg, K. Li, L. Fei-Fei, What does classifying more than 10,000 image categories tell us?, in: Proceedings of the 11th European conference on Computer vision: Part V, ECCV'10, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 71–84. URL http://dl.acm.org/citation.cfm?id=1888150.1888157 5

[27] A. Binder, K.-R. Mller, M. Kawanabe, On taxonomies for multi-class image categorization., International Journal of Computer Vision 99 (3) (2012) 281–301. 5

[28] Y. Feng, M. Lapata, Visual information in semantic representation, in: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Los Angeles, California, 2010, pp. 91–99. 5

[29] C. Silberer, V. Ferrari, M. Lapata, Models of semantic representation with visual attributes, in: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Sofia, Bulgaria, 2013, pp. 572–582. 5

[30] S. K. Divvala, A. Farhadi, C. Guestrin, Learning everything about anything: Webly-supervised visual concept learning, in: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2014. 6, 7, 8, 20

[31] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: CVPR, 2005, pp. 886–893. 8

[32] S. Belongie, J. Malik, J. Puzicha, Shape matching and object recognition using shape contexts, IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2001) 509–522. 8

[33] C. D. Manning, P. Raghavan, H. Schütze, Introduction to information retrieval, Vol. 1, Cambridge University Press Cambridge, 2008. 11

[34] A. Frome, Y. Singer, F. Sha, J. Malik, Learning globally-consistent local distance functions for shape-based image retrieval and classification, in: ICCV, 2007, pp. 1–8. 12

[35] Z. Harris, Distributional structure, Word 10 (23) (1954) 146–162. 13

[36] J. Mitchell, M. Lapata, Composition in distributional models of semantics, Cognitive Science 34 (8) (2010) 1388–1439. 14

[37] W. Blacoe, M. Lapata, A comparison of vector-based representations for semantic composition, in: Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL '12, Association for Computational Linguistics, Stroudsburg, PA, USA, 2012, pp. 546–556. 14

[38] S. Schaeffer, Graph clustering, Computer Science Review 1 (1) (2007) 27–64. 14

[39] J. Pearl, Probabilistic reasoning in intelligent systems: networks of plausible inference, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988. 14

[40] J. S. Yedidia, W. T. Freeman, Y. Weiss, Understanding belief propagation and its generalizations, Exploring artificial intelligence in the new millennium 8 (2003) 236–239. 14

[41] C.-C. Chang, C.-J. Lin, LIBSVM: a library for support vector machines, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm (2001). 16

[42] B. A. Yanikoglu, E. Aptoula, C. Tirkaz, Sabanci-okan system at imageclef 2012: Combining features and classifiers for plant identification, in: CLEF (Online Working Notes/Labs/Workshop), 2012. 23

[43] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: Convolutional architecture for fast feature embedding, arXiv preprint arXiv:1408.5093. 24

[44] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: F. Pereira, C. Burges, L. Bottou, K. Weinberger (Eds.), Advances in Neural Information Processing Systems 25, Curran Associates, Inc., 2012, pp. 1097–1105. 24

[45] C. Lampert, H. Nickisch, S. Harmeling, Attribute-based classification for zero-shot visual object categorization, Pattern Analysis and Machine Intelligence, IEEE Transactions on 36 (3) (2014) 453–465. 24, 25