Karlsruher Institut für Technologie

Semantic Attributes for Transfer Learning in Visual Recognition

zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

der KIT-Fakultät für Informatik
des Karlsruher Instituts für Technologie (KIT)
genehmigte

Dissertation
von

Ziad Al Halah

Tag der mündlichen Prüfung:	5. Februar 2018
Hauptreferent:	Prof. Dr. Rainer Stiefelhagen Karlsruher Institut für Technologie
Korreferent:	Prof. Dr. Christoph Lampert Institute of Science and Technology Austria

Abstract

Energized with the rise of deep learning models, artificial intelligence made large strides in bringing machine understanding to the realm of human performance. However, in order to generalize well these models rely heavily on the availability of thousands of manually labeled examples. Additionally, whenever a new task is encountered the learning starts from scratch and the expensive process of collecting additional training data is repeated. This significantly limits the scalability of such models.

On the other hand, we - humans - do not learn new tasks in isolation. We have the remarkable ability to utilize previously obtained knowledge in solving new problems. This skill is known as *transfer learning*. It enables us to learn faster, better and with few examples. Therefore, there is great interest to mimic this skill by machines especially in domains where training data is scarce or not available.

In this thesis we study transfer learning from a vision perspective. Specifically, we investigate how to carry on visual recognition (*e.g.* object or action classification) when there are few or no training samples. A promising solution in that direction is the semantic attribute framework. Here, visual categories are described in terms of intermediate attributes like color, pattern and shape. These attributes can be learned from a disjoint set of samples. Moreover, since they have a dual interpretation (visual and semantic), language can be effectively leveraged to guide the transfer process. That is, given a novel visual category, a linguistic description can be utilized to compose and transfer relevant attributes and construct the category model without the need of any training images. In this work, we pursue this framework and introduce novel solutions on how to model and transfer semantic attributes, how to automatically associate them with visual categories

and how to discover them from free-form linguistic descriptions. To that end, we address the attribute-based recognition from four vantage points.

First, unlike the prevailing model where attributes are learned in a global manner, we propose a hierarchical approach to learn the attributes at various abstraction levels. Furthermore, we demonstrate how the structure among the categories can be effectively leveraged to guide the learning and transfer process to construct more discriminative models for novel categories. A thorough evaluation shows a significant improvement of our model over the global approach especially in fine-grained recognition.

Second, while in the prevailing attribute-based transfer approach the user supervises the mapping between the attributes and the categories, in this work we propose to automatically establish the link among the two without user intervention. Our model captures the semantic relations coupling attributes and objects to predict their associations and select which attributes to transfer in an unsupervised manner.

Third, we circumvent the requirement of a predefined attribute vocabulary. We propose to leverage encyclopedia articles describing object categories in a free-form text to discover a set of discriminate, salient and diverse attributes. By alleviating the need of user defined vocabulary, our model unlocks the capabilities of attribute-based frameworks for large-scale data.

Finally, we introduce a novel real world application of semantic attributes. We propose the first approach to learn fashion styles and forecast their popularity in the future. We show that semantic attributes provides interpretable fashion styles and lead to better forecast of visual styles popularity compared to other representation.

Kurzzusammenfassung

Angetrieben durch den Erfolg von Deep Learning Verfahren wurden in Bezug auf künstliche Intelligenz erhebliche Fortschritte im Bereich des Maschinenverstehens gemacht. Allerdings sind Tausende von manuell annotierten Trainingsdaten zwingend notwendig, um die Generalisierungsfähigkeit solcher Modelle sicherzustellen. Darüber hinaus muss das Modell jedes Mal komplett neu trainiert werden, sobald es auf eine neue Problemklasse angewandt werden muss. Dies führt wiederum dazu, dass der sehr kostenintensive Prozess des Sammelns und Annotierens von Trainingsdaten wiederholt werden muss, wodurch die Skalierbarkeit solcher Modelle erheblich begrenzt wird. Auf der anderen Seite bearbeiten wir Menschen neue Aufgaben nicht isoliert, sondern haben die bemerkenswerte Fähigkeit, auf bereits erworbenes Wissen bei der Lösung neuer Probleme zurückzugreifen. Diese Fähigkeit wird als *Transfer-Learning* bezeichnet. Sie ermöglicht es uns, schneller, besser und anhand nur sehr weniger Beispiele Neues zu lernen. Daher besteht ein großes Interesse, diese Fähigkeit durch Algorithmen nachzuahmen, insbesondere in Bereichen, in denen Trainingsdaten sehr knapp oder sogar nicht verfügbar sind.

In dieser Arbeit untersuchen wir Transfer-Learning im Kontext von Computer Vision. Insbesondere untersuchen wir, wie visuelle Erkennung (z.B. Objekt- oder Aktionsklassifizierung) durchgeführt werden kann, wenn nur wenige oder keine Trainingsbeispiele existieren. Eine vielversprechende Lösung in dieser Richtung ist das Framework der semantischen Attribute. Dabei werden visuelle Kategorien in Form von Attributen wie Farbe, Muster und Form beschrieben. Diese Attribute können aus einer disjunkten Menge von Trainingsbeispielen gelernt werden. Da die Attribute eine doppelte, d.h. sowohl visuelle als auch semantische, Interpretation haben, kann Sprache effektiv genutzt werden, um den Übertragungsprozess zu steuern. Dies bedeutet, dass Modelle für

eine neue visuelle Kategorie nur anhand der sprachlichen Beschreibung erstellt werden können, indem relevante Attribute selektiert und auf die neue Kategorie übertragen werden. Die Notwendigkeit von Trainingsbildern entfällt durch diesen Prozess jedoch vollständig. In dieser Arbeit stellen wir neue Lösungen vor, semantische Attribute zu modellieren, zu übertragen, automatisch mit visuellen Kategorien zu assoziieren, und aus sprachlichen Beschreibungen zu erkennen. Zu diesem Zweck beleuchten wir die attributbasierte Erkennung aus den folgenden vier Blickpunkten:

- Anders als das gängige Modell, bei dem Attribute global gelernt werden müssen, stellen wir einen hierarchischen Ansatz vor, der es ermöglicht, die Attribute auf verschiedenen Abstraktionsebenen zu lernen. Wir zeigen zudem, wie die Struktur zwischen den Kategorien effektiv genutzt werden kann, um den Lern- und Transferprozess zu steuern und damit diskriminative Modelle für neue Kategorien zu erstellen. Mit einer gründlichen experimentellen Analyse demonstrieren wir eine deutliche Verbesserung unseres Modells gegenüber dem globalen Ansatz, insbesondere bei der Erkennung detailgenauer Kategorien.
- In vorherrschend attributbasierten Transferansätzen überwacht der Benutzer die Zuordnung zwischen den Attributen und den Kategorien. Wir schlagen in dieser Arbeit vor, die Verbindung zwischen den beiden automatisch und ohne Benutzereingriff herzustellen. Unser Modell erfasst die semantischen Beziehungen, welche die Attribute mit Objekten koppeln, um ihre Assoziationen vorherzusagen und unüberwacht auszuwählen welche Attribute übertragen werden sollen.
- Wir umgehen die Notwendigkeit eines vordefinierten Vokabulars von Attributen. Statt dessen schlagen wir vor, Enyzklopädie-Artikel zu verwenden, die Objektkategorien in einem freien Text beschreiben, um automatisch eine Menge von diskriminanten, salienten und vielfältigen Attributen zu entdecken. Diese Beseitigung des Bedarfs eines benutzerdefinierten Vokabulars ermöglicht es uns, das Potenzial attributbasierter Modelle im Kontext sehr großer Datenmengen vollends auszuschöpfen.
- Wir präsentieren eine neuartige Anwendung semantischer Attribute in der realen Welt. Wir schlagen das erste Verfahren vor, welches automatisch Modestile lernt, und vorhersagt, wie sich ihre Beliebtheit in naher Zukunft entwickeln wird. Wir zeigen, dass semantische Attribute interpretierbare Modestile liefern und zu einer besseren Vorhersage der Beliebtheit von visuellen Stilen im Vergleich zu anderen Darstellungen führen.

Contents

Chapter 1

Introduction

We, humans, are equipped with an extraordinary vision system which is refined through long years of evolution. In just a blink of an eye, we are capable of recognizing the categories of many objects present in an image (Potter et al., 2014; Thorpe et al., 1996). A longer look will allow us to recognize the detailed visual properties of these objects and the surrounding scene. If we are to study the given image, we are able to not only infer all visual information available but also go beyond that and infer the relations between these objects and their current status. We will also be able to provide a detailed description of the scene and employ our prior knowledge of the world and imagination to even write a story incorporating the various visual elements in the image.

Computer vision made large strides of improvements and demonstrated a remarkably high level of performance especially within the last few years. There are models that already outperform or at the level of human expert performance on tasks like object classification (He et al., 2015), traffic sign reading (Stallkamp et al., 2012), face recognition (O'Toole et al., 2007), age estimation (Han et al., 2015), lip reading (Chung et al., 2016), predicting photos' geolocation (Weyand et al., 2016), and playing Atari games (Mnih et al., 2015). Many of these development came with the re-emergence of *deep artificial neural networks*. Fueled with large-scale training data sets and advances in computation hardware, deep learning established itself as one of the most successful learning framework in vision by reaching unprecedented level of visual analysis capabilities. However, such deep models are characterized by millions of parameters. This, in return, demands the availability of immense amounts of training data in order to obtain a reliable estimate of these parameters. Hence, such models are known to perform quite poorly when training

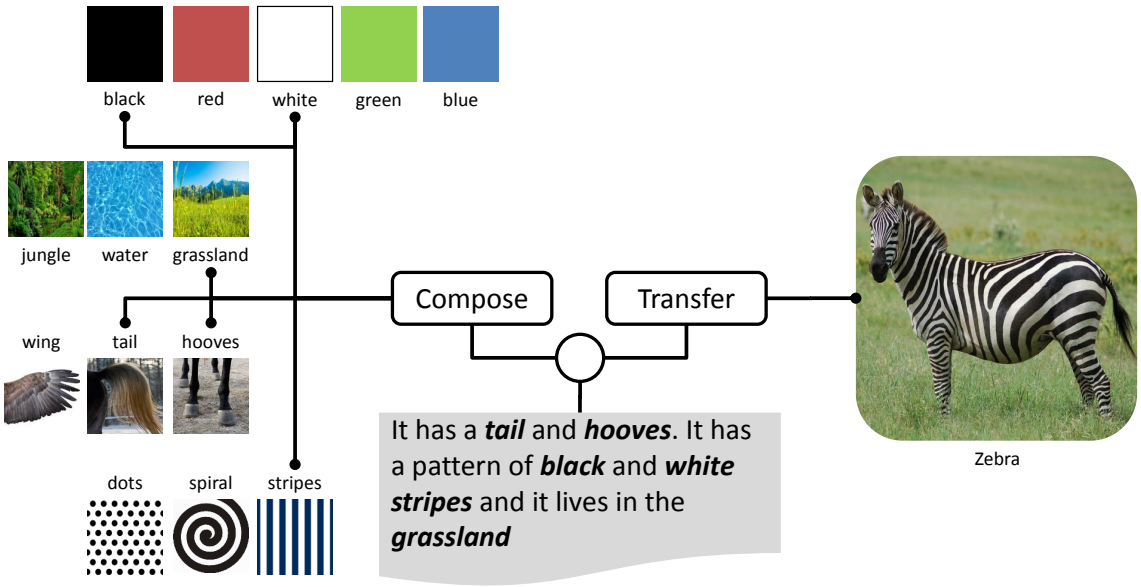


Figure 1.1: By leveraging language and vision modalities, we are able to recognize novel categories even when there are no training data. For example, a linguistic description helps us to identify the familiar visual-semantic concepts that we need to compose and transfer to construct a model for a new visual category.

data is scarce or unavailable. Collecting and labeling data, while it is an option to remedy such cases, is very expensive. Additionally, we are faced with this situation whenever we want to extend our model to a new task or a new domain. Hence, this renders the continuous data collection option infeasible and hard to maintain.

Children, on the other hand, are capable of performing these visual tasks effortlessly and in a manner that is still not reachable yet by our best algorithms. We, humans, can learn new categories using only a handful of examples. In fact, we can even construct mental models of object categories using no training examples at all. We are very good at decomposing problems into simpler units and leveraging different modalities in our learning process. For example, a child through its learning process would learn how an animal such as a horse, a giraffe or a lion looks like. Additionally, the child will learn what an animal is composed of (*e.g.* parts, shape, color, pattern) and assign names to these visual properties (see Figure 1.1). Then, when faced with a new *unseen* category like zebra, the child will only require a linguistic description (*e.g.* a zebra is an animal that looks like a horse but has black and white stripes) in order to construct a model for class zebra. Although the child has never seen such an animal before, through efficient

knowledge representation and leveraging both vision and language modalities to transfer that knowledge, the child will easily recognize zebras in her/his next visit to the zoo.

This ability to tap to our previous skills and experiences, detecting the aspects of similarities to the current task at hand, and efficiently apply such knowledge for solving the new problem is known as *knowledge transfer* or *transfer learning* (Pratt, 1993; Woodworth and Thorndike, 1901). It gives us the advantage of learning new concepts faster and with a high initial performance while at the same time using only few trials or examples. Therefore, there is a growing interest in the Vision and Machine Learning Community to employ transfer learning to improve models generalization and overcome the scarcity of training data.

In this thesis we study the knowledge transfer process from a computer vision perspective. Specifically, we focus on visual-semantic knowledge representation in transfer learning and scene understanding. Such a representation has the unique property of being both visually detectable and human understandable (Figure 1.1). Thus, it can be effectively learned from visual data and transfered to novel tasks guided with a form of semantic description (*i.e.* using the language modality to guide the transfer process). That is, we tackle questions like: What type of knowledge representation is best suited for transfer learning? How can we guide the transfer process to select the most suitable knowledge units to transfer to a new task? How to automatically establish the semantic link between a novel task and the previously obtained knowledge without user intervention (*i.e.* a user given semantic description)? How to discover and learn this visual semantic knowledge from natural language sources without supervision?

1.1 Thesis organization and contributions

The thesis is organized as follows:

Chapter 2: Background and related work. In this chapter, we start by introducing the main concepts and terminology used throughout this thesis. We cover mainly two topics: 1) semantic attributes, and 2) transfer learning. Moreover, we discuss related work to our research in these respective fields and highlight the contributions of this work.

Chapter 3: Semantic knowledge representation in transfer learning. We analyze in this chapter the benefit of using semantic representations in comparison to low-level

features for transfer learning. We consider semantics at different abstraction levels and study the impact of these representations on the performance and robustness of the transfer framework when considering aspects like parameter and instance transfer. Furthermore, we introduce a novel hierarchical knowledge representation based on the embedded structure in the semantic attribute space. Finally, we provide evaluation results of the proposed framework on challenging transfer settings that demonstrates the effectiveness of our approach compared to state-of-the-art.

Chapter 4: Hierarchical transfer of semantic attributes. In this chapter, we introduce an approach that leverages the embedded structure between the categories in the source set to learn and transfer attributes. We propose to capture the intra-attribute variations in a novel hierarchical model that expands the source knowledge with additional abstraction levels of the attributes, from the most specific that distinguish one class from another to the most general that are learned over all categories. Furthermore, we introduce a guided transfer approach that can choose the appropriate attributes to be shared with an unseen class. Finally, we evaluate the proposed model on three challenging data sets each with a different granularity of object categories and highlight the potentials of our model.

Chapter 5: Predicting class-attribute associations. We aim in this chapter to alleviate the need for user predefined mappings of classes and attributes. We propose a novel approach that models semantic relations which couples classes with their corresponding attributes. Hence, given only the name of an unseen class, the learned relationship model is used to automatically predict the class-attribute associations. Consequently, our model also facilitates transferring attributes across data sets without any user supervision. We show in the evaluation that integrating knowledge from multiple sources results in a significant improvement in performance.

Chapter 6: Automatic attribute discovery from natural language. In this chapter, we move a step further and propose a model that not only automatically maps classes and attributes but also discovers the attribute vocabulary itself, thus circumventing any need for human supervision in the attribute-based framework. Our proposed model utilizes online text corpora to automatically discover a salient and discriminative vocabulary that correlates well with the human concept of semantic attributes. Moreover, we propose a deep convolutional model to optimize class-attribute associations with a linguistic prior that accounts for noise and missing data in text. Finally, in a thorough evaluation we

demonstrate that our model is able to efficiently discover and learn semantic attributes at a large scale.

Chapter 7: Attributes for fashion analysis. We introduce in this chapter a real world application of semantic attributes. We propose the first approach to predict the future popularity of styles discovered from fashion images in an unsupervised manner. Our model leverages semantic attributes to discover garment styles at a large scale. Based on these styles, we train a forecasting model to represent their trends over time. The resulting model can hypothesize which styles will become popular in the future, discover style dynamics (trendy vs. classic), and name the key visual attributes that will dominate tomorrow's fashion. Furthermore, we show in the evaluation that fashion forecasting benefits greatly from the attribute-based representation, much more than textual or meta-data cues surrounding products. The work in this chapter was done in collaboration with the University of Texas at Austin.

Chapter 8: Conclusion. We conclude the thesis with a summary of the key contributions presented in this work and discussion of future directions.

1.2 List of publications

The work presented in this thesis spans mainly contributions from the following peer-reviewed publications:

1. Z. Al-Halah, R. Stiefelhagen, and K. Grauman, "Fashion Forward: Forecasting Visual Style in Fashion," in IEEE International Conference on Computer Vision (ICCV), 2017.
2. Z. Al-Halah and R. Stiefelhagen, "Automatic Discovery, Association Estimation and Learning of Semantic Attributes for a Thousand Categories," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.
3. Z. Al-Halah, L. Rybok, and R. Stiefelhagen, "Transfer Metric Learning for Action Similarity using High-Level Semantics," Pattern Recognition Letters, vol. 72, pp. 82–90, 2016.
4. Z. Al-Halah, M. Tapaswi, and R. Stiefelhagen, "Recovering the Missing Link: Predicting Class-Attribute Associations for Unsupervised Zero-Shot Learning," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

5. Z. Al-Halah and R. Stiefelhagen, "How to Transfer? Zero-Shot Object Recognition via Hierarchical Transfer of Semantic Attributes," in IEEE Winter Conference on Applications of Computer Vision (WACV), 2015.
6. Z. Al-Halah, L. Rybok, and R. Stiefelhagen, "What to Transfer? High-Level Semantics in Transfer Metric Learning for Action Similarity," in International Conference on Pattern Recognition (ICPR), 2014. (*Best Student Paper Award*)

Additionally, the following publications were part of my PhD research but outside the scope of this thesis:

1. M.-L. Haurilet, M. Tapaswi, Z. Al-Halah, and R. Stiefelhagen, "Naming TV Characters by Watching and Analyzing Dialogs," in IEEE Winter Conference on Applications of Computer Vision (WACV), 2016.
2. E. Ghaleb, M. Tapaswi, Z. Al-Halah, H. K. Ekenel, and R. Stiefelhagen, "Accio: A Data Set for Face Track Retrieval in Movies Across Age," in ACM on International Conference on Multimedia Retrieval (ICMR), 2015.
3. T. Gehrig*, Z. Al-Halah*, H. K. Ekenel, and R. Stiefelhagen, "Action Unit Intensity Estimation using Hierarchical Partial Least Squares," in IEEE International Conference on Automatic Face and Gesture Recognition (FG), 2015. (* = equal contribution)
4. L. Rybok, B. Schauerte, Z. Al-Halah, and R. Stiefelhagen, "Important Stuff, Everywhere! Activity Recognition with Salient Proto-Objects as Context," in IEEE Winter Conference on Applications of Computer Vision (WACV), 2014.
5. Z. Al-Halah, T. Gehrig, and R. Stiefelhagen, "Learning Semantic Attributes via a Common Latent Space," in International Conference on Computer Vision Theory and Applications, 2014.

Chapter 2

Background and Related Work

In this chapter, we provide background coverage of concepts and terms used throughout this thesis. Additionally, we give an overview of prior work related to the problems tackled in this work which can be roughly split into two main topics: 1) attribute-based visual recognition and 2) transfer learning. Both of these topics witnessed increased popularity in the last years. However, this chapter is not intended to provide an exhaustive coverage of all key contributions in these fields. We focus rather on work most related to ours and highlight our own contributions in terms of commonalities and differences to literature.

We start in Section 2.1 by introducing the main concepts and characteristics of semantic attributes. This is followed with a discussion of related work to our contributions in modeling attributes (Section 2.1.1), predicting their associations to object categories (Section 2.1.2), automatic discovery of attribute vocabulary (Section 2.1.3) and semantic attributes applications (Section 2.1.4). Consequently, we provide a brief introduction to transfer learning in Section 2.2. Then, we review and discuss related work to our own contribution in tackling the transfer aspects of what to transfer? (Section 2.2.1), how to transfer? (Section 2.2.2) and when to transfer? (Section 2.2.3). Finally, we discuss our contributions to the transfer framework of zero-shot learning in Section 2.2.4.

2.1 Semantic attributes

A key aspect of this thesis is the leverage of semantic representation for knowledge transfer and visual recognition. Semantic representation, *i.e.* a knowledge representation

which has a clear association with language, is a natural form of representation that facilitates transferring across tasks with human feedback and guidance. Additionally, there is evidence that we humans rely on such representation in learning concepts from the surrounding world using a form of semantic memory (Patterson et al., 2007). The semantic memory in the human brain is composed of distributed regions such that each region captures specific type of attributes like visual-, motion- and linguistic-properties. These regions are connected with direct neuroanatomical pathways that enable direct activation of representations between modalities of all semantic categories (Patterson et al., 2007). This view of semantic memory supports our ability to easily name objects based on a visual description (*e.g.* what is the name of the big animal that has big ears and a trunk?) or to describe sensory properties of an object (*e.g.* how does the cake taste? what is the sound of a dog?). In other words, this representation allow us to easily transfer and reuse previously learned concepts (*e.g.* gray, big ears and trunk) activated by a linguistic description to compose a new model for a target task (*e.g.* recognition of elephant images). Next, we address one of the most popular forms of semantic representations, semantic attributes.

Semantic attributes are considered a form of intermediate representation between low-level image features and category labels. Furthermore, each dimension in this representation is associated with a semantic concept, *i.e.* it is human understandable, while at the same time is visually detectable, *i.e.* it can be learned and detected by machines. This mid-level representation transcends the categories borders; hence, it forms a shared layer based on a joint vocabulary that describes the various visual aspects of all classes.

Formally, let $\mathcal{C} = \{c_i\}$ be a set of categories. Each object $o_j \in c_i$ can be described with a vector $\mathbf{a}$ of M semantic attributes where each attribute $a_m \in \mathbf{a}$ is assigned a label from the attribute vocabulary $\mathcal{A}$. In the following, we provide more details about some aspects of the attribute-based recognition framework.

Semantics. Probably the most important aspect of semantic attributes is their human interpretability. Many typical visual features like HoG (Dalal and Triggs, 2005) or SIFT (Lowe, 1999) capture low-level properties like edges and salient points. More advanced models like bag-of-words (Leung and Malik, 2001), the constellation model (Fergus et al., 2003) and the deformable part model (Felzenszwalb et al., 2008) capture more abstract and salient visual properties of categories like parts or certain edge configurations. However, unlike attributes we usually can not assign a label for each dimension in these representations. This is due to the unsupervised approach used in learning these

representations, hence they are fitted to discover repetitive pattern of features which is not necessarily interpretable. Nonetheless, it is possible that certain dimensions in these representations can be correlated with a label. For example, some hidden representations of deep neural networks are known to capture semantic properties of objects (Zeiler and Fergus, 2014). Once this happens one can consider that feature to be a semantic attribute since it is nameable.

The semantic aspect of the attributes enables a natural interface between a user and a machine at a fine-grained semantic level. For example, the machine can generate a description for the user of an unfamiliar category based on images (Hendricks et al., 2016) or the user can supply the machine with a textual attribute-based object description to compose a model for categories with no training data (Lampert et al., 2009). Additionally, a more elaborate inference about the properties of objects can be supported by attributes to detect unusual aspects of an object (*e.g. a car with wings*) or missing properties (*e.g. a car without wheels*) (Saleh et al., 2013).

Usually, categories have various aspects of semantic attributes along the visual ones. For example, an attribute can be related to: i) language: like type and form of linguistic descriptions that can be applied to the object; ii) motor-specific: like forms of movement; iii) sensory: like texture, taste and smell; iv) sound: like pitch and tone. However, in this thesis we use the term *semantic attributes* to refer only to those that are directly detectable from or correlate with a visual signal.

Some of the early work toward semantic attributes in computer vision go back to Yanai and Barnard (2005), where they measured the visualness of words using web images queried by certain concepts and assigned these words to regions in the image. Similarly, Ferrari and Zisserman (2008) and Van De Weijer et al. (2009) also proposed to learn simple semantic concepts like colors and textures. Moreover, Kumar et al. (2008) introduced a face retrieval approach that accounts for certain face-related attributes like *gender*, *age*, *hair-color* or *race*. However, it was till the seminal work of Lampert et al. (2009) and Farhadi et al. (2009) that attributes as an intermediate semantic representation shareable across categories are presented to the computer vision community. This new prospective enabled some novel applications of semantic attributes, like learning to recognize new visual concepts with no training images as we will discuss later in Section 2.2.4.

Annotations. Attribute annotations can be defined at three levels:

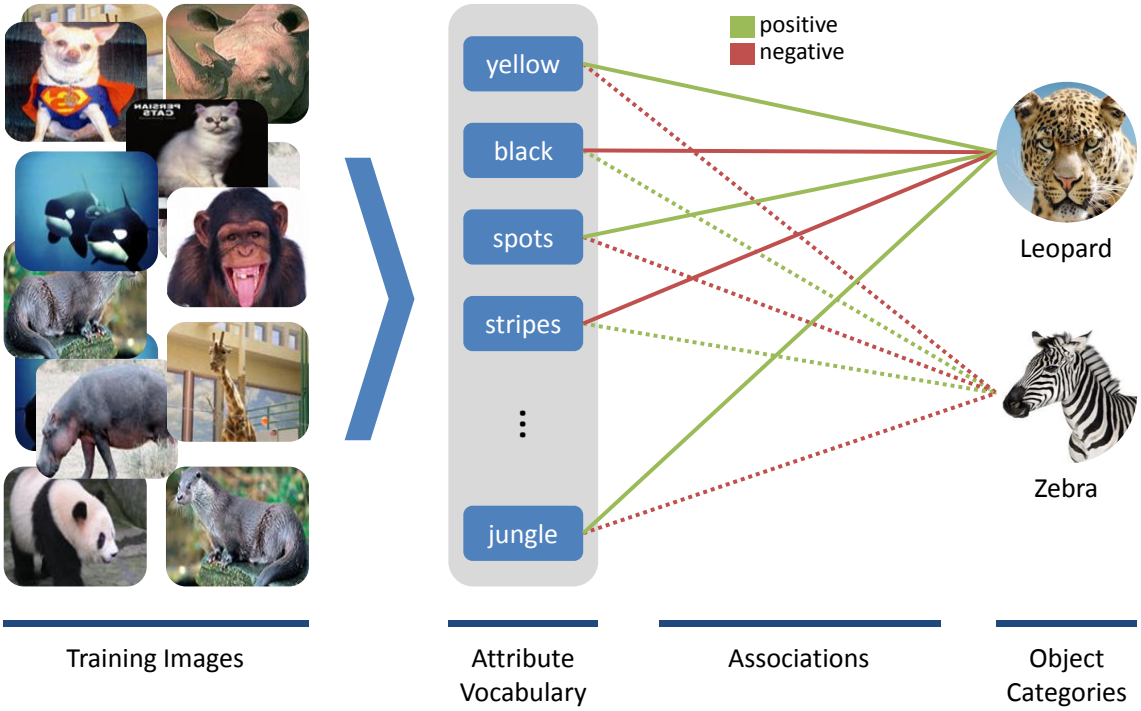


Figure 2.1: In the common Direct Attribute Prediction (DAP) framework, a predefined attribute vocabulary is learned directly from training images. The models of the object categories are then constructed based on a predefined associations that define the mapping to the semantic attributes.

1. instance-level: the annotations are collected at the instance level, *e.g.* per object bounding-box (Farhadi et al., 2009) or image region (Wah and Belongie, 2013).

2. image-level: here the whole image is annotated with attributes without any spatial information (Patterson and Hays, 2012).

3. class-level: in this case, the category is annotated with attributes such that all instances of a certain category share the same attribute annotation (Lampert et al., 2009).

These three different annotation levels differ significantly in their associated cost and precision. While it is quite cost effective to provide annotations at the class level, these annotations usually result in learning less precise attribute models since many correlations are usually incorporated in the learning process. On the other hand, the instance- and image-level annotations help in learning more precise models by spatially grounding the relative image areas. However, they are much more expensive to obtain. This results usually in much smaller data sets to train the attribute models.

Visual recognition pipeline. Attributes can be incorporated in various ways for visual analysis depending on the application. However, the most prominent and intuitive approach to incorporate attributes in the recognition pipeline is the Direct Attribute Prediction approach (DAP) (Farhadi et al., 2009; Lampert et al., 2009). In DAP, attributes are learned and inferred directly from image samples. Then, a category model is constructed by composing the respective attribute models based on the given class-attribute associations (see Figure 2.1).

A probabilistic realization of DAP using binary attributes is first introduced by Lampert et al. (2009). Let $p(a_m|x)$ be the probability of attribute a_m in sample x. Then a class $c_i \in \mathcal{C}$ can be inferred as:

$$p(c_i|\mathbf{a}) = \frac{p(c_i)p(\mathbf{a}|c_i)}{p(\mathbf{a})}. \tag{2.1}$$

Since each class is described by an attribute vector $\mathbf{a}^{c_i}$ in a deterministic way, then:

$$p(c_i|\mathbf{a}) = \frac{p(c_i)[[\mathbf{a} = \mathbf{a}^{c_i}]]}{p(\mathbf{a}^{c_i})}, \tag{2.2}$$

where $[[\cdot]]$ is the Iverson's bracket: *i.e.* $[[P]] = 1$ if condition P is true and 0 otherwise. Accordingly, the probability of class c_i in image x can be defined as:

$$p(c_i|x) = \sum_{a \in \{0,1\}^M} p(c_i|a)p(a|x) = \frac{p(c_i)}{p(\mathbf{a}^{c_i})} \prod_{m=1}^{M} p(a_m^{c_i}|x). \tag{2.3}$$

For $p(\mathbf{a})$ a factorial distribution is assumed, *i.e.* $p(\mathbf{a}) = \prod_m p(a_m)$. Both class prior $p(c_i)$ and attribute prior $p(a_m)$ can be assumed to be uniform distributions and consequently do not affect the classification decision or they can be estimated from the training data.

Our contribution. In this thesis, we make several contributions to the attribute-based visual recognition framework. Specifically, we propose novel models 1) to learn attributes at different abstraction levels; 2) to predict class-attribute associations; 3) discover attribute vocabulary automatically; and finally 4) we introduce a novel application of attributes in the domain of fashion analysis. In the following, we discuss all these contributions in details.

2.1.1 Abstraction level

Attributes are usually referred to as a *mid-level* representation. That is, they capture a finer-grain of semantics than categories. Some of the most common types of attributes are: i) *parts* (*e.g.* leg, wheel and wing); ii) *color* (*e.g.* red, blue and green); iii) *pattern* (*e.g.* stripes, dots and squares). However, the level of representation is relevant to the task at hand. That is for scene or action recognition, for example, object categories may as well be considered as attributes. For example, since *a desert is the habitat of camels*, in this case the object category *camel* can be considered an attribute of the scene category *desert*.

Nonetheless, even when considering a certain target task, attributes are usually learned in a global manner from all classes available in the source and at a single level of abstraction (*e.g.* Farhadi et al., 2009; Lampert et al., 2009; Liu et al., 2011a; Parikh and Grauman, 2011). Context plays a major role in the visual depiction of an attribute. For example, a *fluffy* towel and a *fluffy* dog have different visual realizations of *fluffiness*. Similarly, an *old* book versus an *old* car; and a *small* house versus a *small* horse lead to different depictions of attributes *old* and *small* respectively. Such methods can not cope with the high variations within each of the attributes. Thus, taking context into consideration while learning attributes might help us reach a better representation. One way to handle such variations is to jointly model objects, attributes and their correlations like in Wang and Mori (2010) and Liu et al. (2011a). Another way is to explicitly learn class-specific attributes. In Farhadi et al. (2009) a set of attributes are learned per class as an intermediate step for feature selection in order to reduce attribute correlations. Yu et al. (2013) propose to learn data-driven attributes at the category-level to better discriminate the classes. However, data-driven attributes usually carry no semantic meaning; thus, their approach requires user interaction when performing semantic related tasks like zero-shot learning. In Zhang et al. (2013) the concepts in ImageNet are augmented with a set of semantic and data-driven attributes. These are used along with the hierarchy to learn a better similarity metric for content-based image retrieval. Chen and Grauman (2014) and Misra et al. (2017) propose to learn class specific attributes and infer extra classifiers of class-attribute pairs with no training data.

Our contribution. We propose in Chapter 4 to explicitly model the intra-attribute variations at different abstraction levels. However, rather than just using class-specific attributes, we expand the notion of the attributes to cover the spectrum from the most abstract (*i.e. global attributes*) to the most specific (*i.e. class-based attributes*) driven by

the embedded relations between the categories. Moreover, our model does not require any extra annotations to learn these attributes and at the same time it outperforms other attribute-based baselines with a wide margin.

2.1.2 Associations

The associations define the mapping between categories and attributes. That is, they state which attribute connected to which class and how. There are three types of attribute associations:

1. *binary*: this is the most common type of an association and it describes whether an attribute exists or not for a certain object, *i.e.* $\mathbf{a} \in \{0,1\}^M$, (*e.g.* Lampert et al., 2009).
2. *continuous*: these associations assign a real value for each attribute, *i.e.* $\mathbf{a} \in [0,1]^M$, that captures the frequency or the strength of that attribute for a certain object, (*e.g.* Wah et al., 2011).
3. *relative*: unlike both previous types which represent attributes in the absolute sense, the relative attribute capture how this attribute appears in one object in relation to another. That is, $\mathbf{a}^{ij} \in \{-1,0,1\}^M$ such that if $a_m^{ij} = 1$ then $a_m^i > a_m^j$ (*e.g.* *the elephant is bigger than the ant*) and vice versa when $a_m^{ij} = -1$. If $a_m^{ij} = 0$ then both objects exhibit attribute a_m at the same level, (*e.g.* Parikh and Grauman, 2011).

Predefined associations. The vast majority of attribute-based approaches rely on the underlying assumption that for each class the complete information about attribute associations are manually defined (Farhadi et al., 2009) or imported from expert-based knowledge sources (Lampert et al., 2009; Wah et al., 2011). This is a hindering assumption to the scalability of the attribute-based framework since the common user is unlikely to have such a knowledge or is simply unwilling to manually set hundreds of associations for each new category.

Association prediction. Towards simplifying the required user involvement, Yu et al. (2013) reduce the level of user intervention by asking the operator to select the most similar known classes to a given novel class and then inferring its expected attributes. In a different direction, other approaches focused on predicting the class-attribute associations automatically. Mensink et al. (2014) and Rohrbach et al. (2010) propose an unsupervised

approach to automatically learn the class-attribute association strength by using text-based semantic relatedness measures and co-occurrence statistics obtained from web-search hit counts. However, as web data is noisy, class and attribute terms can appear in documents in different contexts which are not necessarily related to the original attribute relation we seek.

Our contribution. In Chapter 5, we propose a novel approach to automatically predict attribute associations using semantic relations. Our approach captures the diverse relations between classes and attributes as bilinear operators with low rank factorization. This renders our model to be adequate for cases when training data is relatively sparse. Our model can predict class-attribute associations with high accuracy, outperforming state-of-the-art with a significant margin.

2.1.3 Vocabulary

The attribute vocabulary $\mathcal{A}$ plays a vital rule in the robustness and performance of the attribute-based model. Issues like diversity, coverage and descriptiveness control the quality of the defined semantics. Thus, the vocabulary is usually defined by an expert in the field or through crowd sourcing. Normally, such a vocabulary needs to be picked with consideration of the target task. For example, for object classification, the vocabulary needs to discriminate well between the target categories while at the same time is shared among them in order to capture common properties.

Annotation cost. Bearing these previous criteria in mind, collecting attribute annotations proved to be very expensive. This clearly limits the scalability of attribute-based approaches to large number of categories. Thus, most available attribute data sets (Farhadi et al., 2009; Lampert et al., 2009; Patterson and Hays, 2012; Wah et al., 2011) are limited in terms of the number of attributes, categories or images. Patterson and Hays (2016) estimate the cost of annotating $84,000$ images of 29 object categories with 196 attributes for the COCO data set (Lin et al., 2014) using crowdsourcing to be around $110,000$ US Dollars.

Attribute discovery. There were few attempts in the literature to automatically obtain an attribute vocabulary. Rohrbach et al. (2010, 2011) mine attributes by crawling the WordNet (Miller, 1995) ontology. Specifically, they track the "has-part" relations in WordNet to extract "part" attributes. On the other hand, Ferrari and Zisserman (2008),

Chen et al. (2013) and Divvala et al. (2014) use the top ranked images returned by web search engines queried with a certain vocabulary to estimate the "visualness" of words. Berg et al. (2010) and Sun et al. (2015) sample pairs of (image, description) from the Internet to automatically find a set of visual attributes. Similarly, Vittayakorn et al. (2016a) use both image-based textual descriptions as well as a set of image tags provided by users in social media to identify the attribute vocabulary.

Our contribution. In this work, we circumvent the need of user supervision to define and annotate attributes. We propose in Chapter 6 an unsupervised end-to-end approach to automatically mine and learn semantic attributes for thousands of categories. Different from previous work, our approach does not require images aligned with textual descriptions or tags. Furthermore, we do not rely on a predefined ontology such as WordNet or target only a specific type of attributes like "parts". Instead, we use textual description at the category level in form of encyclopedia entries to extract a compact set of attributes. Additionally, our model is able to discover *salient*, *discriminate*, and *diverse* set of attributes that generalize well across categories.

2.1.4 Applications

Due to their favorable properties, semantic attributes witnessed wide spread utilization in many computer vision applications. For example, attributes are successfully employed in face analysis (Kumar et al., 2011), object classification (Farhadi et al., 2010), anomaly detection (Saleh et al., 2013), image retrieval (Douze et al., 2011), visual comparison (Parikh and Grauman, 2011), image (Kulkarni et al., 2011) and video (Rohrbach et al., 2013b) captioning, person re-identification (Shi et al., 2015), image segmentation (Zheng et al., 2014b), zero-shot learning (Lampert et al., 2009) and visual question answering (Wu et al., 2017). In this thesis, we introduce a new application for semantic attributes in the domain of *fashion visual analysis*.

Fashion is a fascinating domain for computer vision. Not only does it offer a challenging testbed for fundamental vision problems—human body parsing (Yamaguchi et al., 2012, 2013), cross-domain image matching (Chen et al., 2015c; Huang et al., 2015; Kiapour et al., 2015; Liu et al., 2012b), and recognition (Bossard et al., 2012; Chen et al., 2012; Kiapour et al., 2014; Liu et al., 2016)—but it also inspires new problems that can drive a research agenda, such as modeling visual compatibility (Iwata et al., 2011; Veit et al., 2015), interactive fine-grained retrieval (Kovashka et al., 2012; Yu and Grauman, 2015),

or reading social cues from what people choose to wear (Chen et al., 2015a; Kwak et al., 2013; Simo-Serra et al., 2015; Song et al., 2011). At the same time, the space has potential for high impact: the global market for apparel is estimated at $3 Trillion USD (FashionUnited 2017). It is increasingly entwined with online shopping, social media, and mobile computing—all arenas where automated visual analysis should be synergetic.

Our contribution. In contrast to prior work, we tackle yet a novel fashion analysis problem and introduce in Chapter 7 the *first* approach to forecast the popularity of fashions styles in the future. In the following, we discuss the relevant work in visual fashion analysis to our work in that direction.

Attributes in fashion. Descriptive visual attributes are naturally amenable to fashion tasks, since garments are often described by their materials, fit, and patterns (*denim*, *polka-dotted*, *tight*). Attributes are used to recognize articles of clothing (Bossard et al., 2012; Liu et al., 2016), retrieve products (Di et al., 2013; Huang et al., 2015), and describe clothing (Chen et al., 2012, 2015c). Relative attributes (Parikh and Grauman, 2011) are explored for interactive image search with applications to shoe shopping (Kovashka et al., 2012; Yu and Grauman, 2015). While often an attribute vocabulary is defined manually, useful clothing attributes are discoverable from noisy meta-data on shopping websites (Berg et al., 2010) or neural activations in a deep network (Vittayakorn et al., 2016b). Unlike prior work, we use inferred visual attributes as a conduit to discover fine-grained fashion styles from unlabeled images.

Learning styles. Limited work explores representations of visual *style*. Different from recognizing an article of clothing (*e.g.* *sweater*, *dress*) or its attributes (*e.g.* *blue*, *floral*), styles entail the higher-level concept of how clothing comes together to signal a trend. Early methods explore supervised learning to classify people into style categories, e.g., biker, preppy, Goth (Kiapour et al., 2014; Veit et al., 2015). Since identity is linked to how a person chooses to dress, clothing can be predictive of occupation (Song et al., 2011) or one's social "urban tribe" (Kwak et al., 2013; Murillo et al., 2012). Other work uses weak supervision from meta-data or co-purchase data to learn a latent space imbued with style cues (Simo-Serra and Ishikawa, 2016; Veit et al., 2015). In contrast to prior work, we pursue an unsupervised approach for discovering visual styles from data, which has the advantages of i) facilitating large-scale style analysis, ii) avoiding manual definition of style categories, iii) allowing the representation of finer-grained styles, and

iv) allowing a single outfit to exhibit multiple styles. Unlike concurrent work from Hsiao and Grauman (2017) that learns styles of outfits, we discover styles for individual garments and, more importantly, predict their popularity in the future.

Discovering trends. Beyond categorizing styles, a few initial studies analyze fashion *trends*. A preliminary experiment plots frequency of attributes (floral, pastel, neon) observed over time (Vittayakorn et al., 2015). Similarly, a visualization shows the frequency of garment meta-data over time in two cities (Simo-Serra et al., 2015). The system from Vittayakorn et al. (2016a) predicts when an object was made. The collaborative filtering recommendation system of (He and McAuley, 2016) is enhanced by accounting for the temporal dynamics of fashion, with qualitative evidence it can capture popularity changes of items in the past (i.e., Hawaiian shirts gained popularity after 2009). A study by Chen et al. (2015a) looks for correlation between attributes popular in New York fashion shows versus what is seen later on the street. Whereas all of the above center around analyzing *past* (observed) trend data, we propose to forecast the *future* (unobserved) styles that will emerge. To our knowledge, our work is the first to tackle the problem of visual style forecasting, and we offer objective evaluation on large-scale data sets.

Text as side information. Text surrounding fashion images can offer valuable side information. Tag and garment type data can serve as weak supervision for style classifiers (Simo-Serra and Ishikawa, 2016; Simo-Serra et al., 2015). Purely textual features (no visual cues) are used to discover the alignment between words for clothing elements and styles on the fashion social website Polyvore (Vaccaro et al., 2016). Similarly, extensive tags from experts can help learn a representation to predict customer-item match likelihood for recommendation (Bracher et al., 2016). Our method can augment its visual model with text, when available. While *adding* text improves our forecasting, we find that text alone is inadequate; the visual content is essential.

2.2 Transfer learning

Transfer learning (TL) is the ability to leverage experiences and skills obtained previously via a training process to a new task or domain. This feature is an important characteristic of the learning process of human beings. We do not learn tasks in isolation, rather we try to project the experiences we gather throughout our lives to facilitate the learning of

a new task. In the following, we introduce the main concepts, benefits and challenges of transfer learning. For a thorough discussion of this domain and its applications, we refer the reader to the extensive survey by Pan and Yang (2010).

Terminology. In transfer learning we distinguish between two domains (see Figure 2.2a):

1. *Source*: The source domain is where the approach extracts knowledge and accumulates experience through learning various source tasks. It is common to assume that the source domain has enough training samples to develop a model that performs well on the source tasks.

2. *Target*: The target domain is where a new task(s) is defined that is different from those we have encountered before in the source domain. It is in the target where we want to leverage our knowledge from the source domain to learn the target task. It is common to assume that the target domain has few or no training samples for the target task.

Benefits of transfer learning. The ability to transfer gives us the advantage of an initial high performance and to learn faster when handling a new task while using only few trials (or examples). Figure 2.2b shows the three main advantages of transfer learning (Torrey and Shavlik (2009)):

1. *Higher start*: Even before any training on the target, a model that integrate transfered knowledge is expected to have a high initial performance compared to an "ignorant" random model.

2. *Higher slop*: Learning while exploiting previous knowledge should lead to faster improvement in performance through time compared to models learning from scratch in the target.

3. *Higher asymptote*: Finally, with transfer learning, a higher final performance can be achievable since leveraging prior knowledge might improve the model generalization properties to unseen target data.

Negative transfer. The main goal of transfer learning is to improve learning in the target domain. However, if the performance of the target task degrades with the transfered knowledge compared to a target model that learns from scratch then this is referred to as negative transfer. This negative effect of transfer learning might happen, for example, when trying to exploit source knowledge that is not relevant to the target task.

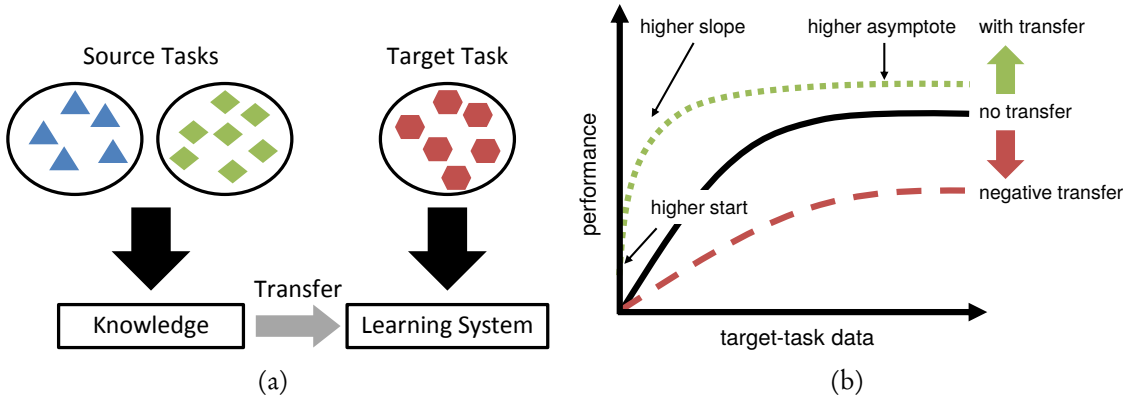


Figure 2.2: In transfer learning the knowledge acquired from the source tasks is transferred to a new target task (a). Transfer learning helps in improving the learning process in target to have a higher initial start, higher slope and better final performance (b). (Figure is adapted from Torrey and Shavlik (2009))

Aspects of transfer learning. A knowledge transfer method usually tries to tackle one or more of the following questions (Pan and Yang (2010)):

1. *What to transfer?* This addresses the type of knowledge most suitable to be transferred across domains. Hence, an important feature of the transferred knowledge is its ability to encode information that is usable and shareable between tasks.

2. *How to transfer?* This addresses the process used to incorporate the transferred knowledge from the source domain in the learning of the target task.

3. *When to transfer?* The source and the target tasks might be very different, and transferring knowledge between them may be harmful and hinders the learning of the target task (negative transfer). Thus, it is important to have a mechanism in order to find out when previous experiences are applicable to the target and when they are not.

In the following, we highlight our own contributions in the context of these three transfer aspects and in the transfer task of zero-shot learning.

2.2.1 What to transfer?

There are three common directions in the literature in regard to the type of knowledge used to transfer across domains (Pan and Yang, 2010): 1) *Instance transfer* where some data samples from source are reused in training the target task (*e.g.* Dai et al., 2007; Fink,

2005; Jiang and Zhai, 2007; Lam et al., 2010; Zadrozny, 2004); 2) *Representation transfer* where a feature encoding that is learned in the source is shared with the target (*e.g.* Dai et al., 2009; Farhadi et al., 2009; Gatys et al., 2016; Liu et al., 2011b; Pan et al., 2011); 3) *Parameter transfer* where the parameters of a model trained in the source are incorporated in the training of the target model, for example as a prior or for initialization (*e.g.* Cao et al., 2010; Fei-Fei et al., 2006; Nater et al., 2011; Oquab et al., 2014; Stark et al., 2009; Tommasi et al., 2014).

Our contribution. In this work, we argue that knowledge representation plays a vital role in the robustness and success of a transfer learning approach. Depending on the type of information encoded in the representation, the transfer approach (whether instance-, representation- or parameter-based transfer) may exhibit different properties and robustness against negative transfer. This aspect is usually ignored in the literature where the focus is only on one of the previously listed directions (*e.g.* Fei-Fei et al., 2006; Lam et al., 2010; Liu et al., 2011b). In Chapter 3, we propose a generic transfer metric learning framework that allows us to consider all three transfer directions jointly. Specifically, we study the impact of knowledge representation on transfer performance and robustness against negative transfer when considering different transfer schemes jointly. We highlight the effectiveness of high-level semantic representations like attributes (Farhadi et al., 2009; Lampert et al., 2009), class similarity (Bart and Ullman, 2005) and our own hierarchical representation against low-level feature representations (Dalal and Triggs, 2005; Laptev et al., 2008; Sivic and Zisserman, 2009). Furthermore, we analyze the effect of semantic correlations and source domain relatedness on transfer performance.

2.2.2 How to transfer?

There is a wide range of approaches that are employed in transfer between source and target domains, like support vector machines (Aytar and Zisserman, 2011; Tommasi et al., 2014), deep neural networks (Gatys et al., 2016; Oquab et al., 2014), boosting (Dai et al., 2007; Yao and Doretto, 2010), graphical models (Dai et al., 2009), metric learning (Zha et al., 2009; Zhang and Yeung, 2010). The choice of model is usually influenced by the type of knowledge that is transfered across domains and the type of the target task.

Our contribution. We consider in this work the attribute-based zero-shot learning for object classification as our main target task. In this context, we introduce three novel models on how to transfer semantic attributes between source and target. We

propose: 1) a guided transfer approach based on class ontology (Chapter 4); 2) a relation-based transfer approach (Chapter 5); and 3) an approach that leverages free-form textual description for learning source-target mappings (Chapter 6). We will discuss these contributions in detail in Section 2.2.4.

2.2.3 When to transfer?

As discussed earlier, not all knowledge in the source is usually transferable to the target. In fact, forcing incompatible knowledge on the target task will likely result in a negative transfer effect (Rosenstein et al., 2005). Thus, a guarding mechanism is needed to decide when it is appropriate to transfer and from which source. This question is usually addressed jointly with the multiple source transfer problem for efficient selection and combination of sources (Jie et al., 2011; Kuzborskij et al., 2017; Tommasi et al., 2010). These approaches use the small amount of data in target to estimate the similarity between target and source categories to decide from which source and when to transfer across domains. However, in extreme settings as zero-shot learning such a transfer scheme is not applicable since there are no target data available during learning to estimate the similarity.

Our contribution. In Chapter 5 we introduce a simple yet effective mechanism to select which knowledge to transfer to a target category. The proposed model leverages its confidence in the predicted semantic link between the source and the target to select which attributes are transferable and which are not.

2.2.4 Zero-shot learning

A common scenario where transfer learning proved to be quite beneficial is when training data for the target task is scarce or not available. Going back to Figure 2.2b, one can see that the highest margin of improvement expected from transfer learning lies in the scarce target data regime (higher start and higher slope). Although, an improvement is also expected when we see larger target training data (higher asymptote), it will practically diminish as we get progressively more data. Thus, the scarce data regime represents a good testbed to evaluate transfer learning approaches.

One of the most popular test settings is zero-shot learning (ZSL) (Larochelle et al., 2008). In ZSL, we target the extreme case of having no training data at all (*i.e.* zero-shot) for the

target task. That is, the transfer approach should rely solely on the obtained knowledge from the source to create a model for the target task.

Formally, let $\mathcal{C} = \{1, 2, 3 \dots N\}$ be a set of categories defined over a data set $\mathcal{D} = \{(x_i, y_i) : x_i \in \mathcal{X} \text{ and } y_i \in \mathcal{C}\}$. In ZSL, $\mathcal{C}$ is split into a disjoint sets of *seen* categories $\mathcal{C}^s$ and *unseen* ones $\mathcal{C}^u$. Where $\mathcal{C}^s \cap \mathcal{C}^u = \varnothing$ and $\mathcal{C}^s \cup \mathcal{C}^u = \mathcal{C}$. Then at training time, we only *see* training examples from $\mathcal{D}^s = \{\mathcal{X}^s, \mathcal{C}^s\}$ and the task is to create a model for the recognition of the unseen classes $\mathcal{C}^u$. This formulation of zero-shot learning we refer to as the conventional setup since it is the most common in the literature. However, there are some other variations of this task which we present next.

Generalized ZSL. The previous definition of ZSL can be seen to be quite restrictive since we assume at test time we only need to differentiate among the unseen classes. However, when considering a realistic situation for recognition in the wild, both seen and unseen classes can appear at test time. Hence, the generalized ZSL setting (Scheirer et al., 2013) does not pose such a strong assumption on the evaluation, and performance is measured when both $\mathcal{C}^s$ and $\mathcal{C}^u$ are considered as candidate labels at test time (Bendale and Boult, 2016; Chao et al., 2016; Socher et al., 2013; Xian et al., 2017a). This setting is quite helpful in shedding light on the bias of the transfer approaches toward the seen classes which were solely used for training.

Transductive ZSL. Here the unlabeled data from the target $\mathcal{X}^u$ is considered to be available at training time and used along with the source labeled data $\{\mathcal{X}^s, \mathcal{C}^s\}$ (*e.g.* Fu et al. (2014b); Kodirov et al. (2015); Rohrbach et al. (2013a); Zhang and Saligrama (2015)) in contrast to the conventional setup where only data from seen classes are exploited (*i.e.* inductive ZSL).

Unless otherwise specified, throughout this thesis we use the term zero-shot learning to refer to the inductive conventional setup.

Side information. In order to carry on ZSL, we need to define a mapping between the seen and the unseen classes to identify how to transfer the knowledge between the two domains. This is usually accomplished with the help of additional information, for example:

1. Semantic attributes: Here, the unseen class is described with a predefined set of semantic attributes, *e.g.* a zebra has stripes, is black and white, has legs and eyes (Farhadi et al., 2009).

2. Class similarities: Here, the unseen class is described by its similarity to other seen classes, *e.g.* a zebra looks like a horse or a donkey but not like a fish or a duck (Bart and Ullman, 2005).

3. Ontology: Here, the unseen class is related to its supercategory or siblings in an ontology, *e.g.* a zebra is an equine (Rohrbach et al., 2011).

4. Articles (Textual Description): Here, the unseen class is described with a free-form text, *e.g.* a zebra is described with this Wikipedia article (Elhoseiny et al., 2013).

These different types of side information require various levels of user involvement in the mapping. For example, attributes require user intervention both at the level of defining the vocabulary and the associations for the unseen classes, while in case of articles, only a description in free-form text is provided for the approach to figure out what and how to transfer to the target.

In this thesis, we introduce multiple attribute-based models with different characteristics for zero-shot learning. We split these models into two main groups based on the level of supervision needed for ZSL.

Supervised ZSL

Perhaps one of the most popular and successful approaches for supervised ZSL is the attribute-based model (Farhadi et al., 2009; Lampert et al., 2009). In this approach, semantic attributes are leveraged as the knowledge units to be transferred between the seen and unseen classes. Since attributes form an intermediate layer between class labels and low-level features (Figure 2.1), they can be effectively used to tackle the case of disjoint train and test categories encountered in ZSL. This is usually carried out by directly learning and transferring attribute classifiers (*e.g.* Farhadi et al., 2009; Jayaraman and Grauman, 2014; Lampert et al., 2013; Rohrbach et al., 2010), or by leveraging the attribute representation to learn new embedding space (*e.g.* Akata et al., 2015; Al-Halah et al., 2014a; Bucher et al., 2016; Changpinyo et al., 2017; Palatucci et al., 2009; Xian et al., 2016; Zhang et al., 2017). A common property of these models is that attributes are learned and transferred in a generic matter to the unseen classes. However, as we argued in Section 2.1.1, attributes exhibit a large variance in their visual representation depending on the described category (*i.e.* the context). In Chapter 4, we propose a novel

supervised ZSL model that leverages the hierarchical structure among the seen classes to capture these differences and to guide the attribute transfer to an unseen class.

Hierarchies represent an attractive structure for knowledge transfer and they have been exploited previously in various ways, for example for parameter transfer (Rohrbach et al., 2011; Salakhutdinov et al., 2010; Shahbaba and Neal, 2007), representation transfer (Al-Halah et al., 2014b) and annotation transfer (Guillaumin and Ferrari, 2012). Of particular relevance to our work is the joint modeling of hierarchy and attributes. Akata et al. (2013) propose an embedding approach that leverages both hierarchical labels along global attributes as side information to learn a joint latent space of visual features and semantics. On the other hand, Deng et al. (2014) propose a hierarchy and exclusion graph which is learned over the various object categories. The graph models binary relations among the classes like mutual exclusion and overlap. They also model similar relations between object categories and global attributes for ZSL.

Our contribution. Differently, we propose a model that exploits the hierarchical structure of the categories in two aspects: 1) We leverage the hierarchy to automatically propagate annotations and learn attributes at different levels of abstraction; Then 2) we use it in guiding the transfer process to select the most promising knowledge source of attributes to share with novel classes. We show that our explicit guided transfer model leads to learning and transferring more discriminative knowledge to the unseen classes and, consequently, to improved ZSL performance especially in the context of fine-grained categories.

Unsupervised ZSL

A drawback of the supervised approach to ZSL is the required user intervention during testing to provide the semantic mapping between the seen and unseen classes (*e.g.* by defining the class-attribute mappings). This is clearly limits the scalability of such models. Hence, there is an increasing interest in conducting ZSL by tapping to an alternative knowledge source where this mapping can be inferred automatically without human supervision. One such an alternative source is lexical ontologies. For example, Rohrbach et al. (2011) propose to leverage the WordNet ontology to transfer visual models between the categories whereas Akata et al. (2015) use the same ontology to generate a hierarchical representation of classes for their joint embedding framework. However, these ontologies

are usually incomplete and they do not contain all categories, especially in the fine-grain scenario.

Word embedding. A different direction leverages powerful word embeddings (Huang et al., 2012; Mikolov et al., 2013) to establish the semantic link between seen and unseen categories. These word embeddings are learned from large-scale text corpora in an unsupervised manner such that words that appear in similar context get similar vector representation (Mikolov et al., 2013). For example, Frome et al. (2013) and Norouzi et al. (2014) propose deep neural models that embed the images in the space of the vector representation of the categories labels. Then, test images are embedded in the vector space and mapped to the closest representation of an unseen class for unsupervised ZSL.

Our contribution. In Chapter 5, we take advantage of these word embeddings in a different manner. We propose a relational model that leverage the word embedding of unseen categories to predict their class-attribute mappings, thus alleviating the need for user intervention. We show that our approach results in a more discriminative ZSL model and outperforms competitor approaches that rely directly on the word embeddings with a wide margin.

Articles. On-line articles or free-form textual description provide yet another knowledge source for ZSL. This valuable and massive source attracted recently more interest in the vision community. Here, each category is associated with an article similar to those found in Wikipedia. Then a form of domain adaptation between vision and language is leveraged to directly predict the classifier weights for an unseen classes based on its textual description embedding (Ba et al., 2015; Elhoseiny et al., 2013; Qiao et al., 2016).

Our contribution. In Chapter 6, we tap to a similar knowledge source to bridge the gap between language and vision. However, in contrast to the previous approaches, our objective is to automatically discover an explicit set of semantic attributes that is compact, discriminative and best describes the categories in our data. Our model benefits from the more discriminative nature of attributes compared to the generic textual embedding while at the same time it does not require any user intervention whether for defining the semantic vocabulary or for setting the class-attribute associations.

Chapter 3

Semantic Knowledge Representation in Transfer Learning

Knowledge representation plays a vital role in our ability in capturing the richness of the surrounding world, how we infer and reason about problems, and how we communicate and transfer information among ourselves. In this chapter, we focus on the latter aspect of a representation, *i.e.* the impact of knowledge representation on transfer learning. Hence we tackle the question *What to transfer?*.

Under *What to transfer?*, one can identify three main directions:

1. Representation transfer, where the goal is to learn a "good" feature representation in the source domain that can be shared with the target. It is important in this case that the representation encodes common knowledge that is not specific to the source task only (*e.g.* Liu et al., 2011b; Pan et al., 2011).
2. Parameter transfer, where parameters or priors learned from the source task are shared with or used to regularize the parameters or priors of the target task model (*e.g.* Nater et al., 2011; Zhang and Yeung, 2010).
3. Instance transfer, where all or some of the samples from the source domain are re-used in the learning of the target task in order to overcome the low number of target training samples (*e.g.* Lam et al., 2010).

It is common in the literature to approach these transfer directions separately (*e.g.* Lam et al., 2010; Liu et al., 2011b; Nater et al., 2011), however they are not mutually exclusive. For example, the choice of the feature representation and how the knowledge is modeled

will eventually impact the efficiency of all three approaches: the representation, instance and parameter transfer. Hence, unlike previous works, we propose in this chapter to consider these directions jointly and analyze their mutual effect. Therefore, we introduce a transfer metric learning framework that enables us to easily and seamlessly integrate and analyze these three transfer approaches. Specifically, we analyze the performance of several semantic representations in comparison to low-level features as a potential similarity spaces for metric learning. We choose action similarity as our target task where given two video samples the goal is to infer whether they exhibit the same action or not. Moreover, we consider semantics that encode aspects of the visual concepts at different levels of abstraction. This give us the opportunity to inspect the interplay of the abstraction level encoded in the representation and the transfer performance. Furthermore, we study the influence of the similarity space representation on the various transfer directions when considered jointly and their resilience to the negative transfer effect.

Contributions. The contributions of our work in this chapter are as follow: i) We show the benefits of using high-level semantics for transfer metric learning; ii) We propose a novel hierarchical knowledge representation that encodes the embedded semantic structure of category similarities in the attribute space, and show its superior performance to other semantic models; iii) We introduce a generic framework for transfer metric learning that improves the transfer performance and reduces the negative transfer effect; iv) We suggest a realistic and challenging evaluation protocol for transfer learning, where the target domain is much more diverse and complex than the source domain.

Publications. This chapter is based on our work that is published in Al-Halah et al. (2014b) and Al-Halah et al. (2016a).

3.1 Semantic similarity spaces

Most metric learning approaches used for object and action recognition are based on low-level features (*e.g.* Davis et al., 2007; Guillaumin et al., 2009; Zha et al., 2009; Zhang and Yeung, 2010). However, we believe that semantics at different levels of complexity can be a better representation for the transfer of source knowledge across domains. While the feature similarity space is usually high-dimensional and dependent on the data distribution in the source domain, the semantic similarity space is lower dimensional, concise,

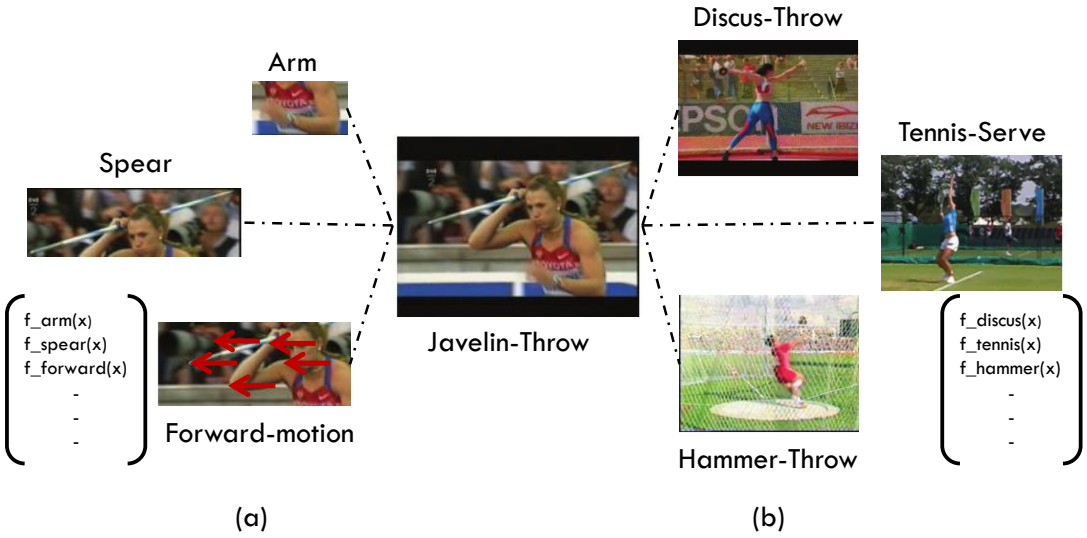


Figure 3.1: The attribute-based representation (a) captures some fine-grained visual properties of an action, such as motion pattern, body parts, and objects; while the category-based representation (b) encodes the overall similarity of a certain action to the various categories.

and more robust to changes in the data distributions between target and source domains. In the following, we will describe the two most common semantic spaces that are used as an intermediate representation, the *attribute similarity space* and the *category similarity space*. In the former, instances are represented by their visual properties (Figure 3.1a), and in the latter, by their resemblance to other previously learned categories (Figure 3.1b). Furthermore, we introduce a third and novel similarity space, the *hierarchical similarity space* (Figure 3.2). Here, the instances are represented by a hierarchical structure, that captures their visual properties at different resolution levels.

3.1.1 Attribute similarity space

Attributes define an intermediate representation between low-level features and high level categories (Farhadi et al., 2009; Lampert et al., 2009). Semantic attributes describe an entity regarding its visual appearance (*e.g. is-round*), parts (*e.g. has-ears*), and motion patterns (*e.g. forward-motion*). Hence, they can be easily shared across categories and even used to predict unseen classes if the classes can be described in terms of the same vocabulary. In the attribute similarity space $\mathcal{A}$, the different semantic attributes span the bases of the space, where each axis encodes the presence of one of the attributes as well as its intensity (or confidence for binary attributes) in a certain data instance, see Figure 3.1a.

Samples that belong to the same category are close to each other in $\mathcal{A}$ since they share the same properties, and they will form a tight cluster of points that are distinguishable from other samples of different categories. Therefore, the lower the distance between points in $\mathcal{A}$, the more attributes they have in common, and consequently, the more similar they are conceptually. The samples in the d-dimensional feature space $\mathcal{X}^d$ are mapped to space $\mathcal{A}$ using:

$$\begin{aligned} f_{\mathcal{A}} &: \mathcal{X}^d \rightarrow \mathcal{A}^n \text{ and} \\ f_{\mathcal{A}}(x) &= [f_{a_1}(x), f_{a_2}(x), \ldots, f_{a_n}(x)]^T, \end{aligned} \tag{3.1}$$

where $f_{a_i}(x)$ is the prediction score of attribute a_i on instance x, and n is the number of defined attributes.

3.1.2 Category similarity space

Humans do not only use visual properties to describe entities in their environment, but also inter-class relationships. Consider for example the action class *triple-jump*; it can be described as an action similar to the classes *run* and *jump*. This intra-class similarity pattern is not specific to a certain sample of *triple-jump*, rather it characterizes all samples that belong to this category. In that sense, the category similarity space $\mathcal{C}$ provides a meaningful semantic space to compare different actions in terms of their similarity patterns to previously learned categories (Bart and Ullman, 2005). In $\mathcal{C}$, the bases are spanned by the predefined categories, where each axis encodes the resemblance of a sample to a learned category, see Figure 3.1b. Samples from the feature space are mapped to $\mathcal{C}$ using:

$$\begin{aligned} f_{\mathcal{C}} &: \mathcal{X}^d \rightarrow \mathcal{C}^m \text{ and} \\ f_{\mathcal{C}}(x) &= [f_{c_1}(x), f_{c_2}(x), \ldots, f_{c_m}(x)]^T, \end{aligned} \tag{3.2}$$

where $f_{c_i}(x)$ is the prediction score of category c_i on instance x, and m is the number of categories.

3.1.3 Hierarchical similarity space

A common property of the previously defined spaces is that both of them represent semantics at a single layer of resolution. That is, both of them ignore the implicit

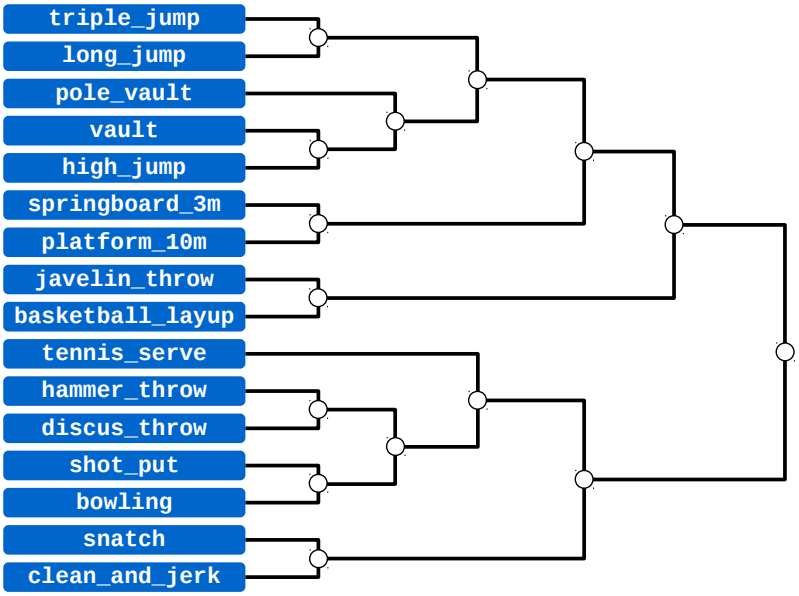
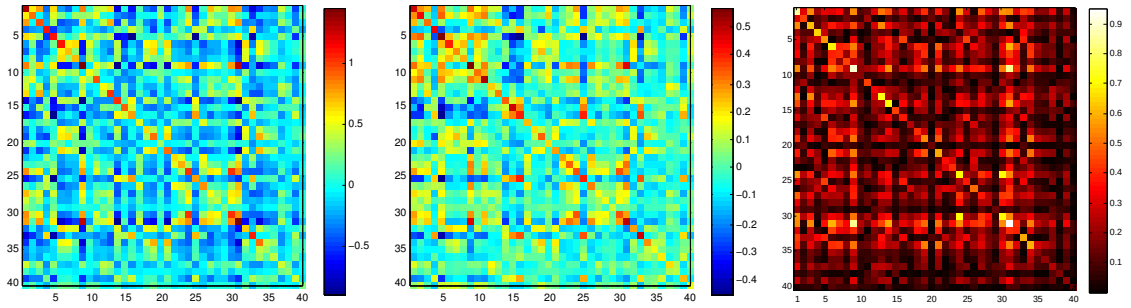


Figure 3.2: The learned hierarchical representation of action classes in Olympic Sports where actions are grouped based on their intra-class similarity in the attribute space.

structure that exists in the semantic space. Such structure allows us to have semantics depicted at various levels of resolution or complexity, which enriches the knowledge obtained in the source domain and provides a better semantic representation of samples. Consider for example the action categories *walk*, *jump* and *jump-forward*. Since the latter class is partially similar to the former ones, it would be better represented by a super-class consisting of the other two categories, *i.e.* by *walk-jump*. Then learning the common pattern between these two classes could provide a higher category of semantics that improves the classification performance for *jump-forward*.

Hence, we propose to learn the structure of this hierarchical model by exploiting the similarity between categories in the attribute space. Attributes correspond to observable properties of the categories, and the more attributes are shared between a couple of categories, the higher is the overall visual similarity between the pairs. Thus, assuming that each of the action categories is described with a vector of semantic attributes of length n ($\mathbf{a}^{c_i} = \{a_j\}_1^n$), we can exploit this representation by defining a distance function f to group categories close to each other based on their similarity in the attribute space (Figure 3.2), *i.e.*:

$$f : \mathcal{C} \times \mathcal{C} \rightarrow \mathbb{R} : f(c_i, c_j) = \mathrm{d}(\mathbf{a}^{c_i}, \mathbf{a}^{c_j}), \tag{3.3}$$



(a) (b) (c)

Figure 3.3: The correlations of semantic attributes on (a) Olympic Sports and (b) ASLAN and (c) their absolute differences.

where $\mathrm{d}(\cdot,\cdot)$ is a distance function in the attribute space.

We construct a hierarchical representation by applying an agglomerative hierarchical clustering algorithm over the attribute representation of the classes to get a dendrogram depicting the hierarchical clustering result. The dendrogram is then used to construct the final action hierarchy by interpreting the action classes as leaf nodes and the intermediate clusters at different similarity threshold levels as inner nodes. The sub- and super-cluster relations are translated to *is-a* relations in the tree structure. For our case of using binary attributes to describe the various action classes, we use a hierarchical k-means clustering algorithm with $f(c_i, c_j) = \| \mathbf{a}^{c_i}, \mathbf{a}^{c_j} \|_1$ to capture the similarity in $\mathcal{A}^n$. Then instances $x \in \mathcal{X}^d$ are represented in the hierarchical similarity space $\mathcal{H}$ using:

$$\begin{aligned} f_{\mathcal{H}} &: \mathcal{X}^d \rightarrow \mathcal{H}^k \text{ and} \\ f_{\mathcal{H}}(x) &= [f_{n_1}(x), f_{n_2}(x), \ldots, f_{n_k}(x)]^T, \end{aligned} \tag{3.4}$$

where $f_{n_i}(x)$ is the prediction score of node i in the hierarchy, and k is the number of nodes.

The node classifiers are trained in a child-vs-parent manner (Marszalek and Schmid, 2007), *i.e.* if $\mathrm{pos}(n_i) = \bigcup \mathrm{pos}(n_j)$ is the positive set of node n_i, where $n_j \in \mathrm{child}(n_i)$, then the classifier f_{n_i} is trained on $\mathrm{pos}(n_i)$ as the positive set against $\{\mathrm{pos}(n_p) \setminus \mathrm{pos}(n_i)\}$ as the negative set, where $n_p = \mathrm{parent}(n_i)$.

3.2 Decorrelated normalized space

It is important to notice that when the semantic similarity spaces are learned, also the correlations of the semantics are implicitly modeled in theses spaces. While some of the correlations are meaningful and correspond to real world co-occurrences (*e.g.* eyes & nose, weak & small), many of these correlations come from the sampling bias of the data sets. Most likely, these correlations are significantly different between the source and the target domain since they arise from the respective semantics distribution in the sampled data from each domain. For example, Figure 3.3 shows the respective different correlations of semantic attributes in two data sets. Since it is hard to separate real world correlations from artificial ones without referring to an oracle, maintaining such knowledge in the representation when transferring across domains will likely result in a negative transfer effect (see Section 3.5.5). Therefore, it is quite important to eliminate the correlations learned in the source domain from the semantic spaces in order to restrain the negative transfer.

The decorrelation of the semantic similarity space $\mathcal{S}$ ($\mathcal{S} \in \{\mathcal{A}, \mathcal{C}, \mathcal{H}\}$) can be efficiently achieved using the whitening transformation. Such a transformation has been successfully used before for attribute decorrelation (Al-Halah et al., 2014a) and for removing co-occurrence patterns from the bag-of-words model (Jegou and Chum, 2012), for example. The correlations are modeled by the covariance matrix $\mathbf{\Omega} = \mathbf{Y}\mathbf{Y}^T$ where $\mathbf{Y}$ represents the data matrix from space $\mathcal{S}$. By transforming $\mathbf{\Omega}$ to the identity matrix, $\mathbf{Y}$ is whitened and the data is transformed to a space $\tilde{\mathcal{S}}$ where the bases are decorrelated and each given the same importance.

The whitening transformation $\mathbf{W}$ of $\mathcal{S}$ is obtained by analyzing the covariance matrix $\mathbf{\Omega}$ such that:

$$\mathbf{W} = \mathbf{V}\mathbf{\Sigma}^{-1/2} \text{ and } \mathbf{\Omega} = \mathbf{V}\mathbf{\Sigma}\mathbf{V}^T, \tag{3.5}$$

where $\mathbf{\Sigma}$ is a diagonal matrix having the eigenvalues of $\mathbf{\Omega}$ as its diagonal elements ($\mathbf{\Sigma}_{ii} = \lambda_i$). $\mathbf{V}$ contains in its columns the relevant eigenvectors of the covariance matrix. To have a robust estimation of $\mathbf{W}$, we ignore the eigenvectors in $\mathbf{V}$ that correspond to very small eigenvalues ($\lambda_i < \theta$), *i.e.*:

$$\tilde{\mathbf{W}} = \tilde{\mathbf{V}}\tilde{\mathbf{\Sigma}}^{-1/2} \text{ where } \tilde{\mathbf{\Sigma}}_{ii} \geq \theta. \tag{3.6}$$

Furthermore, we normalize the vectors in the truncated whitened space by their norms. Thus, the samples representation in $\mathcal{S}$ is transformed to the decorrelated normalized space $\mathcal{S}_{dns}$ using:

$$f_{\mathcal{S}_{dns}}(x) = \tilde{\mathbf{W}}^T y / \|\tilde{\mathbf{W}}^T y\|_2 \text{ where } y = f_{\mathcal{S}}(x) \tag{3.7}$$

3.3 Metric learning

In order to measure similarity between samples in the different semantic spaces, we need to learn an appropriate metric. For that purpose we use the Logistic Discriminant based Metric Learning (LDML) from Guillaumin et al. (2009) to adapt to the positive and negative similarity relations in the target data set.

In LDML, the probability of two data points (x_i, x_j) to be similar (*i.e.* belong to the same category) based on a Mahalanobis distance is modeled using the sigmoid function $(\sigma(z) = (1 + exp(-z))^{-1})$ as follows:

$$p(y_i = y_j | \mathbf{x}_i, \mathbf{x}_j; \mathbf{M}, b) = \sigma(b - d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j)), \tag{3.8}$$

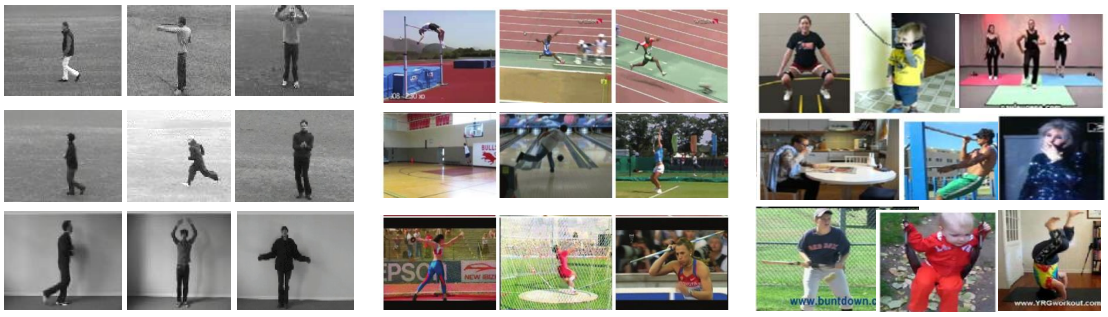
where b is a bias term and $d_{\mathbf{M}}(\cdot, \cdot)$ is the Mahalanobis distance based on matrix $\mathbf{M} \in \mathbb{R}^{d \times d}$:

$$d_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j). \tag{3.9}$$

Then the problem is formulated as a standard logistic discriminant model where the maximum log-likelihood is used to optimize the parameters of the model. The log-likelihood is defined as:

$$\mathcal{L} = \sum_n t_n \ln(p_n) + (1 - t_n) \ln(1 - p_n), \tag{3.10}$$

where $t_n = 1$ when data pair is similar (*i.e.* $y_i = y_j$) and 0 otherwise, and p_n as defined in Eq. 3.8. LDML has a convex optimization objective which guarantees an optimum global solution. However, our approach is not restricted to a certain metric learning method as we will show later in the evaluation (Section 3.5.5).



(a) KTH (b) Olympic Sports (c) ASLAN

Figure 3.4: Samples from the three data sets used in our evaluation for action similarity. While KTH has only 6 classes, Olympic Sports has 16 and ASLAN has 432 different categories.

3.4 Evaluation setup

Data sets. We evaluate our framework using three publicly available data sets:

- **Olympic Sports** (Niebles et al., 2010), which contains 781 videos of 16 action classes collected from YouTube, like *hammer-throw*, *tennis-serve* and *triple-jump*. We use the attribute annotations provided by Liu et al. (2011a), where the actions are labeled with 40 semantic attributes describing motion, pose and objects, such as *lift-something*, *throw-away*, *two-arms-open* and *outdoor* (Figure 3.4b).

- **ASLAN** (Kliper-Gross et al., 2012), which is collected for the main task of comparing actions (similar/not-similar). It has 432 action classes with more than 3600 video samples and each class has on average 8.5 video samples with more than 100 classes having only one sample each (Figure 3.4c). The data set has two views (evaluation splits). View-1 has 1800 pairs splits into 1200 for training and 600 for testing, and we use this view for most of our experiments. View-2 has 6000 pairs with 10-fold cross validation setup which we use for full-scale testing.

- **KTH** (Schüldt et al., 2004), which contains six basic action classes (*i.e. boxing*, *clapping*, *waving*, *jogging*, *running*, and *walking*). In our experiments the classes are described with 10 semantic attributes by Liu et al. (2011a) (Figure 3.4a).

Transfer setup. In our experimental settings, we use Olympic Sports (or KTH) as source and ASLAN as target data set. This addresses a realistic and very difficult scenario for transfer learning that has been overlooked in previous studies. Collecting and labeling

samples for actions is time consuming and expensive. Consequently, the labeled data (source set) tends to be small and simple in terms of diversity and coverage compared to the target. Our evaluation setup tackles this very challenging problem because of the high diversity in ASLAN compared to Olympic Sports (432 to 16 different classes).

Features. As a video descriptor, we use the bag-of-words (BoW) model based on histograms of oriented gradients and optical flow (HOGHOF) from Laptev et al. (2008) with a vocabulary of size 4000. We use that BoW model to train the different classifiers, presented in Section 3.1, on the training split of Olympic Sports. The features are preprocessed with a power transform (Arandjelovic and Zisserman, 2012) with $\alpha = 0.3$ before training a linear support vector machine. The parameters of the SVM classifiers are estimated using a 5-fold cross validation. For the decorrelated normalized space, we set $\theta = 10^{-8}$. To simulate a real transfer learning problem, we do no further training of classifiers or the BoW model on the target set (ASLAN), and only the similarity metric is adapted from the available training data of the target set to infer a reasonable comparison metric in each of the semantic similarity spaces. The threshold of similarity is automatically learned using a linear SVM trained on the distances between training pairs.

3.5 Experiments

We first evaluate our transfer metric learning framework for the three transfer types: representation-, instance- and parameter-transfer (see Figure 3.5). Then we analyze the effect of using different source domains in Section 3.5.4 and the impact of the proposed semantic space decorrelation approach on transfer performance in Section 3.5.5. Finally, we go beyond the case of scarce training data in target and conduct a full scale evaluation on the target data set and compare to a set of common metric learning methods with no transfer (Section 3.5.6).

3.5.1 Representation transfer

We first test the performance of different semantic spaces compared to the common low-level similarity space. We learn the different knowledge representations on Olympic Sports and transfer them to ASLAN where we use the View-1 training/testing split as defined by Kliper-Gross et al. (2012). Furthermore, we vary the number of training pairs

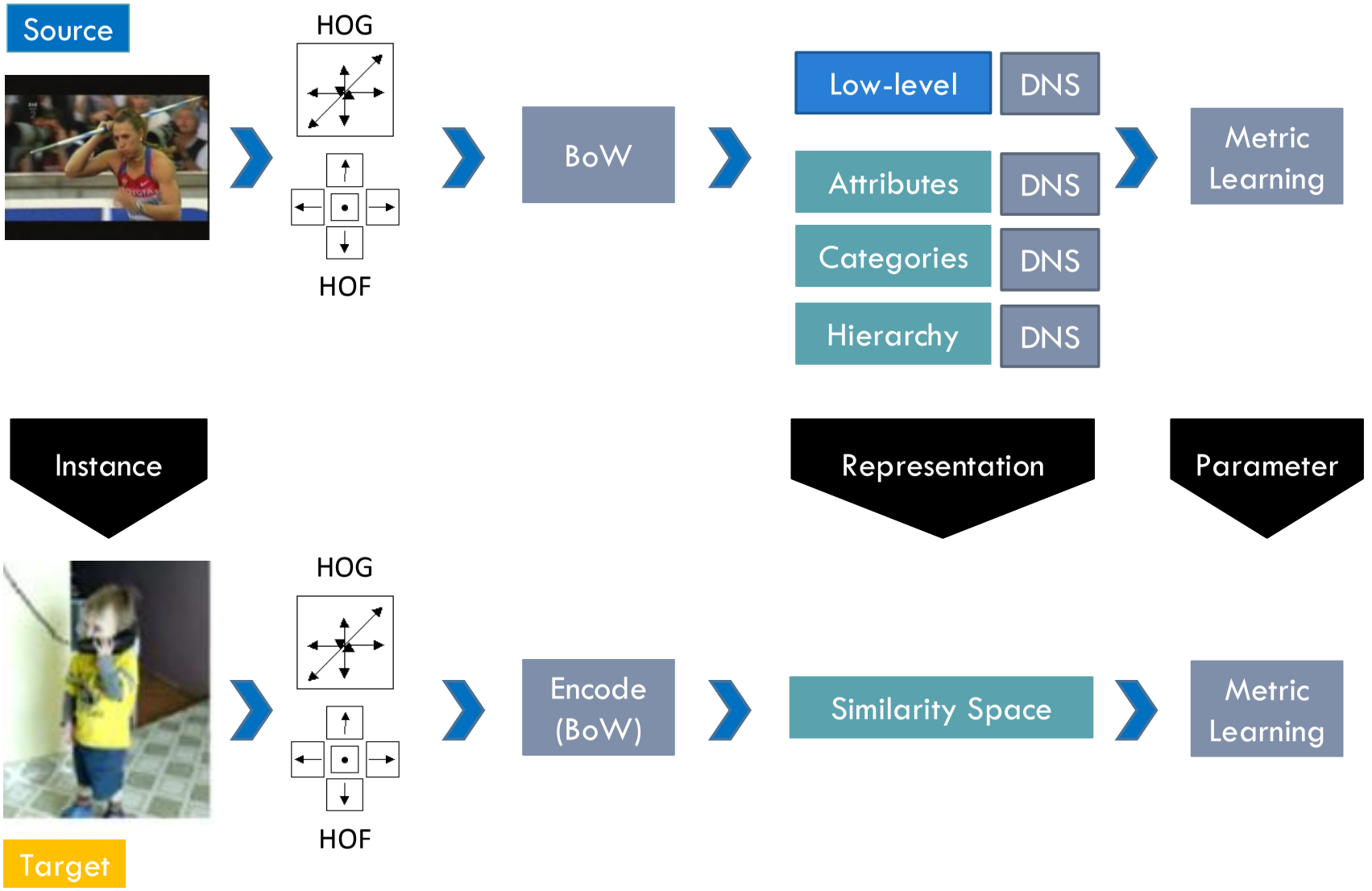


Figure 3.5: Our transfer metric learning framework along with the instance-, representation- and parameter-transfer approaches.

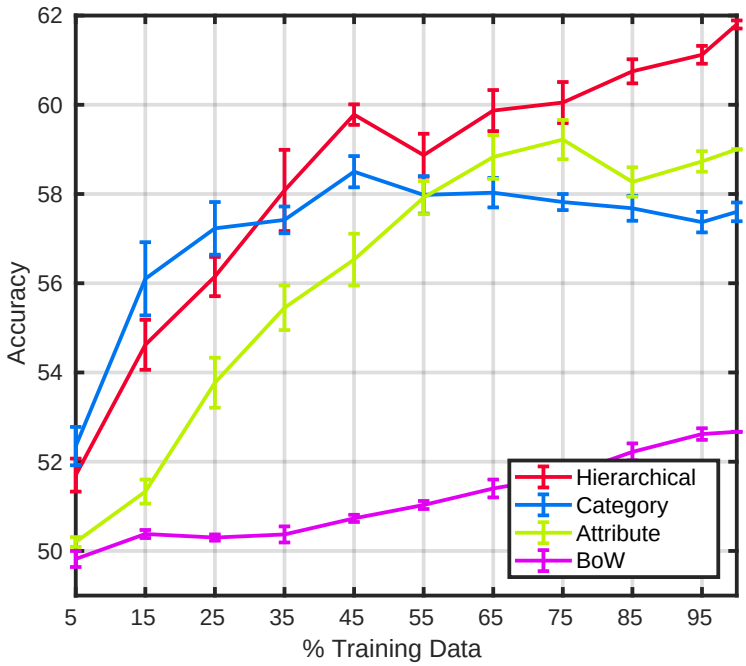
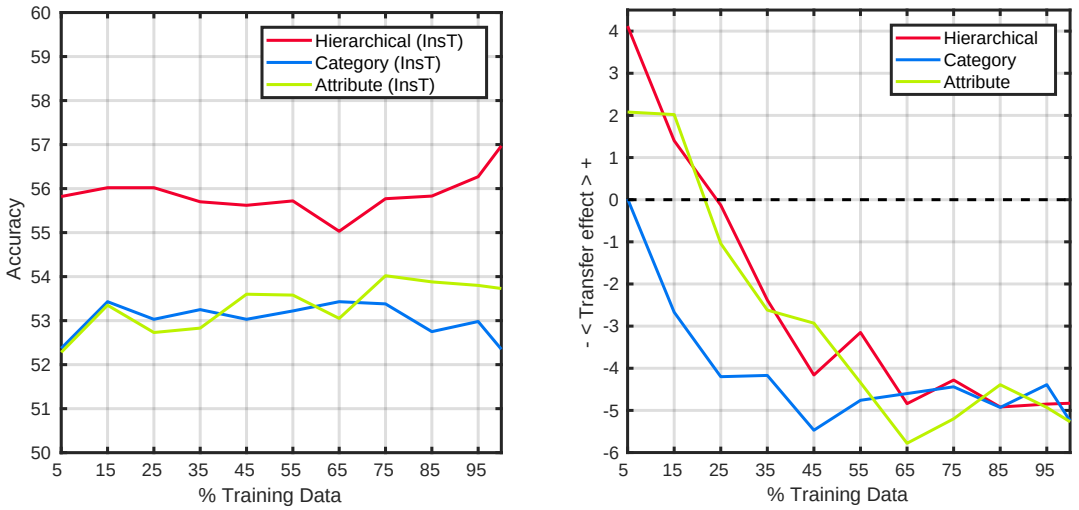


Figure 3.6: Overall performance of different semantic similarity spaces regarding various sizes of the target training set. The transferred high-level semantics clearly outperform the low-level representation.

of similar and dissimilar actions from 5% to 100% of the target training set. For each run, a random subset of the target training pairs is selected to learn the similarity and then we evaluate the model on the test split. This is repeated 10 times, and we report the average accuracy and standard error of similarity classification as seen in Figure 3.6. For the feature space, we first reduce the dimensionality of the features to 128 using principle component analysis since it is intractable to use the full feature vector with LDML (Guillaumin et al., 2009).

In Figure 3.6 it's interesting to see that all semantic spaces exhibit a higher start, a higher slope and a higher asymptote compared to the low-level features space. Moreover, the hierarchical and category similarity spaces outperform the attribute space when the training data is scarce. However, when more than half of the training data is available, the attribute space seems to do better than the category space while the proposed hierarchical model outperforms both. This confirms our previous hypothesis on the importance of high-level semantics and their ability to generalize well when transferred to other domains.

Another interesting aspect of the high-level semantics is their scalability. The high-level representation is much more compact than its low-level counterpart. For example,



(a) (b)

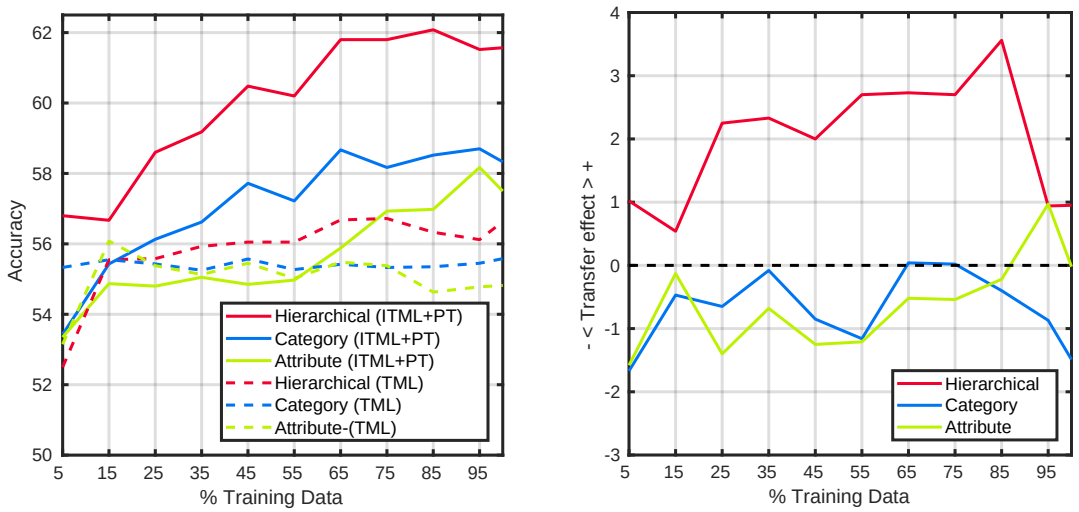
Figure 3.7: The performance of the different knowledge representations when using instance transfer (a) and its negative transfer effect, *i.e.* the difference in performance when not using instance transfer (b).

in our case the dimensionality of semantic representations ranges between 16 and 40 while the low-level feature vectors are of 4000 dimensions. Consequently, the semantic representations are more scalable to big data sets since the computation cost of most of the metric learning algorithms is heavily impacted by the representation dimensionality.

Moreover, adding new concepts for the attribute and category similarity spaces results in a linear expansion in the dimensionality of the similarity space where only the new concept classifiers need to be trained. Adding a new concept to the hierarchical space is equivalent to inserting a leaf node to a binary tree. It requires the retraining of the ancestors of that leaf node which is of logarithmic complexity in term of the number of nodes in the tree. This cost is usually lower than trying to increase the descriptiveness of the low-level features which usually results in much higher computation cost. For instance, adding a new cluster to the bag-of-words requires rerunning the clustering algorithm over all samples again.

3.5.2 Instance transfer

In this transfer setup, a random group of training pairs from the source (Olympic Sports) are added to the training set in the target (ASLAN). Similar to the previous experiment, the representation is learned in the source and we vary the size of the target's training



(a) (b)

Figure 3.8: (a) Comparison of the proposed parameter transfer approach (ITML+PT) to the state-of-the-art (TML) and (b) its robustness using the various semantic representations to negative transfer effect.

set. The metric is learned from the transfered instances from source along with the available target training instances. We report the similarity accuracy and we also analyze the transfer effect (positive or negative) as the difference in performance (measured by accuracy) between using instance transfer and without using it.

We see in Figure 3.7b that when the target's data is too small (less than 25% of the training pairs), both the hierarchical and attribute spaces take advantage of the additional transferred samples from the source. However, when the size of target training set increases, the transferred instances prevent the metric learning to adapt to the actual data distribution of the target. Hence, it produces a significant negative transfer for all semantic spaces. Nonetheless, the hierarchical representation still maintains higher performance compared to the other alternatives (Figure 3.7a).

This type of transfer introduces a significant change in the data distribution of the target training set which is not reflected in the test set, resulting in performance deterioration. It also shows how the target and source sets are different and how the transfer settings are challenging.

3.5.3 Parameter transfer

In parameter transfer, the parameters learned in the source domain are used to regularize or to aid the learning task in the target. The similarity metric learning method LDML does not allow for parameter transfer in its formulation. Hence, we propose instead a simple parameter transfer approach based on the information-theoretic metric learning (ITML) from Davis et al. (2007). The metric learning problem in ITML is defined as:

$$\min_{\mathbf{M}} \ \mathrm{KL}(p(x, \mathbf{M}_0) \parallel p(x, \mathbf{M})), \tag{3.11}$$

where KL is the Kullback-Leibler divergence between two Gaussian distributions corresponding to a prior matrix $\mathbf{M}_0$ and the learned one $\mathbf{M}$. Additionally, some constraints on the distances are incorporated in learning the metric:

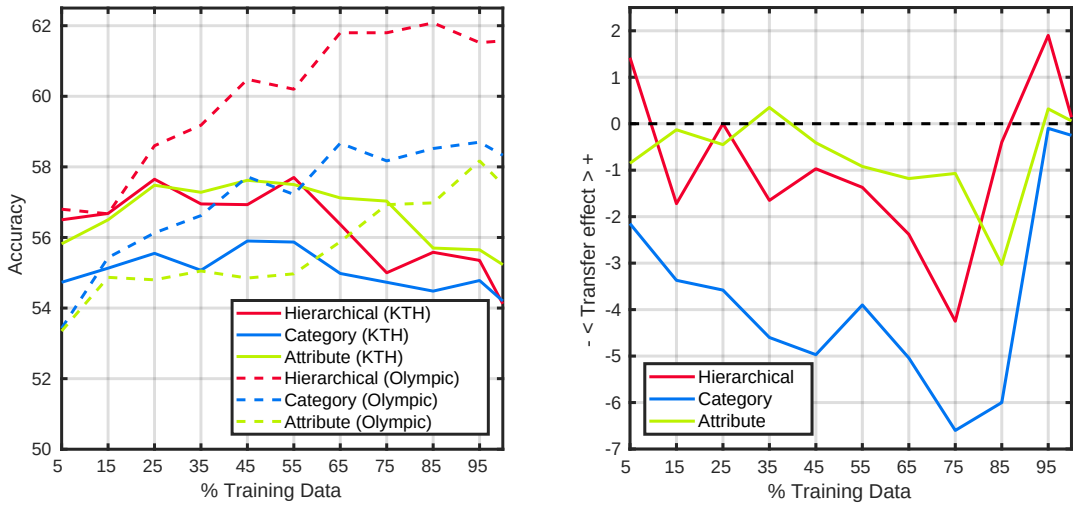
$$\begin{aligned} d_{\mathbf{M}}(\mathbf{v}_i, \mathbf{v}_j) &\leq u \quad if\ (\mathbf{v}_i, \mathbf{v}_j) \in \mathbf{S} \\ d_{\mathbf{M}}(\mathbf{v}_i, \mathbf{v}_j) &\geqslant l \quad if\ (\mathbf{v}_i, \mathbf{v}_j) \in \mathbf{D}, \end{aligned} \tag{3.12}$$

where u and l are the upper and lower bound of distances between similar ($\mathbf{S}$) and dissimilar ($\mathbf{D}$) pairs respectively.

In Eq. 3.11, the common assumption is that the data is Gaussian distributed and the prior $\mathbf{M}_0$ is either set to the inverse of the covariance matrix or the identity matrix $\mathbf{I}$ (euclidean metric). In contrast, we suggest to adapt ITML to carry on parameter transfer by setting the prior to be the metric learned in the source data set ($\mathbf{M}_0 = \mathbf{M}_{source}$). In other words, following Eq. 3.11, the metric learning in the target set is regularized to be close to the source metric ($\mathbf{M}_{source}$) while at the same time satisfying the constraints on the pair distances in the target set (Eq. 3.12).

We evaluate the parameter transfer setting by learning first the similarity metric for each of the three semantic spaces (Section 3.1) in the source set (Olympic Sports) and transfer that metric using Eq. 3.11 to the target set (ASLAN). The metric in Olympic Sports is learned by randomly generating 1500 pairs of similar and dissimilar actions in the source, and then using the standard proposed framework to learn the similarity. During testing, we use the same settings as described in Section 3.5.1.

We compare ITML with the proposed parameter transfer approach (ITML+PT) to state-of-the-art transfer metric learning (TML) from Zhang and Yeung (2010). The



(a) (b) b

Figure 3.9: Performance of parameter transfer when using KTH as the source set (a) instead of Olympic Sports and (b) its negative transfer effect.

parameters for both ITML and TML are set following the recommendations suggested by Davis et al. (2007) and Zhang and Yeung (2010), respectively.

Interestingly, our ITML+PT approach outperforms state-of-the-art TML as seen in Figure 3.8a. TML seems to have a saturated performance after using just 15% of the training set and slightly profits from the different semantic representations. ITML+PT, on the other hand, clearly takes advantage of the characteristics of the different similarity spaces and has a higher initial performance. This can be due to the formulation of TML as a special case of multi-task metric learning, and the assumption that the tasks (source and target) share a common data distribution which is not the case here.

We analyze the transfer effect (as in Section 3.5.2) as the difference in performance between using the parameter transfer and without (i.e. setting $\mathbf{M}_0 = \mathbf{I}$ in Eq. 3.11). While the hierarchical representation evidently benefits from parameter transfer, both the attribute and category similarity representations show a negative transfer effect (Figure 3.8b). As motivated in the introduction of this chapter, it seems that the robustness of the model against negative transfer is increased when the level of semantic knowledge encoded in it is higher.

3.5.4 Varying the source set

While Olympic Sports contains videos collected from YouTube with a lot of variations (like camera motion, occlusion and varying background), KTH contains only simple motion patterns and is recorded with a uniform background (see Figure 3.4). In this experiment, we test the effect of replacing Olympic Sports with the more simpler KTH data set as the source of the transfer metric learning.

We use a similar setup as in Section 3.5.3. A similarity matrix ($\mathbf{M}_{source}$) is learned in KTH (the source set) from a set of randomly generated pairs of action samples. Then, $\mathbf{M}_{source}$ is transferred using ITML+PT for metric learning in ASLAN.

Figure 3.9a shows the performance of the transfer process when using KTH against Olympic Sports as the source set. In general, KTH-based transfer performs worse than the alternative source data set. Another observation is that the KTH-based transfer performance curves do not monotonically increase as its Olympic-based counterparts. The performance of the various KTH-based knowledge representations start to exhibit a drop when the training set in the target gets bigger than 50%. This is expected since KTH contains much less variation in its samples. Hence, it is harder to extract rich semantic representations and learn useful similarity relations. This is evident in Figure 3.9b, where the difference in performance against using $\mathbf{M}_0 = \mathbf{I}$ (*i.e.* no parameter transfer) for the KTH-based transfer is shown. Clearly, the similarity relations learned among the classes and attributes in KTH do not generalize well to ASLAN and a significant negative transfer is produced. Nonetheless, when the training set in the target is tiny ($\leq 15\%$) the transferred knowledge from the very simple source (KTH) aides the learning process in the target, performing on par with their Olympic Sports counterparts. This suggests, that transferring semantics from simple sources may still be beneficial under harsh transfer setting (*i.e.* extremely scarce target training data).

3.5.5 Semantic space decorrelation

Here, we evaluate the impact of the proposed decorrelated normalized space (DNS) on the transfer process effectiveness. Similar to Section 3.5.1, we use Olympic Sports as source for representation learning and transfer them to ASLAN. We test our framework with and without the DNS transformation.

Space / Metric	ITML	LDML	KISSME	Cov^{-1}	L_2	SVM
$\mathcal{H}$	58.38	54.23	55.50	51.03	52.85	**57.58**
$\mathcal{H}_{dns}$	**60.62**	**61.80**	**60.98**	**56.98**	**56.33**	56.90
$\mathcal{A}$	55.08	57.80	55.50	50.87	54.00	57.50
$\mathcal{A}_{dns}$	**57.52**	**59.00**	**58.42**	**56.37**	**54.83**	**57.73**
$\mathcal{C}$	57.65	57.50	56.77	54.17	54.50	53.17
$\mathcal{C}_{dns}$	**59.82**	**57.60**	**61.20**	**57.23**	**55.50**	**57.63**
$\mathcal{X}$	55.38	**58.95**	49.83	49.67	50.00	50.00
$\mathcal{X}_{dns}$	**56.07**	52.67	**54.33**	**53.00**	**56.53**	**56.05**

Table 3.1: The effect of the decorrelated normalized space (DNS) on the performance of popular metric learning methods.

Furthermore, since our framework is not restricted to a specific metric learning approach, we test (along with LDML from Section 3.3) two state-of-the-art metric learning methods: ITML (Davis et al., 2007) and KISSME (Martin et al., 2012); and two commonly used metrics: the Mahalanobis distance using the inverse of the covariance (Cov^{-1}) and the Euclidean distance (L_2). Additionally, we train an SVM on the element wise multiplication of the training pairs $[x_1 . * x_2]$ as the sixth approach for learning similarities (using the absolute difference $|x_1 - x_2|$ or the concatenation of the previous two produced inferior performance). We use all training pairs available in the target and report the accuracy of the different knowledge representations with and without using DNS.

In Table 3.1, we see that in most of the cases (22 out of 24), the decorrelated space increased the performance of the transfer metric (up to 7% absolute increase). DNS is quite generic, and it improves the performance of most of the metric learning approaches. Even when using simple metrics like L_2 and Cov^{-1}, DNS helps to learn a better similarity metric.

On the other hand, both the category and the hierarchical spaces appear to perform better than the attribute model; and the best performance (61.80%) is obtained by using our hierarchical model with LDML.

3.5.6 Full scale evaluation

It is common in transfer learning literature to focus only on the case when the training data in the target is scarce. However, considering the scenario of a large training set in

the target is also beneficial. Evaluating in such settings helps us to put the transfer metric learning method in perspective to standard methods that learn knowledge representation in the target set and have enough information to adapt well to the target data distribution.

For that purpose, we evaluate on ASLAN View-2 which has 6000 pairs of similar and dissimilar actions. We follow the benchmark setup suggested by Kliper-Gross et al. (2012). That is, a 10-fold cross validation is carried out on View-2 and the performance is reported in terms of average accuracy and area under receiver operating characteristic (ROC) curve. For an in-target representation modeling, we compare to the approaches of Kliper-Gross et al. (2012). They propose to extract three feature types: HOG, HOF, and HNF (Laptev et al., 2008); and learn a BoW model of size 5000 for each to represent video samples. They use 12 different similarity metrics to compare actions based on each of these three representations and their combination. We report in Table 3.2 the results of their best single similarity metric and the results of using the combination of the 12 metrics as stated by Kliper-Gross et al. (2012).

We notice in Table 3.2 that the transfer metric method performs as well as the methods that are based on a representation learned in target domain. Even when 12 different similarities and 3 feature representations are combined, the gain in performance of the in-target method is only 1.7% in accuracy. This is an impressive performance for the transfer metric learning approach, bearing in mind the diversity of the target compared to the source set (432 to 16 classes) and that the data representation learned in the source was never adapted to model changes in the target domain. Furthermore, the performance of the different semantic spaces in the transfer metric approach follows the complexity level of semantics encoded in the model. The proposed hierarchical representation has the best performance, followed by the category, and the attribute spaces.

3.6 Summary and discussion

We proposed a generic framework for transfer metric learning and showed the influence of knowledge representation on different transfer options. In our experiments, we demonstrated that high-level semantics have better transfer properties and encode richer transferable knowledge in comparison to low-level features. Furthermore, we introduced a hierarchical representation that models the embedded structure of category similarities in the attribute space. The proposed hierarchical model performed best and was more robust to negative transfer effect. In addition, different metric learning methods benefit

from the proposed transfer framework. We evaluated on very challenging settings where the target set is much more complex and diverse in comparison to the source set. Nonetheless, we showed that even when the knowledge source is limited, transfer learning can still be beneficial if an appropriate semantic representation is used. Finally, a large-scale evaluation showed impressive results of the transfer approach; the performance is on par with methods that use feature representations learned in the target domain.

Discussion. In our analysis of the mutual effect of the different transfer options, we adopted simple yet effective transfer schemes. Nevertheless, these options can be extended to further improve the transfer performance and robustness. For example, in instance-transfer we adopted a simple model that transfers a fixed set of source instances to the target. This set is given the same weight in learning as the target samples. However, adopting a more dynamic approach that can assign variable weights to transferred samples (Dai et al., 2007) might reduce or eliminate the negative transfer effect that we saw in our experiments. Similarly, when multiple sources are available it would be beneficial to have a mechanism to select the most suitable source to transfer from (Tommasi et al., 2014; Yao and Doretto, 2010) or to decide beforehand whether to transfer or not.

Representation Learning in Source w/ Transfer Learning (Ours)	$\mathcal{H}_{dns}$	$\mathcal{C}_{dns}$	$\mathcal{A}_{dns}$	$\mathcal{X}_{dns}$
#Dimension	30	16	40	128
LDML	$\mathbf{59.18 \pm 0.98 (62.16)}$	$57.85 \pm 1.02 (60.57)$	$57.30 \pm 0.58 (60.85)$	$56.97 \pm 0.69 (60.15)$
Representation Learning in Target w/o Transfer Learning (Kliper-Gross et al., 2012)	HOG	HOF	HNF	HOG+HOF+HNF
#Dimension	5000	5000	5000	3×5000
$\sqrt{\sum(x_1. * x_2)}$	$58.55 \pm 0.80 (61.59)$	$56.82 \pm 0.57 (58.56)$	$\mathbf{58.87 \pm 0.89 (62.16)}$	$60.08 \pm 1.08 (63.89)$
Hellinger	$53.22 \pm 0.61 (54.19)$	$53.77 \pm 0.72 (56.00)$	$53.77 \pm 0.73 (55.80)$	$54.83 \pm 0.90 (57.18)$
Chi-Square	$53.28 \pm 0.69 (54.42)$	$53.42 \pm 0.62 (55.79)$	$53.87 \pm 0.72 (55.97)$	$54.97 \pm 0.97 (57.13)$
12 Similarities	$59.78 \pm 0.82 (63.20)$	$56.68 \pm 0.56 (58.97)$	$59.47 \pm 0.66 (63.30)$	$\mathbf{60.88 \pm 0.77 (65.30)}$

Table 3.2: Large scale evaluation on View-2 of the ASLAN data set. Our transfer metric learning with *in-source* representation learning performs in line with standard metric learning approaches that have the advantage to adapt well to the target data distribution when the target data is abundant.

Chapter 4

Hierarchical Transfer of Semantic Attributes

Structured representation of concepts and objects is part of the human understanding of the surrounding world. We usually try to combine objects into certain groups based on a common criteria like functionality or visual similarity. This helps us to better learn the commonality as well as the differences in and across groups. Moreover, we saw in the previous chapter the influence of semantic representation on transfer learning and how encoding structured information with various abstraction levels in the representation impacts the transfer performance and its robustness against negative transfer.

However, in the prevailing approach of attribute-based recognition (like DAP), the attributes are learned from all seen classes and then they are shared or transfered to classify unseen ones (Section 2.1). Such a global learning approach of attributes clearly does not account for the high intra-attribute variance that may exist in the data. Learning an attribute from all seen classes results in capturing the visual concept in a quite abstract manner and ignores the fine visual properties of the attribute that help in discriminating a sub group of classes from another. Hence, subsets of classes that share similar attributes cannot be distinguished easily (*e.g.* fine-grained categories).

Consider for example the attribute *beak* in Figure 4.1. The global model of *beak* would learn that a beak is an elongated extension at a certain position relative to the head. That is, it ignores the fine and distinctive differences among the *wide curved-end beak* of the albatross species; the *long thin beak* of the hummingbirds; and the *round short beak* of the jays. In other words, the global model is incapable of capturing the rich diversity that

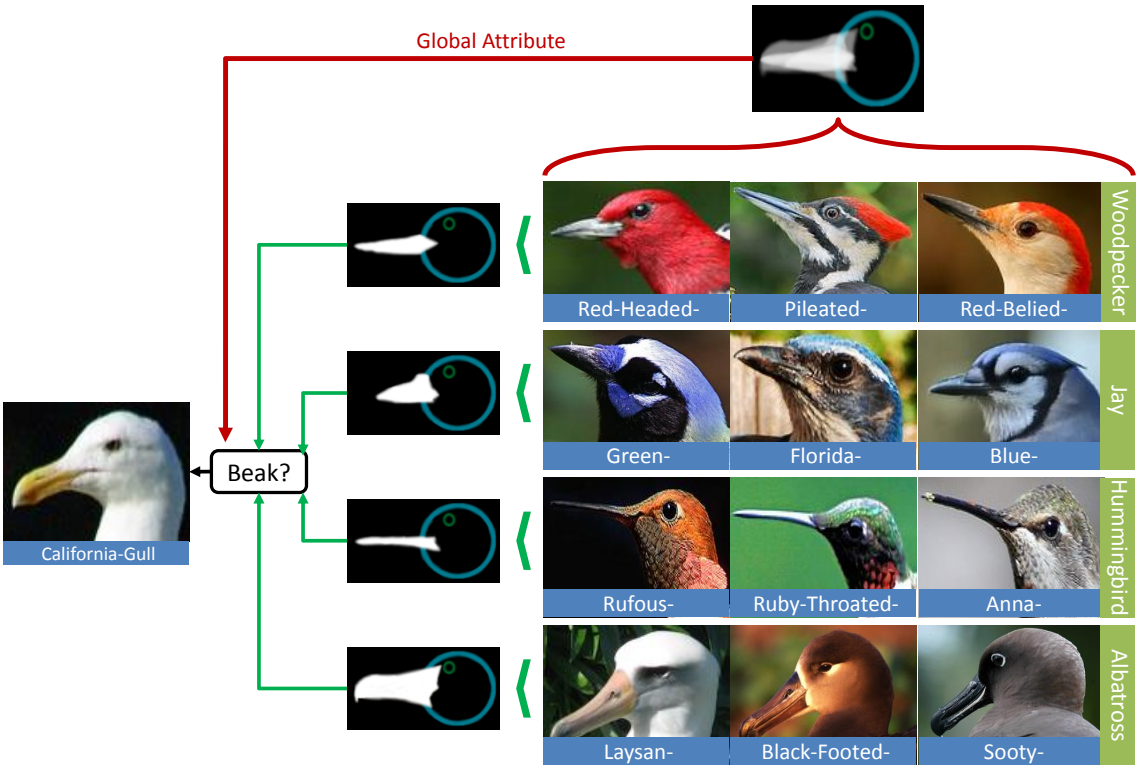


Figure 4.1: The high intra-attribute variance is better represented at different semantic levels of abstraction. This helps in directing the transfer process to identify the most suitable source of knowledge to share with a novel class.

already exists in the source set. Hence, more generic and less discriminative attributes are usually transferred to the unseen class. On the other hand, an approach that can learn these different versions of the same attribute (*beak*) is quite beneficial. Such an approach would create a richer knowledge repository from which more suitable and discriminative attributes can be shared and transferred across categories. Moreover, the object hierarchy groups the classes based on their overall visual similarity; thus provides a natural way to guide the transfer process to share information from the knowledge sources that will most likely contain relative information. Accordingly, knowing that both *Gull* and *Albatross* are *Seabirds*, it is intuitive and probably more discriminative to describe the beak of the *California-Gull* as an *albatross-like-beak*.

Contributions. In this chapter, we propose a novel approach to model attributes at different levels of abstraction, from the most specific that distinguish one class from another to the most generic that are learned over all categories. Our model can leverage the structured knowledge in the source to select and transfer the suitable attributes across the different abstraction levels to be shared with an unseen class. We present a simple

yet efficient approach that outperforms the global attributes model with up to 15% in zero-shot accuracy. Furthermore, we show in the evaluation that our model takes advantage of the category granularity in the source set and shows a higher margin of improvement as the granularity get finer.

Publication. This chapter is based on our work that is published in Al-Halah and Stiefelhagen (2015b).

4.1 Overview

Let $\mathcal{C} = \{c_k\}_{k=1}^{K}$ be the set of seen categories, *i.e.* with training examples, and $\mathcal{Z} = \{z_l\}_{l=1}^{L}$ the unseen categories such that $\mathcal{C} \cap \mathcal{Z} = \varnothing$. A set of semantic attributes $\mathcal{A} = \{a_m\}_{m=1}^{M}$ describe all classes in $\mathcal{C} \cup \mathcal{Z}$. A directed acyclic graph $\mathcal{H} = (\mathcal{N}, \mathcal{E})$ defines a hierarchy over the classes, with nodes $\mathcal{N} = \{n_i\}_{i=1}^{I}$ and edges $\mathcal{E} = \{e_{ij} : n_i, n_j \in \mathcal{N}\}$.

Starting with a set of classes $\mathcal{C}$, global attributes $\mathcal{A}$ and a hierarchy $\mathcal{H}$ in the category space, our approach constitutes three main steps (see Figure 4.2): 1) we automatically populate the hierarchy with additional attribute labels (Section 4.2); 2) model these attributes to capture subtle differences between similar categories (Section 4.3); 3) finally, we use a hierarchy-guided transfer to select the proper attributes to share with a novel class (Section 4.4).

4.2 Attribute label transfer

Since we start with only the attribute labels of the main classes, the inner nodes of the hierarchical representation are unlabeled. Hence, we need to populate the hierarchy with attribute labels in order to obtain the attribute-based representation of all nodes. We adopt here the intuitive idea that if a sub-category has attribute a_i then the super category has the attribute as well. For example, the super-category *Dog* has attributes *meet-teeth* and *small* since *German-Shepherd* and *Chihuahua* have these attributes, respectively (see Figure 4.2a). Note, this does not mean that all dog instances now have all attributes that appear in its sub-categories, rather that these instances may exhibit a subset of these *active* attributes. In other words, the label transfer does not change the underlaying attribute labels of the instances but rather setup the active attributes through the hierarchy to guide the learning and the transfer process afterwards.

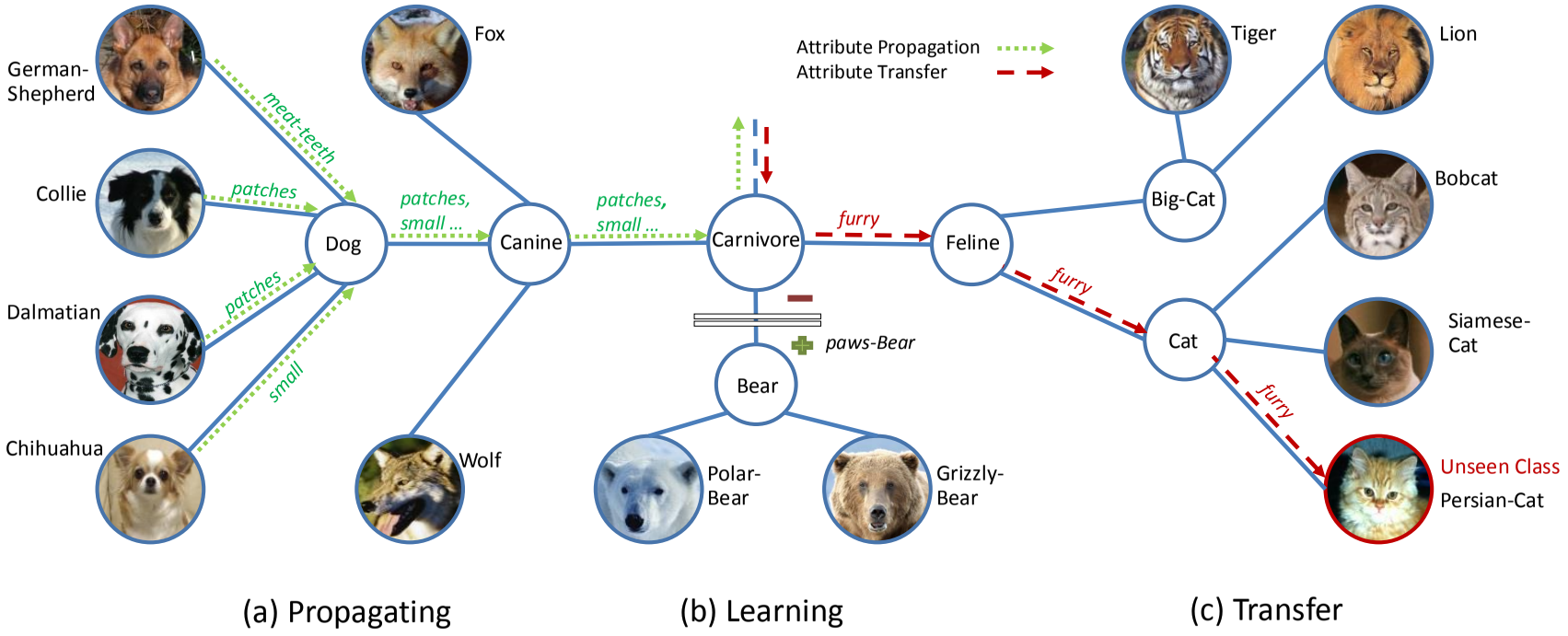


Figure 4.2: Illustrative figure of our hierarchical transfer model. We leverage the hierarchy in many aspects: (a) attribute label propagation, (b) modeling the intra-attribute variations and (c) guiding the transfer process to automatically select and share the most relevant knowledge source for a novel class.

To that end, the attribute label transfer can be carried out by exploiting the hierarchy of the object categories and populating the labels from the leaf node to the root in a bottom-up fashion. Formally, the active attribute a_m for a node n_j in $\mathcal{H}$ is defined as:

$$a_m^{n_j} = 1 \text{ if } \exists\, a_m^{n_i} = 1 \text{ and } n_i \in \text{child}(n_j), \tag{4.1}$$

such that $\text{child}(n_j)$ is the set of all children of node n_j, *i.e.* $\text{child}(n_j) = \{n_i : e_{ji} \in \mathcal{H}\}$. Following this scheme, the root of $\mathcal{H}$ will have all attributes active.

4.3 Learning attributes at different levels of abstraction

Having all nodes in $\mathcal{H}$ populated with attribute annotations, now we want to learn the fine differences among the various abstraction levels of the attributes. That is, we want to learn a model that discriminates the *paws* of *Bears* from the *paws* of other *Carnivores* and another one that discriminates those from the *paws* of other *Mammals*, etc. This leads naturally to adopting a child-vs-parent learning scheme (Section 3.1.3) for our attribute models.

Let $S^+(a_m^{n_j})$ be the support set of attribute a_m for node n_j. A support set for a node's attribute contains all data samples labeled with that attribute at that node level of the hierarchy. In other words:

$$\text{S}^+(a_m^{n_j}) = \bigcup_{n_i \in \text{child}(n_j)} \text{S}^+(a_m^{n_i}) \cup \text{lbl}(a_m^{n_j}). \tag{4.2}$$

where $\text{lbl}(a_m^{n_j})$ is the set of samples from class $n_j \in \mathcal{C}$ with attribute a_m, and if $n_j \notin \mathcal{C}$ then $\text{lbl}(a_m^{n_j}) = \varnothing$:

$$\text{lbl}(a_m^{n_j}) = \{x_i : y_i = n_j \text{ and } a_m^{x_i} = 1\}. \tag{4.3}$$

The support set definition is recursive. That is, S^+ for the inner node n_j will expand till it includes all the samples that are labeled with that particular attribute a_m from all the leaf nodes of the sub-tree rooted by n_j.

To train a discriminative model for a_m at a certain level in the hierarchy n_i, we use the following positive T_+ and negative T_- training sets:

$$T^+ = \text{S}^+(a_m^{n_i}) \quad T^- = \text{S}^+(a_m^{n_j}) - \text{S}^+(a_m^{n_i}), \text{ such that } n_i \in \text{child}(n_j). \tag{4.4}$$

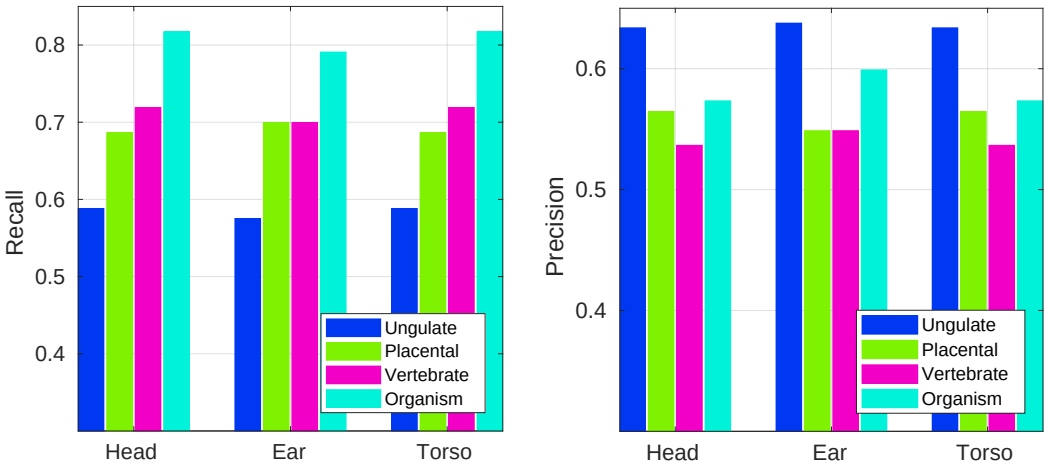


Figure 4.3: Recall and precision primacy of attributes classifiers *Head*, *Ear* and *Torso* at different abstraction levels when evaluated on test samples corresponding to the leaf node "Horse". Attribute models learned close to the leaves of the hierarchy (*e.g.* "Ungulate") tend to have higher precision and lower recall compared to those learned at a higher abstraction level.

Going back to our example from Figure 4.2b. A model for attribute a^{Bear}_{paws} will have all samples with $a^{Polar-Bear}_{paws}$ and $a^{Grizzly-Bear}_{paws}$ in the positive set T^+ while other *paws* samples from class *Carnivore* and excluding those from *Bear* in the negative set T^-. Note that at the root n_r we have no parent node and the attributes at this level are learned using the common 1-vs-all scheme. Hence, the models are trained to discriminate one attribute a_m from the rest of the attributes which map naturally to the notion of global attributes.

4.4 Hierarchical transfer

Now that we have learned attribute models at different levels of abstraction, we need to identify the most suitable ones to transfer for a new unseen category. We leverage here the structure in the category space $\mathcal{H}$ to guide the knowledge transfer process. Consider for example the unseen class *Persian-Cat* in Figure 4.2c. Rather than transferring the generic global attribute *furry* to the *Persian-Cat*, we are able now to select a fine-grained version of that attribute that is probably more suitable for this category, *e.g.* a^{Cat}_{furry} or a^{Feline}_{furry}.

Analyzing the properties of the learned attribute models throughout the hierarchy, we notice that those that are closer to the leaf nodes have higher precision and lower recall from the ones closer to the root (see Figure 4.3). This is an expected property since the intra-attribute variance increases as we move up in the hierarchy. While the attribute models at the lower levels in $\mathcal{H}$ captures the fine differences in the attribute that distinguish few classes from the others, the ones at the higher levels in the hierarchy captures the common visual properties of the attribute among larger and more diverse classes. We find out, similar to Zweig and Weinshall (2007), that a model which is an ensemble of classifiers with such opposite recall and precision properties has better performance than the constituent classifiers. Hence, rather than transferring an attribute model from a specific abstraction level to the unseen category (*e.g.* $a^{Cat}_{furry} \rightarrow a^{Persian-Cat}_{furry}$), we transfer an ensemble model specific for the unseen category which is constructed from the various abstraction levels in $\mathcal{H}$.

Let $s_{n_i}(a^{n_i}_m|\mathbf{x})$ be the prediction score of attribute a_m from node n_i given image $\mathbf{x}$, and $\mathrm{anc}(n_i)$ is the set of ancestor nodes of n_i given $\mathcal{H}$. Then the attribute model for the unseen class $z_l \in \mathcal{Z}$ is defined as:

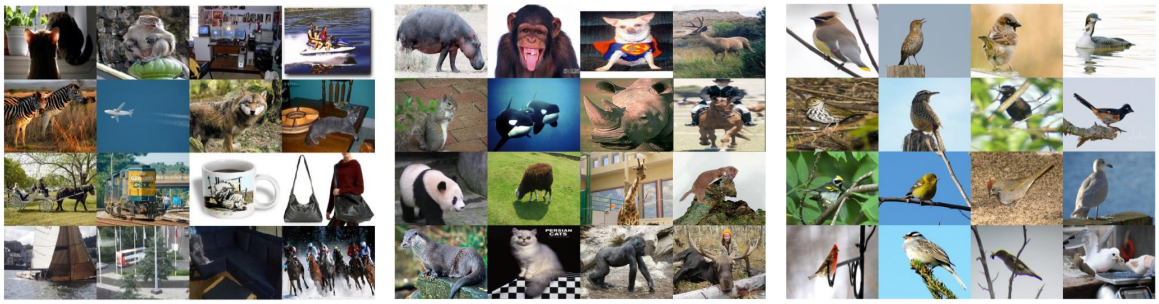
$$s_{z_l}(a^{z_l}_m|\mathbf{x}) = \frac{\sum\limits_{n_i \in \mathrm{anc}(z_l)} [[a^{z_l}_m = a^{n_i}_m]]\; s_{n_i}(a^{n_i}_m|\mathbf{x})}{\sum\limits_{n_i \in \mathrm{anc}(z_l)} [[a^{z_l}_m = a^{n_i}_m]]}, \tag{4.5}$$

where $[[\cdot]]$ is the Iverson bracket which takes the value of 1 when the condition within the bracket is true and 0 otherwise.

Consequently, to predict category z_l in image x we average pool all attribute predictions of z_l:

$$s(z_l|\mathbf{x}) = \frac{\sum\limits_{m=1}^{M} [[a^{z_l}_m = 1]]\; s(a^{z_l}_m|\mathbf{x})}{\sum\limits_{m=1}^{M} [[a^{z_l}_m = 1]]}. \tag{4.6}$$

Similar to Rohrbach et al. (2011), we find that using normalized scores in Eq. 4.6 improves performance. Hence, the prediction scores are normalized to have zero mean and unit standard deviation.



(a) aPascal/aYahoo (aPY) (b) Animals with Attributes (AwA) (c) CUB-200-2011 Birds (CUB)

Figure 4.4: Samples from the three data sets with varying object granularity. While aPY has 32 categories of vehicles, animals and artifacts, AwA has 50 animal classes and CUB has 200 species of birds.

4.5 Evaluation setup

Data sets. We evaluate our Hierarchical Attribute Transfer model (HAT) for zero-shot learning on three data sets with varying granularity of classes (Figure 4.4). This give us the chance to see how the performance of the proposed HAT model varies with regards to the complexity of the embedded knowledge in the source set.

1. *The aPascal/aYahoo (aPY)* (Farhadi et al. (2009)) contains two subsets. The first (aP) uses $12,695$ images and 20 categories from PASCAL VOC 2008 (Everingham et al. (2008)). The second (aY) has 12 disjoint classes and $2,644$ images collected from the Yahoo image search engine. Per-image labels of 64 binary attributes are provided for both subsets. In the predefined zero-shot split, aP is used for training and aY for testing.

2. *The Animals with Attributes (AwA)* (Lampert et al. (2009)) consists of $30,475$ images of 50 classes of animals. They are described with 85 semantic attributes on the class level. The 50 classes have a predefined split for zero-shot learning, where 40 classes are used for training and 10 for testing.

3. *The CUB-200-2011 Birds (CUB)* (Wah et al. (2011)) has 200 bird classes and $11,788$ images. Each image is labeled with 312 attributes. Unlike the previous two, there is no predefined split for this data set. For our experiments, we randomly select 150 classes for training and 50 for testing.

Data set	Attributes	Per-Image	Classes	Seen	Unseen	Images
aPY	64	✓	32	20	12	15,339
AwA	85		50	40	10	30,475
CUB	312	✓	200	150	50	11,788

Table 4.1: Statistics of the three data sets used in our evaluation.

Hierarchy. We learn the object hierarchies using the WordNet ontology (Miller, 1995). By querying the ontology with the category labels we extract a tree that brings the classes into a hierarchical ordering. Subsequently, we prune the hierarchy to remove intermediate nodes that have a single child.

Deep features. Motivated by the impressive success of deep Convolutional Neural Networks (CNN), we encode the images using a CNN-based deep representation to train the attribute classifiers. We use the CNN model CNN-M2K provided by Chatfield et al. (2014) which has a structure similar to AlexNet (Krizhevsky et al., 2012). We also use the BVLC implementation (Jia et al., 2014) of GoogLeNet (Szegedy et al., 2014) which has a much deeper architecture. Both networks are pre-trained on ILSVRC-2012 (Deng et al., 2009) for object classification. Note, one should be careful when using pre-trained deep representation from ImangeNet ILSVRC for zero-shot learning since samples from some of the unseen categories might have been already used in training the neural network as is the case for AwA (see Xian et al. (2017b) for more details).

To extract the deep representation, we follow the best practice found in Chatfield et al. (2014). The image is resized to 256×256. Then 5 image crops obtained from center and corners with their flipped versions are fed to the CNN. The output of the last hidden layer is extracted, sum-pooled and L2-normalized to be used as our deep-features to train the different models.

Classifiers. For all our attribute classifiers, we use L2-regularized logistic regression (Fan et al., 2008). The classifiers parameters estimated using 5-folds cross validation.

4.6 Experiments

In the following experiments, we first evaluate the performance of the deep features, proposed in the previous section, against the common shallow features for attribute prediction (Section 4.6.1). Then, we provide a thorough evaluation of our model for

Features	aPP	aPY	AwA	CUB
Shallow	84.12	70.91	71.16	60.78
CNN-M2K	92.82	**80.73**	78.64	76.03
GoogLeNet	**93.63**	80.03	**79.47**	**76.59**

Table 4.2: Attribute prediction performance (mean AUC) using *deep* and *shallow* feature representations.

zero-shot learning (Section 4.6.2) and consider various ZSL setups like the influence of attribute labeling level, *i.e.* image-based vs. class-based (Section 4.6.4); recognition of unseen classes with unknown attribute associations (Section 4.6.5); and the impact of category granularity in source on ZSL performance (Section 4.6.3).

4.6.1 Attribute prediction with deep embeddings

We first evaluate the performance of the deep features that we use in learning attribute classifiers. We compare the attribute prediction performance using the deep features proposed in Section 4.5 against shallow features. Here, we use the "shallow" term to refer to all manually crafted features like SIFT (Lowe, 1999), SURF (Bay et al., 2008), HOG (Dalal and Triggs, 2005), *etc*. We use the precomputed features provided by Farhadi et al. (2009) and Lampert et al. (2009) as a shallow representation of 9751 and 10940 dimensions for aPY and AwA respectively. For CUB, we encode the images with Fisher vectors (Perronnin and Dance, 2007) based on SIFT and Color descriptors followed by dimensionality reduction using principle component analysis (PCA) (Wold et al., 1987) to get a 6456 dimensional vector which we use as the shallow representation.

We consider two evaluation setups in aPY: 1) within-category attribute prediction (aPP), *i.e.* the evaluation is done on the aPascal testing set; 2) across-category prediction (aPY), *i.e.* we evaluate on the disjoint aYahoo set. In both cases, the attributes are learned using the aPascal training set. For AwA and CUB, we use the zero-shot testing setups defined in (Section 4.5).

In Table 4.2, we see the performance of the three representations in terms of mean area under the receiver operating characteristic curve (AUC) of the attribute predictions. The deep-feature models constantly outperform their counterpart across all data sets with an increase between 7% and 15%. Although the CNN model was learned for object classification on a different data set; still the automatically learned features by the CNN

Model	Features	aPY	AwA	CUB
Global Attributes				
IAP (Lampert et al., 2013)	Shallow	16.9	42.2	-
DAP (Lampert et al., 2013)	Shallow	19.1	41.4	-
SJE (Akata et al., 2015)	CNN-4K	-	45.9	30.0
SJE (Akata et al., 2015)	GoogLeNet	-	52.0	37.8
DAP	CNN-M2k	31.9	54.0	33.7
DAP	GoogLeNet	35.5	59.9	36.7
ENS	CNN-M2k	43.1	57.7	37.3
ENS	GoogLeNet	42.8	63.5	39.4
Hierarchy /and Global Attributes				
AHLE (Akata et al., 2013)	Shallow	-	43.5	17.0
HEX (Deng et al., 2014)	DECAF	-	44.2	-
$\text{SJE}^{\text{WordNet}}$ (Akata et al., 2015)	GoogLeNet	-	51.2	20.6
AMP(SR+SE) (Fu et al., 2015)	CNN-4K	-	66.0	-
Ours				
HAT	DECAF	-	48.9	-
HAT	CNN-M2k	**46.3**	68.8	48.6
HAT	GoogLeNet	45.4	**74.9**	**51.8**

Table 4.3: Zero-shot multi-class accuracy on the three data sets. We compare with state-of-the-art methods under different setting: 1) using global attributes and 2) using a combination of attributes and hierarchical information.

layers are quite generic and work well for attribute prediction. CNNs automatically learn to capture features with varying complexity at each layer. While the lower layers learn features like edges and color patches, the higher layers learn much complex structures of the object like parts (Zeiler and Fergus, 2014). Many of these correspond directly to semantic attributes (Escorcia et al., 2015). Moreover, the non-linear image encoder learned by the CNNs project the samples, by the last hidden layer, to a space where they can be linearly separated according to their categories. This makes the used deep representation quite suitable to our simple linear classifiers.

4.6.2 Zero-shot learning

To populate the hierarchy with attributes (Section 4.2), our model requires class-based attribute descriptions. Hence, for aPY and CUB we average all image-based attribute vectors of each class to calculate the class-attribute occurrence matrices. Then, the binary class-attribute notations are created by thresholding the resulting occurrence matrix at its overall mean value. Along with our Hierarchical Attribute Transfer model (HAT),

Model	Features	aPY	AwA	CUB
DAP	CNN-M2k	**87.3**	88.5	82.2
HAT (ours)	CNN-M2k	87.1	**92.0**	**94.9**
DAP	GoogLeNet	**88.0**	89.8	83.9
HAT (ours)	GoogLeNet	86.7	**94.1**	**95.5**

Table 4.4: Zero-shot mean AUC under ROC curve for the test classes.

we also train and evaluate two common baselines for global attributes using our deep features: 1) The Direct Attribute Prediction model (DAP), where the class prediction is formulated as a MAP estimation (Lampert et al., 2009); 2) The Ensemble model (ENS), that combines the predictions of the attributes using a sum formulation similar to the one we use in Eq. 4.6 but based on global attributes (Rohrbach et al., 2011). We also evaluate our model using the DECAF features (Donahue et al., 2014) provided by Lampert et al. (2009) for direct comparison with some of state-of-the-art methods.

HAT vs. state-of-the-art. In Table 4.3, we report the normalized multi-class accuracy on the three test sets. Compared to methods that leverage global binary attributes and hierarchies (or graphs), our model outperforms the state-of-the-art with a wide margin. In AHLE (Akata et al., 2013) and SJE (Akata et al., 2015), a join embedding model is learned either based on joint modeling of global attributes and hierarchical information with shallow features (Akata et al., 2013) or WordNet-based representations and deep features (Akata et al., 2015). Whereas Deng et al. (2014) leverage the structure in the category space using a hierarchy and exclusion graph (HEX) between objects and attributes. Finally, in Fu et al. (2015) a semantic graph of class prototypes in the attribute space is learned and combined with a probabilistic model to perform zero-shot learning. Nonetheless, our model improves over the best state-of-the-art results by 9% (AwA) and 31% (CUB).

HAT vs. deep global attributes baseline. Compared to our strong baseline with deep features (DAP & ENS), HAT still performs the best in terms of both multi-class accuracy and mean AUC (Table 4.4). Figure 4.5 shows the highest ranking results obtained by the three models for each test class in the AwA data set. While distinctive classes like *Chimpanzee* and *Humpback-Whale* are correctly classified by all models, both DAP and ENS confuse visually similar classes that share many global attributes like *Leopard* & *Persian-Cat* and *Rat* & *Raccoon*, and more samples are wrongly ranked high by this model. To the contrary, HAT learns the fine differences among the shared attributes of

(a) DAP

(b) ENS

(c) HAT (ours)

Figure 4.5: The highest scoring results of DAP and HAT (CNN-M2K) for each test class in AwA. (Best viewed in color)

these classes which helps in discriminating them efficiently. For example, HAT learns the differences between the attributes of *Big-Cat* and *Cat* (Figure 4.2c) which facilitates the separation among the novel classes *Leopard* and *Persian-Cat* (Figure 4.5). Furthermore, we find that normalizing the prediction scores of the novel classes (Eq. 4.6) makes the scores more comparable. Without normalization and with CNN-M2k features, our model achieves accuracies of 38.3% (aPY), 63.1% (AwA) and 44.4% (CUB), and we

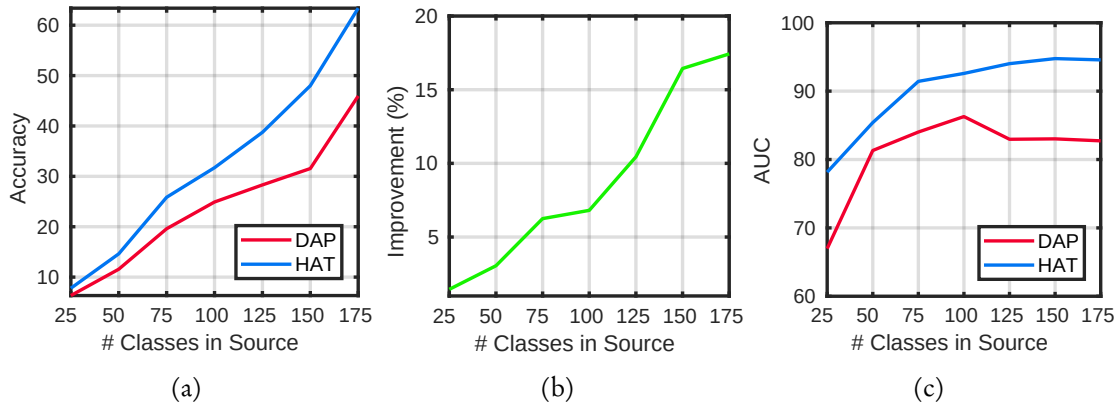


Figure 4.6: The performance of DAP and HAT (ours) in CUB with varying number of classes in the source as demonstrated with (a) multi-class accuracy, (b) absolute improvement of HAT over DAP, and (c) mean AUC.

notice a similar drop in performance for ENS. Normalization improves the accuracy of both the baseline (ENS) and our model (HAT). However, this requires that the test data is available as a batch at test time.

The relative improvement in accuracy of our HAT model over the baselines is 6% (aPY), 18% (AwA) and 31% (CUB). This trend nicely follows the level of granularity of the objects in the data sets. Our model takes advantage of the underlying structure and the commonality among the classes and it is able to distinguish between fine grained classes more efficiently. On the other hand, it is harder for the baseline models (DAP & ENS) to distinguish such fine grained objects using the abstract global attributes.

In the rest of the experiments, we report the results when using the GoogLeNet features for our model and the baselines.

4.6.3 Granularity of the source set

In the following experiment, we vary the complexity of the knowledge contained in the source (the number of seen classes) compared to the target (the unseen classes). This helps to have a better understanding of the characteristics of the different models as the richness of the embedded information in the source changes. We use the CUB data set and start with a random set of 25 classes to be in the source. We gradually increase the source set with additional 25 random classes. At each step, the rest of the 200 classes is used as the target set to conduct zero-shot classification.

In Figure 4.6a we see that when the source is relatively poor and contains less structured knowledge, both DAP and HAT performs at the same level. However, as the source get bigger and more complex HAT consistently outperforms DAP with an increasingly wider margin (Figure 4.6b). Unlike DAP that uses a single layer of global attributes, HAT is able to take advantage of the complexity of information available in the source. HAT captures the commonality among the categories and exploits it to learn and transfer more discriminative attributes to distinguish the unseen categories.

4.6.4 Image versus class level attributes

aPY and CUB provide attribute annotation at the image level which we used to train the models in the previous experiments. We evaluate on these two data sets using class-based attributes similar to those in AwA. We notice that the accuracy of both the baseline and HAT decreases in these settings. Where ENS has 38.3% and 34.6%, the HAT model achieves 40.3% and 48.2% on aPY and CUB. This seems to differ from the findings in Lampert et al. (2013). The reason could be related to the type of features used. In Lampert et al. (2013) a set of shallow features are used which require a relatively larger number of samples to train good attribute classifiers. This in turn results in noisy attribute predictions if there are few image-based annotations of the attribute. In comparison, using the deep features, which have been previously learned with a large-scale data set (like ImageNet), we can learn better attribute classifiers even if the training data in our data sets is relatively small.

4.6.5 Transferring attribute associations

Although this evaluation setup is not possible with the global attribute model, HAT enables us to carry out zero-shot recognition even if the attribute associations of the novel class are unknown. To do that, we again leverage the hierarchy and transfer the attribute associations of the parent node to the novel class. Using this setup, HAT achieves an accuracy of 31.1% (aPY), 59.7% (AwA) and 32.6% (CUB). This drop in performance is reasonable since we are transferring the more generic attribute associations of the parent. Hence, confusion can arise when multiple test classes share the same parent in the hierarchy. Nonetheless, HAT makes it possible to perform attribute-based zero-shot classification when only the label of the novel class is available.

4.7 Summary and discussion

In this chapter, we present a simple yet very effective model for zero-shot object recognition. Our model takes advantage of the embedded structure in the category space to learn attributes at different levels of abstraction. Furthermore, it exploits inter-class relations to provide a guided knowledge transfer approach that can select and transfer the expected relevant attributes to a novel class. The evaluation on three challenging data sets shows the superior performance of the proposed model over the state-of-the-art.

Discussion. In this work we transferred attribute models through the hierarchy with uniform weighting. That is, we assumed that all transferred models are of equal relevance to the novel class. However, considering a different weighting approach might be beneficial. For example, we can consider a decay weighting scheme relevant to the hierarchical distance between the novel class and the source model. Another option would be to incorporate the semantic similarity between the classes (*e.g.* by leveraging a semantic representation like Word2Vec (Mikolov et al., 2013)) in the weighting scheme. Furthermore, in our approach we only consider models of ancestor nodes to transfer, yet those of siblings or near by concepts in the hierarchy might also contribute positively to transfer a more discriminative model. However, this probably needs to be combined with a weighting scheme like the ones discussed earlier to guard against negative transfer.

Chapter 5

Predicting Class-Attribute Associations

Attributes, by creating an intermediate semantic representation layer, lend themselves quite naturally to be suitable knowledge units to share and transfer across categories. We saw in the previous chapter how semantic attributes can be effectively transfered to recognize object categories with zero training samples, *i.e.* zero-shot learning.

However, a major drawback of an attribute-based approach is that user supervision is needed to associate the novel class with the attribute vocabulary. For example, giving a new category "leopard", the user has to establish the semantic link by describing this novel category with the available attributes, *e.g.* the leopard has *paws*, it exhibits a *spotted* pattern but does not live in *ocean*. The semantic description defines which attributes to transfer (*i.e.* *paws*, *spotted* and *ocean*) and how to transfer them (*i.e.* *paws* and *spotted* have positive contribution to classify a leopard while *ocean* has a negative one). This amounts to providing manual class-attribute associations for each new category with a vocabulary size in the range of tens to hundreds of attributes. This is not only time consuming but often requires domain-specific or expert knowledge that the user is unlikely to have. It is more convenient and intuitive for the user to provide just the name of the unseen class rather than a lengthy description. In this chapter, we aim to automatically link a novel category with the visual vocabulary and predict its attribute association without user intervention. Thereby, we answer questions such as: Does the leopard live in the jungle? Does it have a striped pattern? (see Figure 5.1).

Contributions. To this end, we propose a novel approach that models relations coupling classes and attributes and automatically associates an unseen class with our visual vocabulary (*i.e.* the attributes) based solely on the class name. Using the predicted

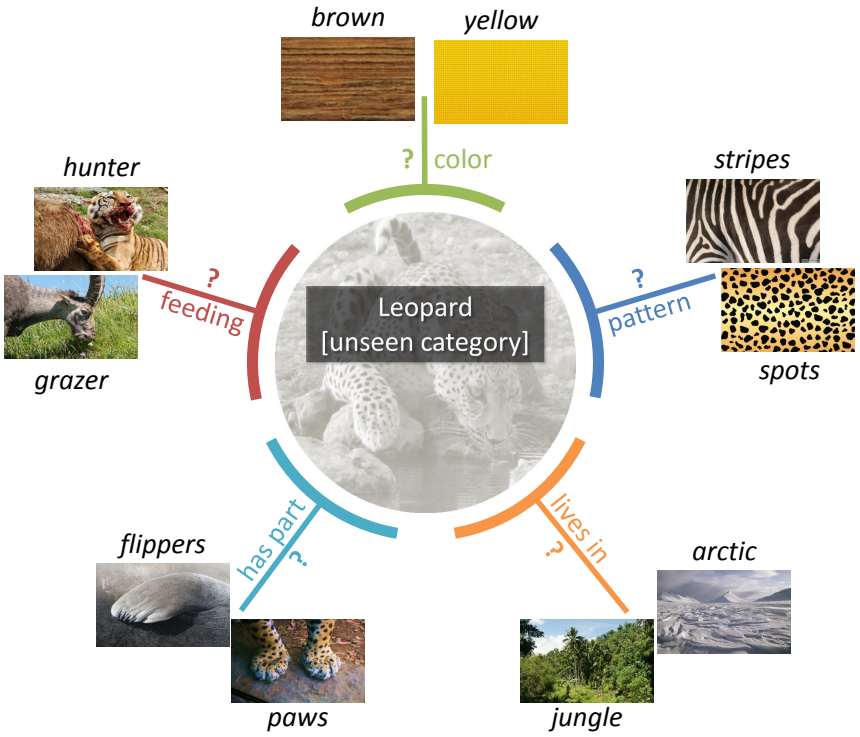


Figure 5.1: Given only the name of an unseen category, here *Leopard*, our method automatically predicts the list of attributes (*e.g.* yellow, spots) associated with the class through relationships (*e.g.* has_color, has_pattern). These predicted associations are leveraged to build category classifiers for zero-shot learning.

relations, we are able to construct a classifier for the novel class and conduct *unsupervised* zero-shot learning. Additionally, our model can automatically select which attributes to transfer that leads to enhanced performance compared to the commonly used generic transfer approach. Moreover, since the proposed approach can automatically establish the semantic link, we demonstrate that our model is able to automatically transfer the visual vocabulary itself across data sets which results in significant performance improvements at no additional cost.

Publication. This chapter is based on our work that is published in Al-Halah et al. (2016b).

5.1 Overview

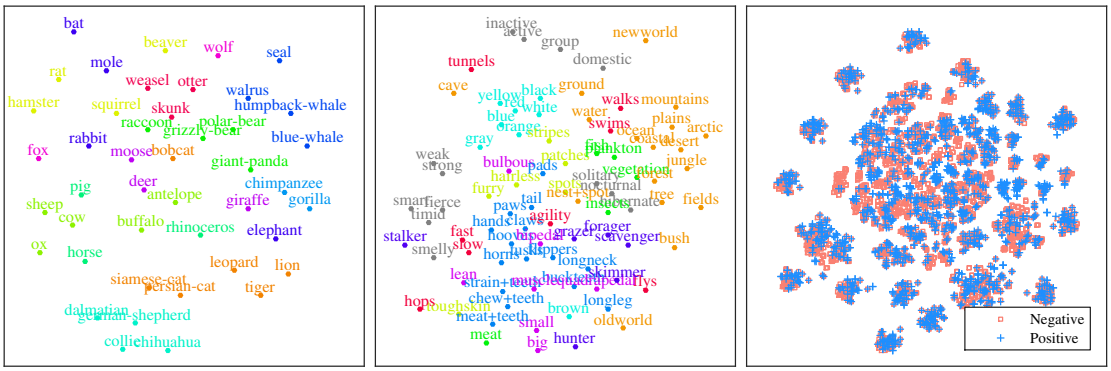
We present an end-to-end approach to automatically predict class-attribute associations and use them for zero-shot learning. We begin by 1) finding suitable vector represen-

tations for words and use the learned embedding as a way to mathematically relate class and attribute names (Section 5.2). These representations form the basis to model semantic relationships between classes and attributes. 2) We formulate the learning of these relations in a tensor factorization framework (see Figure 5.3) and offer key insights to adapt such a model to our problem. 3) In Section 5.4, we define the type of relations used and how to learn them automatically from data. 4) Finally, for an unseen class we show how to predict the set of its most confident attribute associations and carry out zero-shot learning (Section 5.5). We start by defining the notation used throughout this chapter.

Notation. Let $\mathcal{C} = \{c_k\}_{k=1}^{K}$ be a set of seen categories that are described with a group of attributes $\mathcal{A} = \{a_m\}_{m=1}^{M}$. The vector representation of a word is denoted by $\mathrm{v}(\cdot)$, and we use $\mathrm{v}(c_k)$ and $\mathrm{v}(a_m)$ for class c_k and attribute a_m respectively. The categories and attributes are related by a set of relations $\mathcal{R} = \{r_j\}_{j=1}^{N}$ such that $r_j(c_k, a_m) = 1$ if c_k is connected to a_m by relation r_j and 0 otherwise (*e.g.* $\mathrm{has\_color}(\mathrm{sky}, \mathrm{blue}) = 1$, $\mathrm{has\_pattern}(\mathrm{zebra}, \mathrm{spots}) = 0$).Given only the name of an unseen class $z \notin \mathcal{C}$, our goal is to predict the attributes that are associated with the class (*e.g.* $\mathrm{has\_color}(\textit{whale}, \mathrm{blue}) = ?$) and conduct ZSL accordingly.

5.2 Word vector representation

In order to model the relations between classes and attributes, we require a suitable representation that transforms names to vectors and at the same time preserves the semantic connotations of the words. Hereof, we use the skip-gram model presented by Mikolov et al. (2013) to learn vector space embeddings for words. The skip-gram model is a neural network that learns vector representations for words that best help in predicting the surrounding words. Therefore, words that appear in a similar context (neighboring words) are represented with vectors that are close to each other in the embedding space. We employ a context size of 20, use hierarchical softmax, and learn our skip-gram model using asynchronous stochastic gradient descent. We train our model on the Wikipedia corpus to learn a $d = 300$ dimensional word representation. The corpus includes ~ 3.5 billion words which yields a 96 thousand word vocabulary. The vocabulary includes high-frequency words and phrases (appearing more than 300 times) and the labels of classes and attributes used in our data sets.



(a) Class vectors (b) Attribute vectors (c) Class-Attribute displacement vectors

Figure 5.2: t-SNE representation of (a) class embeddings: colors indicate similar classes based on the super category in the WordNet hierarchy (*e.g.* dalmatian, collie, and other dog breeds are all colored in cyan); (b) attribute embeddings: colors indicate attributes which are grouped together to form class-attribute relations (*e.g.* has_color relationship clusters all colors yellow, black, *etc.* which are represented in cyan); and (c) class-attribute pair-wise displacement vectors (*e.g.* $\mathrm{v}(\text{dolphin}) - \mathrm{v}(\text{ocean})$) which show that encoding relationships using vector operations is a difficult task. This figure is best viewed in color.

Figure 5.2 visualizes the obtained word vector representation for few classes and attributes in AwA data set using t-SNE (van der Maaten and Hinton, 2008). Even in such a low-dimension it is clear that classes related to each other appear closer. This is evident for example from the group of dog breeds or feline in Figure 5.2a. Similarly, we also see clusters in the attribute label space corresponding to colors, animal parts, and environment (see Figure 5.2b).

Relations in the word embedding space. The skip-gram embeddings have gained popularity owing to their power in preserving useful linguistic patterns. An empirical evaluation by Mikolov et al. (2013) shows that syntactic/semantic relations can be represented by simple vector operations in the word embedding space. A great example is $\mathrm{v}(\text{king}) - \mathrm{v}(\text{man}) + \mathrm{v}(\text{woman}) \approx \mathrm{v}(\text{queen})$, where $\mathrm{v}(\text{king})$ is the embedding for "king". In other words, the relation between "king" and "man" modeled by their displacement vector is similar to the displacement between "queen" and "woman".

However, can we model the class-attribute relations by simple vector operations? In other words, can we say that $\mathrm{v}(\text{leopard}) - \mathrm{v}(\text{jungle}) \approx \mathrm{v}(\text{whale}) - \mathrm{v}(\text{ocean})$, for example? Figure 5.2c presents the t-SNE representation for the *displacement vectors* between each class-attribute pair (*e.g.* $\mathrm{v}(\text{sky}) - \mathrm{v}(\text{blue})$). We see that displacement vectors for both

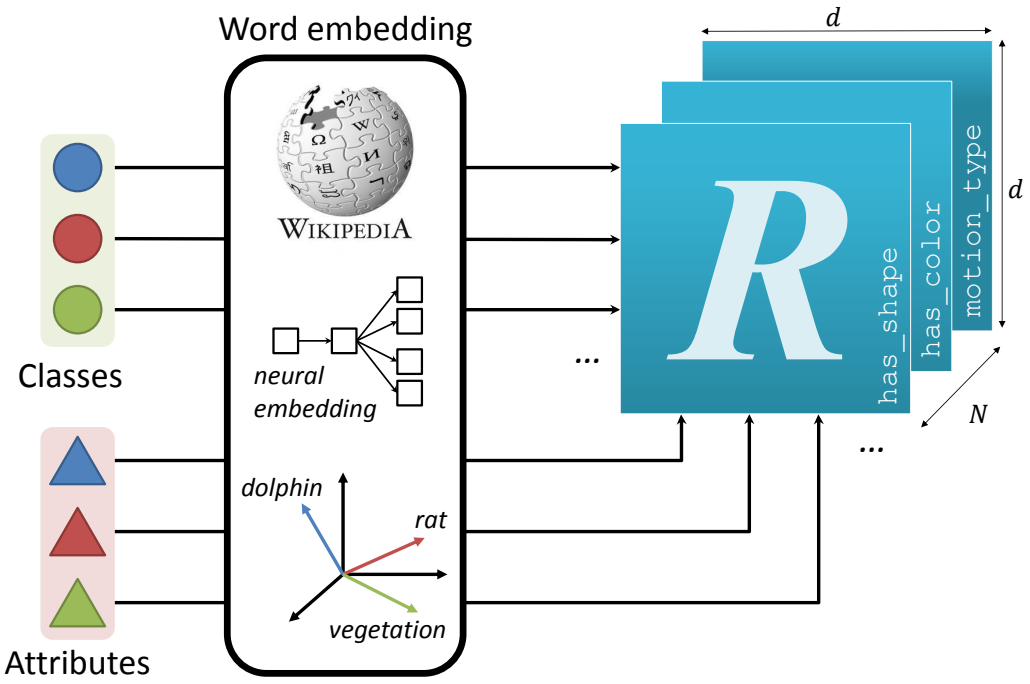


Figure 5.3: Our model couples class and attribute embeddings using the tensor $\mathbf{R}$. Each slice $\mathbf{R}_j$ captures a relationship like has_shape or motion_type. The embeddings are obtained from a neural network trained on a large text corpus.

positive existing relations *and* negative non-existing relations are inseparable. They group together and often even coincide. We empirically show in Section 5.8.1 that class-attribute relations are more complicated and are not easily represented by simple vector operations.

To address this challenge we adopt a more sophisticated and comprehensive method to learn these relations while at the same time effectively leverage the powerful word embedding representation.

5.3 Semantic relations model

We now model the complex relations between categories and their corresponding visual attributes. Leveraging information based on these relations, we can predict the associations between a novel unseen class and our attribute vocabulary and build the corresponding ZSL classifier.

We propose to model the class-attribute relations using a tensor factorization approach (*e.g.* Nickel et al., 2012; Sutskever et al., 2009). We represent the relations using a three

dimensional tensor $\mathbf{R} \in \mathbb{R}^{d \times d \times N}$ where d is the dimension of the word embedding and N is the number of relations (see Figure 5.3). Each slice $\mathbf{R}_j \in \mathbb{R}^{d \times d}$ in the tensor models a relation r_j (*e.g.* has_color) as a bilinear operator. The likelihood of class c_k being associated with attribute a_m through relation r_j is:

$$p(r_j(c_k, a_m)) = \sigma(\mathrm{v}(c_k)^T \mathbf{R}_j \mathrm{v}(a_m)), \tag{5.1}$$

where $\mathrm{v}(x) \in \mathbb{R}^d$ is the vector embedding of word x and $\sigma(\cdot)$ is the logistic function. We learn $\mathbf{R}$ by minimizing the negative log-likelihood of both positive ($\mathcal{P}$) and negative ($\mathcal{N}$) class-attribute associations for each slice $\mathbf{R_j}$:

$$\min_{\mathrm{v}(\mathcal{A}), \mathbf{R_j}} - \sum_{(j,k,m) \in \mathcal{P}} \log(p(t^j_{k,m} = 1)) - \sum_{(j,k,m) \in \mathcal{N}} \log(p(t^j_{k,m} = 0)), \tag{5.2}$$

where $t^j_{k,m} = r_j(c_k, a_m)$.

Note that there are two key components in Eq. 5.2. Firstly, we take advantage of the powerful representation of skip-gram and learn word embeddings on a large text corpus to initialize the representation of our class ($\mathrm{v}(\mathcal{C})$) and attribute ($\mathrm{v}(\mathcal{A})$) entities. This gives our model the ability to generalize well to unseen classes and take advantage of the initial learned similarities among the attributes. Secondly, in our case of zero-shot classification, the novel class name is not available during training and we have no information about how this unseen class is related with the visual attributes. Consequently, we treat the set of categories as an *open* set and fix their embedding $\mathrm{v}(\mathcal{C})$ to the one learned in Section 5.2. On the other hand, visual attributes $\mathcal{A}$ are usually restricted to entities which we have seen before, and for which we have training images and learned models. Hence, we treat $\mathcal{A}$ as a *closed* set which allows us optimize the attribute representation during training by back propagating the gradients to $\mathrm{v}(\mathcal{A})$. This yields improved performance as we will see in the model analysis in Section 5.8).

Limited training data. Learning $\mathbf{R}$ directly from training data is not favorable since the number of class-attribute associations available for training are usually small. For example, a typical data set consisting of 40 categories and 80 attributes yields around 1500 positive associations compared to tens or even hundreds of thousands of parameters in $\mathbf{R}$. Hence, in order to avoid overfitting we build on the ideas of Jenatton et al. (2012) and reduce the number of parameters that are required to be learned, by representing the relation operator $\mathbf{R}_j$ as a combination of L latent factors:

$$\mathbf{R}_j = \sum_{l=1}^{L} \alpha_l^j \Theta_l, \quad \alpha^j \in \mathbb{R}^L \text{ and } \Theta_l \in \mathbb{R}^{d \times d}, \tag{5.3}$$

where α^j is a sparse vector used to weight the contributions of the latent factors Θ. Addiotionally, Θ_l is a rank one factor such that $\Theta_l = \mathbf{u}_l \mathbf{z}_l^T$ and $\mathbf{u}_l, \mathbf{z}_l \in \mathbb{R}^d$. Both α and Θ are learned while minimizing Eq. 5.2 and constraining $\|\alpha^j\|_1 \leq \lambda$. The parameter λ controls the sparsity of α, and hence the extent to which latent factors are shared across relations. Modeling $\mathbf{R}$ with latent factors has the benefit of allowing the learned relations to interact and exchange information through Θ and hence improves the ability of the model to generalize.

5.4 Type of relations

In order to train our model, we need to define the relations that link the classes with the respective attributes. Usually these relations are harvested through the process of collecting and annotating attributes (*e.g.* What color is a bear? What shape is a bus?). We refer to this type of relations as *semantic relations*. However, while some data sets do provide such relation annotations (Farhadi et al., 2009; Wah et al., 2011) others do not (Lampert et al., 2009). An alternative approach that alleviates the need to manual relation annotation is to automatically discover relations by utilizing the word embedding space. As described earlier in Section 5.2, embeddings of semantically related entities tend to be close to each other (see Figure 5.2b). Hence, one can simply group attributes into several relations by clustering their embeddings (*i.e.* N = number of clusters). We refer to this type of relations as *data-driven relations*.

5.5 Inferring binary associations

Given an unseen class z, we infer its associations with the attribute set $\mathcal{A}$:

$$r_j(z, a_m) = \begin{cases} 1 & \text{if } p(r_j(z, a_m)) > t_+ \\ 0 & \text{if } p(r_j(z, a_m)) < t_- \\ \varnothing & \text{otherwise} \end{cases} \quad \forall m, \tag{5.4}$$

where thresholds t_+ and t_- are learned to help select the most confident positive and negative associations while at the same time provide enough discriminative attributes to

predict a novel class. Assignment to $\varnothing$ discards the attribute for ZSL since we are not confident about the type (positive or negative) of the association. In other words, t_+ and t_- allow our model to choose *when to transfer* attributes to the unseen class based on our confidence of the mapping $p(r_j(z, a_m))$. We learn these thresholds using leave-K-class-out cross-validation so as to maximize zero-shot classification accuracy of the held out classes.

Zero-shot learning. The score for unseen class z on image x is estimated based on the predicted attribute associations ($\mathcal{A}^z = \{a_m^z\} \subseteq \mathcal{A}$) using the Direct Attribute Prediction (DAP) (Lampert et al., 2009) method:

$$s(z|x) = \prod_{a_m \in \mathcal{A}^z} p(a_m = a_m^z|x)/p(a_m), \tag{5.5}$$

where $p(a_m|x)$ is the posterior probability of observing attribute a_m in image x. We assume identical class and attribute priors.

5.6 Baselines

We compare our method of predicting class-attribute associations via word vector representations and learned semantic relationships against the state-of-the-art (SOTA) co-occurrence approach and two other baselines based on word embedding space.

Co-occurrence. As in the state-of-the-art methods for associations prediction (Mensink et al., 2014; Rohrbach et al., 2010), we use the Microsoft Bing Search API (Bing, 2016), the Flickr API (Flickr, 2016) and Yahoo Image to obtain hit counts H_{c_k} for classes (*e.g.* "chimpanzee"); H_{a_m} for attributes (*e.g.* "stripes"), and H_{c_k,a_m} jointly for class-attribute pairs (*e.g.* "chimpanzee stripes"). For Flickr, we consider the number of photos using class or attribute names as tags or free text (includes titles, descriptions and tags) and observe that tags typically outperform a free text search. Yahoo Image does not provide a public API, hence we use the hit counts provided by Rohrbach et al. (2010) for AwA. To obtain a hit-count based class-attribute association score, we use the Dice score metric (Mensink et al., 2014) :

$$s^H_{c_k,a_m} = \frac{H_{c_k,a_m}}{H_{c_k} + H_{a_m}}, \tag{5.6}$$

where s^H is the co-occurrence similarity matrix of classes and attributes.

Word embedding space. These methods directly use the word vector representations (Section 5.2) to predict class-attribute associations. We present two approaches using the word embeddings:

(1) C $\rightarrow$ A (Top Q): Let Q be the average number of attributes that are associated with every class in the training set. For each unseen class, we consider a positive association with the Q nearest attributes (in terms of Euclidean distance) using the vector space embedding:

$$a_m^{c_k} = \begin{cases} 1 & \text{if } \mathrm{v}(a_m) \in \mathrm{nearest}_Q(\mathrm{v}(c_k)) \\ 0 & \text{otherwise} \end{cases} \qquad \forall m, \tag{5.7}$$

(2) C $\rightarrow$ A (Similarity): Similar to the co-occurrence method, we construct a similarity matrix between class and attribute labels based on the distances of the vector representation of the two entities:

$$s^W_{c_k,a_m} = \exp(-\|\mathrm{v}(c_k) - \mathrm{v}(a_m)\|_2) \quad \forall c_k, a_m. \tag{5.8}$$

For s^H and s^W, binary associations are obtained by choosing the best threshold over the class-attribute similarity matrix which maximizes the ZSL performance.

In addition to the previous attribute-based baselines, we also examine two category-based baselines for unsupervised ZSL:

(3) C $\rightarrow$ C (Top 1): As we see from Figure 5.2a, similar classes do appear together in the word embedding space. Hence, for each unseen class z we transfer the category classifier of the training set class $c_k \in \mathcal{C}$ which appears closest to it in the vector space:

$$s_T(z|x) = s(c_k|x) \quad \text{such that} \quad c_k = \arg\min_{c_j \in \mathcal{C}}(\|\mathrm{v}(z) - \mathrm{v}(c_j)\|) \tag{5.9}$$

(4) C $\rightarrow$ C (Weighted K): This takes into consideration the similarity of the novel class for all known classes (Bart and Ullman, 2005). We build a classifier as a weighted linear combination of all training classes where the weights are based on distances between their vector representations:

$$s_{wc}(z|x) = \sum_{k=1}^{K} \exp(-\|\mathrm{v}(z) - \mathrm{v}(c_k)\|_2)\ s(c_k|x), \tag{5.10}$$

where $s(c_k|x)$ is the score obtained by the classifier for category c_k on image x.

5.7 Evaluation setup

Data sets. We use two publicly available data sets (see details in Table 4.1):

1. Animals with Attributes (AwA) (Lampert et al., 2009).
2. aPascal/aYahoo (aPY) (Farhadi et al., 2009).

Word embedding. The word embeddings with $d = 300$ dimensional representation are learned using a skip-gram model (Mikolov et al., 2013) trained on the Wikipedia corpus as described earlier in Section 5.2. All the class names and most of the attributes used in our study occur naturally in text and their embeddings are obtained directly from the learned dictionary. However, for few attributes like "two_dimensional_boxy" there is no direct match. In such cases, we automatically split the label to the minimum number of constituted phrases that do appear in the dictionary and take the average embedding of these phrases as the attribute representation *i.e.* $\mathrm{v}(two\_dimensional\_boxy) = \frac{1}{2}(\,\mathrm{v}(two\_dimensional) + \mathrm{v}(boxy)\,)$

Relations. We consider two types of relations for training our model (Section 5.4):

1. Semantic relations (SR): For aPY, we use the 3 predefined relations (*has_material*, *has_shape* and *has_part*). As for AwA, a cursory look at the set of attributes shows us that they can be easily grouped into 9 sets of relationships like *has_color*, *lives_in*, *food_type*, *etc*. Table 5.1 shows the set of semantic relations defined over AwA and their respective attributes.
2. Data-driven relations (DR): For both data sets, we perform hierarchical agglomerative clustering on the word embeddings of the attributes. By analyzing the respective dendrogram, the clustering is stopped at 10 groups of attributes. Table 5.2 shows the learned relations in AwA. We see that many of the data-driven relations can be easily mapped again to their semantic counterparts (*e.g.* $R_{10} = \mathrm{has\_color}$, $R_7 \approx \mathrm{has\_part}$ and $R_9 \approx \mathrm{lives\_in}$).

5.8 Experiments

We evaluate our model at: (1) predicting class-attribute associations (Section 5.8.1) and (2) unsupervised zero-shot learning (Section 5.8.2). This is followed with (3) an in-depth analysis of our model in Section 5.8.3. Furthermore, we demonstrate the ability of our

Semantic Relation	Attribute
has_color	black, white, blue, brown, gray, orange, red, yellow
has_pattern	patches, spots, stripes, furry, hairless, toughskin
has_shape	big, small, bulbous, lean, muscle, bipedal, quadrapedal
has_part	flippers, hands, hooves, pads, paws, longleg, longneck, tail, chewteeth, meatteeth, buckteeth, strainteeth, horns, claws, tusks
movement_type	flys, hops, swims, tunnels, walks, fast, slow, agility
food_type	fish, meat, plankton, vegetation, insects
feeding_style	forager, grazer, hunter, scavenger, skimmer, stalker
lives_in	newworld, oldworld, arctic, coastal, nestspot, desert, bush, plains, forest, fields, jungle, mountains, ocean, ground, water, tree, cave
behavior	fierce, timid, smart, group, solitary, domestic, strong, weak, active, inactive, nocturnal, hibernate, smelly

Table 5.1: The set of defined semantic relations in AwA and their respective attributes.

Data-driven Relation	Attribute
R_1	active, inactive
R_2	fields, ground, group, domestic
R_3	furry, big, smelly, hops, walks, hunter, stalker, bush, fierce, timid, smart
R_4	lean, fast, slow, strong, weak, muscle, agility
R_5	patches, spots, stripes, hairless, small, bulbous, nestspot
R_6	swims, nocturnal, hibernate, fish, plankton, insects, forager, grazer, scavenger, solitary
R_7	flippers, hands, hooves, pads, paws, tail, chewteeth, meatteeth, strainteeth, horns, claws, tusks, bipedal, meat
R_8	toughskin, longleg, longneck, buckteeth, flys, quadrapedal, skimmer, newworld, oldworld
R_9	tunnels, vegetation, arctic, coastal, desert, plains, forest, jungle, mountains, ocean, water, tree, cave
R_{10}	black, white, blue, brown, gray, orange, red, yellow

Table 5.2: The set of learned data-driven relations in AwA and their respective attributes.

model to (4) transfer attributes across data sets without the cost of additional annotations (Section 5.8.4). Finally, (5) we show that the model is generic and can learn different types of relations and not only attribute-based ones (Section 5.8.6). In the following, we refer to our Class-Attribute Association Prediction model as CAAP.

Model	AwA	aPY
Co-Occurrence (Mensink et al., 2014; Rohrbach et al., 2010)		
Bing	41.8 (57.4)	20.9 (69.4)
Yahoo-Img	50.9 (62.5)	-
Flickr	48.7 (63.4)	28.1 (82.3)
Word Embedding		
C → A (Top Q)	41.3 (53.7)	34.2 (74.0)
C → A (Similarity)	41.3 (43.1)	34.2 (77.5)
Ours		
CAAP (SR)	79.1 (78.2)	**76.1 (89.8)**
CAAP (DR)	**79.7 (78.9)**	75.7 (89.6)

Table 5.3: Performance of class-attribute association predictions for unseen classes, presented in mAP and accuracy in parentheses.

5.8.1 Predicting associations

For training of our model CAAP, we generate both positive and negative training triplets using the attribute annotations of the training set (*e.g.* $\text{has\_part}(\text{horse}, \text{tail}) = 1$, $\text{lives\_in}(\text{dolphin}, \text{desert}) = 0$). We estimate the number of latent factors L and λ using 5-folds cross validation. We report the performance to predict all attribute associations for the unseen classes, hence we set $t_- = t_+ = 0.5$.

Table 5.3 presents the mean average precision (mAP) and accuracy for predicting class-attribute associations. Among the co-occurrence methods, Flickr and Yahoo Image search perform better than Bing web search. This can be related to the fact that in the first two methods the search results are grounded from visual information. As demonstrated earlier, the word embedding space is not suitable to directly model the relations and it fails to reliably predict class-attribute associations (see Figure 5.2c). Our method of modeling relations outperforms state-of-the-art by a significant margin (an absolute increase of 19% on AwA and 42% on aPY).

Table 5.4 presents examples of the top 5 confident positive and negative associations. In general, we observe that our model ranks the most distinctive attributes of a category higher (*e.g.* leopard↔fast, chimpanzee↔walk, hippopotamus↔strong). A more detailed insight into the performance of our model for each semantic relation represented by the precision-recall curve is presented in Figure 5.4, and the association accuracy per attribute in Figure 5.5.

Unseen Category	Top Associations	
	Positive	Negative
persian_cat	tail, fast, paws, *active*, furry	orange, yellow, horns, tusks, desert
hippopotamus	strong, *group*, big, walks, ground	claws, flys, red, nocturnal, weak
leopard	fast, lean, oldworld, active, tail	tusks, water, arctic, plankton, weak
humpback_whale	fast, ocean, water, group, fish	red, weak, tunnels, nocturnal, plains
seal	fast, *meatteeth*, *bulbous*, big, toughskin	grazer, tunnels, longleg, hooves, longneck
chimpanzee	walks, group, fast, chewteeth, active	arctic, flippers, red, plankton, strainteeth
rat	furry, active, *chewteeth*, newworld, fast	plankton, yellow, orange, horns, desert
giant_panda	*fast*, walks, *active*, quadrapedal, strong	*domestic*, weak, strainteeth, desert, flys
pig	ground, timid, white, chewteeth, quadrapedal	cave, plankton, orange, desert, yellow
raccoon	fast, newworld, quadrapedal, furry, active	bush, hands, longneck, tusks, desert

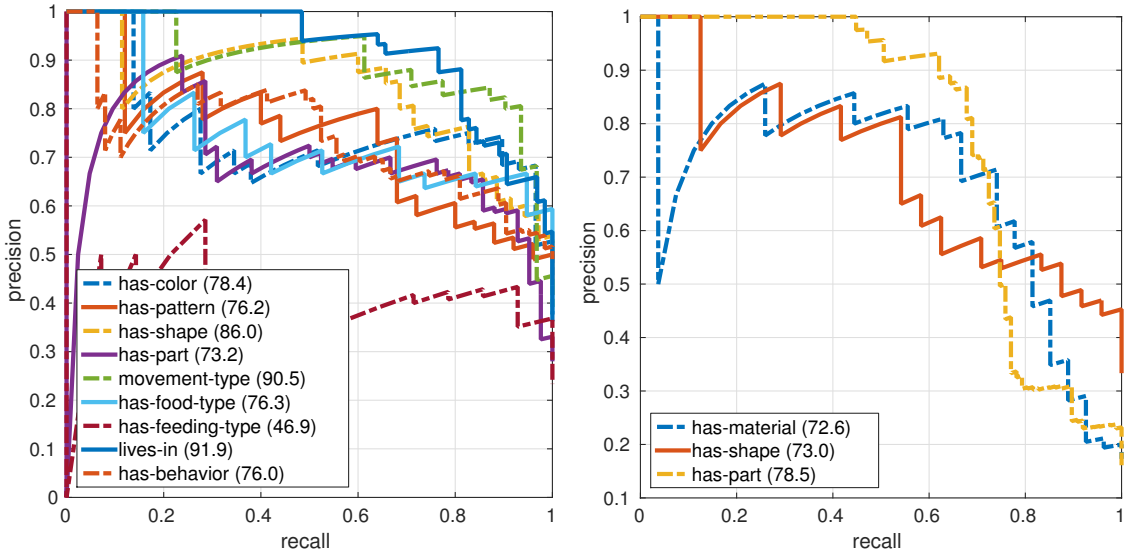
Table 5.4: Examples of predicted class-attribute associations for unseen classes in AwA. Wrong associations are highlighted in gray and italic.

Moreover, both SR and DR models perform at the same level with no substantial difference. Hence, the data-driven approach is a very good alternative for the semantic relations thus even removing the need to provide extra relation annotations for CAAP. In the rest of the experiments, we adopt the DR approach.

5.8.2 Unsupervised zero-shot learning

We now present unsupervised zero-shot classification performance comparing against methods of the previous section which also use predicted class-attribute associations. For all attribute-based approaches, we use the DAP model as described in Section 5.5. We also consider the two additional unsupervised ZSL baselines that leverage class embeddings (C $\rightarrow$ C and C $\rightarrow$ C (Weighted K) from Section 5.6).

Furthermore, most previous works assume that the attribute labels are provided by a human operator for the unseen class. While in this work we circumvent this additional



(a) AwA (b) aPY

Figure 5.4: Prediction performance of individual relations learned by CAAP given by precision-recall curves along with mAP scores (see legend).

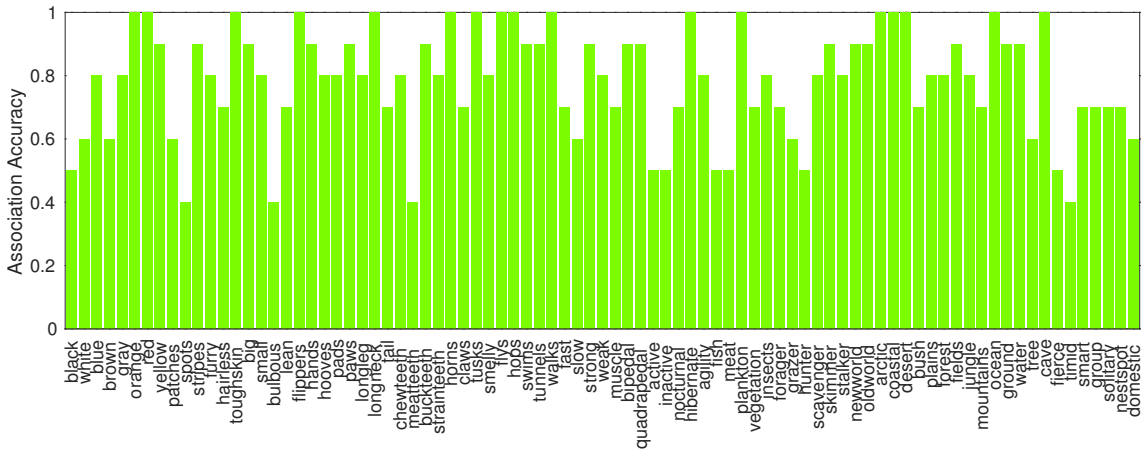
overhead, we present the results of supervised DAP (*i.e.* using ground truth associations) as a reference.

Image features and classifiers. We use the output of the last hidden layer of the public BVLC implementation (Jia et al., 2014) of GoogLeNet (Szegedy et al., 2014) as our 1024 dimensional image features. The deep representation is then used to train linear SVMs (Fan et al., 2008) for the attribute and category classifiers. The SVM parameter C is estimated using 5-folds cross validation. Moreover, the same image features and classifiers are used for the all baselines and our model.

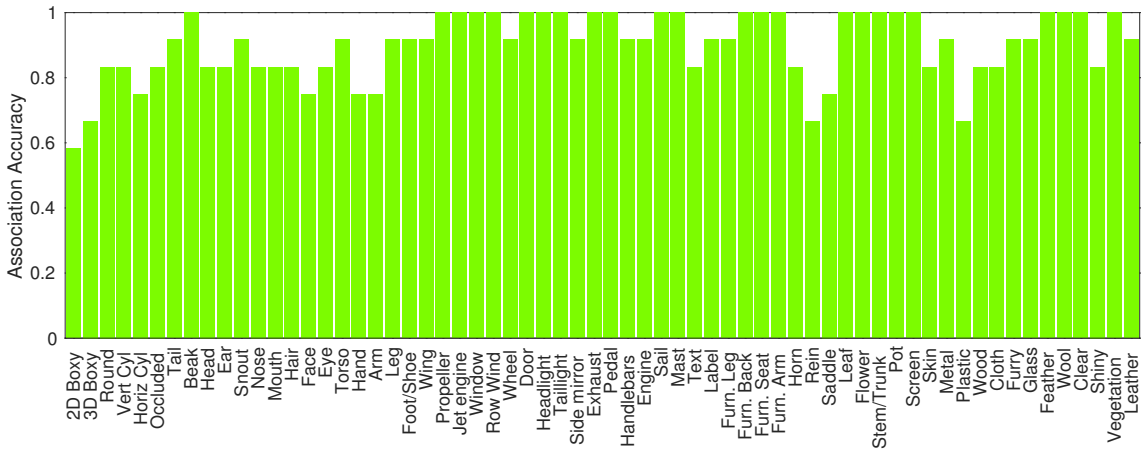
For our model, we estimate the number of latent factors L and λ and additionally learn the thresholds (t_-, t_+) by 5-folds cross validation.

Results. Table 5.5 presents the mean per class accuracy for the test classes used in zero-shot learning. We see again that image-based hit-count information obtained by Yahoo Images or Flickr outperforms general web-based search (Bing). However, they are all far from supervised ZSL performance with ground truth association.

The word embedding methods based on attributes (C $\rightarrow$ A) show poor performance. In comparison, using the classifier of the nearest class (C $\rightarrow$ C (Top 1)) performs well for AwA (48%) and poorly on aPY (15%). An explanation for this is that the unseen classes



(a) AwA



(b) aPY

Figure 5.5: Accuracy of predicting individual attribute associations over the unseen classes.

in AwA are visually close to the train set, while aPY has higher diversity in class types (animals and man-made objects). Building a classifier by weighting all other classes (C $\rightarrow$ C (Weighted K)) shows moderate performance on both data sets.

Our method outperforms all baselines with an accuracy of 67.5% for AwA and 37.0% for aPY . In fact, CAAP performs at the level of supervised DAP in aPY, while for AwA we see impressive performance surpassing the performance of supervised ZSL with ground truth attribute associations.

Figure 5.6 shows the top 5 most confident zero-shot classifications of our model for each of the unseen classes in AwA. Correct predictions are marked with green while the wrong ones in red. Most of the confident classifications are correct.

Model	AwA	aPY
Supervised ZSL		
DAP (Lampert et al., 2013)	59.5	37.1
Unsupervised ZSL		
Co-Occurrence (Mensink et al., 2014; Rohrbach et al., 2010)		
Bing	11.8	13.1
Yahoo-Img	39.8	-
Flickr	44.2	13.8
Word Embedding		
C $\rightarrow$ A (Top Q)	10.2	14.3
C $\rightarrow$ A (Similarity)	26.4	20.4
C $\rightarrow$ C (Top 1)	48.6	15.0
C $\rightarrow$ C (Weighted K)	40.6	22.5
CAAP (ours)	**67.5**	**37.0**

Table 5.5: Zero-shot classification performance presented in mean per-class accuracy.

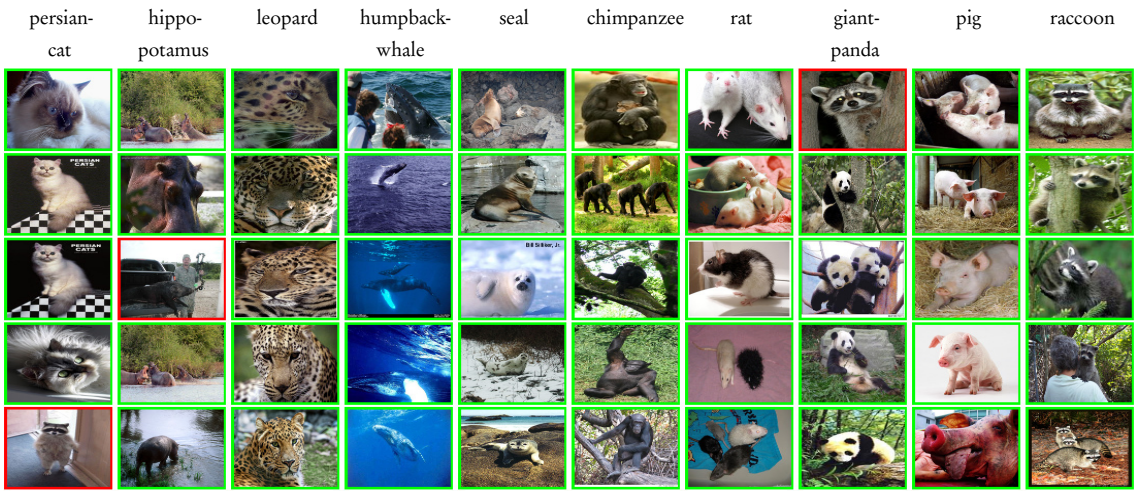


Figure 5.6: The top 5 ranked results of our CAAP model for each unseen class in AwA.

5.8.3 Model analysis

In this section, we study the effect of the different aspects of our model on the final unsupervised ZSL performance (see Table 5.6).

(1) Single relation: In the previous experiments, we used a small set of relations that group similar attributes together. Here, we group all attributes in a single abstract relation ($N = 1$) called *has_attribute* and try to model the class-attribute associations accordingly.

Model	AwA	aPY
Single relation ($N = 1$)	62.6	25.8
Fixed v($\mathcal{A}$)	65.1	28.2
$t_- = t_+ = 0.5$	65.4	25.0
CAAP (full)	67.5	37.0

Table 5.6: The impact of the different hyperparameters of our model on ZSL accuracy.

We observe that in this setting, the absolute drop in accuracy is 5% on AwA while on aPY we see a reduction by 12%.

(2) Fixed attribute embedding: Similar to the category embeddings, we fix the representation of the attributes during learning. Here, the performance on both data sets drops by 2% on AwA and 9% on aPY.

(3) Threshold@0.5: Rather than learning the thresholds (t_-, t_+) we set them both to $t_- = t_+ = 0.5$, *i.e.* all attributes are transferred to the unseen category. In this case, the accuracy drops by 2% on AwA while the performance on aPaY goes down by 12%.

We conclude that updating the attribute representation during learning is beneficial. We notice that attribute pairs like (*big*, *small*) and (*weak*, *strong*) which get initialized with similar embeddings by the skip-gram model are pushed apart by our model to facilitate the learning of the relations. It is also good to learn multiple relations that account for the discrepancies in the attributes rather than an abstract mapping that groups all of them together in one inhomogeneous cluster. Our model learns proper confidence scores on the associations, and ranks most distinctive attributes higher. This leads to better ZSL performance when considering the most confident associations. In general, we notice that results on aPY are more sensitive to changes. This can be related to the large variance in both classes and attributes, since they describe not only animals (like in AwA) but also vehicles and other man-made objects.

Finally, in Figure 5.7a, we present the performance of CAAP during cross validation with respect to the number of latent factors L while $\lambda = 1$. We notice that our model has relatively stable performance with regard of L, especially when the number of latent factor is $L \geq 50$. We observe a similar stable behavior with regard to λ. In Figure 5.7b, we test our model with varying dimensionality d of word vector representation. We see here that d has a high impact on the performance of CAAP when it is very small

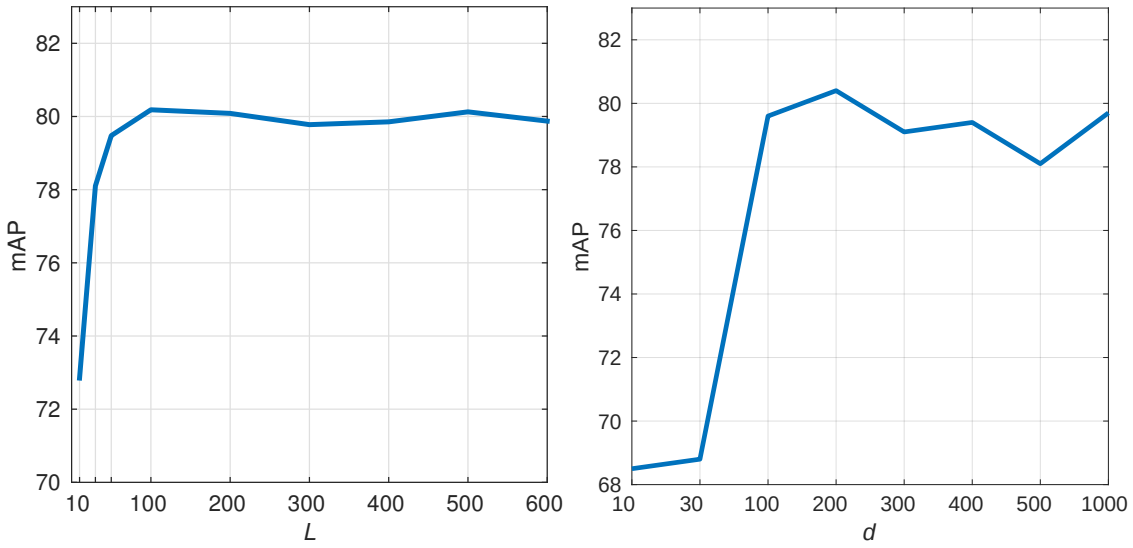


Figure 5.7: The mAP of association prediction of our model in AwA when (a) varying number of latent factors L and (b) varying the dimensionality of the word vector representation d.

($d < 100$). However, even when d is as low as 10 dimensions, our model predicts associations with a mAP of 68.5% outperforming state-of-the-art in Table 5.3.

5.8.4 Attribute transfer across data sets

A major advantage of our approach is the ability to automatically transfer the set of attributes from one data set to another at no additional annotation cost. For example, we can use the 85 attributes of the AwA data set to describe categories from aPY and vice-versa. Most importantly, we do *not* need to manually associate the classes of one data set with the attributes of the other. These associations are automatically obtained through our CAAP model.

In particular, we learn the relations of AwA and aPY jointly without providing any additional associations. Then, for a novel class (from AwA or aPY), we predict its associations to the attribute set $\mathcal{A} = \mathcal{A}^{AwA} \cup \mathcal{A}^{aPY}$. We see in Table 5.7 ($3^{rd}$ row), that CAAP results in a significant improvement surpassing the performance of the manually defined associations on each data set. Especially on aPY, we see a dramatic improvement of 12% in performance. This can be attributed to the fact that roughly half the classes of aPY test set are animals, which benefit strongly from the rich attributes transferred

	Target (*unseen*)		
Source (*seen*)	AwA	aPY	AwA+aPY
AwA	67.5	39.5	37.1
aPaY	10.4	37.0	6.2
AwA+aPaY	**68.6**	**49.0**	**46.8**

Table 5.7: Zero-shot classification accuracy when attributes are transferred across data sets using CAAP. The source set contains the seen classes of the respective data set while the target is the set of unseen ones. A source AwA and Target aPY means classifying the unseen classes of aPY based on their predicted associations with the attributes of AwA.

from AwA. This demonstrates the effectiveness of CAAP in integrating knowledge from different sources without the need for any additional effort.

In Table 5.7, we provide additional evaluation of the attribute transfer by changing the source and target sets. Comparing the two sources AwA and aPY, it is clear that AwA encompasses a richer diversity as it results in good performance for both test sets, while transferring attributes from aPY→AwA results in performance on par with a random classifier. Taking a closer look at the assigned attributes, we notice the following:

(1) AwA→aPY, not only the animal classes but even some man made classes get associated with reasonable attributes. For example, the class “jetski” is positively associated with attributes “water" and “fast"; and class “carriage" with “grazer" and “muscle".

(2) aPY→AwA, the attributes assigned to the animal classes are, in general, correct. However, aPY does not have enough animal-related attributes to distinguish the fine grained categories on AwA. Most of the test classes in AwA are assigned to attributes like “eye", “head" and “leg".

(3) AwA+aPY→AwA+aPY, even on this harder setting where we test on 22 unseen classes (*i.e.* random performance drops to 4.5% as compared to 10% on AwA and 8.3% on aPY), our model generalizes gracefully with 46.8% accuracy.

5.8.5 CAAP versus state-of-the-art

In Table 5.8, the performance of our approach is compared against state-of-the-art in unsupervised ZSL. Both Frome et al. (2013) and Norouzi et al. (2014) use the same word embedding as ours, while Akata et al. (2015) use both GloVe (Pennington et al., 2014) and Word2Vec (Mikolov et al., 2013). Additionally, all methods in Table 5.8

Model	Side Information	AwA	aPaY
DeViSE (Frome et al., 2013)	C	44.5	25.5
Text2Visual (Elhoseiny et al., 2013)	$\text{Text}^{\text{Wiki}}$	55.3	30.2
ConSE (Norouzi et al., 2014)	C	46.1	22.0
SJE (Akata et al., 2015)	C	58.8	-
SJE (Akata et al., 2015)	C + $\text{H}^{\text{WordNet}}$	60.1	-
HAT (Al-Halah and Stiefelhagen, 2015b)	$\text{H}^{\text{WordNet}}$	59.7	31.1
CAAP (ours)	C	**68.6**	**49.0**

Table 5.8: Unsupervised zero-shot learning accuracy of state-of-the-art versus CAAP. The second column shows the type of information leveraged by each model for the unseen classes (C: class label, H: hierarchical representation, and Text: online articles).

use image embedding from GoogLeNet. Still, CAAP outperforms approaches based only on class names (Frome et al., 2013; Norouzi et al., 2014) with more than 20% on both data sets. Approaches like Text2Visual (Elhoseiny et al., 2013), SJE (Akata et al., 2015) make use of additional sources of information like Wikipedia articles or WordNet. Our HAT model from Chapter 4 also leverages the hierarchical structure in the source to propagate attribute associations to unseen classes in an unsupervised fashion (Section 4.6.5). Nonetheless, CAAP outperforms state-of-the-art by 8.5% and 18.8% on AwA and aPY respectively, while only leveraging the name of the unseen class.

5.8.6 Beyond attributes

Although the focus of this work is on predicting class-attribute associations, our model is generic enough and it can learn other types of relations. For example, various approaches in the literature have reported the advantage of incorporating hierarchical information for ZSL (*e.g.* Akata et al., 2015; Rohrbach et al., 2011) and our own work in Al-Halah and Stiefelhagen (2015b).

Our model can also learn hierarchical relations, for example to predict the ancestors of a category. To test this, we query WordNet (Miller, 1995) with each of the data set categories and extract the respective graph relevant to the hypernym links. We then learn the *has_ancestor* relation by generating triplets of the form $\text{has\_ancestor}(\text{horse}, \text{equine}) = 1$ using the information from the extracted graph.

The evaluation reveals that we can predict the ancestor relation of an unseen class with a mAP of 89.8% on AwA and 93.0% on aPY. Interestingly, learning such a hierarchy-

based relation can aid the learning of some attribute-based relations. The model allows the various relations to interact and exchange information at the level of the shared latent factors. For example, in AwA among the improved attribute-based relations, is *has_pattern* (+2.5%), and *feeding_type* (+2.1%). These relations correlate well with the hierarchical information of the classes (*e.g.* carnivores tend to have similar pattern and feeding type). Predicting such a hierarchical relation alleviates the need of a complete hierarchy or manual synonym matching since this can be automatically handled by the word embedding and CAAP model. This keeps user intervention to the minimal requirement of providing class names. We expect that modeling more relations among the classes jointly with class-attribute relations can result in better performance.

5.9 Summary and discussion

Attribute-based ZSL suffers from a major drawback of needing class-attribute associations to be defined manually. To counter this, we present an automatic approach to predict the associations between attributes and unseen classes. We model the associations using a set of relationships linking categories and their respective attributes in an embedding space. Our approach effectively predicts the associations of novel categories and outperforms the state-of-the-art in two tasks; namely association prediction and unsupervised ZSL. Moreover, we demonstrate the ability of our model to transfer attributes between data sets at no cost. The transferred attributes enlarge the size of the description vocabulary, which results in more discriminative classifiers for ZSL yielding an additional boost in performance.

Discussion. In this work, while we leveraged a powerful word embedding that is optimized over a large-scale textual data, we learned our relation model over the attribute data sets from scratch. However, one can also follow a parameter-transfer approach here. The relation model can be trained to predict many semantic relations in large-scale knowledge graphs and then transferred and accommodated to learn and predict the relations in the attribute data sets. Additionally, we proposed a method to select which attribute to transfer to the unseen class based on our confidence in the predicted semantic link. However, once the confidence thresholds of the model are learned during training, they are shared for all target classes. This might lead to cases where an "out of domain" target class still get some attributes transferred. For example, we saw in our cross data set transfer evaluation how some animal-related attributes get transferred to man-made

objects. Hence, incorporating the semantic similarity of the unseen class to the attribute vocabulary to dynamically estimate these threshold at test time might help in guarding against such cases and boost the transfer performance.

Chapter 6

Automatic Attribute Discovery from Natural Language

We saw in the previous chapter how we can circumvent the need for user intervention to associate categories with attributes by predicting them automatically via semantic relations. However, while this overcomes the issue of manually defined associations, we are still reliant on a predefined attribute vocabulary. Such vocabulary is usually manually defined, where several human operators (or domain experts) have carefully engineered this set of attributes with certain properties in mind like discrimination and shareability.

This is clearly a major obstacle for attribute-based approaches to scale to a large number of categories. The cost associated with providing such annotations is prohibitive which limits the available attribute data sets either in the number of classes, attributes or images (see Section 2.1.3). Additionally, this non-trivial and expensive work is needed again when moving across data sets or expanding the current set with new categories. On the other hand, modern machine learning frameworks, like deep learning, require tens of thousands of training samples in order to generalize well. Hence, to employ such powerful frameworks for attribute-based visual recognition, large-scale training data is essential.

To tackle this problem, we propose in this chapter an end-to-end unsupervised attribute learning approach. Our goal is to automatically mine an attribute vocabulary and predict their associations to object categories in a large-scale setting. We achieve this by utilizing the large text corpora available in the web. Online encyclopedias represent a rich source of information which encode the collective human knowledge over various concepts and

Figure 6.1: An encyclopedia article describing an object category. Many discriminative attributes regarding shape, family and habitat of the object can be identified already in the first few lines of the article. We propose a model that utilizes such a knowledge source to automatically discover and learn visual semantic attributes at a large scale.

categories. It is an active and comprehensive knowledge source that keeps growing at impressive rates (Wikipedia, 2017). Figure 6.1 shows a snippet of an article describing the animal category *Wombat*. One can easily observe that numerous distinctive attributes for this category about its shape, taxonomy and habitat already appear in the introduction of the article. In our approach, we tap to this knowledge source to discover these interesting semantic concepts (attributes) and associate them with images to bridge the gap between language and vision.

Contributions. To that end, our main contributions are: a) We propose a novel attribute mining approach from natural textual descriptions that not only accounts for discrimination but also mines a diverse and salient vocabulary which correlates well with the human concept of semantic attributes. b) We propose a novel approach to associate these mined attributes with classes using a deep convolutional model that leverages visual data to account for the noisy and missing information in text corpora. c) We experimentally demonstrate that our deep attribute model is able to learn and predict attributes with high accuracy on ImageNet, as well as it generalizes well across data sets and outperforms state of the art in zero-shot learning on three benchmarks.

Publication. This chapter is based on our work that is published in Al-Halah and Stiefelhagen (2017).

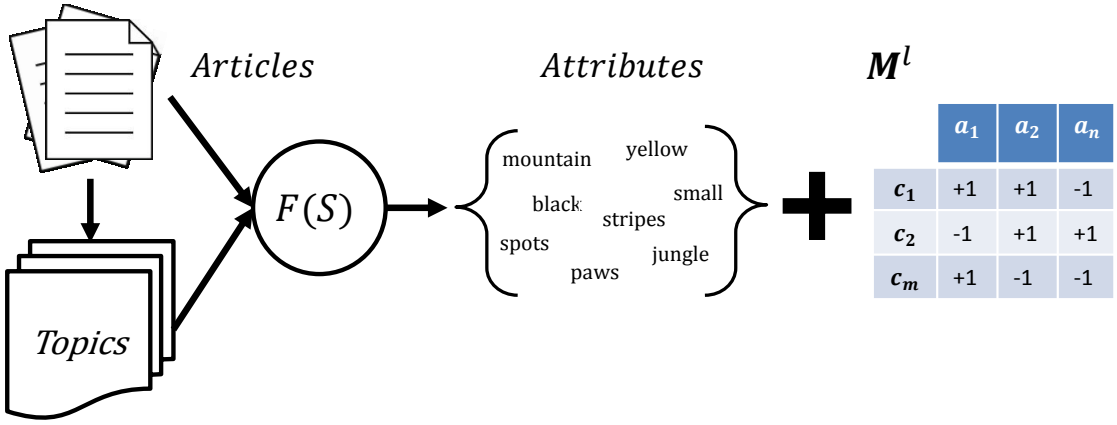


Figure 6.2: Discovering an attribute vocabulary from textual descriptions. Our model leverages text from articles and their underlying latent topics to select a compact, discriminative, diverse and salient set of semantic attributes.

6.1 Overview

In the following, we propose an end-to-end approach for large scale attribute-based visual recognition. Starting with a set of articles describing the object categories, our approach consists of three main steps: 1) We automatically analyze the articles in order to extract an attribute vocabulary with the most salient and discriminative words to describe these categories (Section 6.2). Then, 2) we optimize the class-attribute associations using visual data by a novel deep convolutional model with a linguistic prior and joint optimization of class and attribute predictions (Section 6.3). Finally, 3) we train a deep neural model for large scale attribute classification (Section 6.4). In Section 6.6, we provide an extensive evaluation and analysis of our model.

6.2 Semantic attribute discovery

Let $D = \{d_j\}_{j=1}^{J}$ be a set of text documents describing all object categories $C = \{c_m\}_{m=1}^{M}$ in the data set. For notation simplicity, we assume $|D| = |C|$, *i.e.* there is one document for each category. Let $W = \{w_i\}_{i=1}^{I}$ be the dictionary of words learned from D. Then, our goal is to select a subset vocabulary $A \subseteq W$ that best describes C:

$$A = \arg\max_{S \subseteq W} F(S) \quad \text{where} \quad |S| \leq b, \tag{6.1}$$

where $F(\cdot)$ is a set function that captures the desired properties of the subset S, and b is the size of the sought vocabulary.

Ideally, words in A should: 1) discriminate well between the object categories; 2) describe diverse aspects of the categories rather than focusing only on one or few properties (e.g. only *colors* or *parts*); and 3) represent salient semantic concepts understandable by humans. Next, we describe how we capture these different criteria of S in our objective function (Figure 6.2).

6.2.1 Discrimination

Let $V = \{\mathbf{v}_j = f_v(d_j) : \mathbf{v}_j \in \mathbb{R}^{|W|}\}_{j=1}^{J}$ be a text-based embedding (*e.g.* $f_v(\cdot)$ is based on tf·idf or one-hot encoding) learned over the document set D such that v_j^i captures the word w_i importance in document d_j. We construct an undirected fully connected graph $G(N, E)$. Each node $n_i \in N$ represents a category c_i. Each edge $e_{ij}(i \neq j)$ has a weight:

$$g_{ij}(S) = \sum_{w_k \in S} |v_i^k - v_j^k| \tag{6.2}$$

that captures how well words in S discriminate one class from the others. Additionally, each node has a self loop e_{ii} with a weight

$$g_{ii}(S) = \sum_{j \neq i} \sum_{w_k \notin S} |v_i^k - v_j^k|. \tag{6.3}$$

To capture the discriminative power of a set S, we employ the entropy rate of a random walk X on graph G as defined by Liu et al. (2014) and Zheng et al. (2014a).

In summary, let $g_i(S) = \sum_j g_{ij}(S)$ be the sum of incident weights of node n_i, and the total sum of weights in the graph is $g_T = \sum g_i$. The transition probability among the nodes is set to:

$$p_{ij}(S) = \begin{cases} \frac{g_{ij}(S)}{g_i(S)} & \text{if } i \neq j \\ 1 - \frac{\sum_t g_{it}(S)}{g_i(S)} & \text{if } i = j \end{cases} \tag{6.4}$$

Note that p_{ij} is a set function and the transition probabilities will change when the selected set S changes. The incident weights g_i for each node in the graph are kept constant because of the self loops weight g_{ii}. Let the stationary distribution for the

random walk be $\mu = (\mu_1, \mu_2, \ldots, \mu_{|N|})$, where $\mu_i = \frac{g_i}{g_T}$. Then the entropy rate of a random walk on G is:

$$F_{dis}(S) = -\sum_i \mu_i \sum_j p_{ij}(S) log(p_{ij}(S)) \tag{6.5}$$

The maximization of F_{dis} demands the maximization of p_{ij}, *i.e.* the discrimination among all pairs of classes.

6.2.2 Diversity

Another desired property of a good set of attributes is that it describes various aspects of the categories. That is, we want to encourage diversity among the selected words to reduce the bias towards a specific set of classes and to mine a vocabulary that describes all categories equally well. In order to promote diversity, we first uncover the latent semantic structure among the categories. We leverage here the unsupervised probabilistic topic models, *e.g.* LDA (Blei et al., 2003), to discover underlying themes in the documents.

Let $T = \{T_k\}_{k=1}^K$ be a set of topics learned from documents D and dictionary W. We define the diversity objective criteria as:

$$F_{div}(S) = \sum_{T_k} \sqrt{\sum_{w_i \in S} s(w_i, T_k)} \tag{6.6}$$

where

$$s(w_i, T_k) = \begin{cases} p(w_i|T_k) & \text{if } T_k = \arg\max_{T_j} p(T_j|w_i) \\ 0 & \text{otherwise} \end{cases} \tag{6.7}$$

F_{div} encourages topic diversity in S since adding words that belong to a previously chosen topic will have diminishing gain because of the square root function. It also accounts for word importance for the topic since adding a word with higher $p(w_i|T_k)$ results in a higher gain. Moreover, by encouraging diversity, F_{div} also discourages redundancy. A word and its synonyms are more likely to belong to the same topic, hence they are less likely to be chosen together. That is, F_{div} favors a diverse, less redundant and representative set of words.

Rank	Top Words in Topic
1	instrument play music sound pitch note musical reed player violin make tone range octave bass family key band fiddle hole
2	spaniel english welsh cocker springer show cardigan field pembroke work dock type small sussex average come line variety would century
3	missile target system wing guide flight use force parachute engine know projectile rocket air lift guidance kinetic anti weapon shuttle
⋮	
198	call include allow many time upper consist long much several little last low reach second slow half make follow suitable
199	use make allow would prevent work take give open cause come reduce keep provide way protect help less leave property
200	use century become modern early world work time begin develop could history new war late development introduce part include today

Table 6.1: Ranking of discovered topics according to their significance, *i.e.* how different they are from *junk* topic prototypes. While the top ranked topics capture salient concepts like *music* and *dogs*, the low ranked ones are obscure and have no particular theme.

6.2.3 Saliency

An important aspect of semantic attributes is that they represent *salient* words with relatively clear semantic concepts, *e.g.* "leg", "yellow" and "transparent". Whereas words like "become", "allow" and "various" belong to the background language structure, hence they are usually ambiguous and carry less or no semantics by themselves. Capturing word saliency directly is hard due to word polysemy and since word importance depends on the context. Therefore, we propose to capture this property using the learned underlying topic structure among the documents as a proxy.

One can estimate the significance of a topic by comparing its distribution over the words $p(w|\text{topic})$ and documents $p(d|\text{topic})$ to *junk* topics prototypes (AlSumait et al., 2009). A *junk* topic is one that has uniform distribution over words (*i.e.* it doesn't capture any specific theme) or over documents (*i.e.* it captures the common theme of all documents). By measuring the distance (*e.g.* KL divergence) of the discovered topics to these *junk* prototypes, we can obtain a ranking of the topics regarding their significance.

Table 6.1 shows the highest and lowest ranked topics over a set of documents using topic significance analysis model from AlSumait et al. (2009). The model considers three junk topic prototypes: the uniform distributions over words and documents and the vacuous semantic distribution. It ranks the discovered topics by significance using their weighted average distance to the junk prototypes measured by three metrics: KL-divergence, cosine distance and correlation coefficient. One can see that the top ranked

topics revolve around specific themes like “music”, “dogs” and “military”, while the lowest ranked topics have no theme in particular and are related to the background structure of the language or the documents domain.

Let $\text{insig}(T)$ be the set of $\rho = 10\%$ lowest ranked topics. We define a saliency cost function as:

$$\mathcal{C}(S) = \sum_{w_i \in S} (1 + \gamma \sum_{T_k \in \text{insig}(T)} p(T_k|w_i)), \tag{6.8}$$

where γ controls the contribution of the insignificance score of a word to the cost function. $\mathcal{C}(\cdot)$ favors *salient* words which will have a cost close to 1 while it punishes *junk* words which have a higher probability to give rise to *junk* topics.

6.2.4 Submodular optimization

We formulate the vocabulary selection problem in a submodular knapsack framework (Atamtürk and Narayanan, 2009). A set function F is submodular if it satisfies the decreasing marginal gain condition (Fujishige, 2005) *i.e.*:

$$F(A \cup \{s\}) - F(A) \geq F(B \cup \{s\}) - F(B) \text{ for } A \subseteq B. \tag{6.9}$$

In other words, the benefit of adding a new element s to the set is higher if it happens earlier. All the previous functions F_{dis}, F_{div} and C satisfy the marginal gain condition and are submodular.

Discrimination: A formal proof of the submodularity of the entropy rate of a random walk on G can be found in Liu et al. (2014). Informally, according to the defined edge weights $g_{ij}(\cdot)$ in the constructed graph G, adding a word w to S will result in increasing the edge weights between some node pairs in G. However, this increase will result in a lower increase in the uncertainty of the walk if it happens in a later stage, since then it will be shared with the contributions of other words in S discriminating between the same pairs.

Diversity: Note that $s(w_i, T_k) \geq 0 \ \forall w_i$. Adding a new word w that belongs to a new topic (the outer sum) will result in a higher gain than adding a word to an already existing topic in S (the inner sum). This is due to the fact that the square root function is a

monotonically increasing concave function and thus submodular (Fujishige, 2005). F_{div} is sum of submodular functions and, therefore, it is submodular.

Saliency: Note that $p(T_k|w_i) \geq 0 \ \forall w_i$, therefore $\mathcal{C}(S) > 0$ and is monotonically increasing. Adding a new word w to S in $\mathcal{C}$ will always increase the cost by the same margin, hence $\mathcal{C}$ is submodular.

We formulate our main objective function as:

$$\max_{S \subseteq W} F(S) \quad = F_{dis}(S) + \lambda F_{div}(S) \quad \text{subject to} \quad \mathcal{C}(S) \leq b \tag{6.10}$$

where b is the budget and λ is a hyper-parameters controlling the contribution of F_{div}. $F(\cdot)$ is submodular since it is a linear combination of submodular functions (Fujishige, 2005).

Since $F(\cdot)$ is nondecreasing (*i.e.* $F(S_1) \leq F(S_2) \ \forall S_1 \subseteq S_2 \subseteq W$) and $F(\emptyset) = 0$, then $F(\cdot)$ can be optimized using a greedy algorithm with a guaranteed solution to be within at least a constant fraction $(1 - 1/e)$ of the optimal objective value (Nemhauser et al., 1978). Moreover, we can take advantage of the submodular property of $F(\cdot)$ and use a *lazy* greedy algorithm (Minoux, 1978) to significantly reduce the computation cost. That is, since the gain of adding an item to the set S can only decrease after adding other items, we can adopt a lazy evaluation scheme where only the few most promising candidates are evaluated to select the best. Specifically, we adopt the Cost-Effective Lazy Forward (CELF) greedy selection algorithm (Leskovec et al., 2007). CELF takes into consideration the *benefit to cost* ratio to measure the gain of adding a new element and, at the same time, it maintains the bounds on the solution quality. We start with an empty set $S = \{\}$, then we incrementally add elements to S with maximum gain according to F using *lazy* evaluations.

6.3 Association optimization with a linguistic prior

In the previous step, we have selected the best attribute vocabulary A that describes the different categories $c_i \in C$ in our data set. Having this set of words, we get an

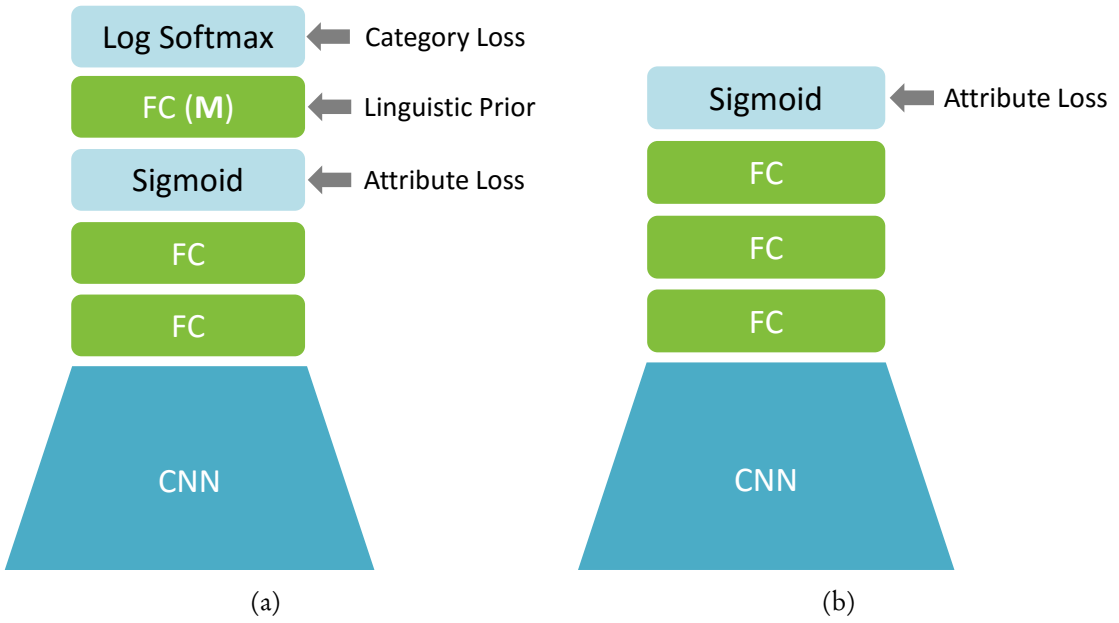


Figure 6.3: (a) The joint optimization of class-attribute associations using a linguistic prior and (b) the deep attribute model architecture.

initial estimate of the class-attribute association matrix $\mathbf{M}^l = [m_{ij}]$ (Figure 6.2) using the text-based embedding $\mathbf{V}$ learned over D.

$$m_{ij} = \begin{cases} +1 & \text{if } v_j^i > 0 \\ -1 & otherwise \end{cases} \tag{6.11}$$

However, this association matrix may contain some noise since $\mathbf{V}$ does not capture the context, and not all relations for a certain category are necessarily represented in the respective text documents. Usually, simple and obvious attributes of a class are omitted from text if they are not interesting enough to mention from the perspective of the author. For example, while most animals have attributes like "head", "leg" or "skin", these are not always mentioned in text when describing the animal unless there is something special about it. Moreover, $\mathbf{V}$ is a bag of words representation, *i.e.* it does not capture the context of the attributes in text. This results in a negative relation like "a tiger *does not* live in ocean" being captured as a positive association between "tiger" and "ocean" (*i.e.* $m_{ocean}^{tiger} = +1$) since $\mathbf{V}$ relies only on the presence of the word in the description.

We propose to improve the initial associations obtained from language by grounding it to visual data using a deep convolutional network model. The network is trained to

predict both attributes and categories while at the same time constraining the weights of the last layer to the initially estimated associations $\mathbf{M}^l$ (see Figure 6.3a). Note that this architecture resembles the direct attribute prediction model DAP where the object class is estimated based on the predicted attributes. We define the training loss function as:

$$\mathcal{L}(x) = \mathcal{L}_c(x) + \beta_1 \mathcal{L}_a(x) + \beta_2 \|\mathbf{M} - \mathbf{M}^l\|_1 \tag{6.12}$$

where $\mathcal{L}_c$ and $\mathcal{L}_a$ are the cross entropy loss of predicting the object category and the binary attributes of a sample x, respectively. $\|\mathbf{M} - \mathbf{M}^l\|_1$ is an entry-wise L_1 regularization term over the weights of the last fully connected layer $\mathbf{M}$ based on the initial association matrix $\mathbf{M}^l$.

By using the linguistic prior we force the network to preserve the semantic link between linguistic and visual data. This prevents the network from finding arbitrarily data-driven associations that can not be estimated anymore from textual description. At the same time, by controlling β_2 we allow for small modification to the associations when there is a strong visual signal supporting change to account for noise and missing information in $\mathbf{M}^l$.

We adopt an AlexNet-like architecture (Krizhevsky et al., 2012) for the joint deep model. That is, we have 5 convolutional layers followed by two fully connected layers and a Sigmoid activation function for attribute prediction, then another fully connected layer with softmax activation for category classification.

At the end of the joint optimization, we get the new binary association matrix of classes and attributes $\mathbf{M}^*$ by thresholding the weights of the last layer $\mathbf{M}$. The optimized associations $\mathbf{M}^*$ redefine the positive and negative label assignments for each attribute which were initially based on $\mathbf{M}^l$.

6.4 Deep attribute model

Finally, given the optimized associations $\mathbf{M}^*$ from the previous step, we train a deep model for attribute prediction (Figure 6.3b). The network has a similar architecture as the one we used for the joint optimization. However, we remove the last layer for the category prediction and add a new fully connected layer before the attribute prediction layer. That is, the network is made of 5 convolutional layers followed by three fully connected layers. The last attribute prediction layer is followed by a Sigmoid

activation function. We use the cross entropy loss to train the network for binary attribute prediction.

Predicting objects. Given an image x, we estimate the corresponding object category using the direct attribute prediction model (DAP) (Lampert et al., 2009). We adopt a summation formulation rather than the probabilistic one since it is more efficient (Al-Halah and Stiefelhagen, 2015b; Rohrbach et al., 2011), especially in our large-scale case. That is, for a class c_m, the estimated prediction score of c_m to appear in image x as:

$$s(c_m|x) = \frac{\sum_i [[a_i^{c_m} = 1]]\ s(a_i|x)}{\sum_i [[a_i^{c_m} = 1]]}. \tag{6.13}$$

where $s(a_i|x)$ is the prediction score of attribute a_i in image x, $a_i^{c_m}$ are the attributes of class c_m, and the classification scores are normalized to have a zero mean and unit standard deviation.

Zero-shot learning. We use the same formulation from Eq. 6.13 for classifying unseen categories in zero-shot learning. However, in this case the associations of the novel class are estimated directly from the textual description.

6.5 Evaluation setup

Data set. Through our experiments, we use the ILSVRC2012 data set from ImageNet (Russakovsky et al., 2015). It contains 1000 categories and more than 1.2 million images.

Articles. To collect the encyclopedia articles, we coded a simple tool to automatically retrieve the articles from Wikipedia. For each synset in the ImageNet data set, the tool queries the Wikipedia API using the different words in the synset to get articles with the same title. We were able to collect about 89% of the articles automatically using the previous method. The rest of the synsets either did not have an exact matching article title in Wikipedia or there were ambiguities in the retrieved articles since multiple ones matched the query. Articles for these synsets were then acquired interactively. In the end we acquired 1100 articles with around 80500 unique words.

Preprocessing. All document are preprocessed to remove non-alphabetic characters, and words are lower cased and stemmed. To avoid bias toward lengthy articles for some

categories, we truncate the articles length to a maximum of 500 words. We extract a tf·idf (term frequency·inverse document frequency) embedding for each document in the set. The tf·idf measures the importance of a word in a document by accounting for how often this word appears in the document and how frequent it appears in all other documents. We use the normalized *tf* and logarithmic *idf* scores (Salton and Buckley, 1988). For each synset, we average the embedding over all its documents to get its final representation.

Implementation details. For the attribute discovery, we learn a set of 200 topics using the Latent Dirichlet Allocation model (Blei et al., 2003). We empirically set $\lambda = 0.001$, $\gamma = 20$ and the maximum number of attributes to discover $b = 1200$. We set the hyperparameters β_1 and β_2 for the joint deep model such that the initial losses from the three terms are of similar magnitudes. Figure 6.4 shows the detailed architecture of the deep neural networks used for the joint and attribute models. All the convolutional and hidden fully connected layers are followed with a batch normalization layer and a ReLU activation function. For the final deep attribute model, we initialize the weights of the convolutional layer from the previous network trained for the joint optimization. All networks are trained using Adam (Kingma and Ba, 2015) for stochastic optimization with an initial learning rate of 0.001 and a weight decay of $5e-4$.

6.6 Experiments

In this section, we provide a thorough evaluation of our model in selecting a set of good attributes, association optimization and predicting semantic attributes. Furthermore, we evaluate our deep attribute model in zero-shot learning and its generalization properties across data sets.

6.6.1 Selecting the attribute vocabulary

We evaluate the quality of the selected attribute vocabulary from two perspectives: 1) the performance of the attribute embedding in capturing object similarity and; 2) the vocabulary saliency.

Attribute-based class embedding. A good attribute representation of categories should capture the similarity among the classes. That is, categories that are visually similar

(a) Deep Joint Model

(b) Deep Attribute Model

Figure 6.4: The detailed network architecture used for (a) the joint and (b) attribute models.

should share most of their attributes and have similar embeddings. To capture the quality of the attribute embedding, we rank the classes based on their similarity in the attribute embedding space. We use the normalized discounted cumulative gain (nDCG) (Siddiquie et al., 2011) to compare among the different methods:

$$\text{nDCG} = \frac{DCG_k}{IDCG_k} \text{ where } \text{DCG}_\text{k} = \sum_{i=1}^{k} \frac{2^{rel_i} - 1}{log_2(i+1)} \tag{6.14}$$

Such that rel_i is the relevance of the i^{th} ranked sample, and the ideal rank score IDCG_k is that for the rank of the classes for each category based on their distances in the ImageNet hierarchy.

As baselines, we consider several common feature selection methods: 1) max-Relevance and min-Redundancy (mRmR) (Peng et al., 2005); 2) Multi-Cluster Feature Selection (MCFS) (Cai et al., 2010); 3) Local Learning-based Clustering method (LLC-fs) (Zeng

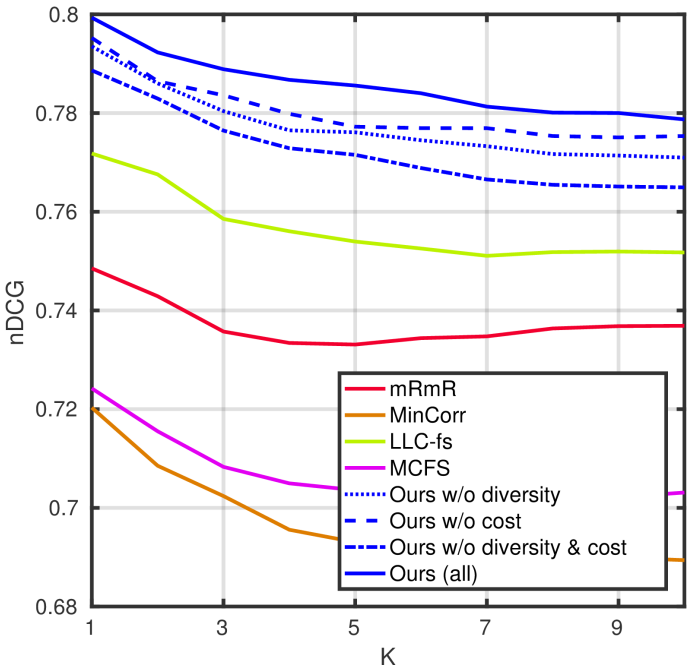


Figure 6.5: The ranking performance of the attribute embedding discovered by our approach against the baselines.

Model	Relevance (%) ↑	Junk (%) ↓	Saliency (%) ↑
mRmR	20.8	53.0	33.9
MinCorr	14.4	20.6	46.9
LLC-fs	29.1	42.9	43.1
MCFS	18.6	13.6	52.5
Ours	**44.5**	**2.6**	**71.0**

Table 6.2: Saliency scores of the selected vocabularies.

and Cheung, 2011); 4) Minimum Correlation (MinCorr) which selects words that have the least correlation with the rest of the vocabulary.

Figure 6.5 shows the ranking quality of all the baselines and our approach up to position $K = 10$ in the ranking list. Our approach outperforms all baselines and produces an embedding that captures the within category similarities. We also consider different variants of our approach by removing some of the optimization terms from Eq. 6.10. Each of the terms used in our submodular optimization contributes positively to the quality of the attribute embedding.

Vocabulary saliency. Here, we explore how the selected vocabulary correlates with human understanding of salient semantic attributes. To that end, we pick 100 synsets that are uniformly distributed in the ImageNet hierarchy. For each category, we select 50 random words from the dictionary with positive tf·idf scores for that class. We asked 5 annotators to classify the association between each class and its 50 words into 4 categories:

1. *positive*: such as "The horse has a tail".
2. *negative*: like "The dolphin does not walk".
3. *unknown*: when the annotator does not have the knowledge to decide the type.
4. *junk*: when the word itself does not carry a clear concept to define an association.

The majority of the annotators agree on 84% of the labels. The labels are distributed as (25.1% *positive*, 47.8% *negative*, 1.9% *unknown* and 25.2% *junk*).

Out of the 4 categories, we are interested in the *positive* and *junk* categories since they describe the semantic saliency of the words. The *negative* and *unknown* types do not deliver much information about the semantics since a word having a negative association might have a positive one with other classes, while the *unknown* reflects the lack of knowledge of the annotator.

We obtain the probability of a word from the annotation vocabulary $w_i \in W^A$ to engage in a positive association $p(+|w_i)$ or being junk $p(J|w_i)$ by marginalizing over all annotators and object classes. We then define the weighted relevance of the selected words S as:

$$Relevance(S) = \frac{\sum_{w_i \in S \cap W^A} p(+|w_i)}{\sum_{w_j \in W^A} p(+|w_j)}, \tag{6.15}$$

and similarly for the junk score:

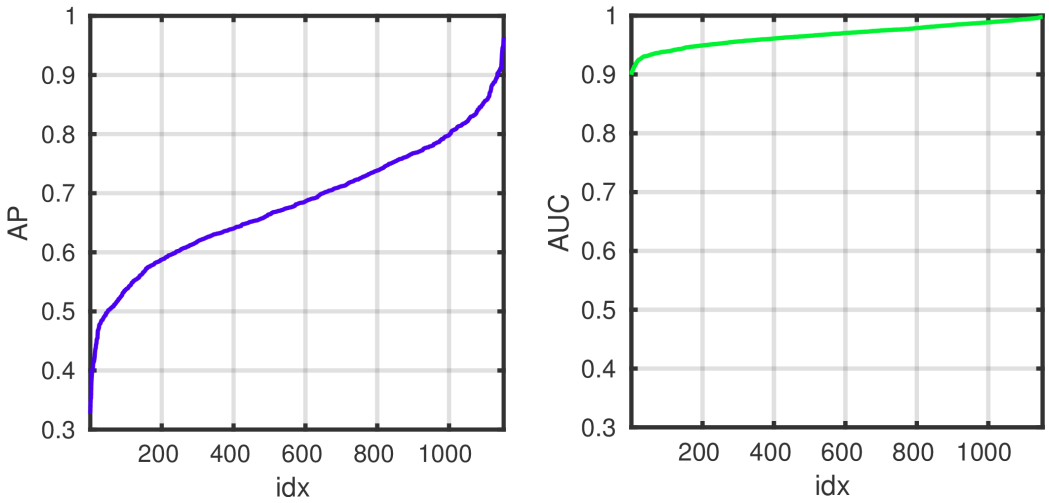
$$Junk(S) = \frac{\sum_{w_i \in S \cap W^A} p(J|w_i)}{\sum_{w_j \in W^A} p(J|w_j)}. \tag{6.16}$$

The final saliency score of S is then defined as the average of both:

$$Saliency(S) = \frac{1}{2}(Relevance(S) + (1 - Junk(S))). \tag{6.17}$$

Model	Attributes		Categories (DAP)	
	Accuracy	AP	Top1	AP
Joint Model				
w/o Linguistic Prior	55.2	22.4	30.4	19.0
w/ Linguistic Prior	**60.3**	**28.9**	**45.2**	**39.5**
Attribute Model				
w/o Association Opt.	74.8	64.1	51.4	48.3
w/ Association Opt.	**76.9**	**68.2**	**55.9**	**54.2**

Table 6.3: Attribute prediction performance.



(a) (b)

Figure 6.6: Performance of individual attributes in average precision (AP) and area under receiver operating characteristic (AUC).

Table 6.2 shows the performance of our approach and the baselines from the previous section. While some of the baselines performed relatively well in getting a good attribute embedding, large portions of the selected words by these methods do not carry a clear semantic concept. Our approach has a much higher relevance score while at the same time the lowest junk score among all baselines. This indicates that the set of attributes discovered by our method correlates well with the human concept of semantic attributes.

6.6.2 Attribute prediction

Having selected a set of salient attributes, we evaluate here the performance of our model in predicting these attributes in images. Table 6.3 shows the attribute prediction accuracy

and average precision (AP). It also reports the object Top1 classification accuracy and the AP based on the predicted attributes and when using the DAP model (Eq. 6.13).

Joint Model. In the first section of Table 6.3, it is interesting to see that regularizing the weights of the last fc layer with the language prior improves the performance of attribute predictions by 5% in accuracy and 6% in AP. At the same time, it results in a boost in object classification Top1 accuracy by 15%. These results show that side information obtained from language has a significant impact on the performance of the deep model. Additionally, the unregularized network learns quite different associations between classes and attributes than those in $\mathbf{M}^l$. Only 13% of the positive associations in this case are shared with those learned from the textual description. This indicates that the semantic link between the attributes and the classes is lost in this model. In contrast, the regularized model preserves the semantics and retains more than 93% of the positive associations in $\mathbf{M}^l$.

Attribute Model. Finally, training the deep attribute model with the optimized associations $\mathbf{M}^*$ results in a better model compared to a one trained directly using $\mathbf{M}^l$. This indicates that our joint model managed to account for some of the noise and missing data in $\mathbf{M}^l$. The deep attribute model trained with $\mathbf{M}^*$ has higher attribute and object prediction performance. Moreover, our deep attribute model achieves 75% Top5 object classification accuracy, by predicting objects through the semantic attribute layer. This is an impressive performance of the attribute model since it almost matches the performance of a deep model with the same architecture trained directly for object classification. By training such a model we get 80% Top5 accuracy.

Figure 6.6 shows the performance of the individual attributes. Around 80% of the attributes can be predicted with an average precision better than 0.6.

6.6.3 Zero-shot learning

An important feature of semantic attributes is their ability to form a shared knowledge layer which can be transfered to unseen classes. We evaluate here the performance of our discovered attributes in classifying unseen classes (*i.e.* zero-shot learning). While there is no standard zero-shot split in ImageNet, there are two common splits used in the literature and defined over the ILSVRC2010 classes, split A from Rohrbach et al. (2011) and B from Mensink et al. (2012). Both of them, split the classes into 800 seen and 200

Model	Split	200 labels	1000 labels
Rohrbach et al. (2011)	A	34.8	-
PST (Rohrbach et al., 2013a)	A	34.0	-
Ours	A	**46.1**	**15.9**
Ours - BT	A	**48.0**	**20.2**
Mensink et al. (2012)	B	35.7	1.9
DeViSE (Frome et al., 2013)	B	31.8	9.0
ConSE (Norouzi et al., 2014)	B	28.5	-
AMP (SR+SE) (Fu et al., 2015)	B	41.0	-
Ours	B	**46.3**	**15.2**
Ours - BT	B	**49.0**	**20.0**
Ours (w/o assoc. opt.)	C	45.8	14.8
Ours	C	**48.1**	**16.9**

Table 6.4: Zero-shot performance (Top5 accuracy) on 200 unseen classes from Imagenet.

unseen categories. We train our model as before while this time we use only the 800 seen classes of the respective split for training and we test on the remaining unseen classes.

Table 6.4 shows the Top5 accuracy of our model over the two splits (A & B). Our deep attribute model outperforms the state-of-the-art by 11% on split A and by 5% on split B.

Furthermore, we analyze the bias of our model toward seen classes similar to Frome et al. (2013). In this test setup, both the seen and unseen labels are considered as candidates when predicting the object category. Our model achieves 15% accuracy on split A & B and shows much less bias compared to state of the art with 6% improvement. Additionally, if we assume the availability of test data as a batch (Ours-BT), we can get a better estimation of the mean and standard deviation for classifiers scores in Eq. 6.13. This results in additional improvement of performance by 3%.

Since in the zero-shot settings, we optimize the associations using only data from the seen categories, we analyze in the last section of Table 6.4 the performance of our model with and without association optimization. Similar to A & B, we define a new split C on ILSVRC2012 with 800 seen and 200 unseen classes. Here again, we find that the association optimization did not result in a bias towards the seen classes, rather it improved the model performance. Overall, we see that optimizing the associations is beneficial in both within and across category prediction.

Model	Side Info.	AwA	aPY
Supervised ZSL			
DAP (Lampert et al., 2013) (AlexNet)	A	54.0	31.9
DAP (Lampert et al., 2013) (GoogLeNet)	A	59.5	37.1
Unsupervised ZSL			
DeViSE (Frome et al., 2013)	W	44.5	25.5
ConSE (Norouzi et al., 2014)	W	46.1	22.0
Changpinyo et al. (2016)	W	57.5	-
CAAP (Al-Halah et al., 2016b)	W	68.6	49.0
HAT (Al-Halah and Stiefelhagen, 2015b)	H	59.7	31.1
SJE (Akata et al., 2015)	H + G	60.1	-
Xian et al. (2016)	H + G + W	66.2	-
EZSL (Romera-Paredes and Torr, 2015)	T	58.5	-
Elhoseiny et al. (2013)	T	55.3	30.2
Qiao et al. (2016)	T	66.5	-
Ours (binary assoc.)	T	77.3	**57.6**
Ours (continuous assoc.)	T	**79.7**	57.5

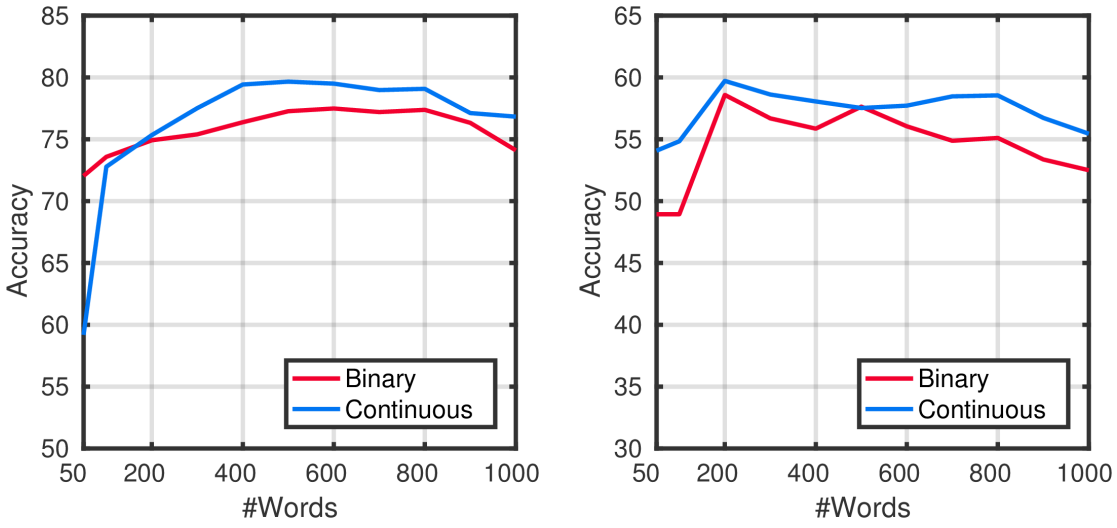
Table 6.5: Zero-shot performance of various models on AwA and aPY. The supervised models use manually defined attributes (A), while the unsupervised approaches rely on other sources like word embeddings such as Word2Vec (W) (Mikolov et al., 2013) and GloVe (G) (Pennington et al., 2014); hierarchy-based information (H) (Miller, 1995) or textual description (T).

6.6.4 Across data sets ZSL

To compare the performance of our model that we learned on ImageNet with a manually selected attribute vocabulary, we evaluate our deep attribute model on two public data sets: 1) Animals with Attributes (AwA) (Lampert et al., 2009); 2) aPascal/aYahoo (aPY) (Farhadi et al., 2009).

We collect articles for each of the unseen categories to extract their associations to our discovered attribute vocabulary. We consider both using the raw continuous associations (*i.e.* tf·idf values) and binary associations. When leveraging continuous associations, we simply weight the attribute predictions in Eq. 6.13 with the respective association weight learned from text. We test our model on the unseen categories on both data sets without any fine tuning of the trained deep model (off-the-shelf).

From Table 6.5 we see that our model outperforms all unsupervised zero-shot approaches. Compared to methods from Elhoseiny et al. (2013) and Qiao et al. (2016) that used



(a) AwA (b) aPY

Figure 6.7: Zero-shot performance with varying textual description lengths and when using binary and continuous associations.

similar type of side information as ours, we have up to 13% improvement. Moreover, our model outperforms a DAP model based on the manually defined attribute vocabulary and using image embeddings from an AlexNet model (Krizhevsky et al., 2012) or even from GoogLeNet (Szegedy et al., 2014). This demonstrates the impressive generalization properties of our model across data sets.

Text Length. Here, we explore the effect of the article length on the prediction performance. We vary the length of the considered section of the articles from 100 to 1000 words. Then, we extract the associations of the unseen classes in AwA and aPY from the truncated articles.

Figure 6.7 shows the performance of the model in this case. We notice that the optimal length of the article increases in correlation with the granularity of the categories in the data set. For AwA which contains only animal classes, on average longer articles (400 to 600 words) are needed to sufficiently extract discriminant associations. In contrast, categories in aPY are easier to separate with shorter articles (200 words). Moreover, we see that most of the important attributes are mentioned quite early in the article, with performance degrading when we consider relatively long articles (more than 800 words). In both data sets, we see that continuous associations outperform their binary counterpart in predicting the categories in most cases.

6.6.5 Discovered Attributes

Using our model, we have discovered and learned 1636 semantic attributes describing 1360 categories with more than 1.2 million images from ImageNet (ILSVRC2010 & ILSVRC2012). This amounts to roughly 2 million class-attribute associations. On average, each attribute is shared between 29 categories, and each category has about 33 active attributes. Some of the *most shared* attributes (with more than 100 categories) are *water*, *black*, *red*, *breed*, *tail*, *metal*, *coat*, *device*, *hunt*, *plastic*, *yellow* and *hair*. Some of the *least shared* attributes (with less than 10 categories) are *cassette*, *cowboy*, *pumpkin*, *sweater*, *convertible*, *ballistic*, *hump*, *axe*, *drilling*, *laundry*, *cash* and *quilt*.

6.7 Summary and discussion

In this chapter, we propose a novel end-to-end approach to discover and learn attributes at a large scale form textual descriptions. Our model discovers a salient, diverse and discriminative set of attribute vocabulary that correlates well with human understanding of semantic attributes. Moreover, in order to account for noise and missing data in the text corpora, we propose to use a linguistic prior in a joint deep model to optimize the class-attribute associations. In an evaluation on ImageNet, we show that our deep attribute model is able to learn and predict semantic attributes with high accuracy for a thousand categories. Our model outperforms the state-of-the-art in unsupervised zero-shot learning and it generalizes well across data sets.

Discussion. In this work, our model relied on a simple method to estimate the initial text-based associations. However, as discussed in Section 6.3, these estimation tend to be quite noisy. Hence, a more advanced NLP approach can help improving the initial estimation. For example, by taking the attribute context in the article into account or by incorporating a form of sentiment analysis, a model can predict whether an attribute is associated positively or negatively with a category. Additionally, our model produced automatic attribute annotation at the category level. However, we saw previously in Section 4.6.4 that image-level annotations lead to better attribute models and, consequently, better transfer performance. Thus, extending our model to produce image-level annotations can be beneficial. Finally, the AlexNet-like architecture used in our deep attribute model can be replaced with deeper and more advanced architectures (*e.g.* Huang et al., 2016; Szegedy et al., 2014) to learn more discriminative

attributes. Specifically, a fully convolutional architecture (*e.g.* He et al., 2016; Simonyan and Zisserman, 2014) might be better suited to learn the semantic attributes since many of them are correlated with local image regions that can be better captured with convolution layers.

Chapter 7

Application: Attributes for Fashion Forecast

We saw in the previous chapters various applications of semantic attributes in visual recognition and transfer learning. Due to their representation properties (Section 2.1), attributes were employed in a wide spectrum of vision applications, like image description, object classification, visual question answering *etc*. In this chapter, we introduce yet a novel real world application of semantic attributes, namely *visual fashion forecasting*.

Our goal is to predict the future popularity of fine-grained fashion styles (Figure 7.1). For example, having observed the purchase statistics for all women's dresses sold on Amazon over the last N years, can we predict what salient visual properties the best selling dresses will have 12 months from now? Given a list of trending garments, can we predict which will remain stylish into the future? Which old trends are primed to resurface, independent of seasonality?

The ability to predict the future of *styles* rather than merely *items* is appealing for applications that demand interpretable models expressing where trends as a whole are headed, as well as those that need to capture the life cycle of collective styles, not individual garments. Computational models able to make such style forecasts would be critically valuable to the fashion industry, in terms of portraying large-scale trends of what people will be buying months or years from now.

A key technical challenge in forecasting fashion is how to represent visual style. Unlike articles of clothing (e.g., sweater, vest), which are well-defined categories handled readily

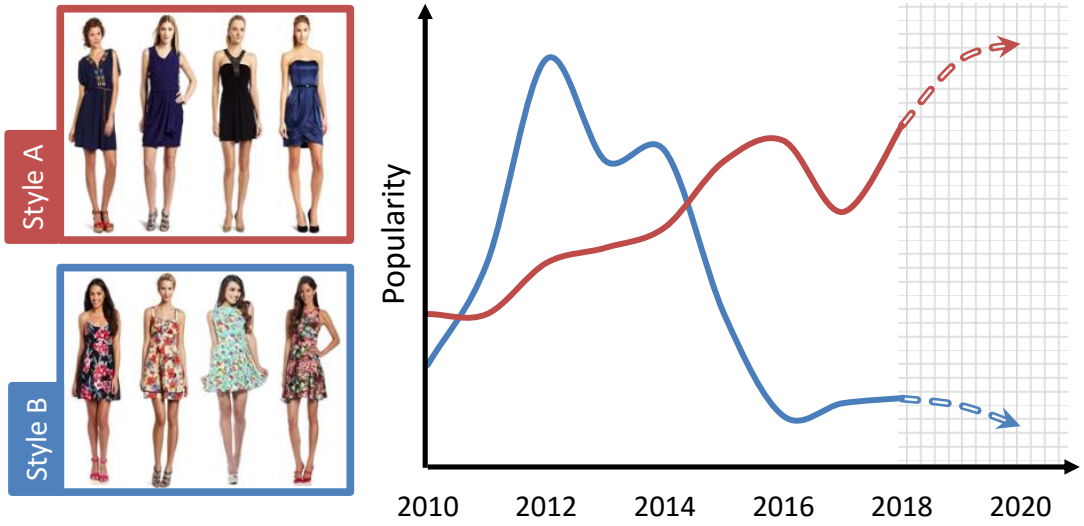


Figure 7.1: We propose to predict the future of fashion based on visual styles.

by today's sophisticated visual recognition pipelines (*e.g.* Bossard et al., 2012; Chen et al., 2012; Liu et al., 2016; Simo-Serra and Ishikawa, 2016), styles are more difficult to pin down and even subjective in their definition. In particular, two garments that superficially are visually different may nonetheless share a style. Semantic attributes represent a good bases to represent fashion styles. They capture visual concepts at various resolutions from the most local (*e.g. v-neck*) to the global (*e.g. casual*). Moreover, they provide us with a semantic description of styles which helps us to better understand what are the key characteristics that distinguish a style from another.

Contributions. To that end, we introduce the first approach that forecasts the popularity of visual styles. We propose a model to discover a vocabulary of latent styles from large scale unlabeled fashion images. Our model is capable of representing trends in the latent styles over time and predicting their popularity in the future. Furthermore, we show how to extract style dynamics (trendy vs. classic vs. outdated), and forecast the key visual attributes that will play a role in tomorrow's fashion—all based on learned *visual* models. Finally, we analyze the tradeoffs of various forecasting models and representations, the latter of which reveals the advantage of unsupervised style discovery based on visual semantic attributes compared to meta-data based and off-the-shelf CNN representations, including those fine-tuned for garment classification.

Publication. This chapter is based on our work that is published in Al-Halah et al. (2017). The work was carried out during a research internship at the University of Texas at Austin and in cooperation with Prof. Kristen Grauman.

7.1 Overview

We propose an approach to predict the future of fashion styles based on images and consumers' purchase data. Our approach 1) learns a representation of fashion images that captures the garments' visual attributes (Section 7.2); then 2) discovers a set of fine-grained styles that are shared across images in an unsupervised manner (Section 7.3); finally, 3) based on statistics of past consumer purchases, constructs the styles' temporal trajectories and predicts their future trends (Section 7.4).

7.2 Elements of fashion

In some fashion-related tasks, one might rely solely on meta information provided by product vendors, *e.g.*, to analyze customer preferences. Meta data such as tags and textual descriptions are often easy to obtain and interpret. However, they are usually noisy and incomplete. For example, some vendors may provide inaccurate tags or descriptions in order to improve the retrieval rank of their products, and even extensive textual descriptions fall short of communicating all visual aspects of a product.

On the other hand, images are a key factor in a product's representation. It is unlikely that a customer will buy a garment without an image no matter how expressive the textual description is. Nonetheless, low level visual features are hard to interpret. Usually, the individual dimensions are not correlated with a semantic property. This limits the ability to analyze and reason about the final outcome and its relation to observable elements in the image. Moreover, these features often reside in a certain level of granularity. This renders them ill-suited to capture the fashion elements which usually span the granularity space from the most fine and local (*e.g.* collar) to the coarse and global (*e.g.* cozy).

Semantic attributes serve as an elegant representation that is both interpretable and detectable in images. Additionally, they express visual properties at various levels of granularity. Specifically, we are interested in attributes that capture the diverse visual elements of fashion, like: *Colors* (*e.g.* blue, pink); *Fabric* (*e.g.* leather, tweed); *Shape* (*e.g.* midi, beaded); *Texture* (*e.g.* floral, stripe); and *Parts* (*e.g.* side-slit, sleeves). These attributes constitute a natural vocabulary to describe styles in clothing and apparel. As discussed in Section 2.1.4, some prior work considers fashion attribute classification (Huang et al., 2015; Liu et al., 2016), though none for capturing higher-level visual styles.

Figure 7.2: The architecture of our deep attribute CNN model.

To that end, we train a deep convolutional model for attribute prediction using the DeepFashion data set (Liu et al., 2016). The data set contains more than $200,000$ images labeled with $1,000$ semantic attributes collected from online fashion websites. Our deep attribute model has an AlexNet-like convolutional neural network (CNN) structure (Krizhevsky et al., 2012). Figure 7.2 shows the details of the network architecture for our attribute prediction model. The model is composed of 5 convolutional layers with decreasing filter sizes from 11×11 to 3×3 followed by 3 fully connected layers and 2 dropout layers with probability of 0.5. Additionally, each convolutional layer and the first two fully connected layers in our model are followed by a batch normalization layer and a rectified linear unit (ReLU). The last attribute prediction layer is followed by a sigmoid activation function. We use the cross entropy loss to train the network for binary attribute prediction. The network is trained using Adam (Kingma and Ba, 2015) for stochastic optimization with an initial learning rate of 0.001 and a weight decay of $5e-4$.

With this model we can predict the presence of $M = 1,000$ attributes in new images:

$$\mathbf{a}_i = f_a(x_i|\theta), \tag{7.1}$$

such that θ is the model parameters, and $\mathbf{a}_i \in \mathbb{R}^M$ where the m^{th} element in $\mathbf{a}_i$ is the probability of attribute a^m in image x_i, *i.e.*, $a_i^m = p(a^m|x_i)$. $f_a(\cdot)$ provides us with a detailed visual description of a garment that, as results will show, goes beyond meta-data typically available from a vendor.

7.3 Fashion style discovery

For each genre of garments (*e.g.*, Dresses or T-Shirts), we aim to discover the set of fine-grained styles that emerge. That is, given a set of images $X = \{x_i\}_{i=1}^{N}$ we want to discover the set of K latent styles $S = \{s_k\}_{k=1}^{K}$ that are distributed across the items in various combinations.

We pose our style discovery problem in a nonnegative matrix factorization (NMF) framework that maintains the interpretability of the discovered styles and scales efficiently to large data sets. First, we infer the visual attributes present in each image using the classification network described above. This yields an $M \times N$ matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$ indicating the probability that each of the N images contains each of the M visual attributes. Given $\mathbf{A}$, we infer the matrices $\mathbf{W}$ and $\mathbf{H}$ with nonnegative entries such that:

$$\mathbf{A} \approx \mathbf{WH} \quad \text{where } \mathbf{W} \in \mathbb{R}^{M \times K},\ \mathbf{H} \in \mathbb{R}^{K \times N}. \tag{7.2}$$

We consider a low rank factorization of $\mathbf{A}$, such that $\mathbf{A}$ is estimated by a weighted sum of K rank-1 matrices:

$$\mathbf{A} \approx \sum_{k=1}^{K} \lambda_k . \mathbf{w}_k \otimes \mathbf{h}_k, \tag{7.3}$$

where $\otimes$ is the outer product of the two vectors and λ_k is the weight of the k^{th} factor (Kolda and Bader, 2009).

By placing a Dirichlet prior on $\mathbf{w}_k$ and $\mathbf{h}_k$, we insure the nonnegativity of the factorization. Moreover, since $||\mathbf{w}_k||_1 = 1$, the result can be viewed as a topic model with the styles learned by Eq. 7.2 as topics over the attributes. That is, the vectors $\mathbf{w}_k$ denote common combinations of selected attributes that emerge as the latent style "topics", such that $w_k^m = p(a_m|s_k)$. Each image is a mixture of those styles, and the combination weights in $\mathbf{h}_k$, when $\mathbf{H}$ is column-wise normalized, reflect the strength of each style for that garment, *i.e.*, $h_k^i = p(s_k|x_i)$.

Note that our style model is unsupervised which makes it suitable for style discovery from large scale data. Furthermore, we employ an efficient estimation for Eq. 7.3 for large scale data using an online MCMC based approach (Hu et al., 2015). At the same time, by representing each latent style s_k as a mixture of attributes $[a_k^1, a_k^2, \ldots, a_k^M]$, we have the ability to provide a semantic linguistic description of the discovered styles in

addition to image examples. Figure 7.4 shows examples of styles discovered for three data sets (genres of products) studied in our experiments.

Finally, our model can easily integrate multiple representations of fashion when it is available by adjusting the matrix $\mathbf{A}$. That is, given an additional view (*e.g.*, based on textual description) of the images $\mathbf{U} \in \mathbb{R}^{L \times N}$, we augment the attributes with the new modality to construct the new data representation $\acute{\mathbf{A}} = [\mathbf{A}; \mathbf{U}] \in \mathbb{R}^{(M+L) \times N}$. Then $\acute{\mathbf{A}}$ is factorized as in Eq. 7.2 to discover the latent styles.

7.4 Forecasting visual style

We focus on forecasting the future of fashion over a 1-2 year time course. In this horizon, we expect consumer purchase behavior to be the foremost indicator of fashion trends. In longer horizons, *e.g.*, 5-10 years, we expect more factors to play a role in shifting general tastes, from the social, political, or demographic changes to technological and scientific advances. Our proposed approach could potentially serve as a quantitative tool towards understanding trends in such broader contexts, but modeling those factors is currently out of the scope of our work.

The temporal trajectory of a style. In order to predict the future trend of a visual style, first we need to recover the temporal dynamics which the style went through up to the present time. We consider a set of customer transactions Q (*e.g.*, purchases) such that each transaction $q_i \in Q$ involves one fashion item with image $x_{q_i} \in X$. Let Q^t denote the subset of transactions at time t, *e.g.*, within a period of one month. Then for a style $s_k \in S$, we compute its temporal trajectory y^k by measuring the relative frequency of that style at each time step:

$$y_t^k = \frac{1}{|Q^t|} \sum_{q_i \in Q^t} p(s_k | x_{q_i}), \tag{7.4}$$

for $t = 1, \ldots, T$. Here $p(s_k|x_{q_i})$ is the probability for style s_k given image x_{q_i} of the item in transaction q_i.

Forecasting the future of a style. Given the style temporal trajectory up to time n, we predict the popularity of the style in the next time step in the future $\hat{y}_{n+1}$ using an exponential smoothing model (Brown and Meyer, 1961):

$$\begin{aligned} \hat{y}_{n+1|n} &= l_n \\ l_n &= \alpha y_n + (1-\alpha) l_{n-1}. \end{aligned} \tag{7.5}$$

Hence, by expanding the recursive definition of $\hat{y}_{n+1|n}$, we get:

$$\hat{y}_{n+1|n} = \sum_{t=1}^{n} \alpha(1-\alpha)^{n-t} y_t + (1-\alpha)^n l_0, \tag{7.6}$$

where $\alpha \in [0,1]$ is the smoothing factor, l_n is the smoothing value at time n, and $l_0 = y_0$. In other words, our forecast $\hat{y}_{n+1}$ is an estimated mean for the future popularity of the style given its previously observed temporal dynamics y_t.

The exponential smoothing model (EXP), with its exponential weighting decay, nicely captures the intuitive notion that the most recent observed trends and popularities of styles have higher impact on the future forecast than older observations. Additionally, due to the range of α, EXP tends to interpolate among previous observations and predicts smooth transitions of popularity which is expected for short-term forecasts of fashion trends. Furthermore, our selection of EXP combined with K independent style trajectories is partly motivated by practical matters, namely the public availability of product image data accompanied by sales rates. EXP is defined with only one parameter (α) which can be efficiently estimated from relatively short time series. In practice, as we will see in results, it outperforms several other standard time series forecasting algorithms, specialized neural network solutions, and a variant that models all K styles jointly (see Section 7.6.2). While some styles' trajectories exhibit seasonal variations (*e.g.* T-Shirts are sold in the summer more than in the winter), such changes are insufficient with regard of the general trend of the style. As we show later, the EXP model outperforms models that incorporate seasonal variations or styles' correlations for our data sets.

7.5 Evaluation setup

Data sets. We evaluate our approach on three data sets collected from *Amazon* by McAuley et al. (2015). The data sets represent three garment categories for women

Figure 7.3: The fashion items are represented with an image, a textual description, and a set of tags.

Data set	#Items	#Transaction
Dresses	19,582	55,956
Tops & Tees	26,848	67,338
Shirts	31,594	94,251

Table 7.1: Statistics of the three data sets from Amazon.

(Dresses and Tops&Tees) and men (Shirts) with around 80,000 unique items. An item in these sets is represented with a picture, a short textual description, and a set of tags (see Figure 7.3). Additionally, it contains the dates each time the item was purchased.

These data sets are a good testbed for our model since they capture real-world customers' preferences in fashion and they span a fairly long period of time. For all experiments, we consider the data in the time range from January 2008 to December 2013. We use the data from the years 2008 to 2011 for training, 2012 for validation, and 2013 for testing. Table 7.1 summarizes the data set sizes.

7.6 Experiments

Our experiments evaluate our model's ability to forecast fashion. We start with analyzing the discovered fashion styles by our model (Section 7.6.1). Then, we quantify its performance against an array of alternative models, both in terms of forecasters (Section 7.6.2)

and alternative representations (Section 7.6.3). We also demonstrate its potential power for providing interpretable forecasts, analyzing style dynamics (Section 7.6.4), and forecasting individual fashion elements (Section 7.6.5).

7.6.1 Style discovery

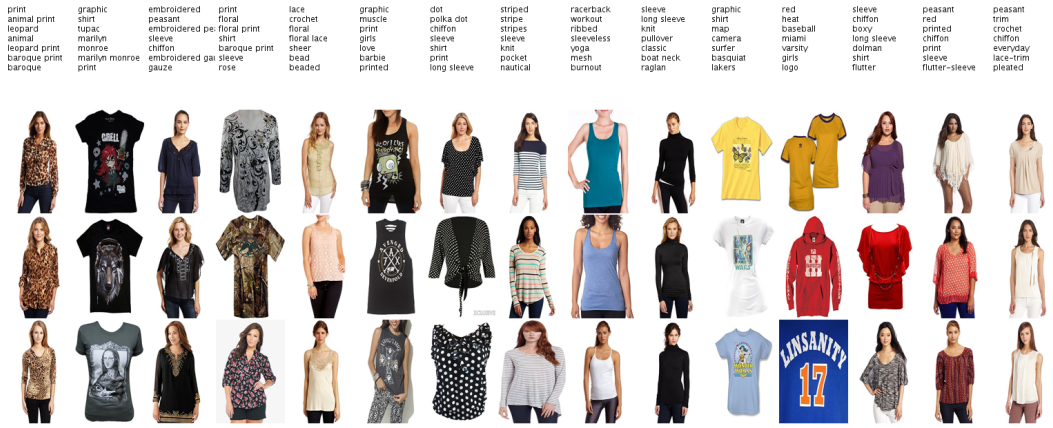
We use our deep model trained on DeepFashion (Liu et al., 2016) (cf. Section 7.2) to infer the semantic attributes for all items in the three data sets, and then learn $K = 30$ styles from each. We found that learning around 30 styles within each category is sufficient to discover interesting visual styles that are not too generic with large within-style variance nor too specific, *i.e.*, describing only few items in our data. Our attribute predictions average 83% AUC on a held-out DeepFashion validation set; attribute ground truth is unavailable for the Amazon data sets themselves.

Figure 7.4 shows 15 of the discovered visual styles in the three data sets along with the 3 top ranked items based on the likelihood of that style in the items $p(s_k|x_i)$, and the most likely attributes per style ($p(a_m|s_k)$). As anticipated, our model automatically finds the fine-grained styles within each genre of clothing. While some styles vary across certain dimensions, there is a certain set of attributes that identify the style signature. For example, color is not a significant factor in the 1^{st} and 3^{rd} styles (indexed from left to right) of Dresses. It is the mixture of shape, design, and structure that defines these styles (*sheath*, *sleeveless* and *bodycon* in 1^{st}, and *chiffon*, *maxi* and *pleated* in 3^{rd}). On the other hand, the clothing material might dominate certain styles, like *leather* and *denim* in the 11^{th} and 15^{th} style of Dresses, respectively. Having a Dirichlet prior for the style distribution over the attributes induces sparsity. Hence, our model focuses on the most distinctive attributes for each style. A naive approach (*e.g.*, clustering) could be distracted by the many visual factors and become biased towards certain properties like color, *e.g.*, by grouping all black clothes in one style while ignoring subtle differences in shape and material.

7.6.2 Style forecasting

Having discovered the latent styles in our data sets, we construct their temporal trajectories as in Section 7.4 using a temporal resolution of months. We compare our approach to several well-established forecasting baselines, which we group in three main categories:

(a) Dresses



(b) Tops & Tees

(c) Shirts

Figure 7.4: The discovered visual styles on (a) Dresses, (b) Tops & Tees and (c) Shirts data sets. Our model captures the fine-grained differences among the styles within each genre and provides a semantic description of the style signature based on visual attributes.

Naïve. These methods rely on the general properties of the trajectory. We consider three simple models:

1) *mean*: the future values are forecasted to be equal to the mean of the observed series, *i.e.* $\hat{y}_{n+1|n} = \frac{1}{n}\sum_{t=1}^{n} y_t$.

2) *last*: the forecast is equal to the last observed value, *i.e.* $\hat{y}_{n+h|n} = y_n$.

3) *drift*: the forecast follows the general trend of the series, *i.e.* $\hat{y}_{n+h|n} = y_n + \frac{h}{n-1}(y_n - y_1)$ where h is the forecast horizon.

Autoregression. These are linear regressors based on the last few observed values' "lags". That is, $\hat{y}_n = b + \sum_i^P \alpha_i y_{n-i} + \epsilon$ where b is a constant, $\{\alpha_i\}$ are the lag coefficients, P is the maximum lag (set by cross validation in our case) and ϵ an error term. We consider several variations (Box et al., 2015):

1) The standard linear autoregression model (*AR*).

2) the autoregression model that accounts for seasonality (*AR+S*): *i.e.* for a series with 12 months seasonality the model will also consider the lag at $n - 12$ along with most recent lags to predict the current value.

3) the vector autoregression (*VAR*) that considers the correlations between the different styles' trajectories.

4) and the autoregressive integrated moving average model *(*ARIMA): it models the temporal trajectory with two polynomials, one for autoregression and the other for the moving average.

Neural Networks. Similar to autoregression, the neural models rely on the previous lags to predict the future; however these models incorporate nonlinearity which make them more suitable to model complex time series. We consider two architectures with sigmoid non-linearity:

1) The feed forward neural network (*FFNN*).

2) and the time lagged neural network (*TLNN*) (Faraway and Chatfield, 1998).

For models that require stationarity (*e.g.* AR), we consider the differencing order as a hyperparamtere for each style. All hyperparameters (α for ours, number of lags for the autoregression, and hidden neurons for neural networks) are estimated over the

Model	Dresses		Tops & Tees		Shirts	
	MAE	MAPE	MAE	MAPE	MAE	MAPE
Naïve						
mean	0.0345	25.50	0.0513	17.61	0.0155	6.14
last	0.0192	8.38	0.0237	8.66	0.0160	5.50
drift	0.0201	9.17	0.0158	5.70	0.0177	6.50
Autoregression						
AR	0.0174	9.65	0.0148	**5.20**	0.0120	4.45
AR+S	0.0210	12.78	0.0177	6.41	0.0122	4.51
VAR	0.0290	20.36	0.0422	14.61	0.0150	5.92
ARIMA	0.0186	13.04	0.0154	5.45	0.0092	3.41
Neural Network						
TLNN	0.0833	35.45	0.0247	8.49	0.0124	4.24
FFNN	0.0973	41.18	0.0294	10.26	0.0109	3.97
Ours	**0.0146**	**6.54**	**0.0145**	5.36	**0.0088**	**3.16**

Table 7.2: The forecast error of our approach compared to several baselines on three data sets.

validation split of the data set. We compare the models based on two metrics: The mean absolute error

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}|e_t|, \tag{7.7}$$

and the mean absolute percentage error

$$\text{MAPE} = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{e_t}{y_t}\right| \times 100. \tag{7.8}$$

Where $e_t = \hat{y}_t - y_t$ is the error in predicting y_t with $\hat{y}_t$.

Forecasting results. Table 7.2 shows the forecasting performance of all models on the test data. Here, all models use the identical visual style representation, namely our attribute-based NMF approach. Our exponential smoothing model outperforms all baselines across the three data sets. Interestingly, the more involved models like ARIMA, and the neural networks do not perform better. This may be due to their larger number of parameters and the relatively short style trajectories. Additionally, no strong correlations among the styles were detected and VAR showed inferior performance. We expect there would be higher influence between styles from different garment categories rather than between styles within a category. Furthermore, modeling seasonality (AR+S) does not improve the performance of the linear autoregression model. We notice that the Dresses

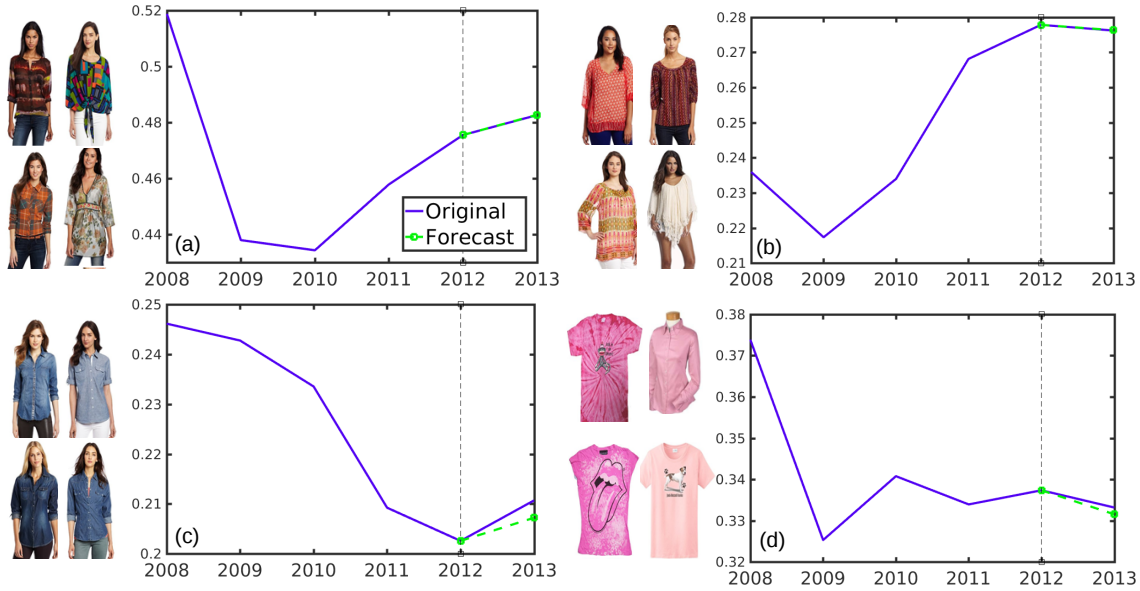


Figure 7.5: The forecasted popularity estimated by our model for 4 styles from the Tops & Tees data set. Our model successfully predicts the popularity of styles in the future and performs well even with challenging trajectories that experience a sudden change in direction like in (c) and (d).

data set is more challenging than the other two. The styles there exhibit more temporal variations compared to the ones in Tops&Tees and Shirts, which may explain the larger forecast error in general. Nonetheless, our model generates a reliable forecast of the popularity of the styles for a year ahead across all data sets. The forecasted style trajectory by our approach is within a close range to the actual one (only 3 to 6 percentage error based on MAPE).

Figure 7.5 visualizes our model's predictions on four styles from the Tops&Tees data set. For trajectories in Figure 7.5a and Figure 7.5b, our approach successfully captures the popularity of styles in year 2013. Styles in Figure 7.5c and Figure 7.5d are much more challenging. Both of them experience a reflection point at year 2012, from a declining popularity to an increase and vice versa. Still, the predictions made by our model forecast this change in direction correctly and the error in the estimated popularity is minor. Moreover, Figure 7.6 shows the style popularity forecasts estimated by baselines from the three forecasting groups in comparison to our approach. The Naive and NN based forecast models seem to produce larger prediction errors. Our model performs the best followed by the Autoregressor (AR).

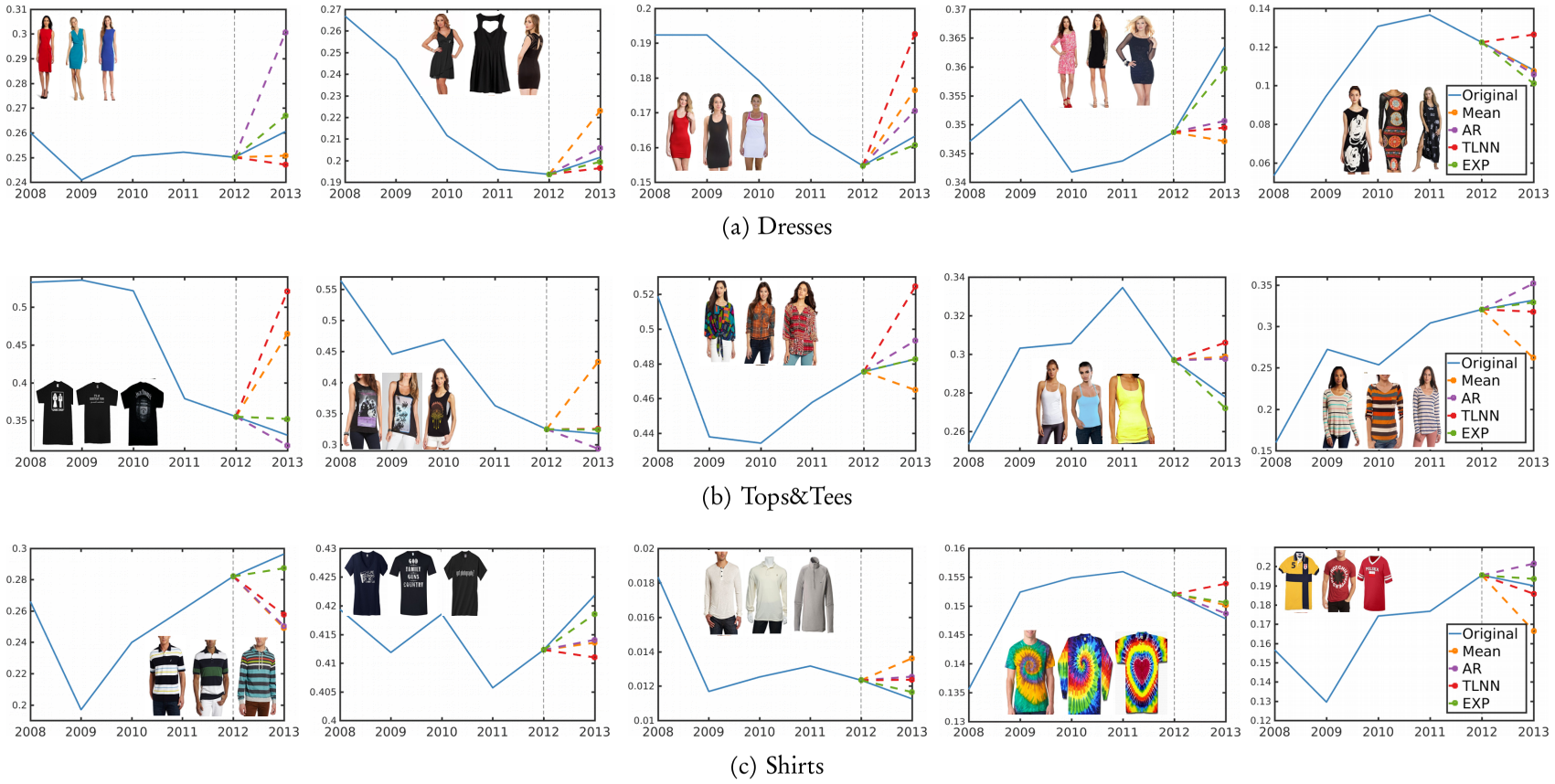


Figure 7.6: The forecasted popularity of the visual styles in (a) Dresses, (b) Tops&Tees and (c) Shirts. Our model (EXP) successfully captures the popularity of the styles in year 2013 with minor errors in comparison to the baselines.

#Styles	Dresses	Tops & Tees	Shirts
15	7.70	6.71	**3.03**
30	**6.54**	**5.36**	3.16
45	8.15	5.98	3.78
70	8.22	5.60	4.10
85	10.66	5.62	4.14

Table 7.3: The forecast error (MAPE) of our approach using varying number of styles.

Number of Styles. Table 7.3 shows the performance of our model in terms of forecasting error when varying the number of styles K between 15 and 85. We notice that increasing K results in introducing more noise in the timeline of the style as some of them does not capture a consistent style in the data and the forecasting error increases. Nonetheless, the variance in performance is still acceptable for the tested K values. Furthermore, we notice that the relative performance of the forecast approaches when varying K is similar to Table 7.2, with EXP performing the best.

From the visual perspective, we see that at $K = 30$ the styles have a coherent visual appearance of mid-level granularity. However, capturing the visual quality of the discovered styles in a quantitative manner is not a trivial task. We believe this is an interesting and important question for future investigation.

7.6.3 Fashion representation

Thus far we have shown the styles discovered by our approach as well as our ability to forecast the popularity of visual styles in the future. Next, we examine the impact of our representation compared to both textual meta-data and CNN-based alternatives.

Meta Information. Fashion items are often accompanied by information other than the images. We consider two types of meta information supplied with the Amazon data sets (Figure 7.3):

1) *Tags*: which identify the categories, the age range, the trademark, the event, *etc*.

2) *Text*: which provides a description of the item in natural language.

For both, we learn a unique vocabulary of tags and words across the data set and represent each item using a bag of words representation. From thereafter, we can employ our NMF and forecasting models just as we do with our visual attribute-based vocabulary.

Consequently, we consider a text-only baseline as well as a multi-modal approach that augments our attribute model with textual cues.

Visual. Attributes are attractive in this problem setting for their interpretability, but how fully do they capture the visual content? To analyze this, we implement an alternative representation based on deep features extracted from a pre-trained CNN. In particular, we train a CNN with an AlexNet-like architecture on the DeepFashion data set to perform *clothing classification*. The ClothingNet model is similar to our attribute model architecture with the last sigmoid layer replaced with a softmax. The network is trained to distinguish 50 categories of garments (*e.g.* *Sweater*, *Skirt*, *Jeans* and *Jacket*) from the DeepFashion data set. On a held-out test set on DeepFashion, the ClothingNet achieves 86.5% Top-5 accuracy.

Since fashion elements can be local properties (*e.g.*, v-neck) or global (*e.g.*, a-line), we use the CNN to extract two representations at different abstraction levels: 1) *FC7*: features extracted from the last hidden layer; 2) *M3*: features extracted from the third max pooling layer after the last convolutional layer. We refer to these as ClothingNet-FC7 and ClothingNet-M3 in the following.

Forecasting results. The textual and visual cues inherently rely on distinct vocabularies, and the metrics applied for Table 7.2 are not comparable across representations. Nonetheless, we can gauge their relative success in forecasting by measuring the distribution difference between their predictions and the ground truth styles, in their respective feature spaces. In particular, we apply the experimental setup of Section 7.6.2, then record the Kullback-Leibler divergences (KL) between the forecasted distribution and the actual test set distribution:

$$\mathrm{KL}(p||q) = \sum_{s_i} p(s_i) \log \frac{p(s_i)}{q(s_i)}. \tag{7.9}$$

For all models, we apply our best performing forecaster from Table 7.2 (EXP).

Table 7.4 shows the effect of each representation on forecasting across all three data sets. Among all single modality methods, our approach is the best. Compared to the ClothingNet CNN baselines, our attribute styles are much more reliable. Upon visual inspection of the learned styles from the CNNs, we find out that they are sensitive to the pose and spatial configuration of the item and the person in the image. This reduces the quality of the discovered styles and introduces more noise in their trajectories. Compared

Model	Dresses		Tops & Tees		Shirts	
	KL	IMP(%)	KL	IMP(%)	KL	IMP(%)
Meta Information						
Tags	0.0261	0	0.0161	0	0.0093	0
Text	0.0185	29.1	0.0075	53.4	0.0055	40.9
Visual						
ClothingNet-FC7	0.0752	-188.1	0.25	-1452.8	0.1077	-1058.1
ClothingNet-M3	0.0625	-139.5	0.0518	-221.7	0.0177	-90.3
Attributes	**0.0105**	**59.8**	**0.0049**	**69.6**	**0.0035**	**62.4**
Multi-Modal						
Attributes + Tags	0.0336	-28.7	0.0099	38.5	0.0068	26.9
Attributes + Text	0.0051	80.5	0.0053	67.1	**0.0014**	**84.9**
Attr + Tags + Text	**0.0041**	**84.3**	**0.0052**	**67.7**	**0.0014**	**84.9**

Table 7.4: Forecast performance for various fashion representations in terms of KL divergence (lower is better) and the relative improvement (IMP) over the Tags baseline (higher is better). Our attribute-based visual styles lead to much more reliable forecasts compared to meta data or other visual representations.

to the tags alone, the textual description is better, likely because it captures more details about the appearance of the item. However, compared to any baseline based only on meta data, our approach is the best. This is an important finding: *predicted* visual attributes yield more reliable fashion forecasting than strong real-world meta-data cues. To see the future of fashion, it pays off to really look at the images themselves.

The bottom of Table 7.4 shows the results when using various combinations of text and tags along with attributes. We see that our model is even stronger, arguing for including meta-data with visual data whenever it is available.

7.6.4 Style dynamics

Having established the ability to forecast visual fashions, we now turn to demonstrating some suggestive applications. Fashion is a very active domain with styles and designs going in and out of popularity at varying speeds and stages. The life cycle of fashion goes through four main stages (Sproles, 1981): 1) introduction; 2) growth; 3) maturity; and finally 4) decline. Knowing which style is at which level of its lifespan is of extreme importance for the fashion industry. Understanding the style dynamics helps companies to adapt their strategies and respond in time to accommodate the customers' needs. Our

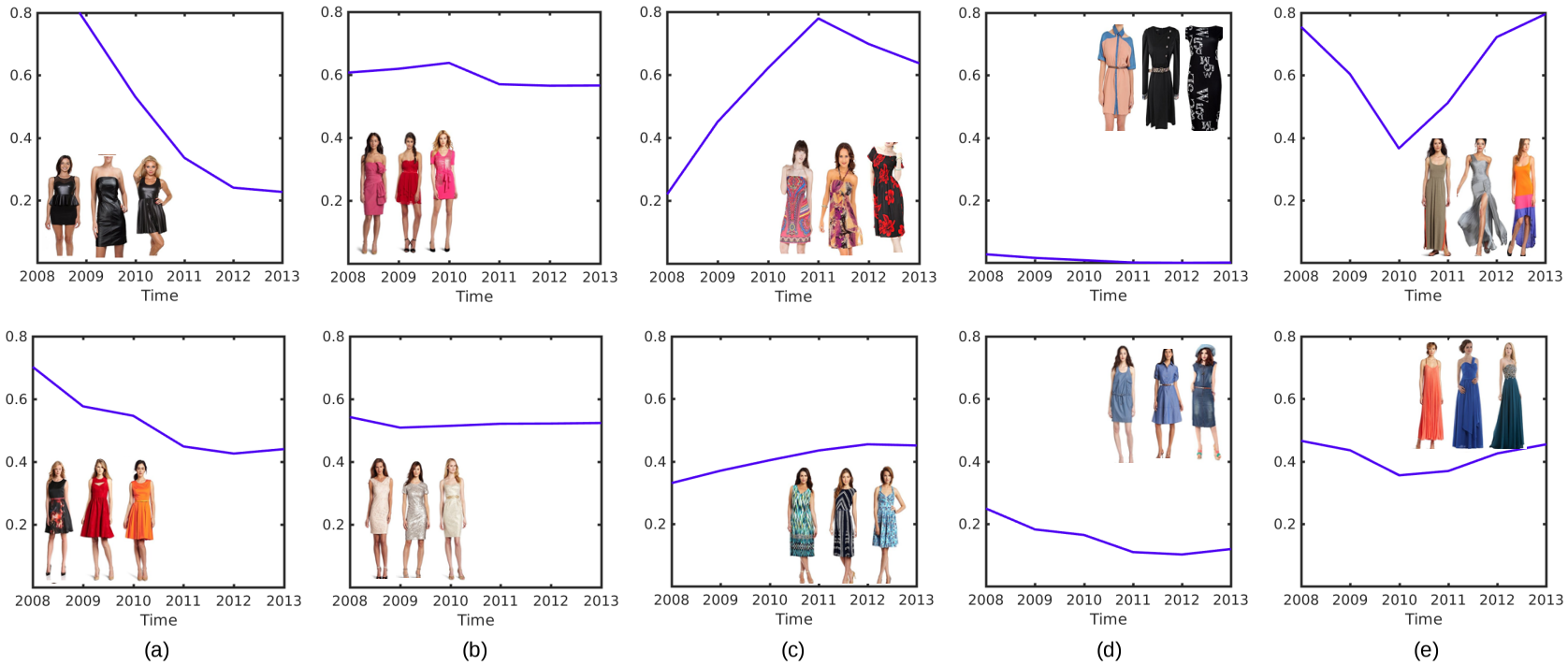


Figure 7.7: Our approach offers the unique opportunity to examine the life cycle of visual styles in fashion. Some interesting temporal dynamics of the styles discovered by our model can be grouped into: (a) out of fashion; (b) classic; (c) in fashion or trending; (d) unpopular; and (e) re-emerging styles.

model offers the opportunity to inspect visual style trends and lifespans. In Figure 7.7, we visualize the temporal trajectories computed by our model for 6 styles from Dresses. The trends reveal several categories of styles:

- *Out of fashion*: styles that are losing popularity at a rapid rate (Figure 7.7a).
- *Classic*: styles that are relatively popular and show little variations through the years (Figure 7.7b).
- *Trending*: styles that are trending and gaining popularity at a high rate (Figure 7.7c).
- *Unpopular*: styles that are currently at a low popularity rate with no sign of improvement (Figure 7.7d).
- *Re-emerging*: styles that were popular in the past, declined, and then resurface again and start trending (Figure 7.7e).

Our model is in a unique position to offer this viewpoint on fashion. For example, using item popularity and trajectories is not informative about the life cycle of the visual style. An item lifespan is influenced by many other factors such as pricing, marketing strategy, and advertising among many others. By learning the latent visual styles in fashion, our model is able to capture the collective styles shared by many articles and, hence, depicts a more realistic popularity trajectory that is less influenced by irregularities experienced by the individual items.

7.6.5 Forecasting elements of fashion

While so far we focused on visual style forecasting, our model is capable of inferring the popularity of the individual attributes as well. Thus, it can answer questions like: what kind of fabric, texture, or color will be popular next year? These questions are of significant interest in the fashion industry (e.g., see the “fashion oracle" World Global Style Network (DeFabio, 2017; WGSN Limited), which thousands of designers rely on for trend prediction on silhouettes, palettes, etc.).

We get the attribute popularity $p(a_m|t)$ at a certain time t in the future through the forecasted popularity of the styles:

$$p(a_m|t) = \sum_{s_k \in S} p(a_m|s_k)p(s_k|t) \tag{7.10}$$

where $p(a_m|s_k)$ is the probability of attribute a_m given style s_k based on our style discovery model, and $p(s_k|t)$ is the forecated probability of style s_k at time t.

(a) Texture (b) Shape

Figure 7.8: Our model can predict the popularity of individual fashion attributes using the forecasted styles as a proxy. The forecasted attributes are shown in color while the ground truth is in black. The attribute size is relative to its popularity rank.

For the 1000 attributes in our visual vocabulary, our model achieves an intersection with ground truth popularity rank at 90%, 84% and 88% for the Top 10, 25 and 50 attributes respectively. Figure 7.8 shows the forecasted *texture* and *shape* attributes for the Dresses test set. Our model successfully captures the most dominant attributes in both groups of attributes, correctly giving the gist of future styles.

7.7 Summary

In this chapter, we propose a novel application of semantic attributes in fashion analysis. We propose a model that discovers fine-grained visual styles from large-scale fashion data in an unsupervised manner. Our model identifies unique style signatures and provides a semantic description for each based on key visual attributes. Furthermore, based on user consumption behavior, our model predicts the future popularity of the styles, and reveals their life cycle and status (*e.g.* in- or out of fashion). We show that vision is essential for reliable forecasts, outperforming textual representations. Finally, fashion is not restricted to apparel; it is present in accessories, automobiles, and even house furniture. Our model is generic enough to be employed in different domains where a notion of visual style is present.

Chapter 8

Conclusion

In this thesis, we explored how visual semantic description can be effectively employed in transfer learning as an intermediate representation in order to share knowledge between source and target domains. In the following, we summarize the key contributions of this thesis then we discuss possible directions for future work.

Semantic representations for transfer learning.
Peer-reviewed publications: Al-Halah et al. (2014b, 2016a)
We started in Chapter 3 with a thorough analysis of the interplay of knowledge representations and the various transfer schemes (*i.e.* representation-, instance- and parameter-transfer). We proposed a hierarchical semantic representation based on attributes along with a transfer framework for action similarity learning. In our experiments, we observed a positive correlation between the abstraction level of knowledge encoded in the representation with its performance and robustness against negative transfer effect. Moreover, decorrelating the semantic space before transfer proved to be crucial for improved performance. Interestingly, we find out that transferring from a relatively simple source domain to a more complex and diverse target domain is still beneficial when labeled data in the target is extremely sparse and when using proper knowledge representation.

Hierarchical transfer of semantic attributes.
Peer-reviewed publication: Al-Halah and Stiefelhagen (2015b)
In Chapter 4, we build on our previous findings and introduce a novel model to incorporate structured knowledge to guide and improve transfer learning. Here, we leverage the structure knowledge in the category space to learn semantic attributes at different levels of abstraction. This effectively enriches our source domain with additional attribute

models with various specificity along the categories spectrum. Additionally, our model employs the inter-class hierarchical relations to guide the whole transfer process by selecting and constructing the most relevant attribute models for an unseen class.

Unsupervised attribute-based ZSL. While attribute-based transfer proved to be very successful in various transfer learning problems, like zero-shot learning for example, they usually required a high level of user supervision. In this work, we systematically reduced the required supervision needed in learning and transferring semantic attributes by proposing novel models for association predictions and vocabulary discovery from text.

Association prediction via semantic relations.
Peer-reviewed publication: Al-Halah et al. (2016b)
In Chapter 5, we circumvent the need for manually defined class-attribute associations by modeling semantic relations of classes and attributes. We propose a bilinear approach that models these relations in the word embedding space. Given a novel class name, our model can accurately predict the class associations with all attributes, hence effectively providing an unsupervised zero-shot learning framework. Moreover, we provide an efficient scheme to select "when to transfer" to the unseen class based on our confidence in the predicted associations which improves the performance of our model even further. The ability to predict association automatically enables our model to easily transfer semantic attributes across data sets without any user intervention or additional labeling.

Automatic discovery of attribute vocabulary.
Peer-reviewed publication: Al-Halah and Stiefelhagen (2017)
In Chapter 6, we go a step further and propose to automatically discover semantic attribute vocabulary from online articles at a large scale. We introduce a novel model that can analyze free form encyclopedia articles describing object categories to mine salient, discriminative and diverse set of semantic attributes. This discovered semantic vocabulary is then associated with categories using a deep model in a joint optimization framework. We show that the vocabulary mined by our model correlated well with human understanding of semantic attributes and that the proposed model can account for noise and missing information in text when optimizing associations. In a large-scale evaluation, our deep attribute model demonstrated high performance and generalized well across data sets.

Forecasting the future of fashion.
Peer-reviewed publication: Al-Halah et al. (2017)
Finally, in Chapter 7 we introduce a novel application for semantic attributes in the domain of fashion analysis. We present the first work to discover fine-grained visual fashion styles and forecast their popularity in the future. Our proposed model leverages semantic attributes to learn fashion styles in unsupervised way and at a large scale. Then, these styles are correlated with customer preferences captured from online shopping services. This enables us to learn the popularity trajectories for each style through time and predict how this popularity will change in the near future. Additionally, the proposed model is capable of revealing the style life cycles and status through time which provides us with a unique opportunity for an in depth analysis of fashion styles. Moreover, we demonstrate that attribute-based representation for fashion helps in producing interpretable styles and leads to the most reliable forecasts, outperforming other forms of visual and textual representations.

8.1 Discussion and open directions

Transfer learning is a key factor in creating models with the ability to generalize well to different tasks and domains. We addressed through this thesis some of the key challenges and aspects of transfer learning in visual recognition, especially when the target training data is scarce or not available. Additionally, we have discussed the properties and some of the limitations of the proposed approaches at the end of the relevant chapter. Next, we discuss a common aspect of the work presented in this thesis. The transfer learning aspects tackled in this work represent a static view of the learning domain. That is, there is a source and target domain and the problem is how to extract relevant knowledge from the source and transfer to the target. However, learning is a continuous and accumulative process. We, humans, go through a lifelong endeavor of learning and adapting to our environment. Not only we can transfer our experience to new tasks but also incorporate the feedback we acquire from learning the new task to expand and update our knowledge. In machine learning, this ability is usually referred to as *lifelong learning* or *learning to learn* (Thrun and Mitchell, 1995). In lifelong learning, the tasks are presented sequentially to the model and the aim is to learn new capabilities while at the same time maintain performance on the old ones. Any practical application in visual recognition requires such an ability since it is infeasible to keep accumulating data and retraining models from scratch whenever a new task is encountered. This direction represents a natural

extension to our work on transfer learning in this thesis. The number of visual categories learned by modern machine learning models is in the range of a 1000 (Russakovsky et al., 2015). However, there are tens of thousands of visual categories (Biederman, 1987) with new visual and semantic concepts appearing every day. Hence, the model should be able to know *what*, *how* and *when to transfer* along with *how to expand its knowledge* in lifelong learning.

Own Publications

Ziad Al-Halah, Rainer Stiefelhagen, and Kristen Grauman. Fashion Forward: Forecasting Visual Style in Fashion. In International Conference on Computer Vision (ICCV), 2017.

Ziad Al-Halah and Rainer Stiefelhagen. Automatic Discovery, Association Estimation and Learning of Semantic Attributes for a Thousand Categories. In Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

Ziad Al-Halah, Makarand Tapaswi, and Rainer Stiefelhagen. Recovering the Missing Link: Predicting Class-Attribute Associations for Unsupervised Zero-Shot Learning. In Conference on Computer Vision and Pattern Recognition (CVPR), 2016.

Monica-Laura Haurilet, Makarand Tapaswi, Ziad Al-Halah, and Rainer Stiefelhagen. Naming TV Characters by Watching and Analyzing Dialogs. In Winter Conference on Applications of Computer Vision (WACV), 2016.

Ziad Al-Halah, Lukas Rybok, and Rainer Stiefelhagen. Transfer Metric Learning for Action Similarity using High-Level Semantics. Pattern Recognition Letters, 2016.

Esam Ghaleb, Makarand Tapaswi, Ziad Al-Halah, Hazim Kemal Ekenel, and Rainer Stiefelhagen. Accio: A Data Set for Face Track Retrieval in Movies Across Age. In International Conference on Multimedia Retrieval (ICMR), 2015.

Tobias Gehrig*, Ziad Al-Halah*, Hazim Kemal Ekenel, and Rainer Stiefelhagen. Action Unit Intensity Estimation using Hierarchical Partial Least Squares. In International Conference on Automatic Face and Gesture Recognition (FG), 2015. (* = equal contribution)

Ziad Al-Halah and Rainer Stiefelhagen. How to Transfer? Zero-Shot Object Recognition via Hierarchical Transfer of Semantic Attributes. In Winter Conference on Applications of Computer Vision (WACV), 2015.

Ziad Al-Halah, Lukas Rybok, and Rainer Stiefelhagen. What to Transfer? High-Level Semantics in Transfer Metric Learning for Action Similarity. In International Conference on Pattern Recognition (ICPR), 2014. (Best Student Paper Award)

Lukas Rybok, Boris Schauerte, Ziad Al-Halah, and Rainer Stiefelhagen. Important Stuff, Everywhere! Activity Recognition with Salient Proto-Objects as Context. In Winter Conference on Applications of Computer Vision (WACV), 2014.

Ziad Al-Halah, Tobias Gehrig, and Rainer Stiefelhagen. Learning Semantic Attributes via a Common Latent Space. In Proceedings of the International Conference on Computer Vision Theory and Applications, 2014.

Bibliography

Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Label-Embedding for Attribute-Based Classification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 24, 59, 60

Z. Akata, S. Reed, D. Walter, H. Lee, and B. Schiele. Evaluation of Output Embeddings for Fine-Grained Image Classification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 23, 24, 59, 60, 83, 84, 105

Z. Al-Halah and R. Stiefelhagen. How to Transfer? Zero-Shot Object Recognition via Hierarchical Transfer of Semantic Attributes. In *Winter Conference on Applications of Computer Vision (WACV)*, 2015b. 51, 84, 97, 105, 129

Z. Al-Halah and R. Stiefelhagen. Automatic Discovery, Association Estimation and Learning of Semantic Attributes for a Thousand Categories. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 88, 130

Z. Al-Halah, T. Gehrig, and R. Stiefelhagen. Learning Semantic Attributes via a Common Latent Space. In *Proceedings of the International Conference on Computer Vision Theory and Applications*, 2014a. 23, 33

Z. Al-Halah, L. Rybok, and R. Stiefelhagen. What to Transfer? High-Level Semantics in Transfer Metric Learning for Action Similarity. In *International Conference on Pattern Recognition (ICPR)*, 2014b. 24, 28, 129

Z. Al-Halah, L. Rybok, and R. Stiefelhagen. Transfer metric learning for action similarity using high-level semantics. *Pattern Recognition Letters*, 2016a. 28, 129

Z. Al-Halah, M. Tapaswi, and R. Stiefelhagen. Recovering the Missing Link: Predicting Class-Attribute Associations for Unsupervised Zero-Shot Learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016b. 66, 105, 130

Z. Al-Halah, R. Stiefelhagen, and K. Grauman. Fashion Forward: Forecasting Visual Style in Fashion. In *International Conference on Computer Vision (ICCV)*, 2017. 110, 131

L. AlSumait, D. Barbará, J. Gentle, and C. Domeniconi. Topic significance ranking of LDA generative models. In *The European Conference on Machine Learning(ECML)*, 2009. 92

R. Arandjelovic and A. Zisserman. Three things everyone should know to improve object retrieval. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 36

A. Atamtürk and V. Narayanan. The submodular knapsack polytope. *Discrete Optimization*, 6:333–344, 2009. 93

Y. Aytar and A. Zisserman. Tabula rasa: Model transfer for object category detection. In *International Conference on Computer Vision (ICCV)*, 2011. 20

J. L. Ba, K. Swersky, S. Fidler, and R. Salakhutdinov. Predicting Deep Zero-Shot Convolutional Neural Networks using Textual Descriptions. In *International Conference on Computer Vision (ICCV)*, 2015. 25

E. Bart and S. Ullman. Single-example learning of novel classes using representation by similarity. In *British Machine Vision Conference (BMVC)*, 2005. 20, 23, 30, 73

H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding (CVIU)*, 110(3):346–359, 2008. 58

A. Bendale and T. E. Boult. Towards open set deep networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 22

T. L. Berg, A. C. Berg, and J. Shih. Automatic Attribute Discovery and Characterization from Noisy Web Data. In *European Conference on Computer Vision (ECCV)*, 2010. 15, 16

I. Biederman. Recognition-by-components: a theory of human image understanding. *Psychological review*, 94(2):115, 1987. 132

Bing. Bing Search API. https://datamarket.azure.com/dataset/bing/search, 2016. 72

D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. *Journal of Machine Learning Research (JMLR)*, 3, 2003. 91, 98

L. Bossard, M. Dantone, C. Leistner, C. Wengert, T. Quack, and L. Van Gool. Apparel Classification with Style. In *Asian Conference on Computer Vision (ACCV)*, 2012. 15, 16, 110

G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015. 119

C. Bracher, S. Heinz, and R. Vollgraf. Fashion DNA: Merging Content and Sales Data for Recommendation and Article Mapping. In *SIGKDD Conference on Knowledge Discovery and Data Mining (KDD) Fashion Workshop*, 2016. 17

R. G. Brown and R. F. Meyer. The fundamental theorem of exponential smoothing. *Operations Research*, 9(5):673–685, 1961. 115

M. Bucher, S. Herbin, and F. Jurie. Improving semantic embedding consistency by metric learning for zero-shot classiffication. In *European Conference on Computer Vision (ECCV)*, 2016. 23

D. Cai, C. Zhang, and X. He. Unsupervised Feature Selection for Multi-Cluster Data. In *SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2010. 99

B. Cao, S. J. Pan, Y. Zhang, D.-Y. Yeung, and Q. Yang. Adaptive transfer learning. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2010. 20

S. Changpinyo, W.-l. Chao, B. Gong, and F. Sha. Synthesized Classifiers for Zero-Shot Learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 105

S. Changpinyo, W.-L. Chao, and F. Sha. Predicting visual exemplars of unseen classes for zero-shot learning. In *International Conference on Computer Vision (ICCV)*, 2017. 23

W.-L. Chao, S. Changpinyo, B. Gong, and F. Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *European Conference on Computer Vision (ECCV)*, 2016. 22

K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman. Return of the Devil in the Details: Delving Deep into Convolutional Nets. In *British Machine Vision Conference (BMVC)*, 2014. 57

C.-Y. Chen and K. Grauman. Inferring Analogous Attributes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 12

H. Chen, A. Gallagher, and B. Girod. Describing Clothing by Semantic Attributes. In *European Conference on Computer Vision (ECCV)*, 2012. 15, 16, 110

K. Chen, K. Chen, P. Cong, W. H. Hsu, and J. Luo. Who are the Devils Wearing Prada in New York City? *arXiv*, 2015a. 16, 17

Q. Chen, J. Huang, R. Feris, L. M. Brown, J. Dong, and S. Yan. Deep Domain Adaptation for Describing People Based on Fine-Grained Clothing Attributes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015c. 15, 16

X. Chen, A. Shrivastava, and A. Gupta. NEIL: Extracting Visual Knowledge from Web Data. In *International Conference on Computer Vision (ICCV)*, 2013. 15

J. S. Chung, A. Senior, O. Vinyals, and A. Zisserman. Lip reading sentences in the wild. *arXiv preprint arXiv:1611.05358*, 2016. 1

W. Dai, Q. Yang, G.-R. Xue, and Y. Yu. Boosting for transfer learning. In *International Conference on Machine Learning (ICML)*, 2007. 19, 20, 46

W. Dai, O. Jin, G.-R. Xue, Q. Yang, and Y. Yu. Eigentransfer: a unified framework for transfer learning. In *International Conference on Machine Learning (ICML)*, 2009. 20

N. Dalal and B. Triggs. Histograms of Oriented Gradients for Human Detection. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005. 8, 20, 58

J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *International Conference on Machine Learning (ICML)*, 2007. 28, 41, 42, 44

C. R. DeFabio. Trend-Forecasting. http://fusion.net/story/305446/wgsn-trend-forecasting-sarah-owen/, 2017. Online; accessed 10 March 2017. 127

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 57

J. Deng, N. Ding, Y. Jia, A. Frome, K. Murphy, S. Bengio, Y. Li, H. Neven, and H. Adam. Large-Scale Object Classification using Label Relation Graphs. In *European Conference on Computer Vision (ECCV)*, 2014. 24, 59, 60

W. Di, C. Wah, A. Bhardwaj, R. Piramuthu, and N. Sundaresan. Style finder: Fine-grained clothing style detection and retrieval. In *Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2013. 16

S. K. Divvala, A. Farhadi, and C. Guestrin. Learning Everything about Anything: Webly-Supervised Visual Concept Learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 15

J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International Conference on Machine Learning (ICML)*, 2014. 60

M. Douze, A. Ramisa, and C. Schmid. Combining attributes and fisher vectors for efficient image retrieval. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 15

M. Elhoseiny, B. Saleh, and A. Elgammal. Write a Classifier: Zero-Shot Learning Using Purely Textual Descriptions. In *International Conference on Computer Vision (ICCV)*, 2013. 23, 25, 84, 105

V. Escorcia, J. C. Niebles, B. Ghanem, and U. Norte. On the relationship between visual attributes and convolutional networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 59

M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2008 (VOC2008) Results, 2008. 56

R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research (JMLR)*, 9:1871–1874, 2008. 57, 78

J. Faraway and C. Chatfield. Time series forecasting with neural networks: a comparative study using the airline data. *Applied Statistics*, pages 231–250, 1998. 119

A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing Objects by their Attributes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 9, 10, 11, 12, 13, 14, 20, 22, 23, 29, 56, 58, 71, 74, 105

A. Farhadi, I. Endres, and D. Hoiem. Attribute-Centric Recognition for Cross-category Generalization. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. 15

FashionUnited 2017. Fashion Statistics. https://fashionunited.com/global-fashion-industry-statistics, 2017. Online; accessed 10 March 2017. 16

L. Fei-Fei, R. Fergus, and P. Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 28(4):594–611, 2006. 20

P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008. 8

R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2003. 8

V. Ferrari and A. Zisserman. Learning Visual Attributes. In *Advances in Neural Information Processing Systems (NIPS)*, 2008. 9, 14

M. Fink. Object classification from a single example utilizing class relevance metrics. In *Advances in Neural Information Processing Systems (NIPS)*, 2005. 19

Flickr. Flickr API. https://www.flickr.com/services/api/flickr.photos.search.html, 2016. 72

A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. DeViSE: A Deep Visual-Semantic Embedding Model. In *Advances in Neural Information Processing Systems (NIPS)*, 2013. 25, 83, 84, 104, 105

Y. Fu, T. M. Hospedales, T. Xiang, Z. Fu, and S. Gong. Transductive multi-view embedding for zero-shot recognition and annotation. In *European Conference on Computer Vision (ECCV)*, 2014b. 22

Z. Fu, T. Xiang, E. Kodirov, and S. Gong. Zero-Shot Object Recognition by Semantic Manifold Distance. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 59, 60, 104

S. Fujishige. *Submodular functions and optimization*, volume 58. Elsevier Science, 2005. 93, 94

L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 20

T. Gehrig*, Z. Al-Halah*, H. K. Ekenel, and R. Stiefelhagen. Action Unit Intensity Estimation using Hierarchical Partial Least Squares. In *International Conference on Automatic Face and Gesture Recognition (FG)*, 2015.

E. Ghaleb, M. Tapaswi, Z. Al-Halah, H. K. Ekenel, and R. Stiefelhagen. Accio: A Data Set for Face Track Retrieval in Movies Across Age. In *International Conference on Multimedia Retrieval (ICMR)*, 2015.

M. Guillaumin and V. Ferrari. Large-scale Knowledge Transfer for Object Localization in ImageNet. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 24

M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? Metric Learning Approaches for Face Identification. In *International Conference on Computer Vision (ICCV)*, 2009. 28, 34, 38

H. Han, C. Otto, X. Liu, and A. K. Jain. Demographic estimation from face images: Human vs. machine performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 37(6):1148–1161, 2015. 1

M.-L. Haurilet, M. Tapaswi, Z. Al-Halah, and R. Stiefelhagen. Naming TV Characters by Watching and Analyzing Dialogs. In *Winter Conference on Applications of Computer Vision (WACV)*, 2016.

K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *International Conference on Computer Vision (ICCV)*, 2015. 1

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 108

R. He and J. McAuley. Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. In *International World Wide Web Conference (WWW)*, 2016. 17

L. A. Hendricks, S. Venugopalan, M. Rohrbach, R. Mooney, K. Saenko, and T. Darrell. Deep compositional captioning: Describing novel object categories without paired

training data. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 9

W.-L. Hsiao and K. Grauman. Learning the Latent "Look": Unsupervised Discovery of a Style-Coherent Embedding from Fashion Images. In *International Conference on Computer Vision (ICCV)*, 2017. 17

C. Hu, P. Rai, C. Chen, M. Harding, and L. Carin. Scalable Bayesian Non-Negative Tensor Factorization for Massive Count Data. In *The European Conference on Machine Learning & Principles and Practice of Knowledge Discovery in Database (ECML PKDD)*, 2015. 113

E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng. Improving word representations via global context and multiple word prototypes. In *Association of Computational Linguistics (ACL)*, 2012. 25

G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten. Densely connected convolutional networks. *arXiv preprint arXiv:1608.06993*, 2016. 107

J. Huang, R. Feris, Q. Chen, and S. Yan. Cross-Domain Image Retrieval With a Dual Attribute-Aware Ranking Network. In *International Conference on Computer Vision (ICCV)*, 2015. 15, 16, 111

T. Iwata, S. Watanabe, and H. Sawada. Fashion Coordinates Recommender System Using Photographs from Fashion Magazines. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2011. 15

D. Jayaraman and K. Grauman. Zero-shot recognition with unreliable attributes. In *Advances in Neural Information Processing Systems (NIPS)*, 2014. 23

H. Jegou and O. Chum. Negative evidences and co-occurrences in image retrieval: the benefit of PCA and whitening. In *European Conference on Computer Vision (ECCV)*, 2012. 33

R. Jenatton, A. Bordes, N. L. Roux, and G. Obozinski. A Latent Factor Model for Highly Multi-relational Data. In *Advances in Neural Information Processing Systems (NIPS)*, 2012. 70

Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv:1408.5093*, 2014. 57, 78

J. Jiang and C. Zhai. Instance weighting for domain adaptation in nlp. In *Association of Computational Linguistics (ACL)*, 2007. 20

L. Jie, T. Tommasi, and B. Caputo. Multiclass transfer learning from unconstrained priors. In *International Conference on Computer Vision (ICCV)*, 2011. 21

M. H. Kiapour, K. Yamaguchi, A. C. Berg, and T. L. Berg. Hipster wars: Discovering Elements of Fashion Styles. In *European Conference on Computer Vision (ECCV)*, 2014. 15, 16

M. H. Kiapour, X. Han, S. Lazebnik, A. C. Berg, and T. L. Berg. Where to Buy It: Matching Street Clothing Photos in Online Shops. In *International Conference on Computer Vision (ICCV)*, 2015. 15

D. P. Kingma and J. L. Ba. ADAM: A Method for Stochastic Optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 98, 112

O. Kliper-Gross, T. Hassner, and L. Wolf. The action similarity labeling challenge. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2012. 35, 36, 45, 47

E. Kodirov, T. Xiang, Z. Fu, and S. Gong. Unsupervised domain adaptation for zero-shot learning. In *International Conference on Computer Vision (ICCV)*, 2015. 22

T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *Society for Industrial and Applied Mathematics (SIAM) Review*, 51(3):455–500, 2009. 113

A. Kovashka, D. Parikh, and K. Grauman. WhittleSearch: Image search with relative attribute feedback. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 15, 16

A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2012. 57, 96, 106, 112

G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby talk: Understanding and generating simple image descriptions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 15

N. Kumar, P. Belhumeur, and S. Nayar. Facetracer: A search engine for large collections of images with faces. In *European Conference on Computer Vision (ECCV)*, 2008. 9

N. Kumar, A. C. Berg, P. N. Belhumeur, and S. Nayar. Describable Visual Attributes for Face Verification and Image Search. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2011. 15

I. Kuzborskij, F. Orabona, and B. Caputo. Scalable greedy algorithms for transfer learning. *Computer Vision and Image Understanding (CVIU)*, 156:174–185, 2017. 21

I. Kwak, A. Murillo, P. Belhumeur, D. Kriegman, and S. Belongie. From Bikers to Surfers: Visual Recognition of Urban Tribes. In *British Machine Vision Conference (BMVC)*, 2013. 16

A. Lam, A. K. Roy-Chowdhury, and C. R. Shelton. Interactive Event Search Through Transfer Learning. In *Asian Conference on Computer Vision (ACCV)*, 2010. 20, 27

C. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 9, 10, 11, 12, 13, 14, 15, 20, 23, 29, 56, 58, 60, 71, 72, 74, 97, 105

C. Lampert, H. Nickisch, and S. Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2013. 23, 59, 63, 80, 105

I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008. 20, 36, 45

H. Larochelle, D. Erhan, and Y. Bengio. Zero-data learning of new tasks. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2008. 21

J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. Cost-effective Outbreak Detection in Networks. In *SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2007. 94

T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International Journal of Computer Vision (IJCV)*, 43 (1):29–44, 2001. 8

T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755, 2014. 14

J. Liu, B. Kuipers, and S. Savarese. Recognizing Human Actions by Attributes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011a. 12, 35

J. Liu, M. Shah, B. Kuipers, and S. Savarese. Cross-View Action Recognition via View Knowledge Transfer. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011b. 20, 27

M.-y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa. Entropy-Rate Clustering: Cluster Analysis via Maximizing a Submodular Function Subject to a Matroid Constraint. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 36, 2014. 90, 93

S. Liu, Z. Song, G. Liu, C. Xu, H. Lu, and S. Yan. Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012b. 15

Z. Liu, S. Qiu, and X. Wang. DeepFashion : Powering Robust Clothes Recognition and Retrieval with Rich Annotations. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 15, 16, 110, 111, 112, 117

D. G. Lowe. Object recognition from local scale-invariant features. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 1999. 8, 58

M. Marszalek and C. Schmid. Semantic Hierarchies for Visual Object Recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007. 32

K. Martin, M. Hirzer, P. Wohlhart, P. M. Roth, and H. Bischof. Large Scale Metric Learning from Equivalence Constraints. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 44

J. McAuley, C. Targett, Q. Shi, and A. van den Hengel. Image-based Recommendations on Styles and Substitutes. In *ACM Special Interest Group on Information Retrieval (SIGIR)*, 2015. 115

T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Metric Learning for Large Scale Image Classification: Generalizing to New Classes at Near-Zero Cost. In *European Conference on Computer Vision (ECCV)*, 2012. 103, 104

T. Mensink, E. Gavves, and C. G. M. Snoek. COSTA: Co-Occurrence Statistics for Zero-Shot Classification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 13, 72, 76, 80

T. Mikolov, G. Corrado, K. Chen, and J. Dean. Efficient Estimation of Word Representations in Vector Space. In *International Conference on Learning Representations (ICLR)*, 2013. 25, 64, 67, 68, 74, 83, 105

G. A. Miller. WordNet: A Lexical Database for English. *Communications of the ACM Vol. 38, No. 11: 39-41.*, 1995. 14, 57, 84, 105

M. Minoux. Accelerated greedy algorithms for maximizing submodular set functions. In *Optimization Techniques*, pages 234–243. Springer, 1978. 94

I. Misra, A. Gupta, and M. Hebert. From red wine to red tomato: Composition with context. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 12

V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015. 1

A. C. Murillo, I. S. Kwak, L. Bourdev, D. Kriegman, and S. Belongie. Urban tribes: Analyzing group photos from a social perspective. In *Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2012. 16

F. Nater, T. Tommasi, H. Grabner, L. Van Gool, and B. Caputo. Transferring Activities: Updating Human Behavior Analysis. In *International Conference on Computer Vision (ICCV) Workshop on Visual Surveillance*, 2011. 20, 27

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14:265–294, 1978. 94

M. Nickel, V. Tresp, and H.-P. Kriegel. Factorizing yago: Scalable machine learning for linked data. In *International World Wide Web Conference (WWW)*, 2012. 69

J. C. Niebles, C.-W. Chen, and L. Fei-Fei. Modeling Temporal Structure of Decomposable Motion Segments for Activity Classification. In *European Conference on Computer Vision (ECCV)*, 2010. 35

M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado, and J. Dean. Zero-Shot Learning by Convex Combination of Semantic Embeddings. In *International Conference on Learning Representations (ICLR)*, 2014. 25, 83, 84, 104, 105

M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 20

A. J. O'Toole, P. J. Phillips, F. Jiang, J. Ayyad, N. Penard, and H. Abdi. Face recognition algorithms surpass humans matching faces over changes in illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 29(9), 2007. 1

M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In *Advances in Neural Information Processing Systems (NIPS)*, 2009. 23

S. J. Pan and Q. Yang. A Survey on Transfer Learning. *IEE Transactions on Knowledge and Data Engineering (TKDE)*, 22:1345–1359, 2010. 18, 19

S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011. 20, 27

D. Parikh and K. Grauman. Relative Attributes. In *International Conference on Computer Vision (ICCV)*, 2011. 12, 13, 15, 16

G. Patterson and J. Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 10, 14

G. Patterson and J. Hays. COCO Attributes: Attributes for People, Animals, and Objects. In *European Conference on Computer Vision (ECCV)*, 2016. 14

K. Patterson, P. J. Nestor, and T. T. Rogers. Where do you know what you know? the representation of semantic knowledge in the human brain. *Nature Reviews. Neuroscience*, 8(12):976, 2007. 8

H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2005. 99

J. Pennington, R. Socher, and C. D. Manning. GloVe : Global Vectors for Word Representation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014. 83, 105

F. Perronnin and C. Dance. Fisher kernels on visual vocabularies for image categorization. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007. 58

M. C. Potter, B. Wyble, C. E. Hagmann, and E. S. McCourt. Detecting meaning in rsvp at 13 ms per picture. *Attention, Perception, & Psychophysics*, 76(2):270–279, 2014. 1

L. Y. Pratt. Discriminability-based transfer between neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 1993. 3

R. Qiao, L. Liu, C. Shen, and A. van den Hengel. Less is more: zero-shot learning from online textual documents with noise suppression. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 25, 105

M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What Helps Where – And Why? Semantic Relatedness for Knowledge Transfer. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. 13, 14, 23, 72, 76, 80

M. Rohrbach, M. Stark, and B. Schiele. Evaluating Knowledge Transfer and Zero-Shot Learning in a Large-Scale Setting. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 14, 23, 24, 55, 60, 84, 97, 103, 104

M. Rohrbach, S. Ebert, and B. Schiele. Transfer learning in a transductive setting. In *Advances in Neural Information Processing Systems (NIPS)*, 2013a. 22, 104

M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele. Translating video content to natural language descriptions. In *International Conference on Computer Vision (ICCV)*, 2013b. 15

B. Romera-Paredes and P. H. Torr. An embarrassingly simple approach to zero-shot learning. In *International Conference on Machine Learning (ICML)*, 2015. 105

M. T. Rosenstein, Z. Marx, L. P. Kaelbling, and T. G. Dietterich. To transfer or not to transfer. In *Advances in Neural Information Processing Systems (NIPS) Workshop on Transfer Learning*, 2005. 21

O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3): 211–252, 2015. 97, 132

L. Rybok, B. Schauerte, Z. Al-Halah, and R. Stiefelhagen. Important stuff, everywhere! Activity recognition with salient proto-objects as context. In *Winter Conference on Applications of Computer Vision (WACV)*, 2014.

R. Salakhutdinov, J. Tenenbaum, and A. Torralba. Learning to Share Visual Appearance for Multiclass Object Detection. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. 24

B. Saleh, A. Farhadi, and A. Elgammal. Object-Centric Anomaly Detection by Attribute-Based Reasoning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 9, 15

G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 1988. 98

W. J. Scheirer, A. de Rezende Rocha, A. Sapkota, and T. E. Boult. Toward open set recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35 (7):1757–1772, 2013. 22

C. Schüldt, I. Laptev, and B. Caputo. Recognizing Human Actions: a local SVM Approach. In *International Conference on Pattern Recognition (ICPR)*, 2004. 35

B. Shahbaba and R. M. Neal. Improving Classification When a Class Hierarchy is Available Using a Hierarchy-Based Prior. *Bayesian Analysis*, 2:221–238, 2007. 24

Z. Shi, T. M. Hospedales, and T. Xiang. Transferring a semantic representation for person re-identification and search. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 15

B. Siddiquie, R. S. Feris, and L. S. Davis. Image ranking and retrieval based on multi-attribute queries. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. 99

E. Simo-Serra and H. Ishikawa. Fashion Style in 128 Floats : Joint Ranking and Classification using Weak Data for Feature Extraction. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 16, 17, 110

E. Simo-Serra, S. Fidler, F. Moreno-Noguer, and R. Urtasun. Neuroaesthetics in Fashion: Modeling the Perception of Fashionability. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 16, 17

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 108

J. Sivic and A. Zisserman. Efficient visual search of videos cast as text retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 31(4):591–606, 2009. 20

R. Socher, M. Ganjoo, C. D. Manning, and A. Y. Ng. Zero-Shot Learning Through Cross-Modal Transfer. In *Advances in Neural Information Processing Systems (NIPS)*, 2013. 22

Z. Song, M. Wang, X.-s. Hua, and S. Yan. Predicting Occupation via Human Clothing and Contexts. In *International Conference on Computer Vision (ICCV)*, 2011. 16

G. B. Sproles. Analyzing fashion life cycles: principles and perspectives. *The Journal of Marketing*, pages 116–124, 1981. 125

J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. Man vs. computer: Benchmarking machine learning algorithms for traffic sign recognition. *Neural Networks*, 32:323–332, 2012. 1

M. Stark, M. Goesele, and B. Schiele. A shape-based object class model for knowledge transfer. In *International Conference on Computer Vision (ICCV)*, 2009. 20

C. Sun, C. Gan, and R. Nevatia. Automatic concept discovery from parallel text and visual corpora. In *International Conference on Computer Vision (ICCV)*, 2015. 15

I. Sutskever, J. B. Tenenbaum, and R. R. Salakhutdinov. Modelling Relational Data using Bayesian Clustered Tensor Factorization. In *Advances in Neural Information Processing Systems (NIPS)*, 2009. 69

C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper with Convolutions. *arXiv:1409.4842v1*, 2014. 57, 78, 106, 107

S. Thorpe, D. Fize, C. Marlot, et al. Speed of processing in the human visual system. *Nature*, 381(6582):520–522, 1996. 1

S. Thrun and T. M. Mitchell. Lifelong robot learning. *Robotics and autonomous systems*, 15(1-2):25–46, 1995. 131

T. Tommasi, F. Orabona, and B. Caputo. Safety in numbers: Learning categories from few examples with multi model knowledge transfer. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. 21

T. Tommasi, F. Orabona, and B. Caputo. Learning categories from few examples with multi model knowledge transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 36(5):928–941, 2014. 20, 46

L. Torrey and J. Shavlik. Transfer Learning. *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, 1:242, 2009. 18, 19

K. Vaccaro, S. Shivakumar, Z. Ding, K. Karahalios, and R. Kumar. The Elements of Fashion Style. In *ACM Symposium on User Interface Software and Technology (UIST)*, 2016. 17

J. Van De Weijer, C. Schmid, J. Verbeek, and D. Larlus. Learning color names for real-world applications. *IEEE Transactions on Image Processing (TIP)*, 18(7):1512–1523, 2009. 9

L. van der Maaten and G. Hinton. Visualizing Data using t-SNE. *Journal of Machine Learning Research (JMLR)*, 9:2579–2605, 2008. 68

A. Veit, B. Kovacs, S. Bell, J. McAuley, K. Bala, and S. Belongie. Learning Visual Clothing Style with Heterogeneous Dyadic Co-occurrences. In *International Conference on Computer Vision (ICCV)*, 2015. 15, 16

S. Vittayakorn, K. Yamaguchi, A. C. Berg, and T. L. Berg. Runway to realway: Visual analysis of fashion. In *Winter Conference on Applications of Computer Vision (WACV)*, 2015. 17

S. Vittayakorn, T. Umeda, K. Murasaki, K. Sudo, T. Okatani, and K. Yamaguchi. Automatic Attribute Discovery with Neural Activations. In *European Conference on Computer Vision (ECCV)*, 2016a. 15, 17

S. Vittayakorn, T. Umeda, K. Murasaki, K. Sudo, T. Okatani, and K. Yamaguchi. Automatic Attribute Discovery with Neural Activations. In *European Conference on Computer Vision (ECCV)*, 2016b. 16

C. Wah and S. Belongie. Attribute-Based Detection of Unfamiliar Classes with Humans in the Loop. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 10

C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical report, California Institute of Technology, 2011. 13, 14, 56, 71

Y. Wang and G. Mori. A Discriminative Latent Model of Object Classes and Attributes. In *European Conference on Computer Vision (ECCV)*, 2010. 12

T. Weyand, I. Kostrikov, and J. Philbin. Planet - photo geolocation with convolutional neural networks. In *European Conference on Computer Vision (ECCV)*, 2016. 1

WGSN Limited. WGSN. https://www.wgsn.com/en/, 2017. Online; accessed 10 March 2017. 127

Wikipedia. Size of Wikipedia. https://en.wikipedia.org/wiki/Wikipedia:Size_of_Wikipedia, 2017. 88

S. Wold, K. Esbensen, and P. Geladi. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1-3):37–52, 1987. 58

R. S. Woodworth and E. L. Thorndike. The influence of improvement in one mental function upon the efficiency of other functions (I). *Psychological Review*, 8(3):247–261, 1901. 3

Q. Wu, C. Shen, P. Wang, A. Dick, and A. van den Hengel. Image captioning and visual question answering based on attributes and external knowledge. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2017. 15

Y. Xian, Z. Akata, G. Sharma, Q. Nguyen, M. Hein, and B. Schiele. Latent Embeddings for Zero-shot Classification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 23, 105

Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *arXiv preprint arXiv:1707.00600*, 2017a. 22

Y. Xian, B. Schiele, and Z. Akata. Zero-shot learning - the good, the bad and the ugly. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017b. 57

K. Yamaguchi, H. Kiapour, L. Ortiz, and T. Berg. Parsing clothing in fashion photographs. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. 15

K. Yamaguchi, H. Kiapour, and T. Berg. Paper doll parsing: Retrieving similar styles to parse clothing items. In *International Conference on Computer Vision (ICCV)*, 2013. 15

K. Yanai and K. Barnard. Image region entropy: A measure of visualness of web images associated with one concept. In *ACM Multimedia (MM)*, 2005. 9

Y. Yao and G. Doretto. Boosting for transfer learning with multiple sources. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010. 20, 46

A. Yu and K. Grauman. Just noticeable differences in visual attributes. In *International Conference on Computer Vision (ICCV)*, 2015. 15, 16

F. X. Yu, L. Cao, R. S. Feris, J. R. Smith, and S.-f. Chang. Designing Category-Level Attributes for Discriminative Visual Recognition. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 12, 13

B. Zadrozny. Learning and evaluating classifiers under sample selection bias. In *International Conference on Machine Learning (ICML)*, 2004. 20

M. D. Zeiler and R. Fergus. Visualizing and Understanding Convolutional Networks. In *European Conference on Computer Vision (ECCV)*, 2014. 9, 59

H. Zeng and Y.-m. Cheung. Feature Selection and Kernel Learning for Local Learning-Based Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2011. 99

Z.-J. Zha, T. Mei, M. Wang, Z. Wang, and X.-S. Hua. Robust Distance Metric Learning with Auxiliary Knowledge. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2009. 20, 28

H. Zhang, Z.-j. Zha, Y. Yang, S. Yan, Y. Gao, and T.-S. Chua. Attribute-augmented Semantic Hierarchy. In *ACM Multimedia (MM)*, 2013. 12

L. Zhang, T. Xiang, and S. Gong. Learning a deep embedding model for zero-shot learning. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 23

Y. Zhang and D.-Y. Yeung. Transfer metric learning by learning task relationships. In *SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2010. 20, 27, 28, 41, 42

Z. Zhang and V. Saligrama. Zero-shot learning via semantic similarity embedding. In *International Conference on Computer Vision (ICCV)*, 2015. 22

J. Zheng, Z. Jiang, R. Chellappa, and P. J. Phillips. Submodular Attribute Selection for Action Recognition in Video. In *Advances in Neural Information Processing Systems (NIPS)*, 2014a. 90

S. Zheng, M. Cheng, J. Warrell, and P. Sturgess. Dense semantic image segmentation with objects and attributes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014b. 15

A. Zweig and D. Weinshall. Exploiting Object Hierarchy: Combining Models from Different Category Levels. In *International Conference on Computer Vision (ICCV)*, 2007. 55