LANGUAGE GUIDED VISUAL PERCEPTION

BY MOHAMED ELHOSEINY

A dissertation submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Doctor of Philosophy

Graduate Program in Computer Science

Written under the direction of

Ahmed Elgammal

and approved by

New Brunswick, New Jersey

October, 2016

ABSTRACT OF THE DISSERTATION

Language Guided Visual Perception

by MOHAMED ELHOSEINY

Dissertation Director: Ahmed Elgammal

People typically learn through exposure to visual facts associated with linguistic descriptions. For instance, teaching visual concepts to children is often accompanied by descriptions in text or speech. In a machine learning context, these observations motivate the question of how this learning process could be computationally modeled to learn visual facts. We explored three settings where we showed that combining language and vision is useful for visual perception in both images and videos.

First, we addressed the question of how to utilize purely textual description of visual classes with no training images, to learn explicit visual classifiers for them. We propose and investigate two baseline formulations, based on regression and domain transfer that predict a classifier. Then, we propose a new constrained optimization formulation that combines a regression function and a knowledge transfer function with additional constraints to predict the classifier parameters for new classes. We also proposed kernelized models which allows defining any two kernel functions in the visual space and text space. We applied the studied models to predict visual classifiers for two fine-grained categorization datasets, and the results indicate successful predictions of our final model against several baselines that we designed.

Second, we modeled video event search as a language&vision problem where we proposed a zero-shot Event Detection method by Multi-modal Distributional Semantic embedding of videos. Our Zero-Shot event detection model is built on top of distributional semantics and

extends it in the following directions: (a) semantic embedding of multimodal information in videos (with focus on the visual modalities), (b) automatically determining relevance of concepts/attributes to a free text query, which could be useful for other applications, and (c) retrieving videos by free text event query (e.g., “changing a vehicle tire”) based on their content. We validated our method on the large TRECVID MED (Multimedia Event Detection) challenge. Using only the event title as a query, our method outperformed the state-of-the-art that uses big descriptions.

Third and motivated by the aforementioned results, we proposed a uniform and scalable setting to learn unbounded number of visual facts. We proposed models that can learn not only objects but also their actions, attributes and interactions with other objects in one unified learning framework and in a never ending way. The training data comes as structured facts in images, including (1) objects (e.g., <boy>), (2) attributes (e.g.,<boy, tall>), (3) actions (e.g., <boy, playing>, and (4) interactions (e.g., <boy, riding, a horse >). We have worked on the scale of 814,000 images and 202,000 unique visual facts. Our experiments show the advantage of relating facts by the structure in the proposed models compared to four designed baselines on bidirectional fact retrieval.

Acknowledgements

I would like to thank my advisor Ahmed Elgammal for the support throughout my Ph.D. struggles. I would like to thank the committee members who over-saw my dissertation. I would like to thank my family, friends and lab-mates for helping me through this stage of my life. It was an exceedingly valuable and joyous experience and it would not have been so if it was not for all of you.

I also so grateful to collaborate with Dr. Hui Cheng, th Dr. Jingen Liu, and Dr. Harpreet Sawhney at SRI International on our work at Multimedia Event Detection. I also can not thank enough Dr. Scott Cohen, Dr. Walter Chang, and Dr. Brian Price from Adobe Research for their tremendous support and collaboration from summer 2015 to summer 2016 in the exciting Sherlock Project.

Dedication

To my family and friends...nothing is impossible with their endless love, support and encouragement. A special feeling of gratitude to my wife, Mai Azab, my sons Adam and Yaseen Elhoseiny, my Mom and Dad.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Computer vision research has reached impressive milestones on standard visual recognition tasks like action classification [63, 64], objects detection [62], classification [83, 147], and segmentation [98]) and on a scale of millions of examples. Standard recognition tasks are rapidly reaching maturity and introduces a motivation for AI Research to work on directly developing methods that can really understand and reason about images where there is a big gap between machine and human intelligence.

The rate by which of visual and unstructured text data is produced has significantly increased and is rapidly accelerating. The goal of this thesis is to develop learning methods that can incorporate unstructured text data to aid visual perception of objects in images and events in videos, and to gain visual knowledge.

1.1 Motivation

Fig. 1.1 shows four images of four different bird categories with a question to determine which one of them is "Parakeet Auklet". It is not hard to see that answering this questions requires an expert knowledge about the appearance of the bird. However, if we got provided a text description shown on the top of Fig. 1.1, the answer will be much easier. The text description provides information that *"Parakeet Auklet" has an orange bill and it is dark above and white below*, which made the task far less challenging. This example shows that people can learn challenging fine-grained visual concepts from descriptions and so we aim to computationally model in this thesis by studying a variety of tasks that connect images and videos to a text description of the visual concept.

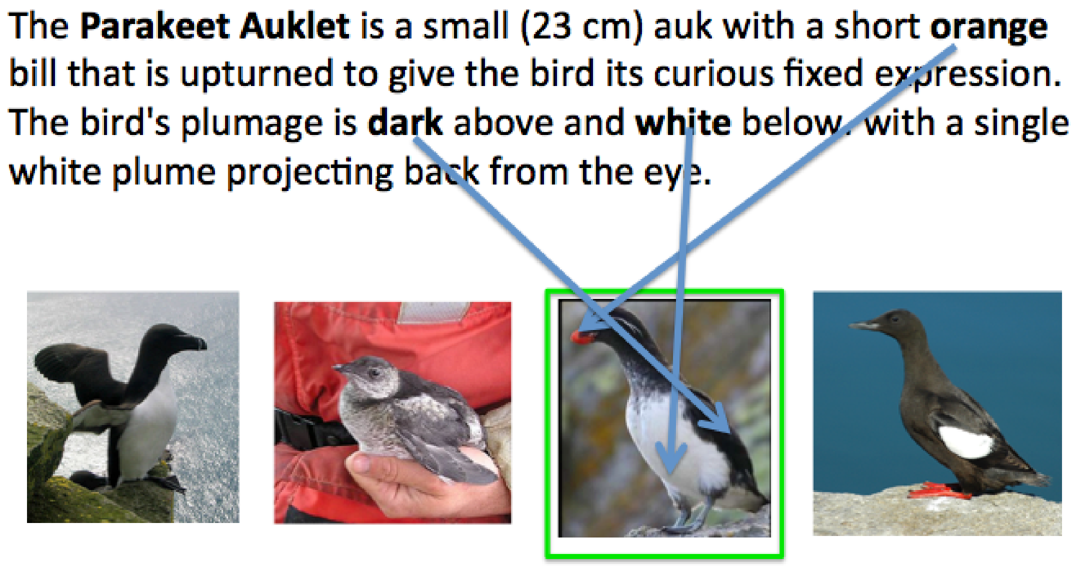


Figure 1.1: Which one of these Birds is Parakeet Auklet

1.2 Background and Related Work

We focus our related work discussion on three related lines of research: “zero/few-shot learning”, “visual knowledge transfer”, and “Language and Vision’

- zero/few-shot learning
- visual knowledge transfer.
- Language and Vision.

’.

Zero/Few-Shot Learning: Motivated by the practical need to learn visual classifiers of rare categories, researchers have explored approaches for learning from a single image (one-shot learning [109, 55, 59, 12]) or even from no images (zero-shot learning). One way of recognizing object instances from previously unseen test categories (the zero-shot learning problem) is by leveraging knowledge about common attributes and shared parts. Typically an intermediate semantic layer is introduced to enable sharing knowledge between classes and facilitate describing knowledge about novel unseen classes, *e.g.* [119]. For instance, given adequately labeled training data, one can learn classifiers for the attributes occurring in the training object categories. These classifiers can then be used to recognize the same attributes in object instances from the novel test categories. Recognition can then proceed on the basis of these learned attributes [89, 53]. Such attribute-based "knowledge transfer" approaches use an intermediate visual attribute representation to enable describing unseen object categories.

Typically attributes [89, 53] are manually defined by humans to describe shape, color, surface material, *e.g.* , furry, striped, *etc.* Therefore, an unseen category has to be specified in terms of the used vocabulary of attributes. Rohrbach *et al.* [135] investigated extracting useful attributes from large text corpora. In [121], an approach was introduced for interactively defining a vocabulary of attributes that are both human understandable and visually discriminative. Huang *et al.* [73] relaxed the attribute independence assumption by modeling correlation between attributes to achieve better zero shot performance, as opposed to prior models.

Similar to the setting of zero-shot learning, we use classes with training data (seen classes) to predict classifiers for classes with no training data (unseen classes). In contrast to attributes based method (e.g., [89, 53]), most of the work in this thesis, relies only on linguistic representation of the visual category/concept which is purely textual without additional human annotation beyond the category description.

Visual Knowledge Transfer: Our work can be seen in the context of knowledge sharing and inductive transfer. In general, knowledge transfer aims at enhancing recognition by exploiting shared knowledge between classes. Most existing research focused on knowledge sharing within the visual domain only, *e.g.* [69]; or exporting semantic knowledge at the level of category similarities and hierarchies, *e.g.* [58, 141]. We go beyond the state-of-the-art to explore cross-domain knowledge sharing and transfer. We explore how knowledge from the visual and textual domains can be used to learn across-domain correlation, which facilitates prediction of visual classifiers from textual description as we explored in Chapter 2 and 3 for objects, Chapter 4 for video events, and Chapter 5 for arbitrary facts in images (e.g., actions, interactions, objects, etc.).

Language and Vision: The relation between linguistic semantic representations and visual recognition has been explored. For example in [28], it was shown that there is a strong correlation between semantic similarity between classes, based on WordNet, and confusion between classes. Linguistic semantics in terms of nouns from WordNet [110] have been used in collecting large-scale image datasets such as ImageNet[29] and Tiny Images [153]. It was also shown that hierarchies based on WordNet are useful in learning visual classifiers, *e.g.* [141].

One of the earliest work on learning from images and text corpora is the work of Barnard *et al.* [11], which showed that learning a joint distribution of words and visual elements facilitates clustering the images in a semantic way, generating illustrative images from a caption, and generating annotations for novel images. There has been an increasing recent interest in the intersection between computer vision and natural language processing with researches that focus on generating textual description of images and videos, *e.g.* [54, 86, 166, 82]. This includes generating sentences about objects, actions, attributes, spatial relation between objects, contextual information in the images, scene information, *etc.* Based on the success of sequence to sequence training of neural nets in machine translation (e.g., [22]), impressive works has been recently proposed for image captioning (e.g., [79, 158, 163, 102]). In contrast, this thesis has a fundamentally different goal and hence focuses on different setting. In terms of the goal, we do not target generating textual description from images, instead we target predicting classifiers from text. In terms of the learning setting, the textual descriptions that we use is at the level of the category and do not come in the form of image-caption pairs, as in typical datasets used for text generation from images, *e.g.* [118].

There are several recent works that studies unannotated text with images. In [61, 150], word embedding language models (*e.g.* [107]) were adopted to represent class names as vectors, which require training using a big text-corpus. Their goal is to embed images into the language space then perform classification. In [44], a similar yet multimodal approach was adopted for Multimedia Event Detection in videos instead of object classification. There are several differences between these methods and the methods developed in this thesis. First, one limitation of the adopted language model is that it produces only one vector per word, which causes problems when a word has multiple meanings. Second, these methods assumes that each class is represented by one or few-words and hence can not represent a class text description that typically contains multiple paragraphs in our setting. Third, our goal is different which is to map the text description to an explicit classifier in the visual domain, *i.e.* the opposite direction of their goal as detailed in Chapter 2. Fourth, these models do not support non-linear classification, supported by the kernelized version proposed in our proposed method which is detailed in Chapter 3. Finally, we comprehensively study describing visual object in different settings. In fine-grained categories, word names are not sufficient which motivates predicting fine grained categories from text descriptions like Wikipedia articles. We study general video event in videos where it is convenient to search for event by the event name as we discuss in Chapter 4. Finally, we developed a language and vision system that is able to jointly learn objects, actions and interactions in one system in a scalable, uniform, and capable of relating facts by structure. The system is also bidirectional, supporting language $\rightarrow$ vision retrieval and also vision $\rightarrow$ language.

..

1.3 Objectives

The first goal of this dissertation is to study how to study the question of how to use purely textual description of object categories with no training images to learn visual classifiers for these categories. This goal implicitly address the problem of how to connect unstructured text descriptions which is a linguistic representation about the object to its images representing its visual representation. The second goal of this dissertation is to go beyond a single modality and study connecting text to a multimodal signal in videos which includes not only visual signal but also speech and text. The third objective by which we conclude our dissertation is how to build a model that can gain visual knowledge by learning facts including not only objects but also its actions and interactions with other objects. We study this setting while giving a careful consideration to generalization (learning from few examples), uniformity (understand attributes, action and interactions in one system), scalability (capability to learn unbounded number of visual facts/concepts).

1.4 Contributions

We mainly investigate how to guide visual perception by language.

In Part I, we propose a novel problem which is predicting visual classifiers of unseen object classes from just a text description for the visual object category. In earlier approach for this task is presented in Chapter 2, which predicts a linear classifier based on combining regression and domain transfer cost function. Chapter 3 presented an improved method which enables kernel classifier prediction such that any kernel function could be defined in both the visual and the text domain. The visual classifiers are predicted in the form defined by the generalized representer theorem [144] as detailed in the chapter.

In Part II (Chapter 4), we propose to guide event detection in videos by language. In particular, we propose a zero-shot Event Detection method by Multi-modal Language embedding of videos. Object, action concepts, as well as other available modalities from videos are embedded into language space. In this work, we extends extended such that videos could be retrieved by free text event query (e.g., ”birthday party”) based on their multimodal content. We embed videos into a distributional semantic space and then measure the similarity between videos and the event query in a free text form. We validated our method on the large TRECVID MED (Multimedia Event Detection) challenge. Using only the event title as a query, our method outperformed the state-of-the-art that uses big descriptions (10% and 1% absolute improvement improvement on ROC AUC (Area under the Curve) and the MAP (Mean Average Precision) metric).

In Part III, we propose a setting where objects, actions and interactions can be modeled simultaneously with a capacity to understand unbounded number of them in a structured way (e.g., objects (e.g., <boy>), (2) attributes (e.g., <boy, tall>), (3) actions (e.g., <boy, playing>), and (4) interactions (e.g., <boy, riding, a horse >)). In Chapter 5, we investigated recent and strong approaches from the multiview learning literature and also introduce two learning representation models. We applied the investigated methods on several datasets that we augmented with structured facts and a large scale dataset of more than 202,000 facts and 814,000 images. Our experiments show the advantage of relating facts by the structure by the proposed models compared to the designed baselines on bidirectional fact retrieval. In Chapter 6 and as a part of our data collection setting for this setting, we present an automatic method for data collection of structured visual facts from images with captions. With a language approach, the proposed method is able to collect hundreds of thousands of visual fact annotations with accuracy of 83% according to human judgment. Our method automatically collected more than 380,000 visual fact annotations and more than 110,000 unique visual facts from images with captions and localized them in images in less than one day of processing time on standard CPU platforms.

.

1.5 Acknowledgements and publications produced As a result the thesis Research

The research work in Chapter 2 was conducted in collaboration with Babak Saleh and Prof. Ahmed Elgammal and resulted in the following publication and presentations:

- [46]: M Elhoseiny, B Saleh, A Elgammal. "Write a classifier: Zero-shot learning using purely textual descriptions." Proceedings of the IEEE International Conference on Computer Vision, 2013.
- [45]: M Elhoseiny, B Saleh, A Elgammal "Heterogeneous Domain Adaptation: Learning Visual Classiers from Textual Description" Visual Domain Adaptation Workshop, ICCV, 2013

The research work in Chapter 3 was conducted in collaboration with Prof. Ahmed Elgammal and Babak Saleh and resulted in the following publication and presentations:

- [43]: M Elhoseiny, A Elgammal, B Saleh, "Write a Classifier: Predicting Visual Classifiers from Unstructured Text Descriptions", TPAMI submission, 2016
- [42]: M Elhoseiny, A Elgammal, B Saleh Tell and Predict: Kernel Classifier Prediction for Unseen Visual Classes from Unstructured Text Descriptions,CVPRW, 2015
- [41]: M Elhoseiny, A Elgammal, "Visual Classifier Prediction by Distributional Semantic Embedding of Text Descriptions", EMNLPW, 2015

The research in Chapter 4 was conducted in collaboration with Dr. Jingen Liu, Dr. Hui Cheng, Dr. Harpreet Sawhney from SRI International and Prof. Ahmed Elgammal. This research resulted in the following conference publication:

- [44]: M Elhoseiny, J Liu, H Cheng, H S Sawhney, and A Elgammal. "Zero-Shot Event Detection by Multimodal Distributional Semantic Embedding of Videos.", AAAI, 2016

The research in Chapter 5 was conducted in collaboration with Dr. Scott Cohen, Dr. Walter Cheng, Dr. Brian Price from Adobe Research and Prof. Ahmed Elgammal. This research resulted in the following article which is in submission:

- [37]: M Elhoseiny, S Cohen, W Chang, B Price, and A Elgammal. ”Sherlock: Scalable Fact Learning in Images”, Arxiv, 2016

The research in Chapter 6 was conducted in collaboration with Dr. Scott Cohen, Dr. Walter Cheng, Dr. Brian Price from Adobe Research and Prof. Ahmed Elgammal. This research resulted in the following publication:

- [36]: M Elhoseiny, S Cohen, W Chang, B Price, and A Elgammal. ”Sherlock: Scalable Fact Learning in Images”, Association of Computation Linguistics, long paper proceedings, Vision and Language Workshop, 2016

During my PhD at Rutgers, I had the privilege to collaborate with other awesome colleagues including Sheng Huang, Tarek El-Gaaly, Amr Bakry, Han Zhang, Sheng Huang, Tao Xu, Dr. Xiaolei Huang, Dr. Shaoting Zhang, and Prof.Dimitris Metaxas in related publications listed below

1. [38]: M Elhoseiny, T El-Gaaly, A Bakry, A Elgammal, ” A Comparative Analysis and Study of Multiview CNN Models for Joint Object Categorization and Pose Estimation” In Proceedings of The 33rd International Conference on Machine Learning, 2016

2. [9]: A Bakry, M Elhoseiny, T El-Gaaly, A Elgammal, ”Digging Deep into the layers of CNNs: In Search of How CNNs Achieve View Invariance”, ICLR, 2016

3. [39]: Mohamed Elhoseiny and Ahmed Elgammal. ”Generalized Twin Gaussian Processes using Sharma-Mittal Divergence,” Machine Learning Journal, 2015.

4. [8]: A Bakry, T El-Gaaly, M Elhoseiny, Ahmed Elgammal. ”Joint Object Recognition and Pose Estimation using a Nonlinear View-invariant Latent Generative Model,” Winter Conference on Applications of Computer Vision (WACV), 2016.

5. [170]: H Zhang, T Xu, M Elhoseiny, X Huang, S Zhang, A Elgammal, D Metaxas, ”SPDA-CNN: Unifying Semantic Part Detection and Abstraction for Fine-grained Recognition”, CVPR, 2016

6. Mohamed Elhoseiny and Ahmed Elgammal. ”Text to Multi-level MindMaps: A Novel Method for Hierarchical Visual Abstraction of Natural Language Text,” Journal of Multimedia Tools and Applications, 2015.

7. [40]: Mohamed Elhoseiny, Ahmed Elgammal. ”Overlapping Domain Cover for Scalable and Accurate Kernel Regression Machines,” The British Machine Vision Conference (BMVC), 2015

8. [74]: S Huang, M Elhoseiny, A Elgammal, D Yang, ”Learning hypergraph-regularized attribute predictors”, CVPR, 2015

Part I

Language Guided Object Recognition from Encyclopedia text

Chapter 2

Write a Classifier: Zero Shot Learning Using Purely Textual Descriptions

The main question we address in this paper is how to use purely textual description of categories with no training images to learn visual classifiers for these categories. We propose an approach for zero-shot learning of object categories where the description of unseen categories comes in the form of typical text such as an encyclopedia entry, without the need to explicitly defined attributes. We propose and investigate two baseline formulations, based on regression and domain adaptation. Then, we propose a new constrained optimization formulation that combines a regression function and a knowledge transfer function with additional constraints to predict the classifier parameters for new classes. We applied the proposed approach on two fine-grained categorization datasets, and the results indicate successful classifier prediction.

2.1 Introduction

One of the main challenges for scaling up object recognition systems is the lack of annotated images for real-world categories. Typically there are few images available for training classifiers for most of these categories. This is reflected in the number of images per category available for training in most object categorization datasets, which, as pointed out in [141], shows a Zipf distribution. The problem of lack of training images becomes even more severe when we target recognition problems within a general category, *i.e.* , fine-grained categorization, for example building classifiers for different bird species or flower types (there are estimated over 10000 living bird species, similar for flowers). Researchers try to exploit shared knowledge between categories to target such scalability issue. This motivated many researchers who looked into approaches that learn visual classifiers from few examples, *e.g.* [28, 55, 12]. This even

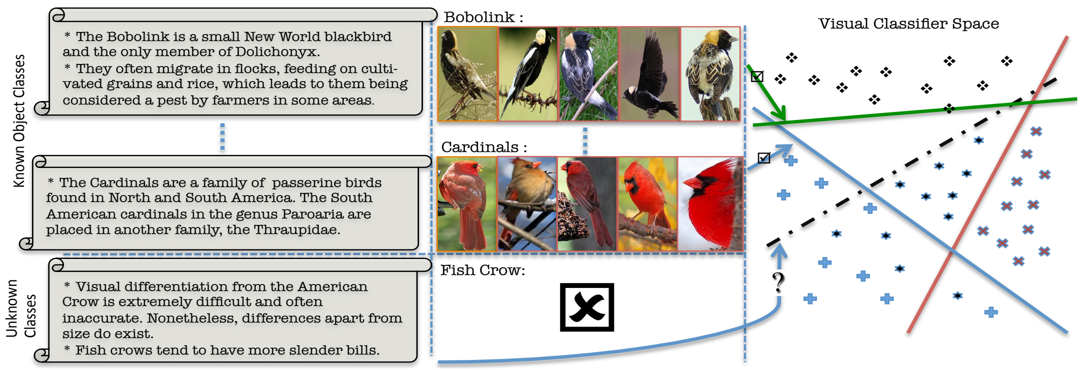


Figure 2.1: Problem Definition: Zero-shot learning with textual description. Left: synopsis of textual descriptions for bird classes. Middle: images for “seen classes”. Right: classifier hyperplanes in the feature space. The goal is to estimate a new classifier parameter given only a textual description

motivated some recent work on zero-shot learning of visual categories where there are no training images available for test categories (unseen classes), *e.g.* [89]. Such approaches exploit the similarity (visual or semantic) between seen classes and unseen ones, or describe unseen classes in terms of a learned vocabulary of semantic visual attributes.

In contrast to the lack of reasonable size training sets for a large number of real world categories, there are abundant of textual descriptions of these categories. This comes in the form of dictionary entries, encyclopedia articles, and various online resources. For example, it is possible to find several good descriptions of a “bobolink” in encyclopedias of birds, while there are only a few images available for that bird online.

The main question we address in this paper is how to use purely textual description of categories with no training images to learn visual classifiers for these categories. In other words, we aim at zero-shot learning of object categories where the description of unseen categories comes in the form of typical text such as an encyclopedia entry. We explicitly address the question of how to automatically decide which information to transfer between classes without the need of human intervention. In contrast to most related work, we go beyond the simple use of tags and image captions, and apply standard Natural Language Processing techniques to typical text to learn visual classifiers.

Similar to the setting of zero-shot learning, we use classes with training data (seen classes) to predict classifiers for classes with no training data (unseen classes).Recent works on zero-shot learning of object categories focused on leveraging knowledge about common attributes and shared parts [89]. Typically, attributes [142, 53] are manually defined by humans and are used to transfer knowledge between seen and unseen classes. In contrast, in our work we do not use any explicit attributes. The description of a new category is purely textual and the process is totally automatic without human annotation beyond the category labels.

The contribution of the paper is on exploring this new problem, which to the best of our knowledge, is not explored in the computer vision community. We learn from an image corpus and a textual corpus, however not in the form of image-caption pairs, instead the only alignment between the corpora is at the level of the category. We propose and investigate two baseline formulations based on regression and domain adaptation. Then we propose a new constrained optimization formulation that combines a regression function and a knowledge transfer function with additional constraints to solve the problem.

Beyond the introduction and the related work sections, the paper is structured as follows: Sec 2.3 introduces the problem definition and proposed baseline solutions. Sec 2.4 describes the solution framework. Sec 2.5 explains the experiments performed on Flower Dataset [116] (102 classes) and Caltech-UCSD dataset [160] (200 classes).

2.2 Related Work

Our proposed work can be seen in the context of knowledge sharing and inductive transfer. In general, knowledge transfer aims at enhancing recognition by exploiting shared knowledge between classes. Most existing research focused on knowledge sharing within the visual domain only, *e.g.* [69]; or exporting semantic knowledge at the level of category similarities and hierarchies, *e.g.* [58, 141]. We go beyond the state-of-the-art to explore cross-domain knowledge sharing and transfer. We explore how knowledge from the visual and textual domains can be used to learn across-domain correlation, which facilitates prediction of visual classifiers from textual description.

Motivated by the practical need to learn visual classifiers of rare categories, researchers have explored approaches for learning from a single image (one-shot learning [109, 55, 59, 12]) or even from no images (zero-shot learning). One way of recognizing object instances from previously unseen test categories (the zero-shot learning problem) is by leveraging knowledge about common attributes and shared parts. Typically an intermediate semantic layer is introduced to enable sharing knowledge between classes and facilitate describing knowledge about novel unseen classes, *e.g.* [119]. For instance, given adequately labeled training data, one can learn classifiers for the attributes occurring in the training object categories. These classifiers can then be used to recognize the same attributes in object instances from the novel test categories. Recognition can then proceed on the basis of these learned attributes [89, 53]. Such attribute-based "knowledge transfer" approaches use an intermediate visual attribute representation to enable describing unseen object categories. Typically attributes are manually defined by humans to describe shape, color, surface material, *e.g.* , furry, striped, *etc.* Therefore, an unseen category has to be specified in terms of the used vocabulary of attributes. Rohrbach *et al.* [135] investigated extracting useful attributes from large text corpora. In [121], an approach was introduced for interactively defining a vocabulary of attributes that are both human understandable and visually discriminative. In contrast, our work does not use any explicit attributes. The description of a new category is purely textual.

The relation between linguistic semantic representations and visual recognition have been explored. For example in [28], it was shown that there is a strong correlation between semantic similarity between classes, based on WordNet, and confusion between classes. Linguistic semantics in terms of nouns from WordNet [110] have been used in collecting large-scale image datasets such as ImageNet[29] and Tiny Images [153]. It was also shown that hierarchies based on WordNet are useful in learning visual classifiers, *e.g.* [141].

One of the earliest work on learning from images and text corpora is the work of Barnard *et al.* [11], which showed that learning a joint distribution of words and visual elements facilitates clustering the images in a semantic way, generating illustrative images from a caption, and generating annotations for novel images. There has been an increasing recent interest in the intersection between computer vision and natural language processing with researches that focus on generating textual description of images and videos, *e.g.* [54, 86, 166, 82]. This includes generating sentences about objects, actions, attributes, patial relation between objects, contextual information in the images, scene information, *etc.* In contrast, our work is different

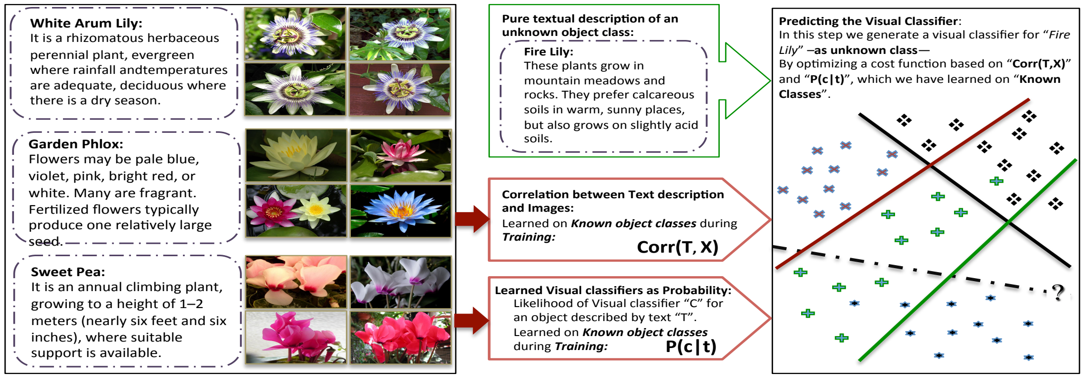


Figure 2.2: Illustration of the Proposed Solution Framework for the task Zero-shot learning from textual description.

in two fundamental ways. In terms of the goal, we do not target generating textual description from images, instead we target predicting classifiers from text, in a zero-shot setting. In terms of the learning setting, the textual descriptions that we use is at the level of the category and do not come in the form of image-caption pairs, as in typical datasets used for text generation from images, *e.g.* [118].

2.3 Problem Definition

Fig 2.1 illustrates the learning setting. The information in our problem comes from two different domains: the visual domain and the textual domain, denoted by $\mathcal{V}$ and $\mathcal{T}$, respectively. Similar to traditional visual learning problems, we are given training data in the form $V = \{(x_i, l_i)\}_N$, where x_i is an image and $l_i \in \{1 \cdots N_{sc}\}$ is its class label. We denote the number of classes available at training as N_{sc}, where sc indicates "seen classes". As typically done in visual classification setting, we can learn N_{sc} binary one-vs-all classifiers, one for each of these classes. Let us consider a typical binary linear classifier in the feature space in the form

$$f_k(\mathbf{x}) = \mathbf{c}_k^\intercal \cdot \mathbf{x}$$

where $\mathbf{x}$ is the visual feature vector amended with 1, and $\mathbf{c}_k \in \mathbb{R}^{d_v}$ is the linear classifier parameters for class k. Given a test image, its class is determined by

$$l^* = \arg\max_k f_k(\mathbf{x})$$

Our goal is to be able to predict a classifier for a new category based only on the learned classes and a textual description(s) of that category. In order to achieve that, the learning process has to also include textual description of the seen classes (as shown in Fig 2.1). Depending on the domain we might find a few, a couple, or as little as one textual description to each class. We denote the textual training data for class k by $\{t_i \in \mathcal{T}\}^k$. In this paper we assume we are dealing with the extreme case of having only one textual description available per class, which makes the problem even more challenging. However, the formulation we propose in this paper directly applies to the case of multiple textual descriptions per class. Similar to the visual domain, the raw textual descriptions have to go through a feature extraction process, which will be described in Sec 2.5. Let us denote the extracted textual feature by $T = \{\mathbf{t}_k \in \mathbb{R}^{d_t}\}_{k=1 \cdots N_{sc}}$.

Given a textual description $\mathbf{t}_*$ of a new unseen category, $\mathcal{C}$, the problem can now be defined as predicting a one-vs-all classifier parameters $c(\mathbf{t}_*)$, such that it can be directly used to classify any test image $\mathbf{x}$ as

$$\begin{aligned} c(\mathbf{t}_*)^\mathsf{T} \cdot \mathbf{x} > 0 &\quad \text{if } \mathbf{x} \text{ belongs to } \mathcal{C} \\ c(\mathbf{t}_*)^\mathsf{T} \cdot \mathbf{x} < 0 &\quad \text{otherwise} \end{aligned} \tag{2.1}$$

In what follows, we introduce two possible frameworks for this problem and discuss potential limitations for them, which leads next to the proposed formulation.

2.3.1 Regression Models

A straightforward way to solve this problem is to pose it as a regression problem where the goal is to use the textual data and the learned classifiers, $\{(\mathbf{t}_k, \mathbf{c}_k)\}_{k=1 \cdots N_{sc}}$ to learn a regression function from the textual feature domain to the visual classifier domain, *i.e.* , a function $c(\cdot) : \mathbb{R}^{d_t} \rightarrow \mathbb{R}^{d_v}$. The question is which regression model would be suitable for this problem? and would posing the problem this way give reasonable results?

A typical regression model, such as ridge regression [72] or Gaussian Process (GP) Regression [127], learns the regressor to each dimension of the output domain (the parameters of a linear classifier) separately, *i.e.* a set of function $c^j(\cdot) : \mathbb{R}^{d_t} \rightarrow \mathbb{R}$. Clearly this will not capture the correlation between the visual and textual domain. Instead, a structured prediction regressor would be more suitable since it would learn the correlation between the input and output domain. However, even a structure prediction model, will only learn the correlation between the textual and visual domain through the information available in the input-output pairs $(\mathbf{t}_k, \mathbf{c}_k)$. Here the visual domain information is encapsulated in the pre-learned classifiers and prediction does not have access to the original data in the visual domain. Instead we need to directly learn the correlation between the visual and textual domain and use that for prediction.

Another fundamental problem that a regressor would face, is the sparsity of the data; the data points are the textual description-classifier pairs, and typically the number of classes can be very small compared to the dimension of the classifier space (*i.e.* $N_{sc} \ll d_v$). In a setting like that, any regression model is bound to suffer from an under fitting problem. This can be best explained in terms of GP regression, where the predictive variance increases in the regions of the input space where there are no data points. This will result in poor prediction of classifiers at these regions.

2.3.2 Knowledge Transfer Models

An alternative formulation is to pose the problem as domain adaptation from the textual to the visual domain. In the computer vision context, domain adaptation work has focused on transferring categories learned from a source domain, with a given distribution of images, to a target domain with different distribution, *e.g.* , images or videos from different sources [164, 139, 85, 35]. What we need is an approach that learns the correlation between the textual domain features and the visual domain features, and uses that correlation to predict new visual classifier given textual features.

In particular, in [85] an approach for learning cross domain transformation was introduced. In that work a regularized asymmetric transformation between points in two domains were learned. The approach was applied to transfer learned categories between different data distributions, both in the visual domain. A particular attractive characteristic of [85], over other domain adaptation models, is that the source and target domains do not have to share the same feature spaces or the same dimensionality.

Inspired by [85], we can formulate the zero-shot learning problem as a domain adaptation. This can be achieve by learning a linear (or nonlinear kernalized) transfer function $\mathbf{W}$ between

$\mathcal{T}$ and $\mathcal{V}$. The transformation matrix $\mathbf{W}$ can be learned by optimizing, with a suitable regularizer, over constraints of the form $\mathbf{t}^\mathsf{T}\mathbf{W}\mathbf{x} \geq l$ if $\mathbf{t} \in \mathcal{T}$ and $\mathbf{x} \in \mathcal{V}$ belong to the same class, and $\mathbf{t}^\mathsf{T}\mathbf{W}\mathbf{x} \leq u$ otherwise. Here l and u are model parameters. This transfer function acts as a compatibility function between the textual features and visual features, which gives high values if they are from the same class and a low value if they are from different classes.

It is not hard to see that this transfer function can act as a classifier. Given a textual feature $\mathbf{t}^*$ and a test image, represented by $\mathbf{x}$, a classification decision can be obtained by $\mathbf{t}_*^\mathsf{T}\mathbf{W}\mathbf{x} \gtrless b$ where b is a decision boundary which can be set to $(l+u)/2$. Hence, our desired predicted classifier in Eq 2.1 can be obtained as $c(\mathbf{t}_*) = \mathbf{t}_*^\mathsf{T}\mathbf{W}$ (note that the features vectors are amended with ones). However, since learning $\mathbf{W}$ was done over seen classes only, it is not clear how the predicted classifier $c(\mathbf{t}_*)$ will behave for unseen classes. There is no guarantee that such a classifier will put all the seen data on one side and the new unseen class on the other side of that hyperplane.

2.4 Problem Formulation

2.4.1 Objective Function

The proposed formulation aims at predicting the hyperplane parameter $\mathbf{c}$ of a one-vs-all classifier for a new unseen class given a textual description, encoded by $\mathbf{t}$ and knowledge learned at the training phase from seen classes. Fig 2.2 illustrates our solution framework. At the training phase three components are learned:

Classifiers: a set of one-vs-all classifiers $\{\mathbf{c}_k\}$ are learned, one for each seen class.

Probabilistic Regressor: Given $\{(\mathbf{t}_k, \mathbf{c}_k)\}$ a regressor is learned that can be used to give a prior estimate for $p_{reg}(\mathbf{c}|\mathbf{t})$ (Details in Sec 2.4.3).

Domain Transfer Function: Given T and V a domain transfer function, encoded in the matrix $\mathbf{W}$ is learned, which captures the correlation between the textual and visual domains (Details in Sec 2.4.2).

Each of these components contains partial knowledge about the problem. The question is how to combine such knowledge to predict a new classifier given a textual description. The new classifier has to be consistent with the seen classes. The new classifier has to put all the seen instances at one side of the hyperplane, and has to be consistent with the learned domain transfer function. This leads to the following constrained optimization problem

$$
\begin{aligned}
\hat{c}(\mathbf{t}_*) =& \underset{\mathbf{c},\zeta_i}{\operatorname{argmin}} \big[\mathbf{c}^\mathsf{T}\mathbf{c} - \alpha \mathbf{t}_*{}^\mathsf{T}\mathbf{W}\mathbf{c} - \beta \ln(p_{reg}(\mathbf{c}|\mathbf{t}_*)) \\
& + \gamma \sum \zeta_i \big] \\
& s.t. : -(\mathbf{c}^\mathsf{T}\mathbf{x}_i) \geq \zeta_i, \;\; \zeta_i \geq 0, \;\;\; i = 1 \cdots N \\
& \mathbf{t}_*{}^\mathsf{T}\mathbf{W}\mathbf{c} \geq l \\
& \alpha, \beta, \gamma, l : \text{hyperparameters}
\end{aligned} \tag{2.2}
$$

The first term is a regularizer over the classifier $\mathbf{c}$. The second term enforces that the predicted classifier has high correlation with $\mathbf{t}_*^\mathsf{T}\mathbf{W}$. The third term favors a classifier that has high probability given the prediction of the regressor. The constraints $-\mathbf{c}^\mathsf{T}\mathbf{x}_i \geq \zeta_i$ enforce all the seen data instances to be at the negative side of the predicted classifier hyperplane with some miss-classification allowed through the slack variables ζ_i. The constraint $\mathbf{t}_*{}^\mathsf{T}\mathbf{W}\mathbf{c} \geq l$ enforces that the correlation between the predicted classifier and $\mathbf{t}_*{}^\mathsf{T}\mathbf{W}$ is no less than l, this is to enforce a minimum correlation between the text and visual features.

2.4.2 Domain Transfer Function

To learn the domain transfer function $\mathbf{W}$ we adapted the approach in [85] as follows. Let $\mathbf{T}$ be the textual feature data matrix and $\mathbf{X}$ be the visual feature data matrix where each feature vector is amended with a 1. Notice that amending the feature vectors with a 1 is essential in our formulation since we need $\mathbf{t}^\mathsf{T}\mathbf{W}$ to act as a classifier. We need to solve the following optimization problem

$$
\min_{\mathbf{W}} r(\mathbf{W}) + \lambda \sum_i c_i(\mathbf{T}\mathbf{W}\mathbf{X}^\mathsf{T}) \tag{2.3}
$$

where c_i's are loss functions over the constraints and $r(\cdot)$ is a matrix regularizer. It was shown in [85], under condition on the regularizer, that the optimal $\mathbf{W}$ in Eq 3.4 can be computed using inner products between data points in each of the domains separately, which results in a kernalized non-linear transfer function; hence its complexity does not depend on the dimensionality of either of the domains. The optimal solution of 3.4 is in the form $\mathbf{W}^* = \mathbf{T}\mathbf{K}_T^{-\frac{1}{2}}\mathbf{L}^*\mathbf{K}_X^{-\frac{1}{2}}\mathbf{X}^\mathsf{T}$,where $\mathbf{K}_T = \mathbf{T}\mathbf{T}^\mathsf{T}$, $\mathbf{K}_X = \mathbf{X}\mathbf{X}^\mathsf{T}$. $\mathbf{L}^*$ is computed by minimizing the following minimization problem

$$\min_{\mathbf{L}}[r(\mathbf{L}) + \lambda \sum_p c_p(\mathbf{K}_T^{\frac{1}{2}}\mathbf{L}\mathbf{K}_X^{\frac{1}{2}})], \tag{2.4}$$

where $c_p(\mathbf{K}_T^{\frac{1}{2}}\mathbf{L}\mathbf{K}_X^{\frac{1}{2}}) = (max(0,(l - e_i\mathbf{K}_T^{\frac{1}{2}}\mathbf{L}\mathbf{K}_X^{\frac{1}{2}}e_j)))^2$ for same class pairs of index i,j, or $= (max(0,(e_i\mathbf{K}_T^{\frac{1}{2}}\mathbf{L}\mathbf{K}_X^{\frac{1}{2}}e_j - u)))^2$ otherwise, where e_k is a vector of zeros except a one at the k^{th} element, and $u > l$ (note any appropriate l, u could work. In our case, we used $l = 2$, $u = -2$). We used a Frobenius norm regularizer. This energy is minimized using a second order BFGS quasi-Newton optimizer. Once L is computed W^* is computed using the transformation above.

2.4.3 Probabilistic Regressor

There are different regressors that can be used, however we need a regressor that provide a probabilistic estimate $p_{reg}(\mathbf{c}|(t))$. For the reasons explained in Sec 2.3, we also need a structure prediction approach that is able to predict all the dimensions of the classifiers together. For these reasons, we use the Twin Gaussian Process (TPG) [16]. TGP encodes the relations between both the inputs and structured outputs using Gaussian Process priors. This is achieved by minimizing the Kullback-Leibler divergence between the marginal GP of the outputs (i.e. classifiers in our case) and observations (i.e. textual features). The estimated regressor output ($\tilde{c}(\mathbf{t}_*)$) in TGP is given by the solution of the following non-linear optimization problem [16] [1].

$$\begin{aligned}\tilde{c}(\mathbf{t}_*) = \operatorname*{argmin}_{\mathbf{c}}[&K_C(\mathbf{c},\mathbf{c}) - 2k_c(\mathbf{c})^\mathsf{T}\mathbf{u} - \eta \log(K_C(\mathbf{c},\mathbf{c}) \\ &- k_c(\mathbf{c})^\mathsf{T}(\mathbf{K}_C + \lambda_c\mathbf{I})^{-1}k_c(c))]\end{aligned} \tag{2.5}$$

where $\mathbf{u} = (\mathbf{K}_\mathsf{T} + \lambda_t\mathbf{I})^{-1}k_t(\mathbf{t}_*)$, $\eta = K_T(\mathbf{t}_*,\mathbf{t}_*) - k(\mathbf{t}_*)^\mathsf{T}\mathbf{u}$, $K_T(\mathbf{t}_l,\mathbf{t}_m)$ and $K_C(\mathbf{c}_l,\mathbf{c}_m)$ are Gaussian kernel for input feature $\mathbf{t}$ and output vector $\mathbf{c}$. $k_c(\mathbf{c}) = [K_C(\mathbf{c},\mathbf{c}_1),\cdots,K_C(\mathbf{c},\mathbf{c}_{N_{sc}})]^\mathsf{T}$. $k_t(\mathbf{t}_*) = [K_T(\mathbf{t}_*,\mathbf{t}_1),\cdots,K_T(\mathbf{t}_*,\mathbf{t}_{N_{sc})}]^\mathsf{T}$. λ_t and λ_c are regularization parameters to avoid

[1] notice we are using $\tilde{\mathbf{c}}$ to denote the output of the regressor, while using $\hat{\mathbf{c}}$ to denote the output of the final optimization problem in Eq 3.9

overfitting. This optimization problem can be solved using a second order, BFGS quasi-Newton optimizer with cubic polynomial line search for optimal step size selection [16]. In this case the classifier dimension are predicted jointly. In this case $p_{reg}(\mathbf{c}|\mathbf{t}_*)$ is defined as a normal distribution.

$$p_{reg}(\mathbf{c}|\mathbf{t}_*) = \mathcal{N}(\mu_c = \tilde{c}(\mathbf{t}_*), \Sigma_c = \mathbf{I}) \tag{2.6}$$

The reason that $\Sigma_c = \mathbf{I}$ is that TGP does not provide predictive variance, unlike Gaussian Process Regression. However, it has the advantage of handling the dependency between the dimensions of the classifiers $\mathbf{c}$ given the textual features $\mathbf{t}$.

2.4.4 Solving for $\hat{c}$ as a quadratic program

According to the definition of $p_{reg}(\mathbf{c}|\mathbf{t}_*)$ for TGP, $\ln p(\mathbf{c}|\mathbf{t}_*)$ is a quadratic term in c in the form

$$\begin{aligned} -\ln p(\mathbf{c}|\mathbf{t}_*) &\propto (\mathbf{c} - \tilde{c}(\mathbf{t}_*))^{\mathsf{T}}(\mathbf{c} - \tilde{c}(\mathbf{t}_*)) \\ &= \mathbf{c}^{\mathsf{T}}\mathbf{c} - 2\mathbf{c}^{\mathsf{T}}\tilde{c}(\mathbf{t}_*) + \tilde{c}(\mathbf{t}_*)^{\mathsf{T}}\tilde{c}(\mathbf{t}_*) \end{aligned} \tag{2.7}$$

We reduce $-\ln p(\mathbf{c}|\mathbf{t}_*)$ to $-2\mathbf{c}^{\mathsf{T}}\tilde{c}(\mathbf{t}_*))$, since 1) $\tilde{c}(\mathbf{t}_*)^{\mathsf{T}}\tilde{c}(\mathbf{t}_*)$ is a constant (*i.e.* does not affect the optimization), 2) $\mathbf{c}^{\mathsf{T}}\mathbf{c}$ is already included as regularizer in equation 3.9. In our setting, the dot product is a better similarity measure between two hyperplanes. Hence, $-2\mathbf{c}^{\mathsf{T}}\tilde{c}(\mathbf{t}_*)$ is minimized. Given $-\ln p(\mathbf{c}|\mathbf{t}_*)$ from the TGP and $\mathbf{W}$, Eq 3.9 reduces to a quadratic program on $\mathbf{c}$ with linear constraints. We tried different quadratic solvers, however the IBM CPLEX solver [2] gives the best performance in speed and optimization for our problem.

2.5 Experiments

2.5.1 Datasets

We used the CU200 Birds [160] (200 classes - 6033 images) and the Oxford Flower-102 [116] (102 classes - 8189 images) image dataset to test our approach, since they are among the largest and widely used fine-grained datasets. We generate textual descriptions for each class in both datasets. The CU200 Birds image dataset was created based on birds that have a corresponding

[2]http://www-01.ibm.com/software/integration/optimization/cplex-optimizer

Wikipedia article, so we have developed a tool to automatically extract Wikipedia articles given the class name. The tool succeeded to automatically generate 178 articles, and the remaining 22 articles was extracted manually from Wikipedia. These mismatches happens only when article title is a different synonym of the same bird class. On the other hand, Flower image dataset was not created using the same criteria as the Bird dataset, so classes of the Flower dataset classes does not necessarily have corresponding Wikipedia article. The tool managed to generate about 16 classes from Wikipedia out of 102, the remaining 86 articles was generated manually for each class from Wikipedia, Plant Database [3], Plant Encyclopedia [4], and BBC articles [5]. We plan to make the extracted textual description available as augmentations of these datasets. Sample textual description can be found in the supplementary material.

2.5.2 Extracting Textual Features

The textual features were extracted in two phases, which are typical in document retrieval literature. The first phase is an indexing phase that generates textual features with tf-idf (Term Frequency-Inverse Document Frequency) configuration (Term frequency as local weighting while inverse document frequency as a global weighting). The tf-idf is a measure of how important is a word to a text corpus. The tf-idf value increases proportionally to the number of times a word appears in the document, but is offset by the frequency of the word in the corpus, which helps to control for the fact that some words are generally more common than others. We used the normalized frequency of a term in the given textual description [143]. The inverse document frequency is a measure of whether the term is common; in this work we used the standard logarithmic idf [143]. The second phase is a dimensionality reduction step, in which Clustered Latent Semantic Indexing (CLSI) algorithm [169] is used. CLSI is a low-rank approximation approach for dimensionality reduction, used for document retrieval. In the Flower Dataset, tf-idf features $\in \mathbb{R}^{8875}$ and after CLSI the final textual features $\in \mathbb{R}^{102}$. In the Birds Dataset, tf-idf features is in $\mathbb{R}^{7086}$ and after CLSI the final textual features is in $\mathbb{R}^{200}$.

[3] http://plants.usda.gov/java/

[4] http://www.theplantencyclopedia.org/wiki/Main_Page

[5] http://www.bbc.co.uk/science/0/

2.5.3 Visual features

We used the Classeme features [154] as the visual feature for our experiments since they provide an intermediate semantic representation of the input image. Classeme features are output of a set of classifiers corresponding to a set of C category labels, which are drawn from an appropriate term list defined in [154], and not related to our textual features. For each category $c \in \{1 \cdots C\}$, a set of training images is gathered by issuing a query on the category label to an image search engine. After a set of coarse feature descriptors (Pyramid HOG, GIST, *etc.*) is extracted, a subset of feature dimensions was selected [154], and a one-versus-all classifier ϕ_c is trained for each category. The classifier output is real-valued, and is such that $\phi_c(x) > \phi_c(y)$ implies that x is more similar to class c than y is. Given an image x, the feature vector (descriptor) used to represent it is the classeme vector $[\phi_1(x), \cdots, \phi_C(x)]$. The Classeme feature is of dimensionality 2569.

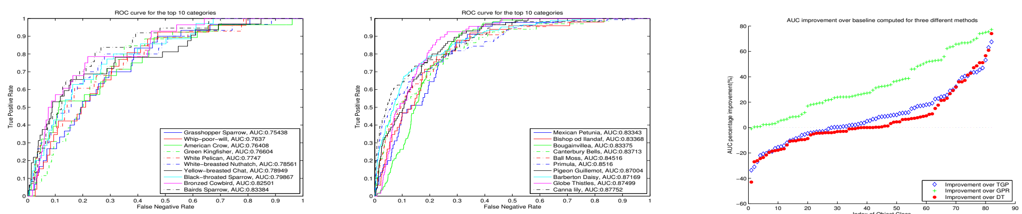


Figure 2.3: **Left and Middle**: ROC curves of best 10 predicted classes (best seen in color) for Bird and Flower datasets respectively, **Right**: AUC improvement over the three baselines on Flower dataset. The improvement is sorted in an increasing order for each baseline separately

2.5.4 Experimental Results

Evaluation Methodology and Metrics: Similar to zero-shot learning literature, we evaluated the performance of an unseen classifier in a one-vs-all setting where the test images of unseen classes are considered to be the positives and the test images from the seen classes are considered to be the negatives. We computed the ROC curve and report the area under that curve (AUC) as a comparative measure of different approaches. In zero-shot learning setting the test data from the seen class are typically very large compared to those from unseen classes. This makes other measures, such as accuracy, useless since high accuracy can be obtained even if all the unseen class test data are wrongly classified; hence we used ROC curves, which are

Table 2.1: Comparative Evaluation on the Flowers and Birds

Approach	Flowers Avg AUC (+/- std)	Birds Avg AUC (+/- std)
GPR	0.54 (+/- 0.02)	0.52 (+/- 0.001)
TGP	0.58 (+/- 0.02)	0.61 (+/- 0.02)
DA	0.62(+/- 0.03)	0.59 (+/- 0.01)
Our Approach	**0.68** (+/- 0.01)	**0.62** (+/- 0.02)

independent of this problem. Five-fold cross validation over the classes were performed, where in each fold 4/5 of the classes are considered as "seen classes" and are used for training and 1/5th of the classes were considered as "unseen classes" where their classifiers are predicted and tested. Within each of these class-folds, the data of the seen classes are further split into training and test sets. The hyper-parameters for the approach were selected through another five-fold cross validation within the class-folds (i.e. the 80% training classes are further split into 5 folds to select the hyper-parameters).

Baselines: Since our work is the first to predict classifiers based on pure textual description, there are no other reported results to compare against. However, we designed three state-of-the-art baselines to compare against, which are designed to be inline with our argument in Sec 2.3. Namely we used: 1) A Gaussian Process Regressor (GPR) [127], 2) Twin Gaussian Process (TGP) [16] as a structured regression method, 3) Nonlinear Asymmetric Domain Adaptation (DA) [85]. The TGP and DA baselines are of particular importance since our formulation utilizes them, so we need to test if the formulation is making any improvement over them. It has to be noted that we also evaluate TGP and DA as alternative formulations that we are proposing for the problem, none of them was used in the same context before.

Results: Table 2.1 shows the average AUCs for the proposed approach in comparison to the three baselines on both datasets. GPR performed poorly in all classes in both data sets, which was expected since it is not a structure prediction approach. The DA formulation outperformed TGP in the flower dataset but slightly underperformed on the Bird dataset. The proposed approach outperformed all the baselines on both datasets, with significant difference on the flower dataset. It is also clear that the TGP performance was improved on the Bird dataset since it has

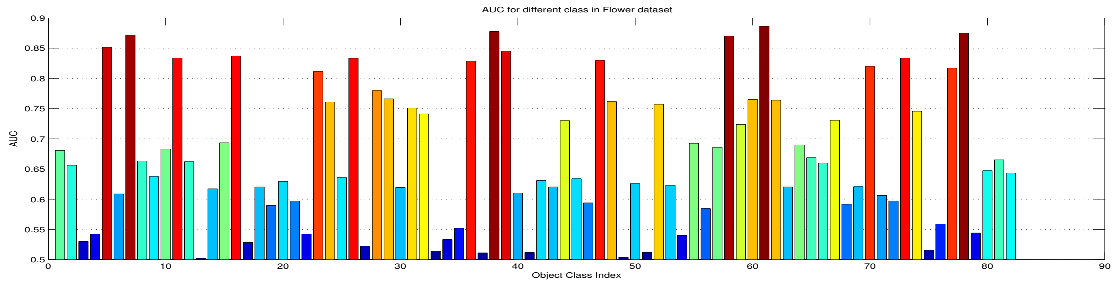


Figure 2.4: AUC of the predicated classifiers for all classes of the flower datasets

Table 2.2: Percentage of classes that the proposed approach makes an improvement in predicting over the baselines (relative to the total number of classes in each dataset

baseline	Flowers (102) % improvement	Birds (200) % improvement
GPR	100 %	98.31 %
TGP	66 %	51.81 %
DA	54%	56.5%

more classes (more points are used for prediction). Fig 2.3 shows the ROC curves for our approach on best predicted unseen classes from the Birds dataset on the Left and Flower dataset on the middle. Fig 2.4 shows the AUC for all the classes on Flower dataset. More results are attached in the supplementary materials.

Fig 2.3, on the right, shows the improvement over the three baseline for each class, where the improvement is calculated as (our AUC- baseline AUC)/ baseline AUC %. Table 2.2 shows the percentage of the classes which our approach makes a prediction improvement for each of the three baselines. Table 2.3 shows the five classes in Flower dataset where our approach made the best average improvement. The point of that table is to show that in these cases both TGP and DA did poorly while our formulation that is based on both of them did significantly better. This shows that our formulation does not simply combine the best of the two approaches but can significantly improve the prediction performance.

To evaluate the effect of the constraints in the objective function, we removed the constraints $-(\mathbf{c}^\mathsf{T}\mathbf{x}_i) \geq \zeta_i$ which try to enforces all the seen examples to be on the negative side of the predicted classifier hyperplane and evaluated the approach. The result on the flower dataset (using one fold) was reduced to average AUC=0.59 compared to AUC=0.65 with the

Table 2.3: Top-5 classes with highest combined improvement in Flower dataset

class	TGP (AUC)	DA (AUC)	Our (AUC)	% Improv.
2	0.51	0.55	0.83	57%
28	0.52	0.54	0.76	43.5%
26	0.54	0.53	0.76	41.7%
81	0.52	0.82	0.87	37%
37	0.72	0.53	0.83	35.7 %

constraints. Similarly, we evaluated the effect of the constraint $\mathbf{t}_*^\top \mathbf{W}\mathbf{c} \geq l$. The result was reduced to average AUC=0.58 compared to AUC=0.65 with the constraint. This illustrates the importance of this constraint in the formulation.

2.6 Conclusion and Future Work

We explored the problem of predicting visual classifiers from textual description of classes with no training images. We investigated and experimented with different formulations for the problem within the fine-grained categorization context. We proposed a novel formulation that captures information between the visual and textual domains by involving knowledge transfer from textual features to visual features, which indirectly leads to predicting the visual classifier described by the text. In the future, we are planning to propose a kernel version to tackle the problem instead of using linear classifiers. Furthermore, we will study predicting classifiers from complex-structured textual features.

Acknowledgment This research was partially funded by NSF award IIS-1218872

Chapter 3

Write a Kernel Classifier

In this paper we propose a framework for predicting kernelized classifiers in the visual domain for categories with no training images where the knowledge comes from textual description about these categories. Through our optimization framework, the proposed approach is capable of embedding the class-level knowledge from the text domain as kernel classifiers in the visual domain. We also proposed a distributional semantic kernel between text descriptions which is shown to be effective in our setting. The proposed framework is not restricted to textual descriptions, and can also be applied to other forms knowledge representations. Our approach was applied for the challenging task of zero-shot learning of fine-grained categories from text descriptions of these categories.

3.1 Introduction

We propose a framework to model kernelized classifier prediction in the visual domain for categories with no training images, where the knowledge about these categories comes from a secondary domain. The side information can be in the form of textual, parse trees, grammar, visual representations, concepts in the ontologies, or any form; see Fig 3.1. Our work focuses on the unstructured text setting. We denote the side information as “privileged” information, borrowing the notion from [157].

Our framework is an instance of the concept of Zero Shot Learning (ZSL)[90], aiming at transferring knowledge from seen classes to novel (unseen) classes. Most zero-shot learning applications in practice use symbolic or numeric visual attribute vectors [87, 89]. In contrast, recent works investigated other forms of descriptions, *e.g.* user provided feedback [159], textual descriptions [46]. It is common in zero-shot learning to introduce an intermediate layer that facilitates knowledge sharing between seen classes, hence the transfer of knowledge to unseen

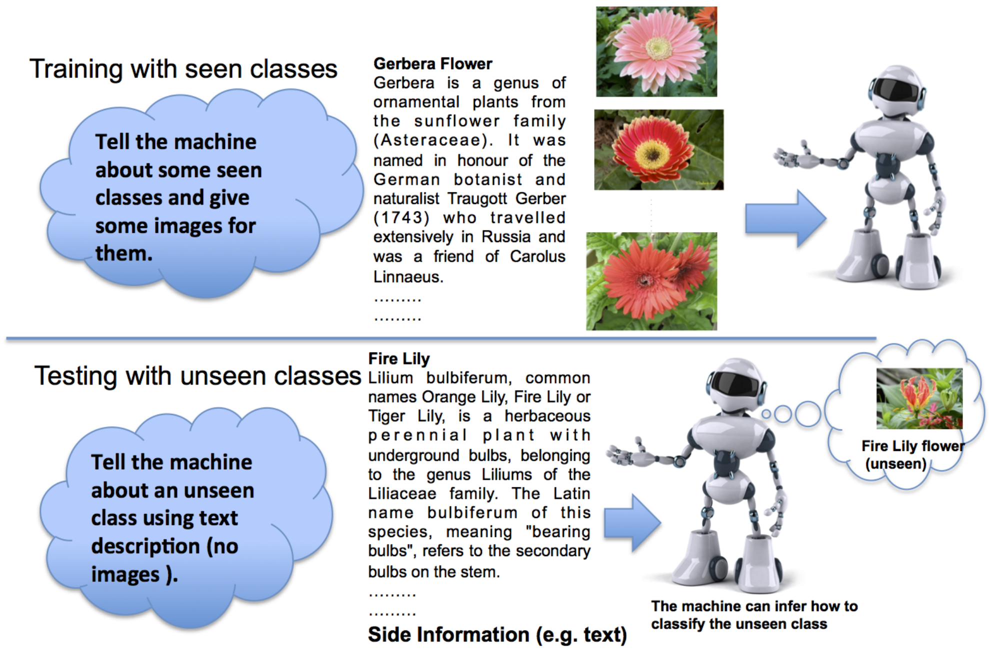


Figure 3.1: Our setting where machine can predict unseen class from pure unstructured text

classes. Typically, visual attributes are being used for that purpose, since they provide a human-understandable representation, which enables specifying new categories [87, 89, 53, 119, 1, 93]. A fundamental question in attribute-based ZSL models is how to define attributes that are visually discriminative and human understandable. Researchers has explored learning attributes from text sources, *e.g.* [129, 130, 135, 14]. Other works have explored interactive methodologies to learning visual attribute that are human understandable, *e.g.* [121].

There are several differences between our proposed framework and the state-of-the-art zero-shot learning approaches. We are not restricted to use attributes as the interface to specify new classes. We can use any “privileged” information available for each category. In particular in this paper we focus on the case of textual description of categories as the secondary domain. This difference is reflected in our zero-shot classification architecture. We learn a domain transfer model between the visual domain and the privileged information domain. This facilitates predicting explicit visual classifiers for novel unseen categories given their privileged information. The difference in architecture becomes clear if we consider, for the sake of argument, attributes as the secondary domain in our framework, although this is not the focus of the paper. In that case we do not need explicit attribute classifiers to be learned as an intermediate layer as typically done in attribute-based ZSL *e.g.* [89, 53, 119], instead the visual classifier are directly learned from the attribute labels. The need to learn an intermediate attribute classifier layer in most attribute-based zero-shot learning approaches dictates using strongly annotated data, where each image comes with attribute annotation, *e.g.* CU-Birds dataset [160]. In contrast, we do not need image-annotation pairs, and privileged information is only assumed at the category level; hence we denote our approach weakly supervised. This also directly facilitates using continuous attributes in our case, and does not assume independent between attributes.

Another fundamental difference in our case is that we predict explicit kernel classifier in the form defined in the representer theorem [144], from privileged information. Explicit classifier prediction means that the output of our framework is classifier parameters for any new category given text description, which can applied to any test image to predict its class. Predicting classifier in kernelized form opens the door for using any kind of side information about classes, as long as kernels can be defined on them. The image features also do not need to be in a vectorized format. Kernelized classifiers also facilitates combining different types of features through a multi-kernel learning (MKL) paradigm, where the fusion of different features can be effectively achieved.

We can summarize the features of our proposed framework, hence the contribution as follows: 1) Our framework explicitly predicts classifiers; 2) The predicted classifiers are kernelized; 3) The framework facilitates any type of "side" information to be used; 4) The approach requires the side information at the class level, not at the image level, hence, it needs only weak annotation. 5) We propose a distributional semantic kernel between text description of visual classes that we show its value in the experiments. The structure of the paper is as follows. Sec 3.2 describes the relation to existing literature. Sec 3.3 and 3.4 explains the learning setting and our formulation. Sec 3.5 presents the proposed distributional semantic kernel for text descriptions. Sec 3.6 shows our experimental results.

3.2 Related Work

We already discussed the relation to the zero-shot learning literature in the Introduction section. In this section, we focus on the relations to other volumes of literature.

There has been increasing interest recently in the intersection between Language and Computer Vision. Most of the work on this area is focused on generating textual description from images [**?**, 86, 118, 166, 112]. In contrast, we focus on generating visual classifiers from textual description or other side information at the category level.

There are few recent works that involved unannotated text to improve visual classification or achieve zero-shot learning. In [61, 117] and [150], word embedding language models (*e.g.* [108]) was adopted to represent class names as vectors. Their framework is based on mapping images into the learned language mode then perform classification in that space. In contrast, our framework maps the text information to a classifier in the visual domain, *i.e.* the apposite direction of their approach. There are several advantages in mapping textual knowledge into the visual domain. To perform ZSL, approaches such as [117, 61, 150] only embed new classes by their category names. This has clear limitations when dealing with fine-grained categories (such as different bird species). Most of fine-grained category names does not exist in current semantic models. Even if they exist, they will end up close to each other in the learned language models since they typically share similar contexts. This limits the discriminative power of such language models. In fact our baseline experiment using these models performed as low as random when applied to fine-grained category; described in Sec 3.6.4. Moreover, our framework directly can use large text description of novel categories. In contrast to [117, 61, 150] which required a vectorized representation of images, our framework facilitates non-linear classification using kernels.

In [46], an approach was proposed to predict linear classifiers from textual description, based on a domain transfer optimization method proposed in [85]. Although both of these works are kernelized, a close look reveals that kernelization was mainly used to reduce the size of the domain transfer matrix and the computational cost. The resulting predicted classifier in [46] is still a linear classifier. In contrast, our proposed formulation predicts kernelized visual classifiers directly from the domain transfer optimization, which is a more general case. This directly facilitates using classifiers that fused multiple visual cues such as Multiple Kernel Learning (MKL).

3.3 Problem Definition

We consider a zero-shot multi-class classification setting on domain $\mathcal{X}$ as follows. At training, besides the data points from $\mathcal{X}$ and the class labels, each class is associated with privileged information in a secondary domain $\mathcal{E}$ in particular, however not limited to, a textual description. We assume that each class $y_i \in Y_{sc}$(training/seen labels), is associated with privileged information $e_i \in \mathcal{E}$. While, our formulation allows multiple pieces of privileged information per class (*e.g.* multiple class-level textual descriptions), we will use one per class for simplicity. Hence, we denote the training as $\mathcal{D}_{train} = \{S_x = \{(x_i, y_i)\}_N, S_e = \{y_j, e_j\}_{N_{sc}}\}$, where $x_i \in \mathcal{X}$, $y_i \in Y_{sc}$, $y_j \in Y_{sc}$, and N_{sc} and N are the number of the seen classes and the training examples/images respectively. We assume that each of the domains is equipped with a kernel function corresponding to a *reproducing kernel Hilbert space* (RKHS). Let us denote the kernel for $\mathcal{X}$ by $k(\cdot,\cdot)$ and the kernel for $\mathcal{E}$ by $g(\cdot,\cdot)$. At the zero-shot time, only the privileged information e_{z^*} is available for each novel unseen class z^*; see Fig 3.1.

The common approach for multi-class classification is to learn a classifier for each class against the remaining classes (i.e., one-vs-all). According to the generalized representer theorem [144], a minimizer of a regularized empirical risk function over an RKHS could be represented as a linear combination of kernels, evaluated on the training set. Adopting the representer theorem on classification risk function, we define a kernel-classifier of class y as follows

$$f_y(x^*) = \sum_{i=1}^{N} \beta_y^i k(x^*, x_i) + b = \boldsymbol{\beta}_y{}^{\mathsf{T}} \mathbf{k}(x^*), \tag{3.1}$$

where $x^* \in \mathcal{X}$ is the test point, $x_i \in S_x$, $\mathbf{k}(x^*) = [k(x^*, x_1), \cdots, k(x^*, x_N), 1]^{\mathsf{T}}$, $\boldsymbol{\beta}_y = [\beta_y^1 \cdots \beta_y^N, b]^{\mathsf{T}}$. Having learned $f_y(x^*)$ for each class y (for example using SVM classifier), the class label of the test point x^* can be predicted as

$$y^* = \arg\max_y f_y(\mathbf{x}^*) \tag{3.2}$$

It is clear that $f_y(\mathbf{x}^*)$ could be learned for all classes with training data $y \in Y_{sc} = y_1 \cdots y_{N_{sc}}$, since there are examples S_x for the seen classes; we denote the kernel-classifier parameters of the seen classes as $\mathcal{B}_{sc} = \{\boldsymbol{\beta}_y\}_{N_{sc}}, \forall y \in Y_{sc}$. However, it is not obvious how to predict $f_{z^*}(\mathbf{x}^*)$ for a new unseen class $z^* \in Y_{us} = z_1 \cdots z_{N_{us}}$. Our main notion is to use the privileged information $e_{z^*} \in \mathcal{E}$, associated with unseen class z^*, and the training data $\mathcal{D}_{train}$ to directly predict the unseen kernel-classifier parameters. Hence, the classifier of z^* is a function of e_{k^*} and $\mathcal{D}_{train}$; *i.e.*

$$f_{z^*}(\mathbf{x}^*) = \boldsymbol{\beta}(e_{z^*}, \mathcal{D}_{train})^{\mathsf{T}} \cdot \mathbf{k}(x^*), \tag{3.3}$$

$f_{z^*}(\mathbf{x}^*)$ could be used to classify new points that belong to an unseen class as follows: 1) one-vs-all setting $f_{z^*}(\mathbf{x}^*) \gtrless 0$; or 2) in a Multi-class prediction as in Eq 3.2.

3.4 Approach

Prediction of $\boldsymbol{\beta}(e_{z^*}, \mathcal{D}_{train})$, which we denote as $\boldsymbol{\beta}(e_{z^*})$ for simplicity, is decomposed into training (domain transfer) and prediction phases.

3.4.1 Domain Transfer

During training, we firstly learn $\mathcal{B}_{sc}$ as SVM-kernel classifiers based on $\mathcal{S}_x$. Then, we learn a domain transfer function to transfer the privileged information $e \in \mathcal{E}$ to kernel-classifier parameters $\boldsymbol{\beta} \in \mathbb{R}^{N+1}$ in $\mathcal{X}$ domain. We call this function $\boldsymbol{\beta}_{DA}(e)$, which has the form of $\mathbf{T}^{\mathsf{T}}\mathbf{g}(e)$, where $\mathbf{g}(e) = [g(e, e_1) \cdots g(e, e_{N_{sc}})]^{\mathsf{T}}$; $\mathbf{T}$ is an $N_{sc} \times N+1$ matrix, which transforms e to kernel classifier parameters for the class e represents.

We aim to learn $\mathbf{T}$, such that $\mathbf{g}(e)^{\mathsf{T}}\mathbf{T}\mathbf{k}(x) > l$ if e and x correspond to the same class, $\mathbf{g}(e)^{\mathsf{T}}\mathbf{T}\mathbf{k}(x) < u$ otherwise. Here l controls similarity lower-bound if e and x correspond to same class, and u controls similarity upper-bound if e and x belong to different classes. In our setting, the term $\mathbf{T}^{\mathsf{T}}\mathbf{g}(e_i)$ should act as a classifier parameter for class i of the training data. Therefore, we introduce penalization constraints to our minimization function if $\mathbf{T}^{\mathsf{T}}\mathbf{g}(e_i)$ is distant from $\boldsymbol{\beta}_i \in \mathcal{B}_{sc}$, where e_i corresponds to the class that $\boldsymbol{\beta}_i$ classifies. Inspired by domain adaptation optimization methods (*e.g.* [85]), we model our domain transfer function as follows

$$\begin{aligned} \mathbf{T}^* = \arg\min_{\mathbf{T}} L(\mathbf{T}) = [\frac{1}{2} r(\mathbf{T}) + \lambda_1 \sum_k c_k(\mathbf{GTK}) + \\ \lambda_2 \sum_{i=1}^{N_{sc}} \|\boldsymbol{\beta}_i - \mathbf{T}^{\mathsf{T}}\mathbf{g}(e_i)\|^2 \end{aligned} \tag{3.4}$$

where, $\mathbf{G}$ is an $N_{sc} \times N_{sc}$ symmetric matrix, such that both the i^{th} row and the i^{th} column are equal to $\mathbf{g}(e_i)$, $e_i \in \mathcal{S}_e$; $\mathbf{K}$ is an $N+1 \times N$ matrix, such that the i^{th} column is equal to $\mathbf{k}(x_i)$, $x_i \in \mathcal{S}_x$. c_k's are loss functions over the constraints defined as $c_k(\mathbf{GTK})) = (max(0, (l - \mathbf{1}_i^{\mathsf{T}}\mathbf{GTK}\mathbf{1}_j)))^2$ for same class pairs of index i and j, or $= r \cdot (max(0, (\mathbf{1}_i^{\mathsf{T}}\mathbf{GTK}\mathbf{1}_j - u)))^2$ otherwise, where $\mathbf{1}_i$ is an $N_{sc} \times 1$ vector with all zeros except at index i, $\mathbf{1}_j$ is an $N \times 1$ vector with all zeros except at index j. This leads to $c_k(\mathbf{GTK}) = max(0, (l - \mathbf{g}(e_i)^\mathsf{T}\mathbf{Tk}(x_j)))^2$ for same class pairs of index i and j, or $= r \cdot (max(0, (\mathbf{g}(e_i)^\mathsf{T}\mathbf{Tk}(x_j) - u)))^2$ otherwise, where $u > l$, $r = \frac{nd}{ns}$ such that nd and ns are the number of pairs (i, j) of different classes and similar pairs respectively. Finally, we used a Frobenius norm regularizer for $r(\mathbf{T})$.

The objective function in Eq 3.4 controls the involvement of the constraints c_k by the term multiplied by λ_1, which controls its importance; we call it $C_{l,u}(\mathbf{T})$. While, the trained classifiers penalty is captured by the term multiplied by λ_2; we call it $C_\beta(\mathbf{T})$. One important observation on $C_\beta(\mathbf{T})$ is that it reaches zero when $\mathbf{T} = \mathbf{G}^{-1}\mathbf{B}^\mathsf{T}$, where $\mathbf{B} = [\boldsymbol{\beta}_1 \cdots \boldsymbol{\beta}_{N_{sc}}]$, since it could be rewritten as $C_\beta(\mathbf{T}) = \|\mathbf{B}^\mathsf{T} - \mathbf{GT}\|_F^2$.

One approach to minimize $L(\mathbf{T})$ is gradient-based optimization using a quasi-Newton optimizer. Our gradient derivation of $L(\mathbf{T})$ leads to the following form

$$\frac{\delta L(\mathbf{T})}{\delta \mathbf{T}} = \mathbf{T} + \lambda_1 \cdot \sum_{i,j} \mathbf{g}(e_i)\mathbf{k}(x_j)^\mathsf{T} v_{ij} + r \cdot \lambda_2 \cdot (\mathbf{G}^2\mathbf{T} - \mathbf{GB}) \tag{3.5}$$

where $v_{ij} = -2 \cdot max(0, (l - \mathbf{g}(e_i)^\mathsf{T}\mathbf{Tk}(x_j)))$ if i and j correspond to the same class, $2 \cdot max(0, (\mathbf{g}(e_i)^\mathsf{T}\mathbf{Tk}(x_j) - u)$ otherwise. Another approach to minimize $L(\mathbf{T})$ is through alternating projection using Bregman algorithm [18], in which $\mathbf{T}$ is updated with respect to a single constraint every iteration.

3.4.2 Classifier Prediction

We propose two ways to predict the kernel-classifier. (1) Domain Transfer (DT) Prediction, (2) One-class-SVM adjusted DT Prediction.

Domain Transfer (DT) Prediction: Construction of an unseen category is directly computed from our domain transfer model as follows

$$\tilde{\boldsymbol{\beta}}_{DT}(\mathbf{e}_{z^*}) = \mathbf{T^*}^\mathsf{T}\mathbf{g}(e_{z^*}) \tag{3.6}$$

One-class-SVM adjusted DT (SVM-DT) Prediction: In order to increase separability against seen classes, we adopted the inverse of the idea of the one class kernel-svm, whose main idea is to build a confidence function that takes only positive examples of the class. Our setting is the opposite scenario; seen examples are negative examples of the unseen class. In order introduce our proposed adjustment method, we start by presenting the one-class SVM objective function.

The Lagrangian dual of the one-class SVM [49] can be written as

$$\begin{aligned} \boldsymbol{\beta}^*_+ =& \underset{\boldsymbol{\beta}}{\text{argmin}} \big[\boldsymbol{\beta}^\mathsf{T} \mathbf{K}^{'} \boldsymbol{\beta} - \boldsymbol{\beta}^T \mathbf{a}\big] \\ & s.t.: \boldsymbol{\beta}^T \mathbf{1} = 1, 0 \leq \boldsymbol{\beta}_i \leq C; i = 1 \cdots N \end{aligned} \tag{3.7}$$

where $\mathbf{K}^{'}$ is an $N \times N$ matrix, $\mathbf{K}^{'}(i,j) = k(x_i, x_j)$, $\forall x_i, x_j \in \mathcal{S}_x$ (*i.e.* in the training data), **a** is an $N \times 1$ vector, $\mathbf{a}_i = k(x_i, x_i)$, C is a hyper-parameter . It is straightforward to see that, if β is aimed to be a negative decision function instead, the objective function becomes in the form

$$\begin{aligned} \boldsymbol{\beta}^*_- =& \underset{\boldsymbol{\beta}}{\text{argmin}} \big[\boldsymbol{\beta}^\mathsf{T} \mathbf{K}^{'} \boldsymbol{\beta} + \boldsymbol{\beta}^T \mathbf{a}\big] \\ & s.t.: \boldsymbol{\beta}^T \mathbf{1} = -1, -C \leq \boldsymbol{\beta}_i \leq 0; i = 1 \cdots N \end{aligned} \tag{3.8}$$

While $\boldsymbol{\beta}^*_- = -\boldsymbol{\beta}^*_+$, the objective function in Eq 3.8 of the one-negative class SVM inspires us with the idea to adjust the kernel-classifier parameters to increase separability of the unseen kernel-classifier against the points of the seen classes, which leads to the following objective function

$$\begin{aligned} \hat{\boldsymbol{\beta}}(\mathbf{e}_{z^*}) =& \underset{\boldsymbol{\beta}}{\text{argmin}} \big[\boldsymbol{\beta}^\mathsf{T} \mathbf{K}^{'} \boldsymbol{\beta} - \zeta \hat{\boldsymbol{\beta}}_{DT}(\mathbf{e}_{z^*})^\mathsf{T} \mathbf{K}^{'} \boldsymbol{\beta} + \boldsymbol{\beta}^T \mathbf{a}\big] \\ & s.t.: \boldsymbol{\beta}^T \mathbf{1} = -1, \hat{\boldsymbol{\beta}}_{DT}^\mathsf{T} \mathbf{K}^{'} \boldsymbol{\beta} > l, -C \leq \boldsymbol{\beta}_i \leq 0; \forall i \\ & C, \zeta, l\text{: hyper-parameters,} \end{aligned} \tag{3.9}$$

where $\hat{\boldsymbol{\beta}}_{DT}$ is the first N elements in $\tilde{\boldsymbol{\beta}}_{DT} \in \mathbb{R}^{N+1}$, $\mathbf{1}$ is an $N \times 1$ vector of ones. The objective function, in Eq 3.9, pushes the classifier of the unseen class to be highly correlated with the domain transfer prediction of the kernel classifier, while putting the points of the seen classes as negative examples. It is not hard to see that Eq 3.9 is a quadratic program in β, which could be solved using any quadratic solver; we used IBM CPLEX. It is worth to mention that, the approach in [46] predicts linear classifiers by solving an optimization problem of size $N + d_X + 1$ variables ($d_X + 1$ linear-classifier parameters and N slack variables); a similar limitation can be found in [61, 150]. In contrast, our objective function in Eq 3.9 solves a quadratic program of only N variables, and predicts a kernel-classifier instead, with fewer parameters. Hence, if very high-dimensional features are used, they will not affect our optimization complexity.

3.5 Distributional Semantic (DS) Kernel for text descriptions

When $\mathcal{E}$ domain is the space of text descriptions, we propose a distributional semantic kernel $g(\cdot, \cdot) = g_{DS}(\cdot, \cdot)$ to define the similarity between two text descriptions . We start by distributional semantic models by [107, 105] to represent the semantic manifold $\mathcal{M}_s$, and a function $vec(\cdot)$ that maps a word to a $K \times 1$ vector in $\mathcal{M}_s$. The main assumption behind this class of distributional semantic model is that similar words share similar context. Mathematically speaking, these models learn a vector for each word w_n, such that $p(w_n|(w_{n-L}, w_{n-L+1}, \cdots, w_{n+L-1}, w_{n+L})$ is maximized over the training corpus, where $2 \times L$ is the context window size. Hence similarity between $vec(w_i)$ and $vec(w_j)$ is high if they co-occurred a lot in context of size $2 \times L$ in the training text-corpus. We normalize all the word vectors to length 1 under L2 norm, i.e., $\|vec(\cdot)\|^2 = 1$.

Let us assume a text description D that we represent by a set of triplets $D = \{(w_l, f_l, vec(w_l)), l = 1 \cdots M\}$, where w_l is a word that occurs in D with frequency f_l and its corresponding word vector is $vec(w_l)$ in $\mathcal{M}_s$. We drop the stop words from D. We define $\mathbf{F} = [f_1, \cdots, f_M]^\mathsf{T}$ and $\mathbf{V} = [vec(w_1), \cdots, vec(w_M)]^\mathsf{T}$, where $\mathbf{F}$ is an $M \times 1$ vector of term frequencies and $\mathbf{V}$ is an $M \times K$ matrix of the corresponding term vectors.

Given two text descriptions D_i and D_j which contains M_1 and M_2 terms respectively. We compute $\mathbf{F}_i$ $(M_i \times 1)$ and $\mathbf{V}_i$ $(M_i \times K)$ for D_i and $\mathbf{F}_j$ $(M_j \times 1)$ and $\mathbf{V}_j$ $(M_j \times K)$ for D_j. Finally $g_{DS}(D_i, D_j)$ is defined as

$$g_{DS}(D_i, D_j) = \mathbf{F}_i^\mathsf{T} \mathbf{V}_i \mathbf{V}_j^\mathsf{T} \mathbf{F}_j \tag{3.10}$$

One advantage of this similarity measure is that it captures semantically related terms. It is not hard to see that the standard Term Frequency (TF) similarity could be thought as a special case of this kernel where $vec(w_l)^\mathsf{T} vec(w_m) = 1$ if $w_l = w_m$, 0 otherwise, i.e., different terms are orthogonal. However, in our case the word vectors are learnt through a distributional semantic model which makes semantically related terms have higher dot product $(vec(w_l)^\mathsf{T} vec(w_m))$.

3.6 Experiments

3.6.1 Datasets and Evaluation Methodology

We validated our approach in a fine-grained setting using two datasets: 1) The UCSD-Birds dataset [160], which consists of 6033 images of 200 classes. 2) The Oxford-Flower dataset [116], which consists of 8189 images of 102 flower categories. Both datasets were amended with class-level text descriptions extracted from different encyclopedias which is the same descriptions used in [46]; see samples in the supplementary materials. We split the datasets to 80% of the classes for training and 20% of the classes for testing, with cross validations. We report multiple metrics while evaluating and comparing our approach to the baselines, detailed as follows

Multiclass Accuracy of Unseen classes (MAU): Under this metric, we aim to evaluate the performance of the unseen classifiers against each others. Firstly, the classifiers of all unseen categories are predicted. Then, an instance x^* is classified to the class $z^* \in Y_{us}$ of maximum confidence for x^* of the predicted classifiers; see Eq 3.2.

AUC: In order to measure the discriminative ability of our predicted one-vs-all classifier for each unseen class, against the seen classes, we report the area under the ROC curve. Since unseen class positive examples are few compared to negative examples, a large accuracy could be achieved even if all unseen points are incorrectly classified. Hence, AUC is a more consistent measure. In this metric, we use the predicted classifier of an unseen class as a binary separator against the seen classes. This measure is computed for each predicted unseen classifier and the average AUC is reported. This is the only measure addressed in [46] to evaluate the unseen classifiers, which is limiting in our opinion.

$|N_{sc}|$ *to* $|N_{sc}+1|$ *Recall:* Under this metric, we aim to check how the learned classifiers of the seen classes confuse the predicted classifiers, when they are involved in a multi-class classification problem of $N_{sc}+1$ classes. We use Eq 3.2 to predict label of an instance x^*, such that the unknown label $y^* \in Y_{sc} \cup l_{us}$, such that l_{us} is the label of the unseen class. We compute the recall under this setting. This metric is computed for each predicted unseen classifier and the average is reported.

3.6.2 Comparisons to Linear Classifier Prediction

We compared our proposed approach to [46], which predicts a linear classifier for zero-shot learning from textual descriptions ($\mathcal{E}$ space in our framework). The aspects of the comparison includes 1) whether the predicted kernelized classifier outperforms the predicted linear classifier 2) whether this behavior is consistent on multiple datasets. We performed the comparison on both Birds and Flower dataset. For these experiments, in our setting, domain $\mathcal{X}$ is the visual domain and domain $\mathcal{E}$ is the textual domain, *i.e.* , the goal is to predict classifiers from pure textual description. We used the same features on the visual domain and the textual domains as [46]. That is, for the visual domain, we used classeme features [154], extracted from images of the Bird and the Flower datasets. Classeme is a 2569-dimensional features, which correspond to confidences of a set of one-vs-all classifiers, pre-trained on images from the web, as explained in [154], not related to either the Bird nor the Flower datasets. The rationale behind using these features in [46] was that they offer a semantic representation. For the textual domain, we used the same textual feature extracted by [46]. In that work, tf-idf (Term-Frequency Inverted Document Frequency)[143] features were extracted from the textual articles were used, followed by a CLSI [169] dimensionality reduction phase.

We denote our DT prediction and one class SVM adjust DT prediction approaches as DT-kernel and SVM-DT-kernel respectively. We compared against the linear classifier prediction by [46]. We also compared against the direct domain transfer [85], which was applied as a baseline in [46] to predict linear classifiers. In our kernel approaches, we used Gaussian rbf-kernel as a similarity measure in $\mathcal{E}$ and $\mathcal{X}$ spaces (*i.e.* $k(d, d') = exp(-\lambda||d - d'||)$).

Recall metric : The recall of our approach is 44.05% for Birds and 40.34% for Flower, while it is 36.56% for Birds and 31.33% for Flower using [46]. This indicates that the predicted classifier is less confused by the classifiers of the seen compared with [46]; see table 3.1 (top part)

MAU metric: It is worth to mention that the multiclass accuracies for the trained seen classifiers are 51.3% and 15.4% using the classeme features on Flower dataset and Birds dataset[1], respectively. Table 3.1 (middle part) shows the average MAU metric over three seen/unseen

[1]Birds dataset is known to be a challenging dataset for fine-grained, even when applied in a regular multiclass setting as it is clear from the 15.4% performance on seen classes

Table 3.1: Recall, MAU, and average AUC on three seen/unseen splits on Flower Dataset and a seen/unseen split on Birds dataset

	Recall-Flower	improvement	**Recall-Birds**	improvement
SVM-DT kernel-rbf	**40.34% (+/- 1.2) %**		**44.05 %**	
Linear Classifier	31.33 (+/- 2.22)%	27.8 %	36.56 %	20.4 %

	MAU-Flower	improvement	**MAU-Birds**	improvement
SVM-DT kernel-rbf	**9.1 (+/- 2.77) %**		**3.4 %**	
DT kernel-rbf	**6.64 (+/- 4.1) %**	37.93 %	**2.95 %**	15.25 %
Linear Classifier	5.93 (+/- 1.48)%	54.36 %	2.62 %	29.77 %
Domain Transfer	5.79 (+/- 2.59)%	58.46 %	2.47 %	37.65 %

	AUC-Flower	improvement	**AUC-Birds**	improvement
SVM-DT kernel-rbf	0.653 (+/- 0.009)		0.61	
DT kernel-rbf	0.623 (+/- 0.01) %	4.7 %	0.57	7.02 %
Linear Classifier	0.658 (+/- 0.034)	- 0.7 %	0.62	-1.61%
Domain Transfer	0.644 (+/- 0.008)	1.28 %	0.56	8.93%

splits for Flower dataset and one split on Birds dataset, respectively. Furthermore, the relative improvements of our SVM-DT-kernel approach is reported against the baselines. On Flower dataset, it is interesting to see that our approach achieved 9.1% MAU, 182% improvement over the random guess performance, by predicting the unseen classifiers using just textual features as privileged information (i.e. $\mathcal{E}$ domain). We also achieved also 13.4%, 268% the random guess performance, in one of the splits (the 9.1% is the average over 3 seen/unseen splits). Similarity on Birds dataset, we achieved 3.4% MAU from text features, 132% the random guess performance (further improved to 224% in next experiments).

AUC metric: Fig 3.2 (top part) shows the ROC curves for our approach on the best predicted unseen classes from the Flower dataset. Fig 3.2 (bottom part) shows the AUC for all the classes on Flower dataset (over three different splits). More results and figures are attached in the supplementary materials. Table 3.1 (bottom part) shows the average AUC on the two datasets, compared to the baselines.

Looking at table 3.1, we can notice that the proposed approach performs marginally similar to the baselines from AUC perspective. However, there is a clear improvement in MAU and Recall metrics. These results show the advantage of predicting classifiers in kernel space. Furthermore, the table shows that our SVM-DT-kernel approach outperforms our DT-kernel

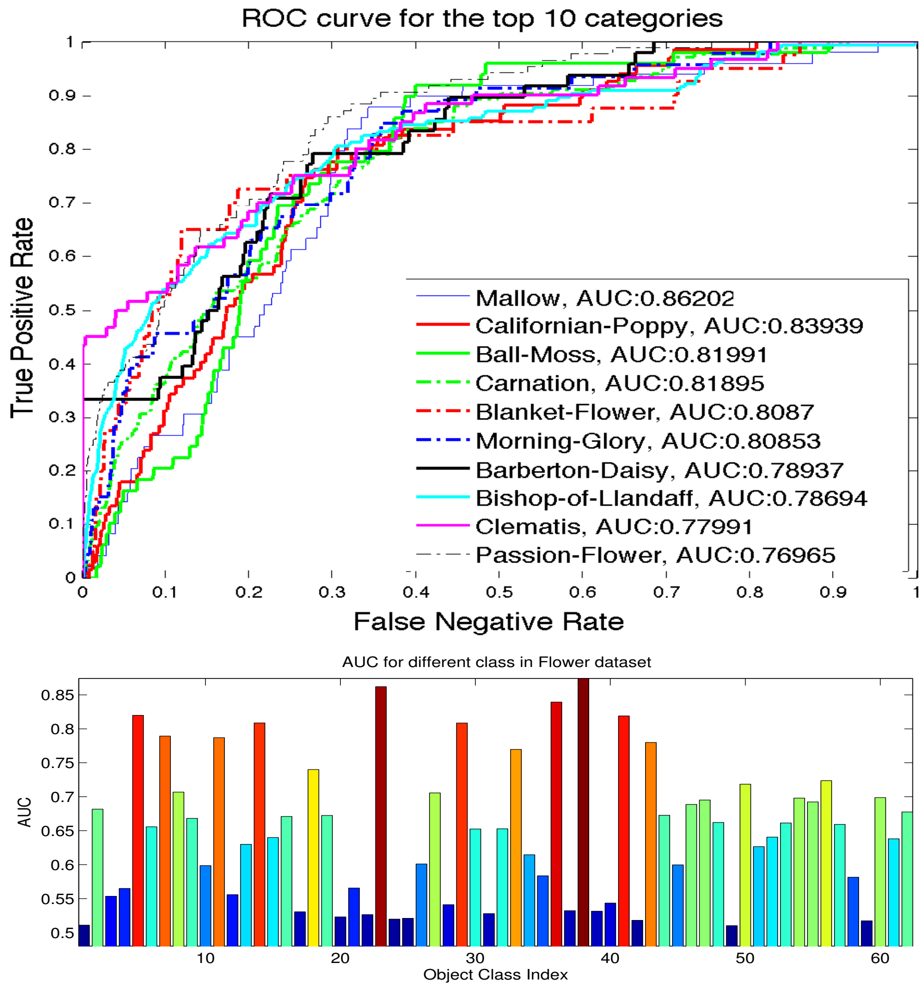


Figure 3.2: AUC of the 62 unseen classifiers the flower data-sets over three different splits (bottom part) and their Top 10 ROC-curves (top part)

model. This indicates the advantage of the class separation, which is adjusted by the SVM-DT-kernel model. More details on the hyper-parameter selection are attached in the supplementary materials.

3.6.3 Multiple Kernel Learning (MKL) Experiment

This experiment shows the added value of proposing a kernelized zero-shot learning approach. We conducted an experiment where the final kernel on the visual domain is produced by Multiple Kernel Learning [66]. For the visual domain, we extracted kernel descriptors for Birds dataset. Kernel descriptors provide a principled way to turn any pixel attribute to patch-level features, and are able to generate rich features from various recognition cues. We specifically

Table 3.2: MAU on a seen-unseen split-Birds Dataset (MKL)

	MAU	improvement
SVM-DT kernel-rbf (text)	**4.10 %**	
Linear Classifier	2.74 %	49.6 %

used four types of kernels introduced by [15] as follows: *Gradient Match Kernels* that captures image variation based on predefined kernels on image gradients. *Color Match Kernel* that describes patch appearance using two kernels on top of RGB and normalized RGB for regular images and intensity for grey images. These kernels capture image variation and visual apperances. For modeling the local shape, *Local Binary Pattern* kernels have been applied.

We computed these kernel descriptors on local image patches with fixed size 16 x 16 sampled densely over a grid with step size 8 in a spatial pyramid setting with four layers. The dense features are vectorized using codebooks of size 1000. This process ended up with a 120,000 dimensional feature for each image (30,000 for each type). Having extracted the four types of descriptors, we compute an rbf kernel matrix for each type separately. We learn the bandwidth parameters for each rbf kernel by cross validation on the seen classes. Then, we generate a new kernel $k_{mkl}(d, d') = \sum_{i=1}^{4} w_i k_i(d, d')$, such that w_i is a weight assigned to each kernel. We learn these weights by applying Bucak's Multiple Kernel Learning algorithm [17]. Then, we applied our approach where the MKL-kernel is used in the visual domain and rbf kernel on the text TFIDF features.

To compare our approach to [46] under this setting, we concatenated all kernel descriptors to end up with 120,000 dimensional feature vector in the visual domain. As highlighted in the approach Sec 3.4, the approach in [46] solves a quadratic program of $N + d_X + 1$ variables for each unseen class. Due to the large dimensionality of data ($d_X = 120,000$), this is not tractable. To make this setting applicable, we reduced the dimensionality of the feature vector into 4000 using PCA. This highlights the benefit of our approach since it does not depend on the dimensionality of the data. Table 3.2 shows MAU for our approach under this setting against [46]. The results show the benefits of having a kernel approach for zero shot learning where kernel methods are applied to improve the performance.

Table 3.3: MAU on a seen-unseen split-Birds Dataset (CNN features, text description)

	MAU	improvement
SVM-DT kernel ($\mathcal{X}$-rbf, $\mathcal{E}$-DS kernel)	**5.35 %**	
SVM-DT kernel ($\mathcal{X}$-rbf, $\mathcal{E}$-rbf on TFIDF)	**4.20 %**	27.3%
Linear Classifier (TFIDF text)	2.65 %	102.0%
[117]	2.3%	132.6%

3.6.4 Multiple Representation Experiment and Distributional Semantic(DS) Kernel

The aim of this experiment is to show that our approach also work on different representations of text and visual domain. In this experiment, we extracted Convolutional Neureal Network(CNN) image features for the Visual domain. We used caffe [75] implementation of [84]. Then, we extracted the sixth activation feature of the CNN since we found it works the best on the standard classification setting. We found this consistent with the results of [34] over different CNN layers. While using TFIDF feature of text description and CNN features for images, we achieved 2.65% for the linear version and 4.2% for the rbf kernel on both text and images. We further improved the performance to 5.35% by using our proposed Distributional Semantic (DS) kernel in the text domain and rbf kernel for images. In this DS experiment, we used the distributional semantic model by [107] trained on GoogleNews corpus (100 billion words) resulting in a vocabulary of size 3 million words, and word vectors of $K = 300$ dimensions. This experiment shows both the value of having a kernel version and also the value of the proposed kernel in our setting. We also applied the zero shot learning approach in [117] which performs worse in our settings; see Table 3.3.

3.6.5 Attributes Experiment

We emphasis that our main goal is not attribute prediction. However, it was interesting for us to see the behavior of our method where side information comes from attributes instead of text. In contrast to attribute-based models, which fully utilize attribute information to build attribute classifiers, our approach do not learn attribute classifiers. In this experiment, our method uses only the first moment of information of the attributes (i.e. the average attribute vector). We

Table 3.4: MAU on a seen-unseen split-Birds Dataset (Attributes)

	MAU	improvement
SVM-DT kernel-rbf	**5.6 %**	
DT kernel-rbf	4.03 %	32.7 %
Lampert DAP	4.8 %	16.6 %

decided to compare to an attribute-based approach from this perspective. In particular, we applied the (DAP) attribute-based model [87, 89], widely adopted in many applications (*e.g.,* [96, 132]), to the Birds dataset. Details weak attribute representation in $\mathcal{E}$ space are attached in the supplementary materials due to space. For visual domain $\mathcal{X}$, we used classeme features in this experiment (like table 3.1 experiment)

An interesting result is that our approach achieved 5.6% MAU (224% the random guess performance); see Table 3.4. In contrast, we get 4.8% multiclass accuracy using DAP approach [87]. In this setting, we also measured the N_{sc} to $N_{sc}+1$ average recall. We found the recall measure is 76.7% for our SVM-DT-kernel, while it is 68.1% on the DAP approach, which reflects better true positive rate (positive class is the unseen one). We find these results interesting, since we achieved it without learning any attribute classifiers, as in [87]. When comparing the results of our approach using attributes (Table 3.4) vs. textual description (Table 3.1)[2] as the privileged information used for prediction, it is clear that the attribute features gives better prediction. This support our hypothesis that the more meaningful the $\mathcal{E}$ domain, the better the performance on $\mathcal{X}$ domain.

3.6.6 Experiments using deep image-sentence similarity

In this experiment, we used a state of the art Model [**?**] for image-sentence similarity by breaking down each text document into sentences and considering it as a positive sentence for all images in the corresponding class. Then we measure the similarities between an image to class by averaging its similarity to all sentences in that class. Images were encoded using VGGNet [148] and sentences were encoded by an RNN with GRU activations [22]. The MAU on

[2]We are refering to the experiment that uses classeme as visual features to have a consistent comparison to here

Birds dataset for this experiments resulted in 3.3% MAU which is better that the Linear Classifier in Table 3.3. However, our kernel method (Eq **??**) over deep features is still performing 2.03% better (i.e. 5.35% MAU).

3.7 Conclusion

We proposed an approach to predict kernel-classifiers of unseen categories textual description of them. We formulated the problem as domain transfer function from the privilege space $\mathcal{E}$ to the visual classification space $\mathcal{X}$, while supporting kernels in both domains. We proposed a one-class SVM adjustment to our domain transfer function to improve the prediction. We validated the performance of our model by several experiments. We applied our approach using different privilege spaces (*i.e.* $\mathcal{E}$ lives in a textual space or an attribute space). We showed the value of proposing a kernelized version by applying kernels generated by Multiple Kernel Learning (MKL) and achieved better results. We also compared our approach with state-of-the-art approaches and interesting findings have been reported.

Part II

Language Guided Video Event Detection

Chapter 4

Zero Shot Event Detection by Multimodal Distributional Semantic Embedding of Videos

We propose a new zero-shot Event Detection method by Multi-modal Distributional Semantic embedding of videos. Our model embeds object and action concepts as well as other available modalities from videos into a distributional semantic space. To our knowledge, this is the first Zero-Shot event detection model that is built on top of distributional semantics and extends it in the following directions: (a) semantic embedding of multimodal information in videos (with focus on the visual modalities), (b) automatically determining relevance of concepts/attributes to a free text query, which could be useful for other applications, and (c) retrieving videos by free text event query (e.g., ”changing a vehicle tire”) based on their content. We embed videos into a distributional semantic space and then measure the similarity between videos and the event query in a free text form. We validated our method on the large TRECVID MED (Multimedia Event Detection) challenge. Using only the event title as a query, our method outperformed the state-of-the-art that uses big descriptions from 12.6% to 13.5% with MAP metric and 0.73 to 0.83 with ROC-AUC metric. It is also an order of magnitude faster.

4.1 Introduction

Every minute, hundreds of hours of video are uploaded to video archival site such as YouTube [68]. Developing methods to automatically understand the events captured in this large volume of videos is necessary and meanwhile challenging. One of the important tasks in this direction is event detection in videos. The main objective of this task is to determine the relevance of a video to an event based on the video content (e.g., feeding an animal, birthday party; see Fig. 4.1). The cues of an event in a video could include visual objects, scene, actions, detected speech (by Automated Speech Recognition(ASR)), detected text (by Optical Character

(a) Grooming an Animal

(1) “brushing dog”, weight = 0.66976

(2) “combing dog”, weight = 0.66419

(3) “clipping nails”, weight = 0.52486

(b) Birthday Party

(1) “cutting cake” concept, weight = 0.7194

(2) “blowing candles” concept, weight = 0.64801

(3) “opening presents” concept, weight = 0.58737

Figure 4.1: Top relevant Concepts from a pre-defined multi-media concept repository and their automatically-assigned weights as a part of our Event Detection method

Recognition (OCR)), and audio concepts (e.g. music and water concepts).

Search and retrieval of videos for arbitrary events using only free-style text and unseen text in particular has been a dream in computational video and multi-media understanding. This is referred as “zero-shot event detection”, because there is no positive exemplar videos to train a detector. Due to the proliferation of videos, especially consumer-generated videos (e.g., YouTube), zero-shot search and retrieval of videos has become an increasingly important problem.

Several research works have been proposed to facilitate performing the zero-shot learning task by establishing an intermediate semantic layer between events or generally categories (i.e., concepts or attributes) and the low-level representation of a multimedia content from the visual perspective. [88] and [52] were the two first to use attribute learning representation for the zero-shot setting for object recognition in still images. Attributes were similarly adopted for recognizing human actions [95]; attributes are generalized and denoted by concepts in this context. Later, [96] proposed Concept Based Event Retrieval (CBER) for videos InTheWild. Even though these methods facilitate zero-shot event detection, they only capture the visual modality and more importantly they assume that the relevant concepts for a query event are manually defined. This manual definition of concepts, also known as semantic query editing, is a tedious task and may be biased due to the limitation of human knowledge. Instead, we aim at automatically generating relevant concepts by leveraging information from distributional semantics.

Recently, several systems were proposed for zero-shot event detection methods [161, 77, 76, 20, 71]. These approaches rely on the whole text description of an event where relevant concepts are specified; see example event descriptions used in these approaches in the Supplementary Materials (SM)[1] (explicitly define the event explication, scene, objects, activities, and audio). In practice, however, we think that typical use of event queries under this setting should be similar to text-search, which is based on few words instead that we model their connection to the multimodal content in videos.

The main question addressed in this paper is how to use an event text query (i.e. just the event title like "birthday party" or "feeding an animal") to retrieve a ranked list of videos based on their content. In contrast to [88, 96], we do not manually assign relevant concepts for a given event query. Instead, we leverage information from a distributional semantic space [107] trained on a large text corpus to embed event queries and videos to the same space, where similarity between both could be directly estimated. Furthermore, we only assume that query comes in the form of an "unstructured" few-keyword query (in contrast to [161, 77, 76]). We abbreviate our method as EDiSE (Event-detection by multimodal Distributional Semantic Embedding of videos).

Contributions. The contributions of this paper can be listed as follows: (1) Studying how to use few-keyword unstructured-text query to detect/retrieve videos based on their multimedia content, which is novel in this setting. We show how relevant concepts to that event query could be automatically retrieved through a distributional semantic space and got assigned a weight associated with the relevance; see Fig. 4.1 "Birthday" and "Grooming an Animal" example events. (2) To the best of our knowledge, our work is the first attempt to model the connection between few keywords and multimodal information in videos by distributional semantics . We study and propose different similarity metrics in the distributional semantic space to enable event retrieval based on (a) concepts, (b) ASR, and (c) OCR in videos. Our unified framework is capable of embedding all of them into the same space; see Fig. 4.2. (3) Our method is also

---

[1] Supplementary Materials (SM) could be found here `https://sites.google.com/site/mhelhoseiny/EDiSE_supp.zip`

very fast, which makes it applicable to both large number of videos and concepts (*i.e.* 26.67 times faster than the state of the art [76]).

4.2 Related Work

Attribute methods for zero-shot learning are based on manually specifying the attributes for each category (e.g., [88, 122]). Other methods focused on attribute discovery [133, 131] and then apply the same mechanism. Recently, several methods were proposed to perform zero shot recognition by representing unstructured text in document terms (*e.g.* [47, 103]) One drawback of the TFIDF [143] in [47] and hardly matching tag terms in [103, 134] is that they do not capture semantically related terms that our model can relate in noisy videos instead of still images. Also, WordNet [111], adopted in [134], does not connect objects with actions (e.g., person blowing candle), making it hard to apply in our setting and heavily depending on predefined information like WordNet.

There has been a recent interest especially in the computational linguistics' community in word-vector representation (e.g., [13]), which captures word semantics based on context. While word-vector representation is not new, recent algorithms (e.g. [107, 105]) enabled learning these vectors from billions of words, which makes them much more semantically accurate. As a result, these models got recently adopted in several tasks including translation [106] and web search [146]. Several computer vision researchers explored using these word-vector representation to perform Zero-Shot learning in the object recognition (e.g. [60, 149, 117]). They embed the object class name into the word-vector semantic space learnt by models like [107]. It is worth mentioning that these zero-shot learning approaches [60, 149] and also the aforementioned work [47] assume that during training, there is a set of training classes and test classes. Hence, they learn a transformation to correlate the information between both domains (textual and visual). In contrast, zero-shot setting of event retrieval rely mainly on the event information without seeing any training events, as assumed in recent zero-shot event retrieval methods (e.g., [25, 77, 161, 96]). Hence, there does not exist seen events to learn such transformation from. Differently, we also model multimodal connection from free text query to video information.

In the context of videos, [161] proposed a method for zero-shot event detection by using the salient words in the whole structured event description, where relevant concept are already defined in the event structured text description; also see Eq. 1 in [161]. Similarly, [25] adopted a Markov-Random-Field language model proposed by [104]. One drawback of this model is that it performs an intensive processing for each new concept. This is since it determines the relevance of the concept to a query event by creating a text document to represent each concept. This document is created by web-querying the concept name and some of its keywords and merging the top retrieved pages. In contrast, our model does not require this step to determine relevance of an event to a query. Once the language model is trained, any concept can be instantly added and captured in our multimodal semantic embedding of videos.

In contrast to both [161] and [25], we focus on retrieving videos only with the event title (i.e., few-words query) and without semantic editing. The key difference is in modeling and embedding concepts to allow zero-shot event retrieval. In [161] and [25], the semantic space is a vector whose dimensionality is the number of the concepts. Our idea is to embed concepts, video information, and the event query into a distributional semantic space whose dimensionality is independent of the number of concepts. This property, together with the semantic properties captured by distributional semantics, feature our approach with two advantages (a) scalability to any concept size. Having new concepts does not affect the representation dimensionality (i.e., in all our experiments concepts, videos, event queries are embedded to M dimensional space; M is few hundreds in our experiments). (b) facilitating automatic determination of relevant concepts given an unstructured short event query: For example, being able to automatically determine that "blowing a candle" concept is a relevant concept to "birthday party" event. [161] and [25] used the complete text description of an event for retrieval that explicitly specifies relevant concepts.

There is a class of models that improve zero-shot Event Detection performance by reranking. Jiang et al. proposed multimodal pseudo relevance feedback [77] and self-paced reranking [76] algorithms. The main assumption behind these models is that all unlabeled test examples are available and the top few examples by a given initial ranking have high top K precision (K $\sim$ 10). This means that reranking algorithms can not update confidence of a video for an event without knowing the confidences of the remaining videos to perform reranking. In contrast, our goal is different which is to directly model the probability of a few-keyword event-query given an arbitrary video. Hence, our work does not require an initial ranking and can compute the conditional probability of a video without any information about other videos. Our method is also 26.67 times faster, as detailed in our experiments.

4.3 Method

4.3.1 Problem Definition

Given an arbitrary event query e and a video v, our objective is to model $p(e|v)$. We start by defining the representation of event query e, the concept set **c**, the video v in our setting.

Event-Query representation e: We use the unstructured event title to represent an event query for concept based retrieval. Our framework also allows additional terms specifically for ASR or OCR based retrieval. While we show retrieval on different modalities, concept based retrieval is our main focus in this work. The few-keyword event query for concept based retrieval is denoted by e_c, while query keywords for OCR and ASR are denoted by e_o and e_a, respectively. Hence, under our setting $e = \{e_c, e_o, e_a\}$.

Concept Set c: We denote the whole concept set in our setting as **c**, which include visual concepts $\mathbf{c}_v$ and audio concepts $\mathbf{c}_d$, i.e., $\mathbf{c} = \{\mathbf{c}_v, \mathbf{c}_d\}$. The visual concepts include object, scene and action concepts. The audio concepts include acoustic related concepts like water sound. We performed an experiment on a set of audio concepts trained on MFCC audio features [26, 97]. However, we found their performance $\approx 1\%$ MAP, and hence we excluded them from our final experiments. Accordingly, our final performance mainly relies on the visual concepts for concept based retrieval; i.e., $\mathbf{c}_d = \emptyset$. We denote each member $c_i \in \mathbf{c}$ as the definition of the i^{th} concept in **c**. c_i is defined by the i^{th} concept's name and optionally some related keywords; see examples in SM. Hence, $\mathbf{c} = \{c_1, \cdots, c_N\}$ is the the set of concept definitions, where N is the number concepts.

Video Representation: For our zero-shot purpose, a video v is defined by three pieces of information, which are video OCR denoted by v_o, video ASR denoted by v_a, and video concept representation denoted by v_c. v_o and v_a are the detected text in OCR and ASR, respectively. We used [115] to extract v_o and [156] to extract v_a. In this paper, we mainly focus on the visual video content, which is the most challenging. The video concept based representation v_c

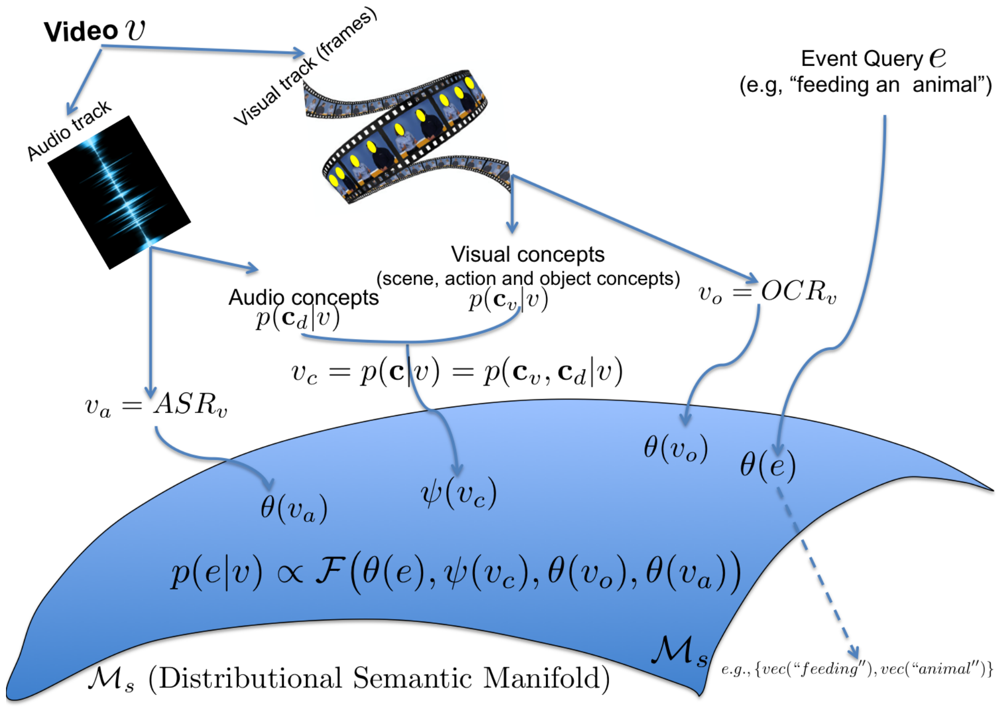


Figure 4.2: EDiSE Approach

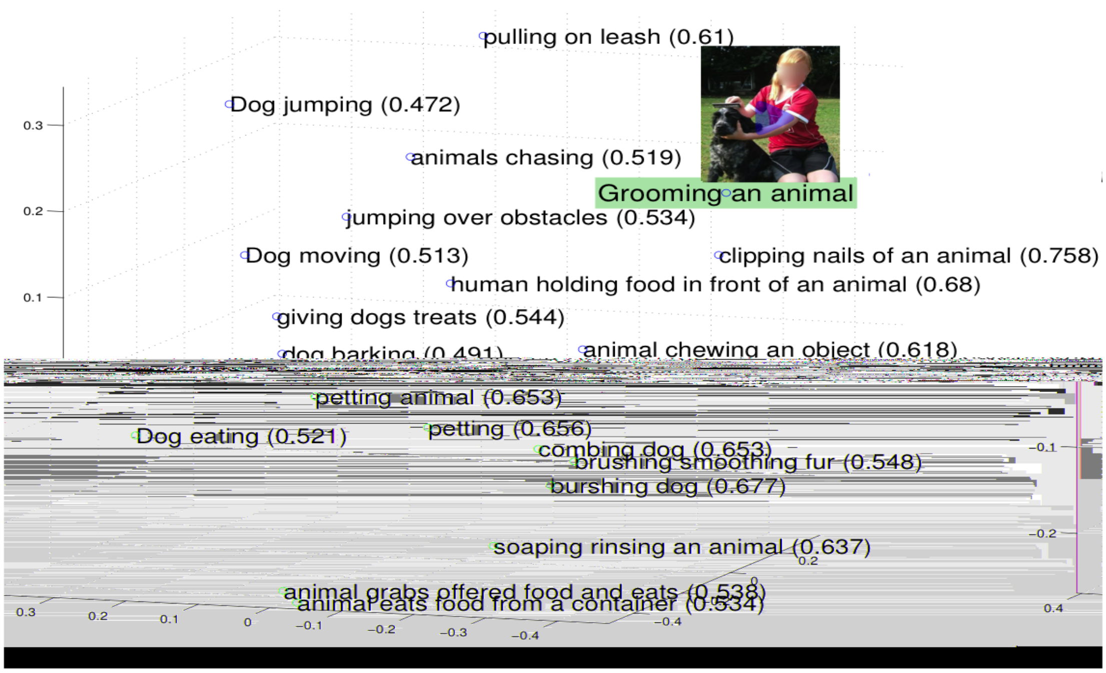


Figure 4.3: PCA visualization in 3D of the "Grooming an Animal" event (in green) and its most 20 relevant concepts in $\mathcal{M}_s$ space using $s_p(\cdot,\cdot)$. The exact $s_p(\theta(\text{"Grooming An Animal"}), \theta(c_i))$ is shown between parenthesis

is defined as

$$v_c = [p(c_1|v), p(c_2|v), \cdots, p(c_N|v)] \tag{4.1}$$

where $p(c_i|v)$ is a conditional probability of concept c_i given video v, detailed later. We denote $p(c_i|v)$ by v_c^i.

In zero-shot event detection setting, we aim at recognizing events in videos without training examples based on its multimedia content including still-image concepts like objects and scenes, action concepts, OCR, and ASR[2]. Given a video $v = \{v_c, v_o, v_a\}$, our goal is to compute $p(e|v)$ by embedding both the event query e and information of video v of different modalities (v_c, v_o, and v_o) into a distributional semantic space, where relevance of v to e could be directly computed; see Fig. 4.2. Specifically, our approach is to model $p(e|v)$ as a function $\mathcal{F}$ of $\theta(e)$, $\psi(v_c)$, $\theta(v_o)$, and $\theta(v_a)$, which are the distributional semantic embedding of e, v_c, v_o, and v_a, respectively

$$p(e|v) \propto \mathcal{F}\big(\theta(e), \psi(v_c), \theta(v_o), \theta(v_a)\big) \tag{4.2}$$

We remove the stop words from e, v_o, v_a before applying the embedding $\theta(\cdot)$. The rest of this section is organized as follows. First, we present the distributional semantic manifold and the embedding function $\theta(\cdot)$ which is applied to e, v_a, v_o, and the concept definitions $\mathbf{c}$ in our framework. Then, we show how to determine automatically relevant concepts to an event title query and assign a relevance weight to them, as illustrated in Fig. 4.1. We present this concept relevance weighting in a separate section since it might be generally useful for other applications. Finally, we present the details of $p(e|v)$ where we derive v_c embedding (*i.e.* $\psi(v_c)$), which is based on the proposed concept relevance weighting.

4.3.2 Distributional Semantic Model & $\theta(\cdot)$ Embedding

We start by the distributional semantic model by [107, 105] to train our semantic manifold. We denote the trained semantic manifold by $\mathcal{M}_s$, and the vectorization function that maps a word to $\mathcal{M}_s$ space as $vec(\cdot)$. We denote the dimensionality of the real vector returned from $vec(\cdot)$ by M. These models learn a vector for each word w_n, such that $p(w_n|(w_{i-L}, w_{i-L+1}, \cdots, w_{i+L-1}, w_{i+L})$ is maximized over the training corpus; $2 \times L$ is the context window size. Hence similarity

[2] Note that OCR and ASR are not concepts. They are rather detected text in video frames and speech

between $vec(w_i)$ and $vec(w_j)$ is high if they co-occurred a lot in context of size $2 \times L$ in the training text-corpus (i.e., semantically similar words share similar context). Based on the trained $\mathcal{M}_s$ space, we define how to embed the event query e, and **c**. Each of e_c, e_a, and e_o is set of one or more words. Each of these words can be directly embedded into $\mathcal{M}_s$ manifold by $vec(\cdot)$ function. Accordingly, we represent these sets of word vectors for each of e_c, e_a, and e_o as $\theta(e_c)$, $\theta(e_a)$, and $\theta(e_o)$. We denote $\{\theta(e_c), \theta(e_a), \theta(e_o)\}$ by $\theta(e)$. Regarding embedding of **c**, each concept $c^* \in \mathbf{c}$ is defined by its name and optionally some related keywords. Hence, the corresponding word vectors are then used to define $\theta(c^*)$ in $\mathcal{M}_s$ space.

4.3.3 Relevance of Concepts to Event Query

Let us define a similarity function between $\theta(c^*)$ and $\theta(e_c)$ as $s(\theta(e_c), \theta(c^*))$. We propose two functions to measure the similarity between $\theta(e_c)$ and $\theta(c^*)$. The first one is inspired by an example by [107] to show the quality of their language model, where they indicated that vec("king") + vec("woman") − vec("man") is closest to vec("queen"). Accordingly, we define a version of $s(X, Y)$, where the sets X and Y are firstly pooled by the sum operation; we denote the sum pooling operation on a set by an overline. For instance, $\overline{X} = \sum_i x_i$ and $\overline{Y} = \sum_i y_j$, where x_i and y_j are the word vectors of the i^{th} element in X and the j^{th} element in Y, respectively. Then, cosine similarity between $\overline{X}$ and $\overline{Y}$ is computed. We denote this version as $s_p(\cdot, \cdot)$; see Eq. 4.3. Fig. 4.3 shows how $s_p(\cdot, \cdot)$ could be used to retrieve the top 20 concepts relevant to θ("Grooming An Animal") in $\mathcal{M}_s$ space. The figure also shows embedding of the query and the relevant concept sets in 3D PCA visualization. $\theta(e_c =$"Grooming An Animal") and each of $\theta(c_i)$ for the most relevant 20 concepts are represented by their corresponding pooled vectors ($\overline{\theta(e_c)}$ and $\overline{\theta(c_i)}) \forall i$), normalized to unit length under L2 norm. Another idea is to define $s(X, Y)$ as a similarity function between the X and Y sets. For robustness [152], we used percentile-based Hausdorff point set metric, where similarity between each pair of points is computed by the cosine similarity. We denote this version by $s_t(X, Y)$; see Eq. 4.3. We used $l = 50\%$ (i.e., median).

$$s_p(X, Y) = \frac{(\sum_i x_i)^\intercal (\sum_j y_j)}{\|\sum_i x_i\| \|\sum_j y_j\|} = \frac{\overline{X}^\intercal \overline{Y}}{\|\overline{X}\| \|\overline{Y}\|} \tag{4.3}$$

$$s_t(X, Y) = min\{\overset{l\%}{\underset{j}{min}}\, \underset{i}{max} \frac{x_i^\intercal y_j}{\|x_i\| \|\|y_j\|}, \overset{l\%}{\underset{i}{min}}\, \underset{j}{max} \frac{x_i^\intercal y_j}{\|x_i\| \|y_j\|}\}$$

4.3.4 Event Detection $p(e|v)$

In practice, we decomposed $p(e|v)$ into $p(e_c|v)$, $p(e_o|v)$, $p(e_a|v)$, which makes the problem reduces to deriving $p(e_c|v)$ (concept based retrieval), $p(e_o|v)$ (OCR based retrieval), and $p(e_a|v)$ (ASR based retrieval) under $\mathcal{M}_s$. We start by $p(e_c|v)$ then we will how later in this section how $p(e_o|v)$, and $p(e_a|v)$ could be estimated.

Estimating $p(e_c|v)$ **:** In our work, concepts are linguistic meanings that have corresponding detection functions given the video v. From Fig. 4.3, $\mathcal{M}_s$ space could be viewed as a space of meanings captured by a training text-corpus, where only sparse points in that space has a corresponding visual detection functions given v, which are the concepts $\mathbf{c}$ (e.g., "blowing a candle"). For zero shot event detection, we aim at exploiting these sparse points by the information captured by $s(\theta(e_c), \theta(c^i \in \mathbf{c}))$ in $\mathcal{M}_s$ space. We derive $p(e_c|v)$ from probabilistic perspective starting from marginalizing $p(e_c|v)$ over the concept set $\mathbf{c}$

$$p(e_c|v) \propto \sum_{c_i} p(e_c|c_i)p(c_i|v) \propto \sum_{c_i} s(\theta(e_c), \theta(c_i))v_c^i \tag{4.4}$$

where $p(e|c_i)\forall i$ are assumed to be proportional to $s(\theta(e_c), \theta(c_i))$ in our framework. From semantic embedding perspective, each video v is embedded into $\mathcal{M}_s$ by the set $\psi(v_c) = \{\theta_v(c_i) = v_c^i\theta(c_i), \forall c_i \in \mathbf{c}\}$, where $v_c^i\theta(c_i)$ is a set of the same points in $\theta(c_i)$ scaled with v_c^i; $\psi(v_c)$ could be then directly compared with $\theta(e_c)$; see Eq. 4.5

$$\begin{aligned} p(e_c|v) &\propto \sum_{c_i} s(\theta(e_c), \theta(c_i))v_c^i \\ &\propto s'(\theta(e_c), \psi(v_c) = \{\theta_v(c_i), \forall c_i \in \mathbf{c}\}) \end{aligned} \tag{4.5}$$

where $s'(\theta(e_c), \psi(p(\mathbf{c}|v)) = \sum_i s(\theta(e_c), \theta_v(c_i))$ and $s(\cdot,\cdot)$ could be replaced by $s_p(\cdot,\cdot)$, $s_t(\cdot,\cdot)$, or any other measure in $\mathcal{M}_s$ space. An interesting observation is that when $s_p(\cdot,\cdot)$ is chosen, $p(e_c|v) \propto \frac{\overline{\theta(e_c)}^T}{\|\theta e_c\|} \left(\sum_i \frac{\overline{\theta(c_i)}}{\|\theta c_i\|} v_c^i\right)$ which is a direct similarity between $\overline{\theta(e_c)}$ representing the query and the embedding of $\psi(v_c)$ as $\sum_i \frac{\overline{\theta(c_i)}}{\|\theta c_i\|} v_c^i$; see proof in Appendix A. $s_p(\cdot,\cdot)$ performs consistently better than $s_t(\cdot,\cdot)$ in our experiments. In practice, we only include $\theta_v(c_i)$ in $\psi(v_c)$ such that c_i is among the top R concepts with highest $p(e_c|c_i)$. This is assuming that the remaining concepts are assigned $p(e_c|c_i) = 0$ which makes those items vanish; we used R=5. Hence, only a few concept detectors needs to be computed for on v which is a computational advantage.

Estimating $p(e_o|v)$ **and** $p(e_a|v)$ **:** Both v_o and v_a can be directly embedded into $\mathcal{M}_s$ since they are sets of words. Hence, we can model $p(e_o|v)$ and $p(e_a|v)$ as follows

$$p(e_o|v) \propto s_d(\theta(e_o), \theta(v_o)), p(e_a|v) \propto s_d(\theta(e_a)\ ,\ \theta(v_a)) \tag{4.6}$$

where $s_d(X, Y) = \sum_{ij} x_i^T y_j$. We found this similarity function more appropriate for ASR/OCR text since they normally contains more text compared to concept definition. We also exploited an interesting property in $\mathcal{M}_s$ that nearest words to an arbitrary point can be retrieved. Hence, we automatically augment e_a and e_o with the nearest words to the event title in $\mathcal{M}_s$ using cosine similarity before retrieval. We found this trick effective in practice since it automatically retrieve relevant words that might appear in v_o or v_a.

Fusion: We fuse $p(e_c|v)$, $p(e_o|v)$, and $p(e_a|v)$ by weighted geometric mean with focus on visual concepts, i.e. $p(e|v) = \sqrt[w+1]{p(e_c|v)^w \sqrt{p(e_o|v)p(e_a|v))}}$; $w = 6$. $p(e_c|v)$, $p(e_c|v)$, and $p(e_c|v)$ involves the similarity between $\theta(e)$ and each of $\psi(v_c)$, $\theta(v_o)$, and $\theta(v_a)$, leading to Eq. 4.2 view.

4.4 Visual Concept Detection functions ($p(\mathbf{c}|v)$)

We leverage the information from three types of visual concepts in $\mathbf{c}_v$: object concepts $\mathbf{c}_o$, action concepts $\mathbf{c}_a$, and scene concepts $\mathbf{c}_s$. Hence, $\mathbf{c} = \mathbf{c}_v = \{\mathbf{c}_o \cup \mathbf{c}_a \cup \mathbf{c}_s\}$; the list of concepts are attached in SM. We define object and scene concept probabilities per video frame, and action concepts per video chunks. The rest of this section summarizes the concept detection for objects and scenes per frame f, and action concepts per video chunk u. Then, we show how they can be reduced to video level probabilities. Fig. 4.4 shows example high confidence concepts in a "Birthday Party" video.

Object Concepts $p(o_i|f), o_i \in \mathbf{c}_o$**:** We involve 1000 Overfeat [145] object concept detectors which maps to 1000-ImageNet categories. We also adopt the concept detectors of face, car and person from a publicity available detector (i.e., [56])

Scene Concepts $p(s_i|f), s_i \in \mathbf{c}_s$**:** We represented scene concepts ($p(s_i|f)$) as bag of word representation on static features (i.e., SIFT [99] and HOG [24]) with 10000 codebooks. We used TRECVID 500 SIN concepts concepts, including scene categories like "city" and "hall" way; these concepts are provided by provided by TRECVID2011 SIN track.

Action Concepts $p(a_i|u), a_i \in \mathbf{c}_a$**:** We use both manually annotated (i.e. strongly supervised) and automatically annotated (i.e. weekly supervised) concepts; detailed in SM. We have ~500 action concepts; please refer to [96] for the action concept detection method that we adopt.

Video level concept probabilities $p(\mathbf{c}|v)$

We represent probabilities of the $\mathbf{c}_v$ set given a video v by a pooling operation over the the chunks or the frames of the videos similar to [96]. In our experiments, we evaluated both max and average pooling. Specifically, $p(o_i|v) = \rho(\{p(o_i|f_k), f_k \in v\}), p(s_l|v) = \rho(\{p(s_i|f_k), f_k \in v\}), p(a_k|v) = \rho(\{p(a_i|u_k), u_k \in v\}$, where $p(o_i|v)$ and $p(s_l|v)$ are the video level probabilities of for the i^{th} object and the l^{th} scene concepts respectively, pooled over frames $f_k \in v$. $\{f_k \in v\}$ are selected every M frames in v (M= 250). $p(a_k|v)$ is the video level probability of the k^{th} action concept, pooled over a set of video chunks $\{u_k \in v\}$. The chunk size is set to the mean chunk length of all concept training chunks. Finally, ρ is the pooling function. We denote average and max pooling as $\rho_a(\cdot)$ and $\rho_m(\cdot)$, respectively.

4.5 EDiSE Computational Performance Benefits

Here we discuss the computational complexity of concept based EDISE, and ASR/OCR based EDiSE. The fusion part is negligible since it is constant time.

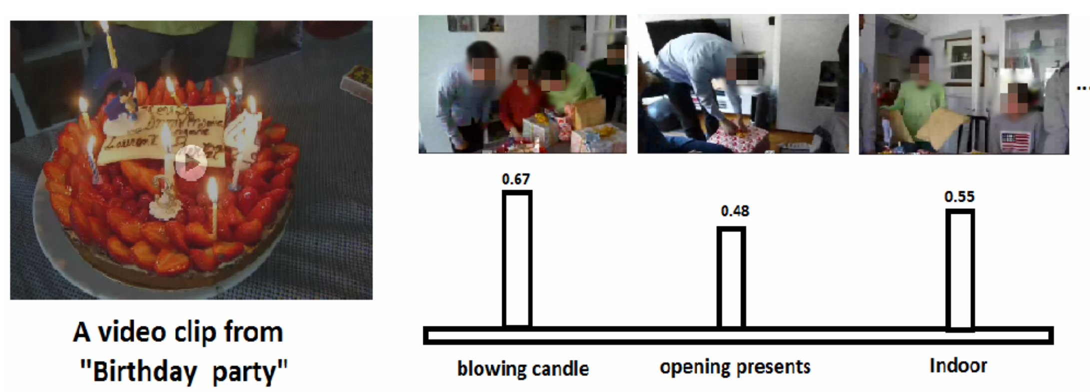


Figure 4.4: Concept probabilities from videos ($p(\mathbf{c}|v)$)

Table 4.1: MED2013 MAP performance on four concept sets (event title query)

TRECVID MED 2013	Ours-Gnews				Ours-Wiki				(Dalton etal, 2013)
	$\rho_m(\cdot)$		$\rho_a(\cdot)$		$\rho_m(\cdot)$		$\rho_a(\cdot)$		
	$s_p(\cdot,\cdot)$	$s_t(\cdot,\cdot)$	$s_p(\cdot,\cdot)$	$s_t(\cdot,\cdot)$	$s_p(\cdot,\cdot)$	$s_t(\cdot,\cdot)$	$s_p(\cdot,\cdot)$	$s_t(\cdot,\cdot)$	
Concepts G1 (152 concepts)	**4.29**	3.94%	2.39%	2.38%	3.14%	2.13%	1.85%	1.70%	2.57%
Concepts G2 (101 concepts)	**1.74**	1.20	1.56%	1.20%	1.09%	0.96%	0.66%	0.60%	1.17%
Concepts G3 (60 concepts)	**1.72**	1.33%	1.28%	1.16%	1.21%	0.88%	0.88%	0.74%	1.54%
Concepts G4 (56 concepts)	**1.22**	0.95	0.84%	0.69%	0.87%	0.76%	0.67%	0.56%	0.83%

Table 4.2: MED2013 full concept set MAP Performance (auto-weighted versus manually-weighted concepts)

Ours (auto-weighted))	(Dalton etal,13)(auto-weighted)	(Dalton etal,13) (manually-weighted)	Overfeat
8.36%	3.40%	7.4%	2.43%

SUN	Object Rank	Classeme	CD^{DT}	$WSC^{D-SIFT}_{YouTube}$
0.48%	0.77%	0.84%	2.28%	3.48%

4.5.1 Concept based EDiSE

The computational complexity for computing $p(e_c|v)$ is mainly linear in the number of videos, denoted by $|V|$. We here detail why computational complexity of $p(e_c|v)$ is almost constant and hence video retrieval is almost $O(|V|)$.

From Eq. 4.5, $p(e_c|v)$ has a computational complexity of $O(N \cdot Q)$ for on e video, where Q is the computational complexity of computing $s(\cdot,\cdot)$ and N is the number of concepts. We detail next the computational complexity of $s_p(\cdot,\cdot)$ and $s_t(\cdot,\cdot)$ for the whole set of videos $|V|$.

Complexity of $p(e_c|v)$ for $s_p(\cdot,\cdot)$

Let's assume that there $\theta(e_c)$ set has $|e_c|$ terms and $\theta(c_i)$ has $|c_i|$ terms. Then, the computational complexity of $s_p(\theta(e_c), \theta(c_i))$ is $O(M(|e_c| + |c_i|)$. $|c_i|$ and and $|e_c|$ are usually few terms in our case (< 10). Hence the computational complexity of $s_p(\theta(e_c), \theta(c_i))$ is $O(M)$, where M is the dimensionality of the word vectors. In our experiments $M = 300$. Given the complexity of $s_p(\theta(e_c), \theta(c_i))$, the computational complexity of $p(e_c|v)$ will be $O(N \cdot M)$, where N is the number of concepts. Hence, the computational complexity for computing $p(e_c|v)$ for $|V|$ videos is $O(|V| \cdot N \cdot M)$. However, for a given event, only few concepts are relevant, which are computed based on $s_p(\theta(e_c), \theta(c_i))$ and only few concepts 5 in our case are sufficient for event zero shot retrieval, retrieved by Nearest Neighbor search of $c_i \in \mathbf{c}$ that is close the e_c. Hence the computational complexity reduced to $O(|V| \cdot M)$, $M = 300$ for the GoogleNews word2vec model that we used. Hence, the computational complexity for $|V|$ videos is basically linear $O(|V|)$, given M is a constant and $M << |V|$.

Complexity of $p(e_c|v)$ for $s_t(\cdot,\cdot)$

The previous argument applies here in all elements except the complexity of the similarity function $s_t(\theta(e_c), \theta(c_i))$, which is $O(M(|e_c| \cdot |c_i|)$. Assuming that $|e_c| \cdot |c_i|$ is bounded by a constant, then the complexity of $|V|$ videos is also $O(|V| \cdot M)$, but with a bigger constant compared to $s_p(\cdot,\cdot)$ (linear in $|V|$ for constant $M << |V|$).

4.5.2 ASR/OCR based EDiSE

The computational complexity of $s_d(\theta(e_o), \theta(v_o))$ and $s_d(\theta(e_a), \theta(v_a))$ are $O(|e_o| \cdot |v_o| \cdot M)$ and $O(|e_a| \cdot |v_a| \cdot M)$, respectively. There is no concepts for ASR/OCR based retrieval. Hence, the computational complexity of $p(e_o, v)$ and $p(e_a|v)$ are $O(|V| \cdot |e_o| \cdot |v_o| \cdot M)$ and $O(|V| \cdot |e_a| \cdot |v_a| \cdot M)$, respectively. Since $|e_o| \ll |V|$, $|v_o| \ll |V|$, $|e_a| \ll |V|$, $|v_a| \ll |V|$, and $M \ll |V|$, the dominating factor in the complexity for both $p(e_o, v)$ and $p(e_a|v)$ will be $|V|$.

4.6 Experiments

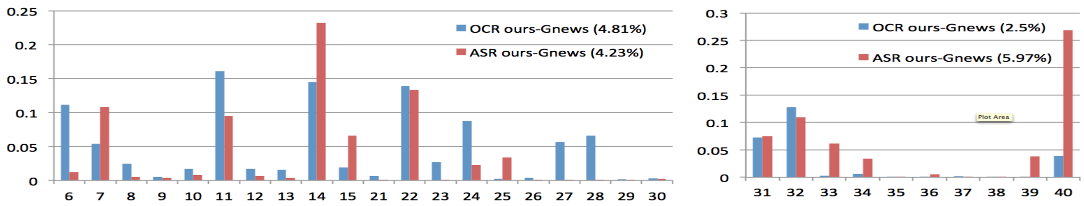
We evaluated our method on the large TRECVID MED [57]. We show the MAP (Mean Average Precision) and ROC AUC performance of the designated MEDTest set [57], containing more than 25,000 videos. Unless otherwise mentioned, our results are on TRECVID MED2013. There are two distributional semantic models in our experiments, trained on Wikipedia and GoogleNews using [107]. The Wikipedia model got trained on 1 billion words resulting in a vocabulary of size of≈120,000 words and word vectors of 250 dimensions. The GoogleNews model got trained on 100 billion words resulting in a vocabulary of size 3 million words and word vectors of 300 dimensions. The objective of having two models is to compare how well our EDiSE method performs depending on the size of the training corpus, used to train the language model. In the rest of this section, we present Concepts, OCR, ASR, and fusion results.

4.6.1 Concept based Retrieval

All the results in this section were generated by automatically retrieved concepts using only the event title. We start by comparing different settings of our method against [25]. We used the language model in [25] for concept based retrieval to rank the concepts. This indicates that $p(e|c_i)$ in Eq. 4.4 is computed by the language model in [104] as adopted in [25], that we compare with under exactly the same setting. For our model, we evaluated the two pooling operations $\rho_m(\cdot)$ and $\rho_p(\cdot)$ and also the two different similarity measures on $\mathcal{M}_s$ space $s_p(\cdot,\cdot)$ and $s_t(\cdot,\cdot)$. Furthermore, we evaluated the methods on both Wikipedia and GNews language models. In order to have conclusive experiments on these eight settings of our model compared to [25], we performed all of them on the four different sets of concepts (i.e. each has the same concept detectors; completely consistent comparison); see Table 4.1. Details about these concept sets are attached in SM.

There are a number of observations. (1) using GNews (the bigger text corpus) language model is consistently better than using the Wikipedia language model. This indicates when the word embedding model is trained with a bigger text corpus, it captures more semantics and hence more accurate in our setting. (2) max pooling $\rho_m(\cdot)$ behaves consistently better than average pooling $\rho_a(\cdot)$. (3) $s_p(\cdot,\cdot)$ similarity measure is consistently better than $s_t(\cdot,\cdot)$, which we see very interesting since this indicates that our hypothesis of using the vector operations on $\mathcal{M}_s$ manifold better represent $p(e|c_i)$. Hence, we recommend finally to use the model trained on the larger corpus, $\rho_m(\cdot)$ for concept pooling, and use $s_p(\cdot,\cdot)$ to measure the performance on $\mathcal{M}_s$ manifold. (4) our model's final setting is consistently better than [25]. The final MED13 ROC AUC performance is 0.834. MAP for MED13 Events 31 to 40 (E31:40) is 5.97%. Detailed figures are attached in SM.

Our next experiment shows the final MAP performance using the recommended setting for our framework on the whole set of concepts, detailed earlier and in SM. Table 4.2 shows our final performance compared with [25] on the same concept detectors. It is not hard to see that our method performs more than double the MAP performance of [25] under the same concept set. Even when manual semantic editing is applied to [25], our performance is still better without semantic editing. We also show the performance on the same events of different concepts (i.e. SUN [123], Object Rank [92], Classeme [155]), and the best performing concepts in [161] (i.e., CD^{DT}, $WSC^{D-SIFT}_{YouTube}$). These numbers are as reported in [161]. The results indicate the value of our concepts and approach compared to [161] and their concepts. We also report our performance using Overfeat concepts only to retrieve videos for the same events.



(a) MED2013 (b) MED2014 (E31:40)

Figure 4.5: ASR & OCR AP Performance (Google News)

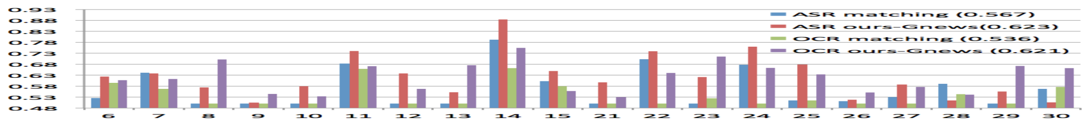


Figure 4.6: ASR & OCR AUCs on MED2013: Ours (GoogleNews) vs keyword Matching (the same query)

This shows the value of involving action and scene concepts compared to only still image concepts like Overfeat for zero-shot event detection. We highlight that the results in [161] uses the whole event description which explicitly includes names of relevant concepts.

4.6.2 ASR and OCR based Retrieval

First, we compared our OCR and ASR retrieval trained on both Wikipedia and GoogleNews language model. Table 4.3 shows that the GoogleNews model MED13 MAP is better than the Wikipedia Model MAP in both ASR and OCR, which is consistent with our concept retrieval results. Figure 4.5 shows the GoogleNews MED13 AP per event for both OCR and ASR. We further show our AP performance on MED14 events 31 to 40 in Fig. 4.5.

In order to show the value our semantic modeling, we computed the performance of string matching method as a baseline, which basically increment the score for every exact match in

Table 4.3: ASR & OCR Retrieval MAP on $\mathcal{M}_s$ using GNews, Wikipedia, and using word matching

	GNews MED2013	Wiki MED2013	matching MED2013
OCR	**4.81%**	3.85%	1.8%
ASR	**4.23%**	1.50%	3.77%

Table 4.4: ASR & OCR MAP performance using GNews corpus compared to [161](prefix E indicates Event)

	MED13	MED13 (word) [161]	MED14(E31:40)
OCR	**4.81%**	4.30%	2.5%

	MED13	MED13 [161]	MED13 [161]-expansion	MED14 (E31:40)
ASR	**4.23%**	3.27%	3.66%	5.97%

the the detected text to words in the query. While, both our model and the matching model use the same query words and ASR/OCR detection, semantic properties captured by $\mathcal{M}_s$ boosts the performance compared to string matching; see table 4.3. This is since semantically relevant terms to the query have a high cosine similarity in $\mathcal{M}_s$ (i.e., high $vec(w_i)^\mathsf{T} vec(w_j)$ if w_i is semantically related to w_j). On the other hand, hard matching basically assumes that $vec(w_i)^\mathsf{T} vec(w_j) = 1$ if$w_i = w_j$, 0 otherwise. We also computed the ROC AUC metric for our method and the hard matching method on ASR and OCR; see Fig. 4.6. For ASR, average AUC is 0.623 for ours and 0.567 for Matching (9.9% gain). For OCR, average AUC is 0.621 for ours and 0.53 for Matching (17.1% gain). We report our GNews model results compared with [161] to indicate that, we achieve state-of-the-art MED13 MAP performance or even better for ASR/OCR; see table 4.4. The table also shows our ASR&OCR MED14 (E31:40) MAP.

4.6.3 Fusion Experiments and Related Systems

In table 4.5, we start by presenting a summary of our earlier ASR/OCR results on MED13 Test. Comparing OCR and ASR performances to Concepts performance, it is not hard to see that OCR/ASR have much lower average AUC zero-shot performance compared to concepts which are visual in our work. This indicates that OCR/ASR produces much higher false negatives compared to visual concepts. When we fused our all OCR and ASR confidences, we achieved 10.7% MAP performance, however, the average AUC performance is as low as 0.67. We achieved lower MAP for our concepts 8.36% MAP but the average AUC performance is as high as 0.834. This indicates that measuring retrieval performance on MAP performance only is not informative, so one approach might achieve a high MAP but lower average AUC and vice versa. We further achieved the best performance of our system by fusing all Concepts, OCR, and ASR to achieve 13.1% MAP and 0.830 average AUC. We found our system achieves better than the state of the art system [161] 4.0% gain in MAP, but significantly in average AUC; see 13.6% gain to [161] in table 4.5.

We also discuss CPRF [165], MMPRF [77], and SPaR [76] reranking systems in contrast to our system that does not involve reranking. The initial retrieval performance is 3.9% MAP without reranking. Interestingly, we achieved a performance of 13.1% MAP also without reranking. The reranking methods assumes high top 5-10 precision of the initial ranking and that all test videos are available. Without any of these assumptions, our system without reranking performs 6.7%, 3.0%, and 0.2% better than CPRF [165], MMPRF [77], and SPaR [76] re-ranking systems; see table 4.5. Unfortunately, ROC AUC performances are not available for these method to compare with. Regarding efficiency, given v_c representation of videos, our concept retrieval experiment on our whole concept set it takes ≈270 seconds on a 16 cores Intel Xeon processor (64GB RAM) to the retrieval task on 20 events altogether. This is more than the time that SPaR [76] takes to rerank one event on an Intel Xeon processor(16GB RAM); see [76]. Since, we detect the MED13 events in ≈270 given v_c representation of videos and as reported in [76], their average detection time per event for MED13 is ≈ 5 minutes assuming feature representation of videos (i.e., 360 seconds per event = 7200 seconds per 20 events). This indicates that our system is 26.67X faster than [76] in MED13 detection. Finally, when we applied SPaR on our output as an initial ranking, we found that it improves MAP (from 13.1% to 13.5%) but hurts ROC AUC (from 0.83 to 0.79). This indicates that reranking has a limited/harmful effect on the performance of our method. We think is since our method already achieve a high performance without re-ranking; see SM for details about the features in this experiment.

4.7 Conclusion

We proposed a method for zero shot event detection by distributional semantic embedding of video modalities and with only event title query. By fusing all modalities, our method outperformed the state of the art on the challenging TRECVID MED benchmark. Based on this notion, we also showed how to automatically determine relevance of concepts to an event based on the distributional semantic space.

Acknowledgements. This work has been supported by the Intelligence Advanced Research Projects Activity (IARPA) via Department of Interior National Business Center contract number D11-PC20066. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of IARPA,

Table 4.5: Fusion Experiments and Comparison to State of the Art Systems

Method	MAP	AUC
Our Concept retrieval (event title query)	8.36%	0.834
Concept retrieval (Dalton etal, 2013) (event title query)	3.4 %	-
Concept retrieval (Dalton etal, 2013) (after manually specifying concepts)	7.4%	-
Our ASR GNews	4.81%	0.623
Our OCR GNews	4.23%	0.621
Our ASR Matching	2.77%	0.567
Our OCR Matching	1.8%	0.536
Our ASR and OCR all fused	10.6	0.670
Our Full (Concepts+ASR+OCR) (No reranking)	**13.1%**	**0.830**
Our Full + SPaR reranking [76]	**13.5%**	**0.790**
Full system [161]	12.6	0.730
Reranking Systems		
Without Reranking [77]	3.9%	-
CPRF [165]	6.4%	-
Full Reranking system MMPRF[77]	10.1%	-
Full Reranking system SPaR[76]	12.9%	-

DOI/NBC, or the U.S. Government. This work is also partially funded by NSF-USA award # 1409683.

Appendix A: Proof $p(e_c|v)$ for $s(\cdot,\cdot) = s_p(\cdot,\cdot)$

We start by Eq. 4.5 while replacing $s(\cdot,\cdot)$ as $s_p(\cdot,\cdot)$.

$$\begin{aligned} p(e_c|v) &\propto \sum_i s_p(\theta(e_c), \theta(c_i)) p(c_i|v) \\ &\propto \sum_i \frac{\overline{\theta(e_c)}^T \overline{\theta(c_i)}}{\|\theta e_c\| \|\theta c_i\|} v_c^i \propto \frac{\overline{\theta(e_c)}^T}{\|\theta e_c\|} \Big(\sum_i \frac{\overline{\theta(c_i)}}{\|\theta c_i\|} v_c^i \Big) \end{aligned} \tag{4.7}$$

which is the dot product between $\frac{\overline{\theta(e_c)}^T}{\|\theta e_c\|}$ representing the event embedding, and $\sum_i \frac{\overline{\theta(c_i)}}{\|\theta c_i\|} v_c^i$ representing the video embedding, which is a function of $\psi(v_c^i) = \{\theta_v(c_i) = \theta(c_i) v_c^i\}$. This equation clarifies our notion of distributional semantic embedding of videos and relating it to event title

Part III

Language Guided Gaining of Visual Knowledge

Chapter 5

Sherlock: Scalable Fact Learning in Images

We study scalable and uniform understanding of facts in images. Existing visual recognition systems are typically modeled differently for each fact type such as objects, actions, and interactions. We propose a setting where all these facts can be modeled simultaneously with a capacity to understand unbounded number of facts in a structured way. The training data comes as structured facts in images, including (1) objects (e.g., <boy>), (2) attributes (e.g., <boy, tall>), (3) actions (e.g., <boy, playing>), and (4) interactions (e.g., <boy, riding, a horse >). Each fact has a semantic language view (e.g., < boy, playing>) and a vBEGIN:VCARD VERSION:3.0 PRODID:-//Apple Inc.//Mac OS X 10.9.5//EN N:Yang;Yi;;; FN:Yi Yang EMAIL;type=INTERNET;type=pref:yangyi05@baidu.com X-ABUID:B749CF42-C614-4271-86A9-4852FFDDD8E9:ABPerson END:VCARD We show that learning visual facts in a structured way enables not only a uniform but also generalizable visual understanding. We propose and investigate recent and strong approaches from the multiview learning literature and also introduce two learning representation models as potential baselines. We applied the investigated methods on several datasets that we augmented with structured facts and a large scale dataset of more than 202,000 facts and 814,000 images. Our experiments show the advantage of relating facts by the structure by the proposed models compared to the designed baselines on bidirectional fact retrieval.

5.1 Introduction

Despite recent significant advances in recognition, image captioning, and visual question answering (VQA), there is still a large gap between humans and machines in the deep image understanding of objects, their attributes, actions, and interactions with one another. The human visual system is able to efficiently gain visual knowledge by learning different types of facts in a never ending way from many or few examples, aided by the ability to generalize from other known facts with related structure. We believe that the most effective and fastest way to close this gap are with methods that possess that following key characteristics:

- **Uniformity**: The method should be able to handle objects ("dog"), attributes ("brown dog"), actions ("dog running") and interactions between objects ("dog chasing cat").
- **Generalization**: The method should be able to generalize to facts that have zero or few examples during training.
- **Scalability**: The method should handle an unbounded number of facts.
- **Bi-directionality**: The method should be able to retrieve a language description for an image, and images that show a given language description of a fact.
- **Structure**: The method should provide a structured understanding of facts, for example that "dog" is the subject and has an attribute of "smiling".

Existing visual understanding systems may be categorized into two trends: (1) fact-level systems and (2) high-level systems. Fact level systems include object recognition, action recognition, attribute recognition, and interaction recognition (e.g., [147], [171], [19], [173], [65], [5]). These systems are usually evaluated separately for each fact type (e.g., objects, actions, interactions, attributes, etc.) and are therefore not uniform. Typically, these systems have a fixed dictionary of facts, assuming that facts are seen during training by at least tens of examples, and treat facts independently. Such methods cannot generalize to learn facts outside of the dictionary and will not scale to an unbounded number of facts, since model size scales with the number of facts. Furthermore, these recognition systems are typically uni-directional, only able to return the conditional probability of a fact given an image. The zero/few-shot learning setting (e.g., [136, 88]), where only a few or even zero examples per fact are available, is typically studied apart from the traditional recognition setting. We are not aware of a unified recognition/few shot learning system that learns unbounded set of facts.

In the second trend, several researchers study tasks like image captioning [79, 158, 163, 102], image-caption similarity [79, 81], and visual question answering [4, 100, 128] with very promising results. These systems are typically learning high-level tasks but their evaluation

Figure 5.1: Visual Facts in Images

does not answer whether these systems relate captions or questions to images by fact-level understanding. Captioning models output sentences and thus can mention different types of facts and, in principle, any fact. However, Devlin et al. [31, 32] reported that 60-70% of the generated captions by LSTM-based captioning methods actually exist in the training data and show that nearest neighbor methods have very competitive performance in captioning. These results call into question both the core understanding and the generalization capabilities of the state-of-the-art caption-level systems.

The limitations of prior settings motivated us to study a fact-level understanding setting, which is more related to the first trend but unified to any fact type and able to learn an unbounded number of facts. This setting allows measuring the gained visual knowledge represented by the facts learnt by any proposed system to solve this task. Our goal is a method that achieves a more sophisticated understanding of the objects, actions, attributes, and interactions between objects, and possesses the desireable properties of scalability, generalization, uniformity, bi-directionality, and structure.

Our approach is to learn a common embedding space in which the language and visual views of a fact are mapped to the same location. The key to our solution achieving the desireable characteristics is to make the basic unit of understanding a structured fact as shown in Fig. 5.1 and to have a structured embedding space in which different dimensions record information about the subject S, predicate P, and object O of a fact.

Using an embedding space approach allows our method to scale as we can submit any (<S,P,O>, image), (<S,P>, image), or (<S>, image) facts to train our embedding network. At test time, it allows for bi-directional retrieval, as we can search for language facts that embed near a given image fact and vice-versa. Retaining the structure of a fact in the embedding space gives our method the chance to generalize to understand an S/SP/SPO from training data on its S, P, and O components, since this information is kept separate. To obtain uniformity, we introduce wildcards "*" into our structured fact representation, e.g. <man,smiling,*> or <dog,*,*> and use a wildcard training loss which ignores the unspecified components of embedded second and first order visual and language facts. Carefully designed experiments show that our uniform method achieves state-of-the-art performance in fact-level bidirectional view retrieval over existing image-sentence correlation methods, other view embedding methods, and a version of our method without structure, while also scaling and generalizing better.

Contributions: (1) We propose a new problem setting to study fact-level visual understanding of unbounded number of facts while considering the aforementioned characteristics. (2) We design and investigate several baselines from the multiview learning literature and apply them on this task. (3) We propose two learning representation models that relate different fact types using the structure exemplified in Fig 5.1. (4) Both the designed baselines and the proposed models embed language views and visual views (images) of facts in a joint space that allows uniform representation of different fact types. We show the value of relating facts by structure in the proposed models compared to the designed baselines on several datasets on bi-directional fact retrieval.

5.2 Related Work

In order to make the contrast against the related work clear, we start by stating the scale of facts we are modeling in this work. Let's assume that |S|, |P|, and |O| denotes the number of unique subjects, unique predicates, and unique objects, respectively; see Fig 5.1. The scale of unique second and third order facts is bounded by $|S| \times |P|$ and $|S| \times |P| \times |O|$ possibilities respectively, which can easily reach millions of facts. The data we collected in this work has thus far reached 202,000 unique facts (814,000 images). We cover five lines of related research (first three are from fact-level recognition literature).

(A) Modeling Visual facts in Discrete Space: Recognition of objects or activities has been typically modeled as a mapping function $g : \mathcal{V} \to \mathcal{Y}$, where $\mathcal{Y}$ is discrete set of classes. The function g has recently been learned using deep learning (e.g., [147, 151]). Different systems are built to recognize each fact type in images by modeling a different $g : \mathcal{V} \to \mathcal{Y}$, where $\mathcal{Y}$ is constrained to objects, (e.g., [147]), attributes (e.g. [171]), attributed objects (<car, red>) [19], scenes (e.g., [173]), human actions (e.g., [65]), and interactions [5]. There are several limitations for modeling recognition as $g : \mathcal{V} \rightarrow \mathcal{Y}$ with $|\mathcal{Y}| \rightarrow \infty$. **(1) Scalability:** Adding a new fact leads to changing the architecture, meaning adding thousands of parameters and re-training the model (e.g., for adding a new output node). For example, if VGGNet [147] is used on the scale of 202,000 facts, the number of parameters in the softmax layer alone is close to 1 billion. **(2) Uniformity:** Modeling each group of facts by a different g requires maintaining different systems, retrain several models as new facts are added, and also doesn't allow learning the correlation among different fact types. However, we aim to uniformally model visual perception. **(3) Generalization:** While most of the existing benchmarks for this setting have at least tens of examples per fact (e.g., imageNet [30]), a more realistic assumption is that there might not be enough examples to learn the new class (the long-tail problem). Several works have been proposed to deal this problem in object recognition settings [174, 140]. However, they suffer from the aforementioned scalability problems as facts increase. **(4) Bi-directionality:** These models are uni-directional from $\mathcal{V}$ to $\mathcal{Y}$. Fig 5.2 shows representatives settings of these methods. The three axes are Scalability, Uniformity, and Generalization. These methods typically study seen classes and hence do not generalize to unseen classes.

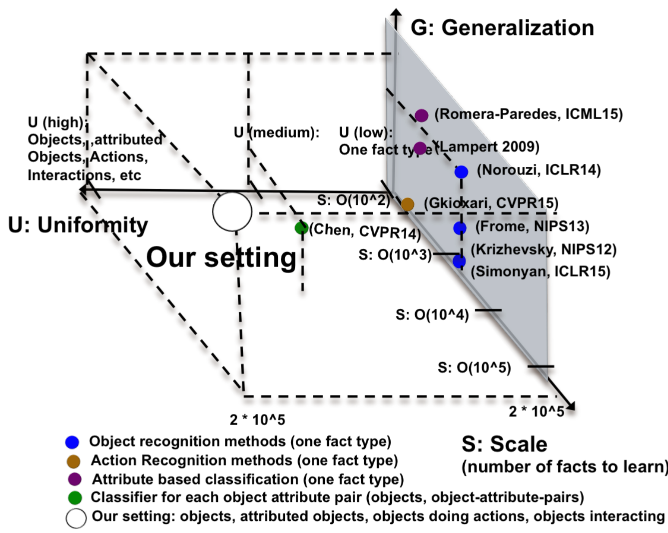


Figure 5.2: Our setting in contrast to the studied fact recognition settings in the literature. Scalability means the number of facts studied in these works. Uniformity means if the setting is applied for multiple fact types. Generalization means the performance of this methods on facts of zero/few images.

(B) Modeling zero/few shot fact learning by semantic representation of classes (e.g., attributes): One of the most successful ideas for learning from few examples per class is by using semantic output codes like attributes as an intermediate layer between features and classes. Formally, g is a composition of two function $g = h(a(\cdot))$, where $a : \mathcal{V} \rightarrow \mathcal{A}$, and $h : \mathcal{A} \rightarrow \mathcal{Y}$ [120]. The main idea is to collect data that is sufficient to learn an intermediate attribute layer, where classes are then represented by these attributes to facilitate zero-shot/few-shot learning. However, Chen *et al*. [19] realized that attribute appearance is dependent on the class, as opposed to these earlier models [120, 88, 52]. Although [19]'s assumption is more realistic, they propose learning different classifiers for each category-attribute pair, which suffers from the same scalability and learning problems pointed out in (A) and is restricted to certain groups of facts (not uniform).

More recent attribute-based zero-shot learning methods embed both images and attributes into a shared space (e.g., Attribute Embedding [1], ESZSL [136]). These methods were mainly studied in the case of zero-shot learning and have shown strong performance. In contrast, we aim at studying the setting where one system that can learn from both facts with many training images and facts with few/no training images. Fig 5.2 shows the contrast between our setting (white circle) and this setting. Although these methods were mainly studied using attributes as a semantic representation and at a much smaller scale of facts, we apply the state of the art ESZSL [136] in order to study the capacity of these models at a much larger scale.

(C) Object Recognition in continuous space using Vision and Language: Recent works in language and vision involve using unannotated text to improve object recognition and to facilitate zero-shot learning. The following group of approaches model object recognition as a function $g(v) = \arg\max_y s(v \in \mathcal{V}, y \in \mathcal{Y})$, where $s(\cdot, \cdot)$ is a similarity function between image v and class y represented by text. In [61], [117] and [150], word embedding language models (e.g., [107]) were adopted to represent class names as vectors. In their setting, the imageNet dataset has 1000 object facts with thousands of examples per class. Our setting has two orders of magnitude more facts with a long-tail distribution. Conversely, other works model the mapping of unstructured text descriptions for classes into a visual classifier [47, 7]. We are extending the visual recognition task to unbounded scale of facts, not only object recognition but also attributes, actions, and interactions in one model; see Fig 5.2 for contrast to our setting.

(D) Image-Caption Similarity Methods: As we illustrated earlier, our goal is fact-level understanding. However, image-caption similarity methods such as [79, 81] are relevant as multi-view learning methods. Although it is a different setting, we found two interesting aspects of these methods to study in our setting. First, how image-caption similarity system trained on image-caption level performs on fact-level understanding. Second, these systems could be retrained in our setting by providing them with fact-level annotation, where every example is a phrase representing the fact and an image (e.g., “person riding horse” and an image with this fact).

(E) MV-CCA : MV-CCA is a recent multiview, scalable, and robust version of the famous CCA embedding method [67]. We apply MV-CCA as a baseline in our setting.

5.3 Problem Definition: Representation and Visual Modifiers

We deal with three groups of facts; see Fig. 5.1. First Order Facts <S,*,*> are object and scene categories (e.g., <baby,*,*>, <girl,*,*>, <beach,*,*>). Second Order Facts <S,P,*> are objects performing actions or attributed objects (e.g., <baby, smiling,*>, <baby, Asian,*>). Third Order Facts <S,P,O> are interactions and positional information (e.g. <baby, sitting_on, high_chair>, <person, riding, horse>). By allowing wild-cards in this structured representation (<baby,*,*>and <baby, smiling,*>), we can not only allow uniform representation of different fact types but also relate them by structure. We propose to model these facts by embedding them into a structured fact space that has three continuous hyper-dimensions ϕ_S, ϕ_P, and ϕ_O

$\phi_S \in \mathbb{R}^{d_S}$**:** The space of object categories or scenes S.

$\phi_P \in \mathbb{R}^{d_P}$**:** The space of actions, interactions, attributes, and positional relations.

$\phi_O \in \mathbb{R}^{d_O}$**:** The space of interacting objects, scenes that interact with S for SPO facts.

where d_S, d_P, and d_O are the dimentionalities corresponding to ϕ_S, ϕ_P, and ϕ_O, respectively. As shown in Fig. 5.3, first order facts like <woman,*,*>, <man,*,*>, <person,*,*> live in a hyper-plane in the $\phi_P \times \phi_O$ space. Second order facts (e.g., <man, walking,*>, <girl, walking,*>) live as a hyper-line that is parallel to ϕ_O axis. Finally, a third order fact like <man, walking, dog> is a point in the $\phi_S \times \phi_P \times \phi_O$ visual perception space. Inspired from the concept of language modifiers, the ϕ_S, ϕ_P, and ϕ_O could be viewed as what we call “visual modifiers”. For example, the second order fact <baby, smiling,* > is a ϕ_P visual modifier for <baby,*,*>,

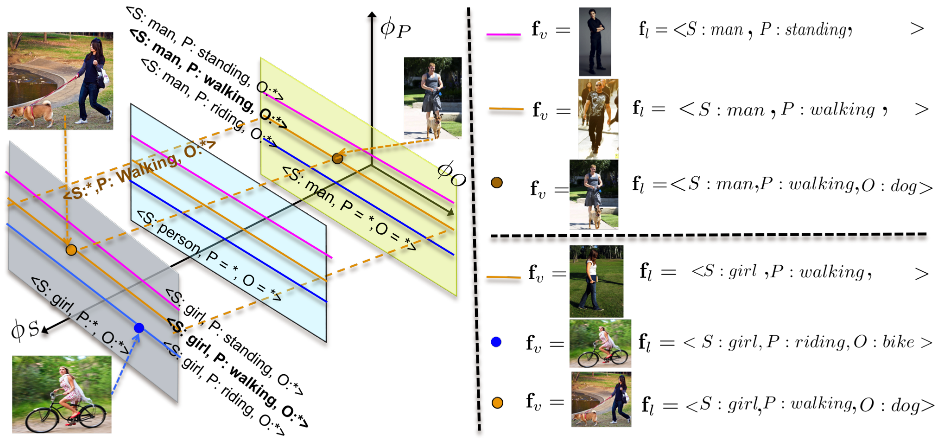


Figure 5.3: Unified Fact Representation and Visual Modifiers Notion

and the third order fact <person, playing, flute> is the fact <person, *, *> visually modified on both ϕ_P and ϕ_O axes. By embedding all language and images into this common space, our algorithm can scale efficiently. Further, this space can be used to retrieve a language view of an image as well as a visual view of a language description, making the model bi-directional. We argue that modeling visual recognition based on this notion gives it a generalization capability. For example is if the model learned the facts <boy>, <girl>, <boy, petting, dog>, <girl, riding, horse>, we would aim at recognizing an unseen fact <boy, petting, horse>. We show these capabilities quantitatively later in our experiments. We model this setting as a problem with two views, one in the visual domain $\mathcal{V}$ and one in the language domain $\mathcal{L}$. Let $\mathbf{f}$ be a structured fact, $\mathbf{f}_v \in \mathcal{V}$ denoting the visual view of $\mathbf{f}$ and $\mathbf{f}_l \in \mathcal{L}$ denoting the language view of $\mathbf{f}$. For instance, an annotated fact with language view $\mathbf{f}_l$ =<S:girl, P:riding, O:bike> would have a corresponding visual view $\mathbf{f}_v$ as an image where this fact occurs; see Fig. 5.4.

Our goal is to learn a representation that covers all the three orders of facts. We denote the embedding functions from a visual view to ϕ_S, ϕ_P, and ϕ_O as $\phi_S^{\mathcal{V}}(\cdot)$, $\phi_P^{\mathcal{V}}(\cdot)$, and $\phi_O^{\mathcal{V}}(\cdot)$, and the structured visual embeddings of a fact $\mathbf{f}_v$ by $\mathbf{v}_S = \phi_S^{\mathcal{V}}(\mathbf{f}_v)$, $\mathbf{v}_P = \phi_P^{\mathcal{V}}(\mathbf{f}_v)$, and $\mathbf{v}_O = \phi_O^{\mathcal{V}}(\mathbf{f}_v)$, respectively. Similarly, we denote the embedding functions from a language view to ϕ_S, ϕ_P, and ϕ_O as $\phi_S^{\mathcal{L}}(\cdot)$, $\phi_P^{\mathcal{L}}(\cdot)$, and $\phi_O^{\mathcal{L}}(\cdot)$, and the structured language embeddings of a fact $\mathbf{f}_l$ as $\mathbf{l}_S = \phi_S^{\mathcal{L}}(\mathbf{f}_l)$, $\mathbf{l}_P = \phi_P^{\mathcal{L}}(\mathbf{f}_l)$, and $\mathbf{l}_O = \phi_O^{\mathcal{L}}(\mathbf{f}_l)$. We denote the concatenation of the visual view hyper-dimensions' embedding as $\mathbf{v}$, and the language view hyper-dimensions' embedding as $\mathbf{l}$; see Eq. 5.1 Third-order facts <S,P,O> can be directly embedded in the structured fact space by Eq. 5.1 with $\mathbf{v} \in \mathbb{R}^{d_S} \times \mathbb{R}^{d_P} \times \mathbb{R}^{d_O}$ for the image view and $\mathbf{l} \in \mathbb{R}^{d_S} \times \mathbb{R}^{d_P} \times \mathbb{R}^{d_O}$ for the

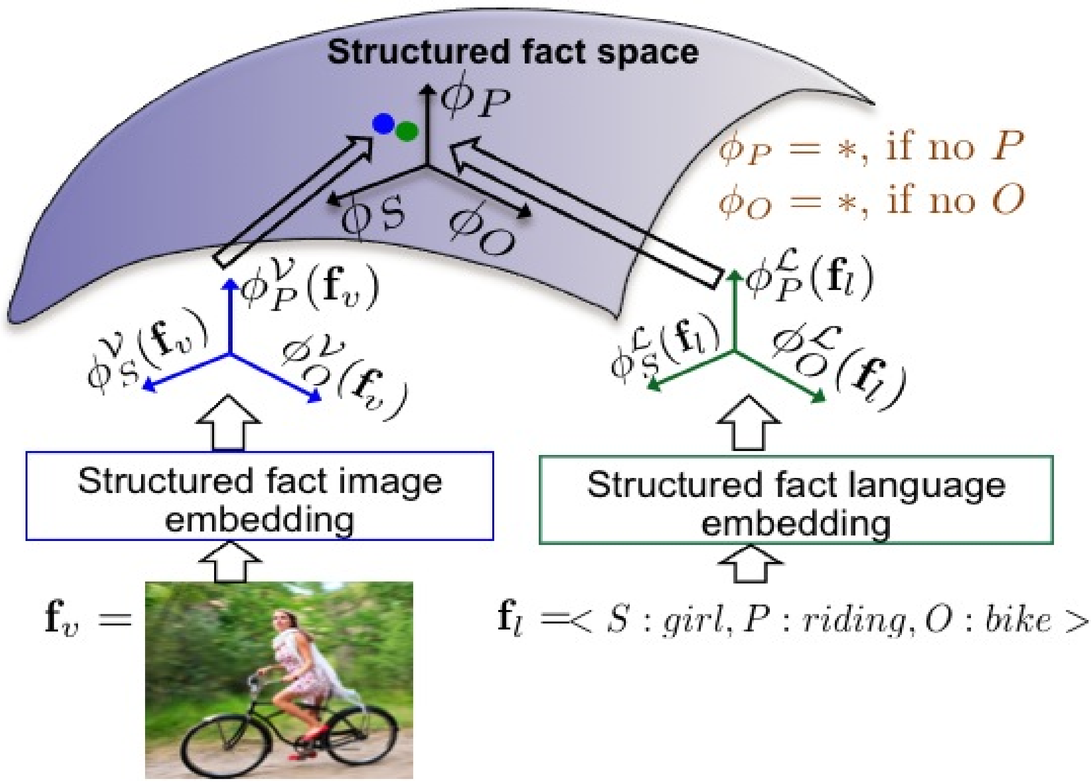


Figure 5.4: Structured Embedding

language view. Based on our "fact modifier" observation, we propose to represent both second and first-order facts as wild cards "$*$", as illustrated in Eq. 5.2, 5.3; see Fig 5.4, 5.3.

$$\textbf{Third-Order Facts <S,P,O>:} \ \mathbf{v} = [\mathbf{v}_S, \mathbf{v}_P, \mathbf{v}_O] \qquad \mathbf{l} = [\mathbf{l}_S, \mathbf{l}_P, \mathbf{l}_O] \tag{5.1}$$

$$\textbf{Second-Order Facts <S, P,*>:} \ \mathbf{v} = [\mathbf{v}_S, \mathbf{v}_P, \mathbf{v}_O = *] \qquad \mathbf{l} = [\mathbf{l}_S, \mathbf{l}_P, \mathbf{l}_O = *] \tag{5.2}$$

$$\textbf{First-Order Facts <S,*,*>:} \ \mathbf{v} = [\mathbf{v}_S, \mathbf{v}_P = *, \mathbf{v}_O = *] \qquad \mathbf{l} = [\mathbf{l}_S, \mathbf{l}_P = *, \mathbf{l}_O = *] \tag{5.3}$$

Setting ϕ_P and ϕ_O to $*$ for first-order facts means that the P and O modifiers are not of interest for first-order facts, which is intuitive. Similarly, setting ϕ_O to $*$ for second-order facts indicates that the O modifier is not of interest for single-frame actions and attributed objects. If an image contains lower order fact such as <man>, then higher order facts such as <man, tall> or <man, walking, dog> may also be present. Hence, the wild cards (i.e. $*$) of the first- and second-order facts are not penalized during training.

5.4 Models

We propose a two-view structured fact embedding model with five properties mentioned in Sec 5.1. Satisfying the first four properties can be achieved by using a generative model $p(\mathbf{f}_v, \mathbf{f}_l)$ that connects the visual and the language views of $\mathbf{f}$, where more importantly $\mathbf{f}_v$ and $\mathbf{f}_l$ inhabit a continuous space. We model $p(\mathbf{f}_v, \mathbf{f}_l) \propto s(\mathbf{v}, \mathbf{l})$, where $s(\cdot, \cdot)$ is a similarity function defined over the structured fact space. We satisfy the fifth property by building our models over the aforementioned structured wild card representation. Our objective is that two views of the same fact should be embedded so that they are close to each other; see Fig 5.4. The question now is how to model and train $\phi^{\mathcal{V}}(\cdot)$ visual functions ($\phi^{\mathcal{V}}_S(\cdot), \phi^{\mathcal{V}}_P(\cdot), \phi^{\mathcal{V}}_O(\cdot)$) and $\phi^{\mathcal{L}}(\cdot)$ language functions ($\phi^{\mathcal{L}}_S(\cdot), \phi^{\mathcal{L}}_P(\cdot), \phi^{\mathcal{L}}_O(\cdot)$) . We model $\phi^{\mathcal{V}}(\cdot)$ as a CNN encoder (e.g., [83, 147]), and $\phi^{\mathcal{L}}(\cdot)$ as RNN encoder (e.g., [107, 124]) due to their recent success as encoders for images and words, respectively. We propose two models for learning facts, denoted by Model 1 and Model 2. Both models share the same structured fact language embedding/encoder but differ in the structured fact image encoder.

We start by defining an activation operator $\psi(\theta, a)$, where a is an input, and θ is a series of one or more neural network layers (may include different layer types, e.g., convolution, pooling, then another convolution and pooling). The operator $\psi(\theta, a)$ applies θ parameters layer by layer to compute the final activation of a using θ subnetwork.

Model 1 (structured fact CNN image encoder): In Model 1, a structured fact is visually encoded by sharing convolutional layer parameters (denoted by θ_c), and fully connected layer parameters (denoted by θ_u); see Fig. 5.5(a). Then W^S, W^P, and W^O transformation matrices are applied to produce $\mathbf{v}_S = \phi^{\mathcal{V}}_S(\mathbf{f}_v)$,$\mathbf{v}_P = \phi^{\mathcal{V}}_P(\mathbf{f}_v)$, and $\mathbf{v}_O = \phi^{\mathcal{V}}_O(\mathbf{f}_v)$. If we define $b = \psi(\theta_u, \psi(\theta_c, \mathbf{f}_v))$, then

$$\mathbf{v}_S = \phi^{\mathcal{V}}_S(\mathbf{f}_v) = W^S\, b, \;\; \mathbf{v}_P = \phi^{\mathcal{V}}_P(\mathbf{f}_v) = W^P\, b, \;\; \mathbf{v}_O = \phi^{\mathcal{V}}_O(\mathbf{f}_v) = W^O\, b. \tag{5.4}$$

Model 2 (structured fact CNN image encoder): In contrast to Model 1, we use different convolutional layers for S than that for P and O, inspired by the idea that P and O are modifiers to S (Fig. 5.5(b)). Starting from $\mathbf{f}_v$, there is a common set of convolutional layers, denoted by θ^0_c, then the network splits into two branches, producing two sets of convolutional layers θ^S_c and θ^{PO}_c, followed by two sets of fully connected layers θ^S_u and θ^{PO}_u. If we define the

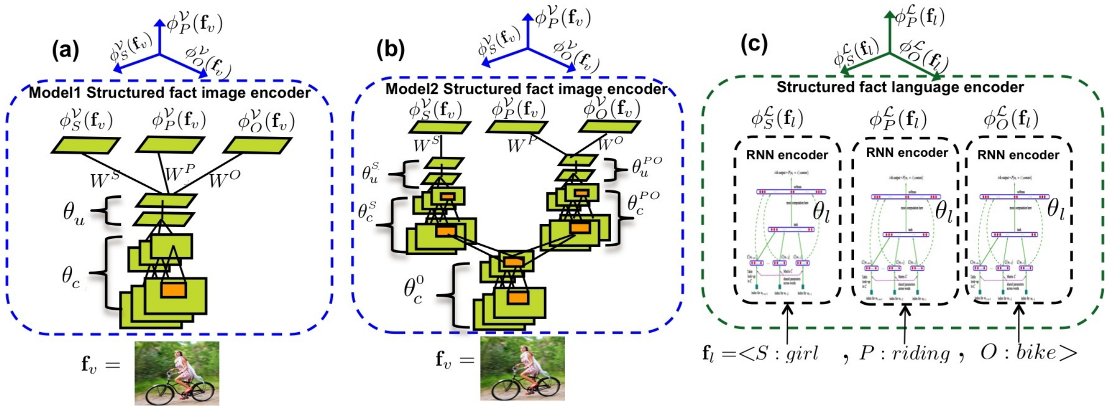


Figure 5.5: Sherlock Models. See Fig. 5.4 for the full picture.

output of the common S,P,O layers as $d = \psi(\theta_c^0, \mathbf{f}_v)$ and the output of the P,O column as $e = \psi(\theta_u^{PO}, \psi(\theta_c^{PO}, d))$, then

$$\mathbf{v}_S = \phi_S^{\mathcal{V}}(\mathbf{f}_v) = W^S\ \psi(\theta_u^S, \psi(\theta_c^S, d)),\ \mathbf{v}_P = \phi_P^{\mathcal{V}}(\mathbf{f}_v) = W^P\ e,\ \ \mathbf{v}_O = \phi_O^{\mathcal{V}}(\mathbf{f}_v) = W^O\ e. \tag{5.5}$$

Structured fact RNN language encoder: The structured fact language view is encoded using RNN word embedding vectors for S, P and, O separately. Hence

$$\mathbf{l}_S = \phi_S^{\mathcal{L}}(\mathbf{f}_l) = \text{RNN}_{\theta_l}(\mathbf{f}_l^S), \mathbf{l}_P = \phi_P^{\mathcal{L}}(\mathbf{f}_l) = \text{RNN}_{\theta_l}(\mathbf{f}_l^P), \mathbf{l}_O = \phi_O^{\mathcal{L}}(\mathbf{f}_l) = \text{RNN}_{\theta_l}(\mathbf{f}_l^O) \tag{5.6}$$

where $\mathbf{f}_l^S$, $\mathbf{f}_l^P$, and $\mathbf{f}_l^O$ are the Subject, Predicate, and Object parts of $\mathbf{f}_l \in \mathcal{L}$. For each of them, the literals are dropped. In our experiments, θ_l is fixed to a pre-trained word vector embedding model (e.g. [107, 124]) for $\mathbf{f}_l^S$, $\mathbf{f}_l^P$, and $\mathbf{f}_l^O$; see Fig 5.5(c).

Loss function: One way to model $p(\mathbf{f}_v, \mathbf{f}_l)$ for Model 1 and Model 2 is to assume that $p(\mathbf{f}_v, \mathbf{f}_l) \propto= \exp(-\text{loss}_w(\mathbf{f}_v, \mathbf{f}_l))$ and minimize the distance $\text{loss}_w(\mathbf{f}_v, \mathbf{f}_l)$ defined as

$$\text{loss}_w(\mathbf{f}_v, \mathbf{f}_l) = w_S^{\mathbf{f}} \cdot D(\mathbf{v}_S, \mathbf{l}_S) + w_P^{\mathbf{f}} \cdot D(\mathbf{v}_P, \mathbf{l}_P) + w_O^{\mathbf{f}} \cdot D(\mathbf{v}_O, \mathbf{l}_O). \tag{5.7}$$

where $D(\cdot,\cdot)$ is a distance function. Thus we minimize the distance between the embeddings of the visual view and the language view. Our solution to penalize wild-card facts is to ignore their wild-card modifiers in the loss. Here $w_S^{\mathbf{f}} = 1$, $w_P^{\mathbf{f}} = 1$, $w_O^{\mathbf{f}} = 1$ for <S,P,O> facts , $w_S^{\mathbf{f}} = 1$, $w_P^{\mathbf{f}} = 1$, $w_O^{\mathbf{f}} = 0$ for <S,P> facts, and $w_S^{\mathbf{f}} = 1$, $w_P^{\mathbf{f}} = 0$, $w_O^{\mathbf{f}} = 0$ for <S> facts. Hence loss_w does not penalize the O modifier for second-order facts or the P and O modifiers for first-order facts, which follows our definition of wild-cards. In this paper, we used $D(\cdot,\cdot)$ as the standard Euclidean distance.

Testing (Two-view retrieval): After training a model (either Model 1 or 2), we embed all the testing $\mathbf{f}_v$s (images) by the learnt models, and similarly embed all the test $\mathbf{f}_l$s as shown in Eq 5.6. For language view retrieval (retrieve relevant facts in language given an image), we compute the distance between the structured embedding of an image $\mathbf{v}$ and all the facts structured language embeddings $\mathbf{l}$s, which indicates relevance for each fact $\mathbf{f}_l$ for the given image. For visual view retrieval (retrieve relevant images given fact in language form), we compute the distance between the structured embedding of the given fact $\mathbf{l}$ and all structured visual embedding of images $\mathbf{v}$s in the test set. For first and second order facts, the wild-card part is ignored while computing the distance.

5.5 Experiments

5.5.1 Data Collection of Structured Facts

In order to train a model that connects the structured fact language view in $\mathcal{L}$ with its visual view in $\mathcal{V}$, we need to collect large scale data in the form of ($\mathbf{f}_v$, $\mathbf{f}_l$) pairs. Large scale data collection is challenging in our setting since it relies on the localized association of a structured language fact $\mathbf{f}_l$ with an image $\mathbf{f}_v$ when such facts occur. In particular, it is a complex task to collect annotations for second-order facts and third-order facts.

We began our data collection by augmenting existing datasets with fact language view labels $\mathbf{f}_l$: PPMI [167], Stanford40 [168], Pascal Actions [50], Sports [70], Visual Phrases [138], INTERACT [6] datasets. The union of these 6 datasets resulted in 186 facts with 28,624 images as broken out in Table 5.1. We also extracted structured facts from the Scene Graph dataset [78] with 5000 manually annotated images in a graph structure from which first-, second-, and third-order relationships can be extracted. We extracted 110,000 second-order facts and 112,000 third-order facts. The majority of these are positional relationships. We also added to the aforementioned data, 380,000 second and third order fact annotation collected from MSCOCO and Flickr30K Entities datasets using a language approach as detailed in [113] in the supplementary. We show later in this section how we use this data to perform several experiments varying in scale to validate our claims. Table 5.2 shows the unique facts in the large scale dataset.

5.5.2 Setup of our Models and the designed Baselines

In our Model 1 and Model 2, θ_l is the GloVE840B RNN model [124] to encode structured facts in the language view.

1. **Model 1**: Model 1 is constructed from VGG-16, where θ_c is built from the layer `conv_1_1` to `pool5`, and θ_u is the two following fully connected layers `fc6` and `fc7` in VGG-16 [147]. Similar to Model 2, W^S, W^P, and W^O are initialized randomly and the rest of the parameters are initialized from VGG-16 trained on ImageNet [30].

2. **Model 2:** The shared layers θ_c^0 match the architecture of the convolutional layers and pooling layer in VGG-16 named `conv_1_1` until `pool3`, and have seven convolution layers. The subject layers θ_c^S and predicate-object layers θ_c^{PO} are two branches of convolution and pooling layers with the same architecture as VGG-16 layers named `conv_4_1` until `pool5` layer, which makes six convolution-pooling layers in each branch. Finally, θ_u^S and θ_u^{PO} are two instances of `fc6` and `fc7` layers in VGG-16 network. W^S, W^P, and W^O are initialized randomly and the rest are initialized from

Table 5.1: Our fact augmentation of six datasets

	Unique language views f_l				**Number of (f_v, f_l) pairs**			
	S .	**SP.**	**SPO .**	**total**	**S**	**SP**	**SPO**	**total images**
INTERACT	0	0	60	60	0	0	3171	3171
VisualPhrases	11	4	17	32	3594	372	1745	5711
Stanford40	0	11	29	40	0	2886	6646	9532
PPMI	0	0	24	24	0	0	4209	4209
SPORT	14	0	6	20	398	0	300	698
Pascal Actions	0	5	5	10	0	2640	2663	5303
Union	**25**	**20**	**141**	**186**	**3992**	**5898**	**18734**	**28624**

Table 5.2: Large Scale Dataset

	S	**SP**	**SPO**	**Total**
Training facts	6116	57681	107472	171269
Testing facts	2733	22237	33447	**58417**
Train/Test Intersection	1923	13043	11774	26740
Test unseen facts	**810**	**9194**	**21673**	**31677**

VGG-16 trained on ImageNet.

3. **Multiview CCA IJCV14 [67] (MV CCA)** : MV CCA expects features from both views. For visual view features, we used VGG16 (FC6). For the language view features, we used GloVE. Since MV CCA does not support wild-cards, we fill the wild-card parts of $\Phi^{\mathcal{L}}(\mathbf{f}_l)$ with zeros for First Order and Second order facts.

4. **ESZSL ICML15 Baseline [136] (ESZSL)**: ESZSL also expects both visual and semantic features for a fact. As in MV CCA, we used VGG16 (FC6) and GloVE.

5. **Image-Sentence Similarity (TACL15 [81]) (MS COCO pretrained)**: We used the theano implementations of this method that were made publically available by the authors [80]. The purpose of applying MS COCO pretrained image-caption models is to show how image-caption trained models perform when applied to fact level recognition in our setting. In order to use these models to measure simility between image and facts in our setting, we provide them with the image and a phrase constructed from the fact language representation. For example <person, riding, horse > is converted to "person riding horse".

6. **Image-Sentence Similarity (TACL15 [81] (retrained)**: In contrast to the previous setting, we retrain these models by providing them our image-fact training pairs where facts are converted to phrases. The results show the value of learning models on the fact level instead of the caption level.

5.5.3 Evaluation Metrics

We present evaluation metrics for both language view retrieval and visual view retrieval. **Metrics for visual view retrieval (retrieving $\mathbf{f}_v$ given $\mathbf{f}_l$):** To retrieve an image (visual view) given a language view (e.g. <S: person, P: riding, O: horse>), we measure the performance by mAP (Mean Average Precision).An image $\mathbf{f}_v$ is considered positive only if there is a pair $(\mathbf{f}_l, \mathbf{f}_v)$ in the annotations. Even if the retrieved image is relevant but such pair does not exist, it is considered incorrect. We also use mAP10, mAP100 variants that compute the mAP based on only the top 10 or 100 retrieved images, which is useful for evaluating large scale experiments.

Metrics for language view retrieval (retrieving $\mathbf{f}_l$ given $\mathbf{f}_v$): To retrieve fact language views given an image. we use top 1, top 5, top 10 accuracy for evaluation. We also used MRR (mean reciprocal ranking) metric which is basically $1/r$ where r is the rank of the correct class. An important issue with our setting is that there might be multiple facts in the same image. Given that there are L correct facts in the given image to achieve top 1 performance these L facts must all be in the top L retrieved facts. Accordingly, top K means the L facts are in the top $L + K - 1$ retrieved facts. A fact language view $\mathbf{f}_l$ is considered correct only if there is a pair $(\mathbf{f}_l, \mathbf{f}_v)$ in the annotations.

It is not hard to see that the aforementioned metrics are very harsh, especially in the large scale setting. For instance, if the correct fact for an image is <S:man,P: jumping>, then an answer <S:person, P:jumping> receives zero credit. Also, the evaluation is limited to the ground truth fact annotations. There might be several facts in an image but the provided annotations may miss some facts. Qualitatively we found the metrics harsh for our large scale experiment. Defining better metrics is future work.

5.5.4 Small and Mid scale Experiments

We performed experiments on several datasets ranging in scale: Stanford40 [168], Pascal Actions [167], Visual Phrases [138], and the union of six datasets described earlier in Table 5.1 in Sec. 5.5.1. We used the training and test splits defined with those datasets. For the union of six datasets, we unioned the training and testing annotations to get the final split. In all these training/testing splits, each fact language view $\mathbf{f}_l$ has corresponding tens of visual views $\mathbf{f}_v$ (i.e., images) split into training and test sets. So, each test image belongs to a fact that was seen by other images in the training set.

Table 5.3 shows the performance of our Model 1, Model 2, and the designed baselines on these four datasets for both view retrieval tasks. We note that Model 2 works relatively better than Model 1 as the scale size increases as shown here when comparing results on Pascal dataset to larger datasets like Stanford40, Visual Phrases, and 6DS. In the next section, we show that Model2 is clearly better than Model 1 in the large scale setting. Our intuition behind this result is that Model 2 learns a different set of convolutional filters in the PO branch to understand action/attributes and interactions which is different from the filter bank learned to discriminate between different subjects for the S branch. In contrast, Model 1 is trained by optimizing one bank of filters for SPO altogether, which might conflict to optimize for both S and PO together; see Fig 5.5.

Learning from image-caption pairs even on big dataset like MSCOCO does not help discriminate between tens of facts as shown in these experiments. However, retraining these models by providing them image-fact pairs makes them perform much better as shown in Table 5.3. Compared to other methods on language view retrieval, we found Model 1 and 2 perform significantly better than TACL15 [81] even when retrained for our setting, especially on PASCAL10, Stanford40, and 6DS datasets which are dominated by SP and SPO facts; see Table 5.1. For

Table 5.3: Small and Medium Scale Experiments

		Language View retrieval			**Visual View retrieval**		
		Top1	**Top 5**	**MRR**	**mAP**	**mAP10**	**mAP100**
Standord40 (40 facts)	**Model2**	**74.46**	**92.01**	**82.26**	73	98.35	92
(11 SP, 29 SPO)	**Model1**	71.22	90.98	82.09	**74.57**	**99.72**	**92.62**
	MV CCA IJCV14	67.74	88.32	76.80	66.00	96.86	86.66
	ESZSL ICML15 [136]	40.89	74.93	56.08	50.9	93.87	78.35
	Image-Sentence TACL15 [81] (COCO pretrained)	33.73	62.62	47.70	26.29	59.68	44.2
	Image-Sentence TACL15 [81] (retrained)	60.86	87.82	72.51	51.9	88.13	74.55
	Chance	2.5	-	-	-	-	-
Pascal Actions (10 facts)	**Model2**	**74.760**	**95.750**	**83.680**	**80.950**	**100.000**	**97.240**
(5 SP, 5 SPO)	**Model1**	74.080	95.790	83.280	80.530	100.000	96.960
	MV CCA IJCV14	59.82	92.78	73.16	33.45	66.52	53.29
	ESZSL ICML15 [136]	44.846	88.864	63.366	54.274	89.968	82.273
	Image-Sentence TACL15 [81] (COCO pretrained)	46.050	86.907	62.796	40.712	88.694	71.078
	Image-Sentence TACL15 [81] (retrained)	60.27	94.66	74.77	50.58	84.65	71.61
	Chance	10	-	-	-	-	-
VisualPhrases (31 facts)	**Model2**	**34.367**	**76.056**	**47.263**	**39.865**	**61.990**	**48.246**
(14 S, 4 SP, 17 SPO)	**Model1**	28.100	75.285	42.534	38.326	65.458	46.882
	MV CCA IJCV14 [67]	28.94	70.61	88.92	28.27	49.30	34.48
	ESZSL ICML15 [136]	33.830	68.264	44.650	33.010	57.861	41.131
	Image-Sentence TACL15 [81] (COCO pretrained)	30.111	64.494	42.777	26.941	49.892	33.014
	Image-Sentence TACL15 [81] (retrained)	32.32	94.72	50.7	28.0	49.89	33.21
	Chance	3.2	-	-	-	-	-
6DS (186 facts)	**Model2**	**69.63**	**80.32**	**70.66**	**34.86**	**61.03**	**50.68**
(25 S, 20 SP, 141 SPO)	**Model1**	68.94	78.74	70.74	34.64	56.54	47.87
	MV CCA IJCV14 [67]	29.84	39.78	32.00	23.93	46.43	36.44
	ESZSL ICML15 [136]	27.53	47.4	58.2	30.7	60.97	47.58
	Image-Sentence TACL15 [81] (COCO pretrained)	15.71	26.84	19.65	9.37	21.58	15.88
	Image-Sentence TACL15 [81] (retrained)	26.13	41.10	30.94	26.17	56.10	40.4
	Chance	0.54	-	-	-	-	-

visual view retrieval, performance is competitive in some of the datasets. We think the reason is due to the structure that makes our models relate all fact types by the visual modifiers notion.

Although ESZSL is applicable in our setting, it is among the worst performing methods in Table 5.3. This could be because ESZSL is mainly designed for Zero-Shot Learning, but each fact has some training examples in these experiments. Interestingly, MV CCA with the chosen visual and language features is among the best methods. Next we compare these methods when number of facts becomes three orders of magnitudes larger and with tens of thousands of testing facts that are unseen in training.

5.5.5 Large Scale Experiment

In this experiment, we used the union of all the data described in Sec. 5.5.1. We further augmented this data with 2000 images for each MS COCO object (80 classes) as first-order facts. We also used object annotations in the Scene Graph dataset as first-order fact annotations with a maximum of 2000 images per object. Finally, we randomly split all the annotations into an 80%-20% split, constructing sets of 647,746 $(\mathbf{f}_v, \mathbf{f}_l)$ training pairs (with 171,269 unique fact language views $\mathbf{f}_l$) and 168,691 $(\mathbf{f}_v, \mathbf{f}_l)$ testing pairs (with 58,417 unique $\mathbf{f}_l$), for a total of $(\mathbf{f}_v, \mathbf{f}_l)$ 816,436 pairs, 202,946 unique $\mathbf{f}_l$. Table 5.2 shows the coverage of different types of facts. There are 31,677 language view test facts that were unseen in the training set (851 <S>, 9,194 <S,P>, 21,673 <S,P,O>). The majority of the facts have only one example; see the supplementary material.

Qualitative results are shown in Fig. 5.6, 5.7 (with many more in the supplementary). In Fig. 5.6, our model's ability to generalize can be seen in the red facts. For example, for the leftmost image our model was able to correctly identify the image as <dog, riding, wave> despite that fact never being seen in our training data. The left images in Fig. 5.7 show the variety of images we can retrieve for the query <airplane, flying>. In the right images in Fig. 5.7, note how our model learns to visually distinguish gender ("man" versus "girl"), and group versus single. It can also correctly retrieve images for facts that were never seen in the training set (<girl, using, racket>). Highlighting the harshness of the metric, Fig. 5.7 also shows that <airplane, flying> has zero AP10 value giving us zero credit since the top images were just annotated as an < airplane>.

<dog, riding, wave>	0.88	<man, on, sand>	0.52	<boat, behind, boat>	0.775	<bananas, over,fruit>	0.68
<one, riding, wave>	0.62	<man,on, beach>	0.51	<boat, beside, boat>	0.765	<bananas,behind, fruit>	0.68
<guy, riding, wave>	0.60	<man, pushing, boat>	0.48	< boat, pulling, boat>	0.753	<pineapple, on, table>	0.59

Figure 5.6: Language View Retrieval examples (red means unseen facts)

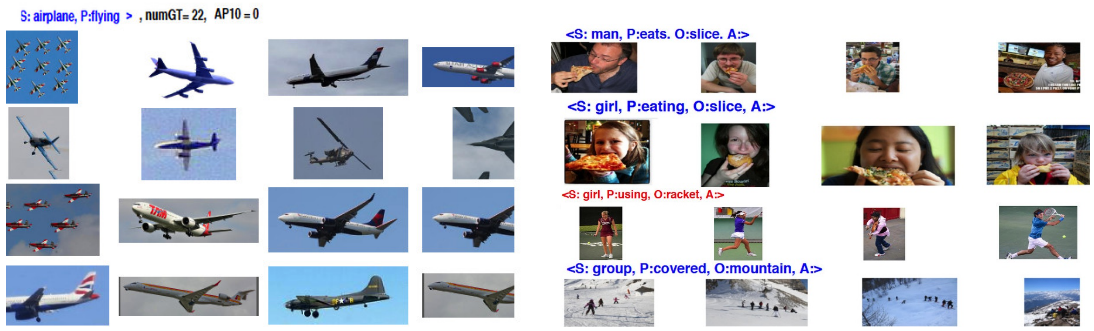


Figure 5.7: Visual View Retrieval Examples (red means unseen facts)

To perform retrieval in both directions, we used the FLANN library [114] to compute the (approximate) 100 nearest neighbors for $\mathbf{f}_l$ given $\mathbf{f}_v$, and vice-versa. Details about the nearest-neighbor database creation and the large scale evaluation could be found in the supplementary. The results in Table 5.4 indicate that Model 2 is better than Model 1 for retrieval from both views, which is consistent with our medium scale results and our intuition. Model 2 is also multiple orders of magnitude better than chance and is also significantly better than the competing methods. To test the value of structure, we ran an experiment where we averaged the S, P, and O parts of the visual and language embedding vectors instead of keeping the structure. Removing the structure leads to a noticeable decrease in performance from 16.39% to 8.1% for the K1 metric; see Table 5.4.

Previous smaller scale experiments are orders of magnitudes smaller and also less challenging since all facts were seen during training. Figure 5.8 shows the effect of the scale on the Top1 performance for language view retrieval task (denoted K1). There is an observable increase on the improvement of Model 2 compared to the baselines in the large scale setting. Additionally, the performance of the image-caption similarity methods degrade substantially. We think this is due to both the large scale of the facts and that the majority of the facts have zero or very few training examples. Interestingly, MV CCA is among the best performing methods in the large scale setting. However, Model 2 and Model 1 outperform MV CCA on both Top1 and Top 5 metrics; see Table 5.4. On the language view retrieval, we have very competitive results to MV CCA but as we have notices several good visual retrieval results for which the metric gives zero-credit.

Figure 5.9 shows the Top10 large scale knowledge view retrieval (K10) results reported in Table 5.4 broken out by fact type and the number of images per fact. These results show that Model 2 generally behaves better with compared other models with the increase of facts. We noticed a slight increase for Model 1 over Model 2.

It is desirable for a method to be able to generalize to understand an SPO interaction from training examples involving its components, even when there are zero or very few training examples for the exact SPO with all its parts S,P and O. Table 5.5 shows the K10 performance for

Table 5.4: Large Scale Experiment

	Language View retrieval %			**Visual view Retrieval %**	
	Top1	**Top 5**	**Top 10**	**mAP100**	**mAP10**
Model 2	**16.39**	**17.62**	**18.41**	0.90	0.90
Model 1	13.27	14.19	14.80	0.73	0.73
Model 2 (Unstructured by SPO average)	8.1	12.4	14.00	0.61	0.62
MV CCA IJCV14 [67]	12.28	12.84	13.15	**1.0**	**1.0**
ESZSL ICML15 [136]	5.80	5.84	5.86	0.4	0.4
Image-Sentence TACL15 [81] (COCO pretrained)	3.48	3.48	3.5	0.021	0.0087
Image-Sentence TACL15 [81] (retrained)	5.87	6.06	6.15	0.29	0.29
Chance	0.0017	-	-	-	-

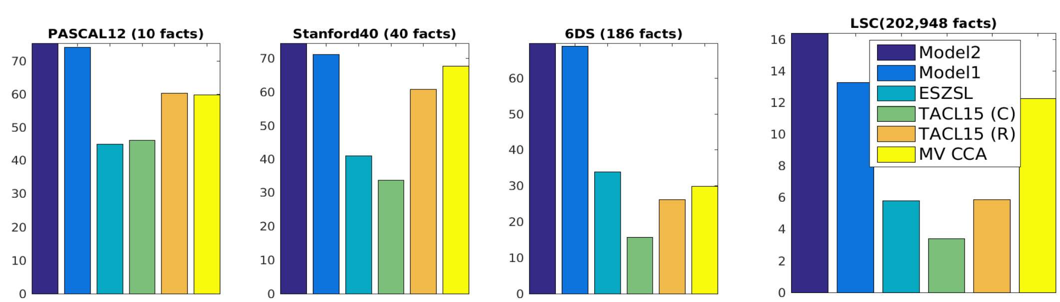


Figure 5.8: K1 Performance Across Different Datasets. These graphs show the advantage of the proposed models as the scale increases from left to right. (R) for TACL15 means the retrained version, (C) means COCO pretrained model; see Sec 5.5.2

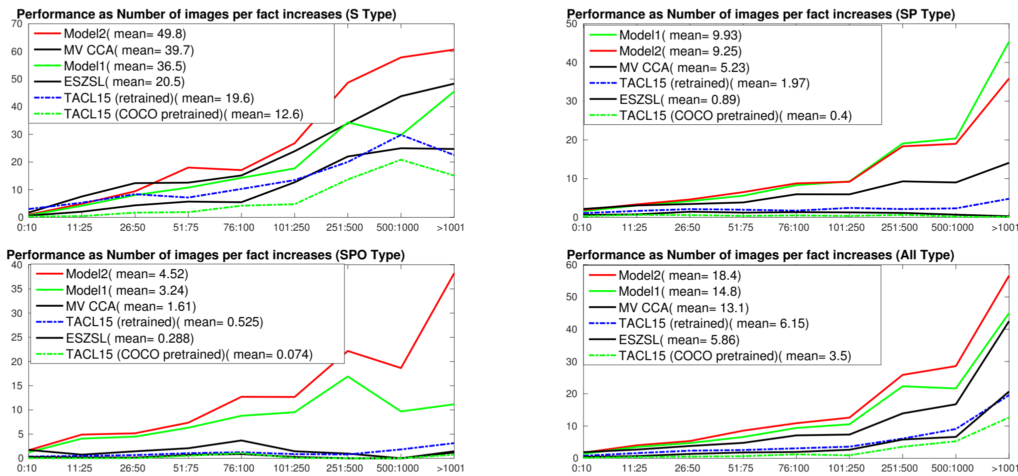


Figure 5.9: K10 Performance (y-axis) versus the number of images per fact (x-axis). Top Left: Objects (S), Top Right: Attributed Objects and Objects performing Actions (SP), Bottom Left: Interactions (SPO), Bottom Right: All Facts.

Table 5.5: Generalization: SPO Facts of less than or equal 5 examples (K10 metric)

Cases	SP≥15, O≥15	PO≥15, ≥15	SO≥15, P≥15	S≥15, PO≥15	SO≥15, PO≥15	SO≥15, SP≥15	S,P,O≥15	S,P,O≥100
NumFacts for this case	**10605**	**9313**	**4842**	**4673**	**1755**	**3133**	**21616**	**12337**
Model2	**2.063**	**2.026**	**3.022**	**2.172**	**3.092**	**2.962**	**1.861**	**2.462**
Model1	1.751	1.357	1.961	1.645	1.684	2.097	1.405	1.666
ESZSL	0.149	0.107	0.098	0.066	0.041	0.038	0.240	0.176
TACL15 (COCO pretrained)	0.013	0.024	0.025	0.019	0.000	0.013	0.034	0.027
TACL15 (retrained)	0.367	0.380	0.473	0.384	0.543	0.586	0.353	0.438
MV CCA	1.221	1.889	1.462	1.273	1.786	1.109	1.853	1.838

SPOs where the number of training examples is ≤ 5. For example, the column **SP≥15, O≥15** means ≤ 5 examples of an SPO that has at least 15 examples for the SP part and for the O part. An example of this case is when we see zero or very few examples of <person, petting, horse>, but we see at least 15 examples of <person, petting, something=dog/cat/etc (not horse)> and at least 15 examples of something interacting with a horse <*,*, horse>. Model2 performs the best in all the listed generalization cases in Table 5.5. We found a similar generalization behavior for SP facts that have no more than 5 examples during training. We add more figures and additional results in the supplementary materials.

5.6 Conclusion

We introduce new setting for learning unbounded number of facts in images, which could be referred to as a model for gaining visual knowledge. The facts could be of different types like objects, attributes, actions, and interactions. While studying this task, we consider Uniformity, Generalization, Scalability, Bi-directionality, and Structure. We investigated several baselines from multi-view learning literature which were adapted to the proposed setting. We proposed learning representation methods that outperform the designed baseline mainly by the advantage of relating facts by structure.

Chapter 6

SAFA: Sherlock Automatic Fact Annotation

Motivated by the application of fact-level image understanding, we present an automatic method for data collection of structured visual facts from images with captions. Example structured facts include attributed objects (e.g., <flower, red>), actions (e.g., <baby, smile>), interactions (e.g., <man, walking, dog>), and positional information (e.g., <vase, on, table>). The collected annotations are in the form of fact-image pairs (e.g.,<man, walking, dog> and an image region containing this fact). With a language approach, the proposed method is able to collect hundreds of thousands of visual fact annotations with accuracy of 83% according to human judgment. Our method automatically collected more than 380,000 visual fact annotations and more than 110,000 unique visual facts from images with captions and localized them in images in less than one day of processing time on standard CPU platforms. We will make the data publically available.

6.1 Introduction

People generally acquire visual knowledge by exposure to both visual facts and to semantic or language-based representations of these facts, e.g., by seeing an image of "a person petting dog" and observing this visual fact associated with its language representation . In this work, we focus on methods for collecting structured facts that we define as structures that provide attributes about an object, and/or the actions and interactions this object may have with other objects. We introduce the idea of automatically collecting annotations for second order visual facts and third order visual facts where second order facts <S,P> are attributed objects (e.g., <S: car, P: red>) and single-frame actions (e.g., <S: person, P: jumping>), and third order facts specify interactions (i.e., <boy, petting, dog>). This structure is helpful for designing machine learning algorithms that learn deeper image semantics from caption data and allow us

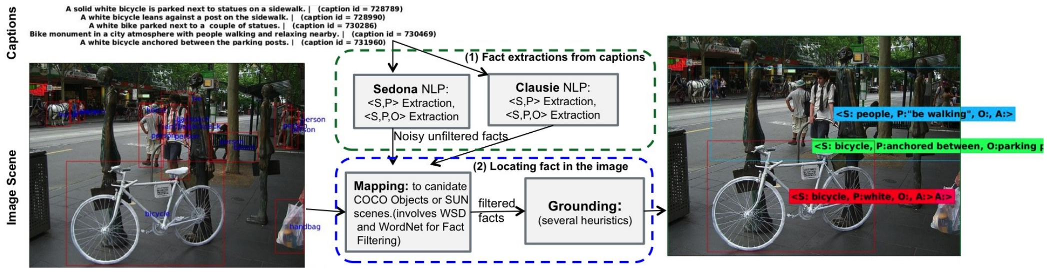


Figure 6.1: Structured Fact Automatic Annotation

to model the relationships between facts. In order to enable such a setting, we need to collect these structured fact annotations in the form of (language view, visual view) pairs (e.g., $<$baby, sitting on, chair$>$ as the language view and an image with this fact as a visual view) to train models.

[21] showed that visual concepts, from a predefined ontology, can be learned by querying the web about these concepts using image-web search engines. More recently, [33] presented an approach to learn concepts related to a particular object by querying the web with Google-N-gram data that has the concept name. There are three limitations to these approaches. (1) It is difficult to define the space of visual knowledge and then search for it. It is further restricting to define it based on a predefined ontology such as [21] or a particular object such as [33]. (2) Using image search is not reliable to collect data for concepts with few images on the web. These methods assume that the top retrieved examples by image-web search are positive examples and that there are images available that are annotated with the searched concept. (3) These concepts/facts are not structured and hence annotations lacks information like “jumping” is the action part in $<$person, jumping $>$, or “man’ and “horse” are interacting in $<$person, riding, horse $>$. This structure is important for deeper understanding of visual data, which is one of the main motivations of this work.

The problems in the prior work motivate us to propose a method to automatically annotate structured facts by processing image caption data since facts in image captions are highly likely to be located in the associated images. We show that a large quantity of high quality structured visual facts could be extracted from caption datasets using natural language processing methods. Caption writing is free-form and an easier task for crowd-sourcing workers than labeling second- and third-order tasks, and such free-form descriptions are readily available in existing image caption datasets. We focused on collecting facts from the MS COCO image caption dataset [94] and the newly collected Flickr30K entities [125]. We automatically collected more than 380,000 structured fact annotations in high quality from both the 120,000 MS COCO scenes and 30,000 Flickr30K scenes.

The main contribution of this paper is an accurate, automatic, and efficient method for extraction of structured fact visual annotations from image-caption datasets, as illustrated in Fig. 6.1. Our approach (1) extracts facts from captions associated with images and then (2) localizes the extracted facts in the image. For fact extraction from captions, We propose a new method called *SedonaNLP* for fact extraction to fill gaps in existing fact extraction from sentence methods like Clausie [27]. SedonaNLP produces more facts than Clausie, especially <subject,attribute> facts, and thus enables collecting more visual annotations than using Clausie alone. The final set of automatic annotations are the set of successfully localized facts in the associated images. We show that these facts are extracted with more than 80% accuracy according to human judgment.

6.2 Motivation

Our goal by proposing this automatic method is to generate language&vision annotations at the fact-level to help study language&vision for the sake of structured understanding of visual facts. Existing systems already work on relating captions directly to the whole image such as [79, 81, 158, 163, 102, 4, 100, 128]. This gives rise to a key question about our work: why it is useful to collect such a large quantity of structured facts compared to caption-level systems?

We illustrate the difference between caption-level learning fact-level learning that motivates this work by the example in Fig 6.1. Caption-level learning systems correlate captions like those on top of Fig. 6.1(top-left) to the whole image that includes all objects. Structured Fact-level learning systems are instead fed with localized annotations for each fact extracted form the image caption; see in Fig. 6.1(right), Fig. 6.6, and 6.7 in Sec. 6.6. Fact level annotations are less confusing training data than sentences because they provide more precise information for both the language and the visual views. **(1)** From the language view, the annotations we generate is precise to list a particular fact (e.g., <bicycle,parked between, parking posts>). **(2)** From the visual view, it provide the bounding box of this fact; see Fig 6.1. **(3)** A third unique part about our annotations is the structure: e.g., <bicycle,parked between, parking posts> instead of “a bicycle parked between parking posts”.

Our collected data has been used to develop methods that learn hundreds of thousands of image facts, as we introduced and studied in [2]. The results shows that fact-level learning is superior compared to caption-level learning like [81], as shown in Table 4 in [2] (16.39% accuracy versus 3.48% for [81]). It further shows the value of the associated structure in the (16.39% accuracy versus 8.1%) in Table 4[2]). Similar results also shown on a smaller scale in Table 3 in [2].

6.3 Approach Overview

We propose a two step automatic annotation of structured facts: (i) Extraction of structured fact from captions, and (ii) Localization of these facts in images. First, the captions associated with the given image are analyzed to extract sets of clauses that are considered as candidate <S,P>, and <S,P,O> facts.

Captions can provide a tremendous amount of information to image understanding systems. However, developing NLP systems to accurately and completely extract structured knowledge from free-form text is an open problem. We extract structured facts using two methods: Clausie [27] and Sedona(detailed later in Sec 6.4); also see Fig 6.1. We found Clausie [27] missed many visual facts in the captions which motivated us to develop Sedona to fill this gap as detailed in Sec. 6.4.

Second, we localize these facts within the image (see Fig. 6.1). The successfully located facts in the images are saved as fact-image annotations that could be used to train visual perception models to learn attributed objects, actions, and interactions. We managed to collect 380.409 high-quality second- and third-order fact annotations (146,515 from Flickr30K Entities, 157,122 from the MS COCO training set, and 76,772 from the MS COCO validation set). We present statistics of the automatically collected facts in the Experiments section. Note that the process of localizing facts in an image is constrained by information in the dataset.

For MS COCO, the dataset contains object annotations for about 80 different objects as provided by the training and validation sets. Although this provides abstract information about objects in each image (e.g., “person”), it is usually mentioned in different ways in the caption. For the “person” object, “man”, “girl”, “kid”, or “child” could instead appear in the caption. In order to locate second- and third-order facts in images, we started by defining visual entities. For the MS COCO dataset [94], we define a visual entity as any noun that is either (1) one of the MS COCO dataset objects, (2) a noun in the WordNet ontology [111, 91] that is an immediate or indirect hyponym of one of the MS COCO objects (since WordNet is searchable by a sense and not a word, we perform word sense disambiguation on the sentences using a state-of-the-art method [172]), or (3) one of scenes the SUN dataset [162] (e.g., a “restaurant”). We expect visual entities to appear either in the S or the O part (if exists) of a candidate fact. This allows us to then localize facts for images in the MS COCO dataset. Given a candidate third-order fact, we first try to assign each S and O to one of the visual entities. If S and O elements are not visual entities, then the fact is ignored. Otherwise, the facts are processed by several heuristics, detailed in Sec 6.5. For instance, our method takes into account that grounding the plural ”men” in the fact <S:men, P: chasing, O: soccer ball > may require the union of multiple ”man” bounding boxes.

In the Flickr30K Entities dataset [125], the bounding box annotations are presented as phrase labels for sentences (for each phrase in a caption that refers to an entity in the scene). A visual entity is considered to be a phrase with a bounding box annotation or one of the SUN scenes. Several heuristics were developed and applied to collect these fact annotations, e.g. grounding a fact about a scene to the entire image; detailed in Sec 6.5.

6.4 Fact Extraction from Captions

We extract facts from captions using Clausie [27] and our proposed SedonaNLP system. In contrast to Clausie, we address several challenging linguistic issues by evolving our NLP pipeline to: 1) correct many common spelling and punctuation mistakes, 2) resolve word sense ambiguity within clauses, and 3) learn a common spatial preposition lexicon (e.g., “next_to”, “on_top_of”, “in_front_of”) that consists of over 110 such terms, as well as a lexicon of over two dozen collection phrase adjectives (e.g., ”group_of”, ”bunch_of”, ”crowd_of”, ”herd_of”). For our purpose, these strategies allowed us to extract more interesting structured facts that Clausie fails at which include (1) more discrimination between single versus plural terms, (2) extracting positional facts (e.g., next_to). Additionally, SedonaNLP produces attribute facts that we denote as <S, A>; see Fig 6.4. Similar to some existing systems OpenNLP [10] and ClearNLP [23], the SedonaNLP platform also performs many common NLP tasks: e.g., sentence segmentation, tokenization, part-of-speech tagging, named entity extraction, chunking, dependency and constituency-based parsing, and coreference resolution. SedonaNLP itself employs both open-source components such as NLTK and WordNet, as well as internally-developed annotation algorithms for POS and clause tagging. These tasks are used to create more advanced functions such as structured fact annotation of images via semantic triple extraction. In our work, we found SedonaNLP and Clausie to be complementary for producing a set of candidate facts for possible localization in the image that resulted in successful annotations.

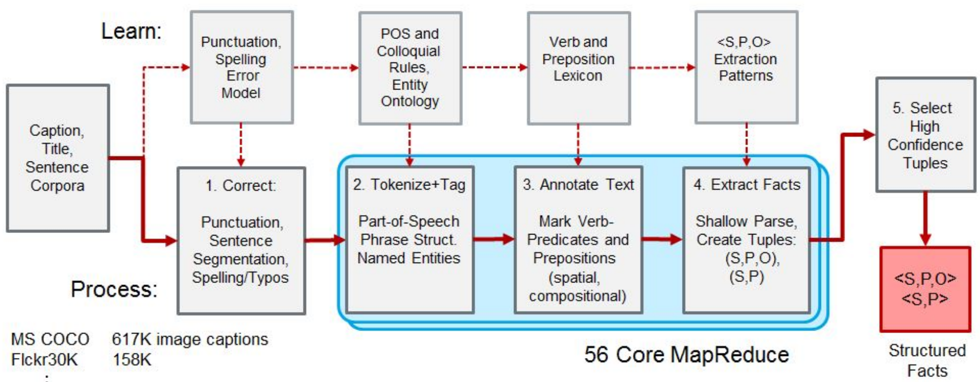


Figure 6.2: SedonaNLP Pipeline for Structured Fact Extraction from Captions

Varying degrees of success have been achieved in extracting and representing structured triples from sentences using <subject, predicate, object> triples. For instance, [137] describe a basic set of methods based on traversing the parse graphs generated by various commonly available parsers. Larger scale text mining methods for learning structured facts for question answering have been developed in the IBM Watson PRISMATIC framework [51]. While parsers such as CoreNLP [101] are available to generate comprehensive dependency graphs, these have historically required significant processing time for each sentence or have traded accuracy for performance. In contrast, SedonaNLP currently employs a shallow dependency

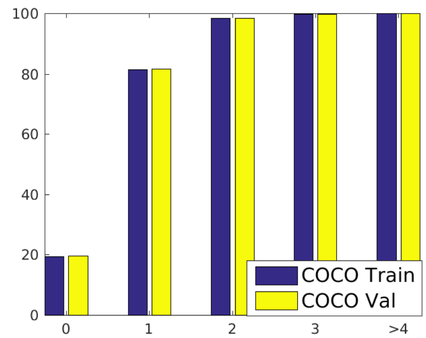


Figure 6.3: Accumulative Percentage of SP and SPO facts in COCO 2014 captions as number of verbs increases

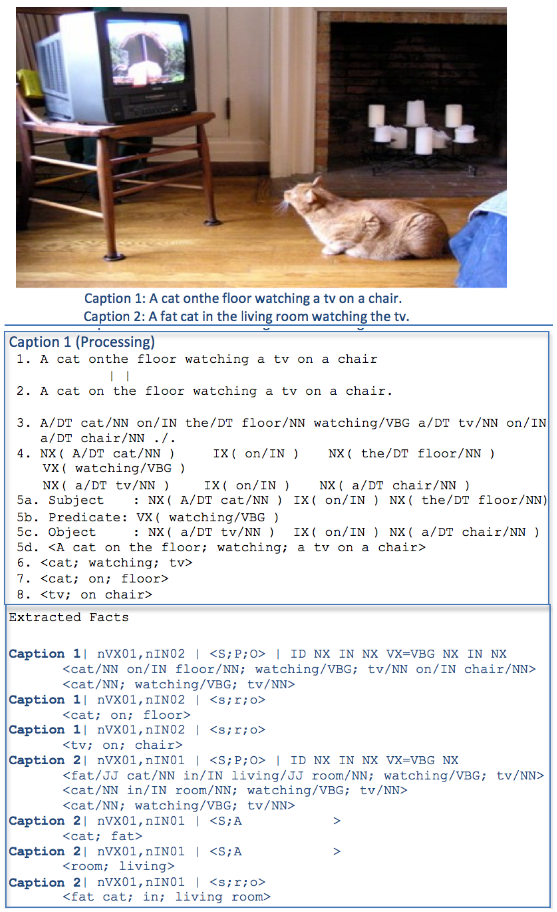


Figure 6.4: Examples of caption processing and <S,P,O> and <S,P> structured fact extractions.

parsing method that runs in some cases 8-9X faster than earlier cited methods running on identical hardware. We choose a shallow approach with high, medium, and low confidence cutoffs after observing that roughly 80% of all captions consisted of 0 or 1 Verb expressions (VX); see Fig. 6.3 for MSCOCO dataset [94]. The top 500 image caption syntactic patterns we observed can be found on our supplemental materials [3]. These syntactic patterns are used to learn rules for automatic extraction for not only <S,P,O>, but also <S,P>, and <S,A>, where <S,P>, are subject-action facts and <S,A> are subject-attribute facts. Pattern examples and statistics for MS COCO are shown in Fig. 6.5.

```
NX VX IN NX              #   1  10.19/10.19 NX(a blue street sign) VX(sitting) IN(under) NX(a camera)
NX VX IN NX IN NX        #   2   9.73/19.92 NX(a brown cat) VX(stares) IN(at) NX(something) IN(in) NX(the field)
NX VX NX IN NX           #   3   7.05/26.97 NX(some sheep) VX(eating) NX(grass) IN(in_front_of) NX(a rock)
NX IN NX IN NX           #   4   5.65/32.62 NX(a) IN(round) NX(blue street sign) IN(with) NX(a white arrow)
NX IN NX VX IN NX        #   5   3.27/35.89 NX(a sign) IN(in_front_of) NX(a fence) VX(laced) IN(with) NX(shrubbery)
NX IN NX                 #   6   3.00/38.89 NX(a orange cat) IN(with) NX(green eyes and long whiskers)
NX VX NX IN NX IN NX     #   7   1.77/40.67 NX(a) VX(very close) NX(shot) IN(of) NX(a cat's face) IN(in_front_of)
                                            NX(the camera)
NX IN NX IN NX IN NX     #   8   1.76/42.43 NX(a toddler reaches) IN(into) NX(a bowl) IN(of) NX(grapes) IN(in)
                                            NX(a sink)
NX IN NX CC NX           #   9   1.70/44.13 NX(a bathroom) IN(with) NX(two sinks mirrors) CC(and) NX(some bottles)
NX IN NX VX NX           #  10   1.69/45.82 NX(a person) IN(on) NX(a skate board) VX(does) NX(a trick)
```

Figure 6.5: Examples of the top observed Noun (NX), Verb (VX), and Preposition (IN) Syntactic patterns.

In SedonaNLP, structured fact extraction was accomplished by learning a subset of abstract syntactic patterns consisting of basic noun, verb, and preposition expressions by analyzing 1.6M caption examples provided by the MS COCO, Flickr30K, and Stony Brook University Im2Text caption datasets. Our approach mirrors existing known art with the addition of internally-developed POS and clause tagging accuracy improvements through the use of heuristics listed below to reduce higher occurrence errors due to systematic parsing errors: (i) Mapping past participles to adjectives (e.g., stained glass), (ii) De-nesting existential facts (e.g., this is a picture of a cat watching a tv.), (iii) Identifying auxiliary verbs (e.g., do verb forms).

In Fig. 6.4, we show an example of extracted <S,P,O> structured facts useful for image annotation for a small sample of MS COCO captions. Our initial experiments empirically confirmed the findings of IBM Watson PRISMATIC researchers who indicated big complex parse trees tend to have more wrong parses. By limiting a frame to be only a small subset of a complex parse tree, we reduce the chance of error parse in each frame [51]. In practice, we observed many correctly extracted structured facts for the more complex sentences (i.e., sentences with multiple VX verb expressions and multiple spatial prepositional expressions) – these facts contained useful information that could have been used in our joint learning model but were conservatively filtered to help ensure the overall accuracy of the facts being presented to our system. As improvements are made to semantic triple extraction and confidence evaluation systems, we see potential in several areas to exploit more structured facts and to filter less information. Our full <S,P,O> triple and related tuple extractions for MS COCO and Flickr30K datasets are available in the supplemental material [3].

6.5 Locating facts in the Image

In this section, we present details about the second step of our automatic annotation process introduced in Sec. 6.3. After the candidate facts are extracted from the sentences, we end up with a set $\mathbf{F}_s = \{\mathbf{f}_l^i\}, i = 1 : N_s$ for statement s, where N_s is the number of extracted candidate fact $\mathbf{f}_l^i, \forall i$ from the statement s using either Clausie [27] or Sedona-3.0. The localization step is further divided into two steps. The mapping step maps nouns in the facts to candidate boxes in the image. The grounding step processes each fact associated with the candidate boxes and outputs a final bounding box if localization is successful. The two steps are detailed in the following subsections.

6.5.1 Mapping

The mapping step starts with a pre-processing step that filters out a non-useful subset of $\mathbf{F}_s$ and produces a more useful set $\mathbf{F}_s^*$ that we try to locate/ground in the image. We perform this step by performing word sense disambiguation using the state-of-the-art method [172]. The word sense disambiguation method provides each word in the statement with a word sense in the wordNet ontology [91]. It also assigns for each word a part of speech tag. Hence, for each extracted candidate fact in $\mathbf{F}_s$ we can verify if it follows the expected part of speech according to [172]. For instance, all S should be nouns, all P should be either verbs or adjectives, and O should be nouns. This results in a filtered set of facts $\mathbf{F}_s^*$. Then, each S is associated with a set of candidate boxes in the image for second- and third-order facts and each O associated with a set or candidate boxes in the image for third-order facts only. Since entities in MSCOCO dataset and Flickr30K are annotated differently, we present how the candidate boxes are determined in each of these datasets.

MS COCO Mapping: Mapping to candidate boxes for MS COCO reduces to assigning the S for second-order and third-order facts, and S and O for third-order facts. Either S or O is assigned to one of the MSCOCO objects or SUN scenes classes. Given the word sense of the given part (S or O), we check if the given sense is a descendant of MSCOCO objects senses in the wordNet ontology. If it is, the given part (S or O) is associated with the set of candidate bounding boxes that belongs to the given object (e.g., all boxes that contain the "person" MSCOCO object is under the "person" wordnet node like "man", 'girl', etc). If the given part (S or O) is not an MSCOCO object or one of its descendants under wordNet, we further check if the given part is one of the SUN dataset scenes. If this condition holds, the given part is associated with a bounding box of the whole image.

Flickr30K Mapping: In contrast to MSCOCO dataset, the bounding box annotation comes for each entity in each statement in Flickr30K dataset. Hence, we compute the candidate bounding box annotations for each candidate fact by searching the entities in the same statement from which the clause is extracted. Candidate boxes are those that have the same name. Similarly, this process assigns S for second-order facts and assigns S and O for second- and third-order facts.

Having finished the mapping process, whether for MSCOCO or Flickr30K, each candidate fact $\mathbf{f}_l^i \in \mathbf{F}_s^*$, is associated with candidate boxes depending on its type as follows.

$<$S,P$>$: Each $\mathbf{f}_l^i \in \mathbf{F}_s^*$ of second-order type is associated with one set of bounding boxes $\mathbf{b}_S^i$, which are the candidate boxes for the S part. $\mathbf{b}_O^i$ could be assumed to be always an empty set for second-order facts.

$<$S,P,O$>$: Each $\mathbf{f}_l^i \in \mathbf{F}_s^*$ of third-order type is associated with two sets of bounding boxes $\mathbf{b}_S^i$ and $\mathbf{b}_S^i$ as candidate boxes for the S and P parts, respectively.

6.5.2 Grounding

The grounding process is the process of associating each $\mathbf{f}_l^i \in \mathbf{F}_s^*$ with an image $\mathbf{f}_v$ by assigning $\mathbf{f}_l$ to a bounding box in the given MS COCO image scene given the $\mathbf{b}_S^i$ and $\mathbf{b}_O^i$ candidate boxes. The grounding process is relatively different for the two dataset due to the difference of the entity annotations.

Grounding: MS COCO dataset (Training and Validation sets)

Table 6.1: Human Subject Evaluation by MTurk workers %

Dataset (responses)	**Q1**		**Q2**		**Q3**						
	yes	**no**	**Yes**	**No**	**a**	**b**	**c**	**d**	**e**	**f**	**g**
MSCOCO train 2014 (4198)	89.06	10.94	87.86	12.14	64.58	12.64	3.51	5.10	0.86	1.57	11.73
MSCOCO val 2014 (3296)	91.73	8.27	91.01	8.99	66.11	14.81	3.64	4.92	1.00	0.70	8.83
Flickr30K Entities2015 (3296)	88.94	11.06	88.19	11.81	70.12	11.31	3.09	2.79	0.82	0.39	11.46
Total	89.84	10.16	88.93	11.07	66.74	12.90	3.42	4.34	0.89	0.95	10.76

Table 6.2: Human Subject Evaluation by Volunteers % (This is another set of annotations different from those evaluated by MTurkers)

Volunteers	**Q1**		**Q2**		**Q3**						
	yes	**No**	**Yes**	**No**	**a**	**b**	**c**	**d**	**e**	**f**	**g**
MSCOCO train 2014 (400)	90.75	9.25	91.25	8.75	73.5	8.25	2.75	6.75	0.5	0.5	7.75
MSCOCO val 2014 (90)	97.77	2.3	94.44	8.75	84.44	8.88	3.33	1.11	0	0	2.22
Flickr30K Entities 2015 (510)	78.24	21.76	73.73	26.27	64.00	4.3	1.7	1.7	0.7	1.18	26.45

In the MS COCO dataset, one challenging aspect is that the S or O can be singular, plural, or referring to the scene. This means that one S could map to multiple boxes in the image. For example, "people" maps to multiple boxes of "person". Furthermore, this case could exist for both the S and the O. In cases where either S or O is plural, the bounding box assigned is the union of all candidate bounding boxes in $\mathbf{b}^i_S$. The grounding then proceeds as follows.

<S,P> facts:

(1) If the computed $\mathbf{b}^i_S = \varnothing$ for the given $\mathbf{f}^i_l$, then $\mathbf{f}^i_l$ fails to ground and is discarded.

(2) If S singular, $\mathbf{f}^i_v$ is the image region that with the largest candidate bounding box in $\mathbf{b}^i_S$.

(3) If S is plural, $\mathbf{f}^i_v$ is the image region that with union of the candidate bounding boxes in $\mathbf{b}^i_S$.

<S,P, O> facts:

(1) If $\mathbf{b}^i_S = \varnothing$ and $\mathbf{b}^i_O = \varnothing$, $\mathbf{f}^i_l$ fails to ground and is ignored.

(2) If $\mathbf{b}^i_S \neq \varnothing$ and $\mathbf{b}^i_O \neq \varnothing$, then bounding boxes are assigned to S and O such that the distance between them is minimized (though if S or O is plural, the assigned bounding box is the union of all bounding boxes for $\mathbf{b}^i_S$ or $\mathbf{b}^i_O$ respectively), and the grounding is assigned the union of the bounding boxes assigned to S and O.

(3) If either $\mathbf{b}_S^i = \varnothing$ or $\mathbf{b}_O^i = \varnothing$, then a bounding box is assigned to the present object (the largest bounding box if singular, or the union of all bounding boxes if plural). If the area of this region compared to the area of the whole scene is greater than a threshold $th = 0.3$, then the $\mathbf{f}_v^i$ is associated to the whole image of the scene. Otherwise, $\mathbf{f}_l^i$ fails to ground and is ignored.

Grounding: Flickr30K dataset The main difference in Flickr30K is that for each entity phrase in a sentence, there is a box in the image. This means there is no need to have cases for single and plural. Since in this case, the word "men" in the sentence will be associated with the set of boxes referred to by "men" in the sentences. We union these boxes for plural words as one candidate box for "men"

We can also use the information that the object box has to refer to a word that is after the subject word, since subject usually occurs earlier in the sentence compared to object. We union these boxes for plural words.

<S,P> facts:

If the computed $\mathbf{b}_S^i = \varnothing$ for the given $\mathbf{f}_l^i$, then $\mathbf{f}_l^i$ fails to ground and is discarded. Otherwise, the fact is assigned to the largest candidate box in if there are multiple boxes.

<S,P, O> facts: <S,P, O> facts are handled very similar to MSCOCO dataset with two main differences.

a) The candidate boxes are computed as described for the case of Flickr30K dataset.

b) All cases are handled as single case, since even plural words are assigned one box based on the nature of the annotations in this dataset.

6.6 Experiments

6.6.1 Human Subject Evaluation

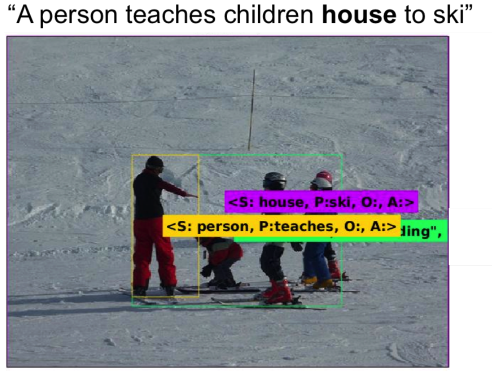
We propose three questions to evaluate each annotation: (Q1) Is the extracted fact correct (Yes/No)? The purpose of this question is to evaluate errors captured by the first step, which extracts facts by Sedona or Clausie. (Q2) Is the fact located in the image (Yes/No)? In some cases, there might be a fact mentioned in the caption that does not exist in the image and is mistakenly considered as an annotation. (Q3) How accurate is the box assigned to a given fact (a to g)? a (about right), b (a bit big), c (a bit small), d (too small), e (too big), f (totally wrong box), g (fact does not exist or other). Our instructions on these questions to the participants can be found in this anonymous url [48].

We evaluate these three questions for the facts that were successfully assigned a box in the image, because the main purpose of this evaluation is to measure the usability of the collected annotations as training data for our model. We created an Amazon Mechanical Turk form to ask these three questions. So far, we collected a total of 10,786 evaluation responses, which are an evaluation of 3,595 ($\mathbf{f}_v, \mathbf{f}_l$) pairs (3 responses/ pair). Table 6.2 shows the evaluation results, which indicate that the data is useful for training, since$\approx$83.1% of them are correct facts with boxes that are either about right, or a bit big or small (a,b,c). We further some evaluation responses that we collected from volunteer researchers in Table 6.2 showing similar results.

Fig. 6.6 shows some successful qualitative results that include four extracted structured facts from MS COCO dataset (e.g., <person, using, phone>, <person, standing>, etc). Fig 6.7 also

Figure 6.6: Several Facts successfully extracted by our method from two MS COCO scenes

"A person teaches children **house** to ski"



<person, teaches>, <house, ski>

Figure 6.7: An example where one of the extracted facts are not correct due to a spelling mistake

show a negative example where there is a wrong fact among the extracted facts (i.e., <house, ski>). The main reason for this failure case is that "how" is mistyped as "house"; see Fig 6.7. The supplementary materials [3] includes all the captions of these examples and also additional qualitative examples.

6.6.2 Hardness Evaluation of the collected data

In order to study how the method behave in both easy and hard examples. This section present statistics of the successfully extracted facts and relate it to the hardness of the extraction of these facts. We start by defining hardness of an extracted fact in our case and its dependency on the fact type. Our method collect both second- and third-order facts. We refer to candidate subjects as all instances of the entity in the image that match the subject type of either a second-order fact <S,P> or a third-order fact <S,P,O>. We refer to candidate objects as all instances in the image that match the object type of a third-order fact <S,P,O>. The selection of the candidate subjects and candidate objects is a part of our method that we detailed in Sec 6.5. We define the hardness for second order facts by the number of candidate subjects and the hardness of third order facts by the number of candidate subjects multiplied by the number of candidate objects.

In Fig 6.8 and 6.9, the Y axis is the number of facts for each bin. The X axis shows the bins that correspond to hardness that we defined for both second and third order fats. Figure 6.8 shows a histogram of the difficulties for all Mturk evaluated examples including both the successful and the failure cases. Figure 6.9 shows a similar histogram but for but for subset of facts verified by the Turkers with Q3 as (about right). The figures show that the method is able to handle difficulty cases even with more than 150 possibilities for grounding. We show these results broken out for MSCOCO and Flickr30K Entities datasets and for each fact types in the supplementary materials [3].

6.7 Conclusion

We present a new method whose main purpose to collect visual fact annotation by a language approach. The collected data help train visual system systems on the fact level with the diversity of facts captured by any fact described by an image caption. We showed the effectiveness of

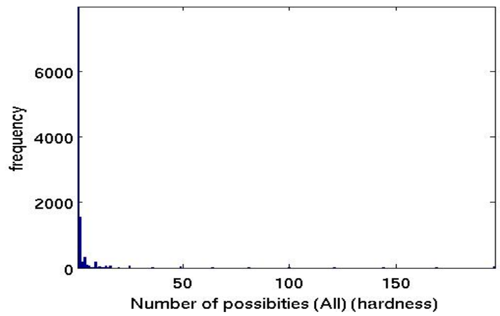


Figure 6.8: (All MTurk Data) Hardness histogram after candidate box selection using our method

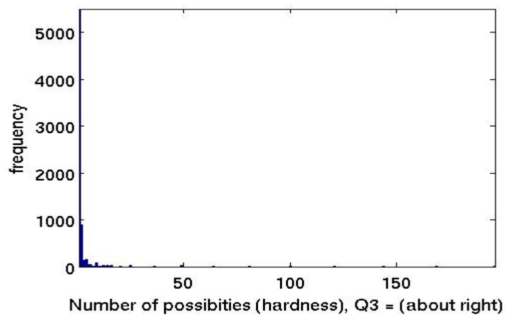


Figure 6.9: (MTurk Data with Q3=about right)Hardness histogram after our candidate box selection

the proposed methodology by extracting hundreds of thousands of fact-level annotations from MSCOCO and Flickr30K datasets. We verified and analyzed the collected data and showed that more than 80% of the collected data are good for training visual systems.

Chapter 7

Conclusion and Future Work

In this thesis, I have explored how to guide visual perception by language on three settings/applications where either zero of very few examples per visual concepts are available.

(A) Write a Classifier In Chapter 2, we showed that zero shot recognition of fine-grained objects is possible from just Wikipedia article describing the visual category. The main idea was to predict a visual classifier from just term frequencies of the Wikipedia articular. We studied several formulations where we showed that the most useful learning component is a transformation matrix W that project the term frequencies of the Wikipedia article into visual classifier space. It could be also improved by a second step with quadratic programming that improve the predicted classifier by sampling negative examples. In Chapter 3, we showed that this transformation could be done in the kernel space, where two arbitrary kernel similarity functions can be use, one between text description and one between images. We also showed the value of using a variant of TFIDF that uses distributional semantics. More recent advances has that regularization that encourage sparsity wikipedia terms, which in turn suppresses the noisy wikipedia terms and hence significantly improve the zero shot performance [126].

(B) Video Event Retrieval from text a Classifier: In Chapter 4, I studied a multimodal setting where an event text query (i.e. just the event title like “birthday party” or “feeding an animal”) to retrieve a ranked list of videos based on their multimodal content. In contrast to [88, 96], I showed that relevant concepts to a given query can be determined automatically by leveraging leverage information from distributional semantic space [107]. The distributional semantic space was trained on a large text corpus to embed event queries and videos to the same space, where similarity between both could be directly estimated. Furthermore, we only assume that query comes in the form of an “unstructured” few-keyword query (in contrast to [161, 77, 76]) which uses a big text articles that explicitly list relevant concepts and yet achieved a better zero-shot performance. This property makes our approach more practical since the typical use of event queries for video search should be similar to text-search (based on few words).

(C) Towards Generalization and Visual Understanding: In (A) and (B), we studied zero-shot learning setting on particular set of classes (A) or events (B). In Chapter 5, we introduce a setting for learning unbounded number offacts in images, which facilitates gaining visual knowledge.While studying this task, we consider Uniformity, Generalization, Scalability, Bi-directionality, and Structure. We investigated several baselines from multi-view learning literature, adapted to our setting. We proposed a structured embedding model that outperform the designed baselines mainly by the advantage of relating facts by structure

My thesis work was an attempt towards better generalization of visual understanding guided by language and encourage future work in several directions. For *(A) Write a Classifier*, we still do not have model that are capable of demonstrating their understanding by relating which sentence in the text description is talking about which part of image (e.g., “orange bill” should be related to the head of the bird in bird images). For *(B) Video Event Retrieval from text a Classifier:* , the average precision on the zero-shot event detection is still 13.1% and improving the results data efficient methods is an open-question to my humble knowledge. For *(C) Towards Generalization and Visual Understanding:* , I just scratched the surface of the problem of rich image understanding. One of the hard questions is building interpretable methods to tackle visual reasoning which is very exciting problem (e.g., inferring hidden facts like mirror for a selfie image). A huge progress has been made in the recent few years in computer vision, yet, there is so much work to be done towards more intelligent machines.

References

[1] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Label-embedding for attribute-based classification. In *CVPR*, 2013.

[2] Anonymous. Sherlock: Scalable fact learning in images.

[3] Anonymous. Autoomatic annotation of structured facts in images- supplementary materials. `https://www.dropbox.com/s/22m6jxvtqhhg10q/supplementary.zip?dl=0`, 2016. [Online; accessed 19-Nov-2015].

[4] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh. Vqa: Visual question answering. In *ICCV*, 2015.

[5] S. Antol, C. L. Zitnick, and D. Parikh. Zero-Shot Learning via Visual Abstraction. In *ECCV*, 2014.

[6] S. Antol, C. L. Zitnick, and D. Parikh. Zero-shot learning via visual abstraction. In *ECCV*. 2014.

[7] J. Ba, K. Swersky, S. Fidler, and R. Salakhutdinov. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *ICCV*, 2015.

[8] A. Bakry, T. Elgaaly, M. Elhoseiny, and A. Elgammal. Joint object recognition and pose estimation using a nonlinear view-invariant latent generative model. In *WACV*, 2016.

[9] A. *Bakry, M. *Elhoseiny, T. *El-Gaaly, and A. Elgammal. Digging deep into the layers of cnns: In search of how cnns achieve view invariance. In *ICLR (* co-first authors)*, 2016.

[10] J. Baldridge. The opennlp project. *URL: http://opennlp.apache.org/index.html,(accessed 2 February 2014)*, 2014.

[11] K. Barnard, P. Duygulu, and D. Forsyth. Clustering art. In *CVPR*, 2001.

[12] E. Bart and S. Ullman. Cross-generalization: Learning novel classes from a single example by feature replacement. In *CVPR*, 2005.

[13] Y. Bengio, H. Schwenk, J.-S. Senécal, F. Morin, and J.-L. Gauvain. Neural probabilistic language models. In *Innovations in Machine Learning*, pages 137–186. 2006.

[14] T. L. Berg, A. C. Berg, and J. Shih. Automatic attribute discovery and characterization from noisy web data. In *ECCV*. 2010.

[15] L. Bo, X. Ren, and D. Fox. Kernel descriptors for visual recognition. In *NIPS*, 2010.

[16] L. Bo and C. Sminchisescu. Twin gaussian processes for structured prediction. *IJCV*, 2010.

[17] S. S. Bucak, R. Jin, and A. K. Jain. Multi-label multiple kernel learning by stochastic approximation: Application to visual object recognition. In *NIPS*, 2010.

[18] Y. Censor and S. Zenios. *Parallel Optimization: Theory, Algorithms, and Applications*. Oxford University Press, USA, 1997.

[19] C.-Y. Chen and K. Grauman. Inferring analogous attributes. In *CVPR*, 2014.

[20] J. Chen, Y. Cui, G. Ye, D. Liu, and S.-F. Chang. Event-driven semantic concept discovery by exploiting weakly tagged internet images. In *ICMR*, 2014.

[21] X. Chen, A. Shrivastava, and A. Gupta. Neil: Extracting visual knowledge from web data. In *ICCV*, 2013.

[22] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *EMNLP*, 2014.

[23] J. D. Choi. Clearnlp, 2014.

[24] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, 2005.

[25] J. Dalton, J. Allan, and P. Mirajkar. Zero-shot video retrieval using content and concepts. In *CIKM*, 2013.

[26] S. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 28(4):357–366, 1980.

[27] L. Del Corro and R. Gemulla. Clausie: clause-based open information extraction. In *WWW*, 2013.

[28] J. Deng, A. C. Berg, K. Li, and L. Fei-Fei. What does classifying more than 10,000 image categories tell us? In *ECCV*. 2010.

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

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

[31] J. Devlin, H. Cheng, H. Fang, S. Gupta, L. Deng, X. He, G. Zweig, and M. Mitchell. Language models for image captioning: The quirks and what works. *arXiv preprint arXiv:1505.01809*, 2015.

[32] J. Devlin, S. Gupta, R. Girshick, M. Mitchell, and C. L. Zitnick. Exploring nearest neighbor approaches for image captioning. *arXiv preprint arXiv:1505.04467*, 2015.

[33] S. K. Divvala, A. Farhadi, and C. Guestrin. Learning everything about anything: Webly-supervised visual concept learning. In *CVPR*, 2014.

[34] 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 *ICML*, 2014.

[35] L. Duan, D. Xu, I. W.-H. Tsang, and J. Luo. Visual event recognition in videos by learning from web data. *TPAMI*, 2012.

[36] M. Elhoseiny, S. Cohen, W. Chang, B. Price, and A. Elgammal. Automatic annotation of structured facts in images. In *ACL 2016 (Vision&Language long-paper workshop proceedings)*, 2016.

[37] M. Elhoseiny, S. Cohen, W. Chang, B. Price, and A. Elgammal. Sherlock: Scalable fact learning in images. *arXiv preprint arXiv:1511.04891*, 2016.

[38] M. Elhoseiny, T. El-Gaaly, C. RUTGERS, C. A. Bakry, and A. Elgammal. A comparative analysis and study of multiview cnn models for joint object categorization and pose estimation. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 888–897, 2016.

[39] M. Elhoseiny and A. Elgammal. Generalized twin gaussian processes using sharma–mittal divergence. *Machine Learning*, 100(2-3):399–424, 2015.

[40] M. Elhoseiny and A. Elgammal. Overlapping domain cover for scalable and accurate regression kernel machines. In *BMVC*, volume 87, pages 28–52, 2015.

[41] M. Elhoseiny and A. Elgammal. Visual classifier prediction by distributional semantic embedding of text descriptions. In *EMNLPW*, 2015.

[42] M. Elhoseiny, A. Elgammal, and B. Saleh. Tell and predict: Kernel classifier prediction for unseen visual classes from unstructured text descriptions. *arXiv preprint arXiv:1506.08529*, 2015.

[43] M. Elhoseiny, A. Elgammal, and B. Saleh. Write a classifier: Predicting visual classifiers from unstructured text descriptions. *IEEE TPAMI submission*, 2016.

[44] M. Elhoseiny, J. Liu, H. Cheng, H. Sawhney, and A. Elgammal. Zero-shot event detection by multimodal distributional semantic embedding of videos. In *AAAI*, 2016.

[45] M. Elhoseiny, B. Saleh, and A. Elgammal. Heterogeneous domain adaptation: Learning visual classifiers from textual description. In *Visual Domain Adaptation Workshop, ICCV*, 2013.

[46] M. Elhoseiny, B. Saleh, and A. Elgammal. Write a classifier: Zero shot learning using purely text descriptions. In *ICCV*, 2013.

[47] M. Elhoseiny, B. Saleh, and A. Elgammal. Write a classifier: Zero-shot learning using purely textual descriptions. In *ICCV*, 2013.

[48] S. Eval. Safa eval instructions (mohamed elhoseiny, scott cohen, walter chang). `https://dl.dropboxusercontent.com/u/479679457/Sherlock_SAFA_eval_Instructions.html`, 2015. [Online; accessed 02-Nov-2015].

[49] P. F. Evangelista, M. J. Embrechts, and B. K. Szymanski. Some properties of the gaussian kernel for one class learning. In *ICANN*, 2007.

[50] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[51] J. Fan, D. Ferrucci, D. Gondek, and A. Kalyanpur. Prismatic: Inducing knowledge from a large scale lexicalized relation resource. In *NAACL HLT*. Association for Computational Linguistics, 2010.

[52] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *CVPR*, 2009.

[53] A. Farhadi, I. Endres, D. Hoiem, and D. A. Forsyth. Describing objects by their attributes. In *CVPR*, 2009.

[54] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. In *ECCV*. 2010.

[55] L. Fe-Fei, R. Fergus, and P. Perona. A bayesian approach to unsupervised one-shot learning of object categories. In *CVPR*, 2003.

[56] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, 2008.

[57] P. Felzenszwalb, D. McAllester, and D. Ramanan. Trecvid 2013 an overview of the goals, tasks, data, evaluation mechanisms and metrics. In *TRECVID 2013*, pages 1–8. NIST, 2013.

[58] R. Fergus, H. Bernal, Y. Weiss, and A. Torralba. Semantic label sharing for learning with many categories. In *ECCV*. 2010.

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

[60] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov, et al. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013.

[61] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013.

[62] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[63] G. Gkioxari, R. Girshick, and J. Malik. Actions and attributes from wholes and parts. 2015.

[64] G. Gkioxari, R. Girshick, and J. Malik. Contextual action recognition with r* cnn. In *ICCV*, 2015.

[65] G. Gkioxari and J. Malik. Finding action tubes. In *CVPR*, 2015.

[66] M. Gonen and E. Alpaydin. Multiple kernel learning algorithms. *JMLR*, 2011.

[67] Y. Gong, Q. Ke, M. Isard, and S. Lazebnik. A multi-view embedding space for modeling internet images, tags, and their semantics. *International journal of computer vision*, 106(2):210–233, 2014.

[68] Google2014. Youtube, howpublished = ”`https://www.youtube.com/yt/press/statistics.html`”, year = 2014, note = ”[online; 11/06/2014]”.

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

[70] A. Gupta. Sports Dataset. `http://www.cs.cmu.edu/~abhinavg/Downloads.html`, 2009. [Online; accessed 15-July-2015].

[71] A. Habibian, T. Mensink, and C. G. Snoek. Composite concept discovery for zero-shot video event detection. In *ICMR*, 2014.

[72] A. E. Hoerl and R. W. Kennard. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 1970.

[73] S. Huang, M. Elhoseiny, A. Elgammal, and D. Yang. Learning hypergraph-regularized attribute predictors. In *CVPR*, 2015.

[74] S. Huang, M. Elhoseiny, A. Elgammal, and D. Yang. Learning hypergraph-regularized attribute predictors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 409–417, 2015.

[75] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *ACM Multimedia*, 2014.

[76] L. Jiang, D. Meng, T. Mitamura, and A. G. Hauptmann. Easy samples first: Self-paced reranking for zero-example multimedia search. In *ACM Multimedia*, 2014.

[77] L. Jiang, T. Mitamura, S.-I. Yu, and A. G. Hauptmann. Zero-example event search using multimodal pseudo relevance feedback. In *ICMR*, 2014.

[78] J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, and L. Fei-Fei. Image retrieval using scene graphs. In *CVPR*, 2015.

[79] A. Karpathy, A. Joulin, and F. F. F. Li. Deep fragment embeddings for bidirectional image sentence mapping. In *Advances in neural information processing systems*, pages 1889–1897, 2014.

[80] J. R. Kiros. Image-sentence tacl15 implementation. `https://github.com/ryankiros/visual-semantic-embedding`, 2015. [Online; accessed 19-Nov-2015].

[81] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *TACL*, 2015.

[82] N. Krishnamoorthy, G. Malkarnenkar, R. Mooney, K. Saenko, U. Lowell, and S. Guadarrama. Generating natural-language video descriptions using text-mined knowledge. *NAACL HLT*, 2013.

[83] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[84] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems (NIPS)*, 2012.

[85] B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *CVPR*, 2011.

[86] 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 *CVPR*, 2011.

[87] C. Lampert, H. Nickisch, and S. Harmeling. Attribute-based classification for zero-shot visual object categorization. *TPAMI*, 36(3):453–465, March 2014.

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

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

[90] H. Larochelle, D. Erhan, and Y. Bengio. Zero-data learning of new tasks. In *AAAI*, 2008.

[91] C. Leacock and M. Chodorow. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database*, 1998.

[92] L.-J. Li, H. Su, L. Fei-Fei, and E. P. Xing. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *NIPS*, 2010.

[93] Z. Li, E. Gavves, T. Mensink, and C. G. Snoek. Attributes make sense on segmented objects. In *ECCV*. 2014.

[94] 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 *ECCV*. Springer, 2014.

[95] J. Liu, B. Kuipers, and S. Savarese. Recognizing human actions by attributes. In *CVPR*, 2011.

[96] J. Liu, Q. Yu, O. Javed, S. Ali, A. Tamrakar, A. Divakaran, H. Cheng, and H. Sawhney. Video event recognition using concept attributes. In *WACV*, 2013.

[97] B. Logan et al. Mel frequency cepstral coefficients for music modeling. In *ISMIR*, 2000.

[98] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[99] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.

[100] M. Malinowski, M. Rohrbach, and M. Fritz. Ask your neurons: A neural-based approach to answering questions about images. In *ICCV*, 2015.

[101] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The stanford corenlp natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, 2014.

[102] J. Mao, W. Xu, Y. Yang, J. Wang, and A. Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). *ICLR*, 2015.

[103] T. Mensink, E. Gavves, and C. G. Snoek. Costa: Co-occurrence statistics for zero-shot classification. In *CVPR*, 2014.

[104] D. Metzler and W. B. Croft. A markov random field model for term dependencies. In *ACM SIGIR*, 2005.

[105] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *ICLR*, 2013.

[106] T. Mikolov, Q. V. Le, and I. Sutskever. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*, 2013.

[107] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013.

[108] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013.

[109] E. G. Miller, N. E. Matsakis, and P. A. Viola. Learning from one example through shared densities on transforms. In *CVPR*, 2000.

[110] G. A. Miller. Wordnet: A lexical database for english. *COMMUNICATIONS OF THE ACM*, 1995.

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

[112] M. Mitchell, J. Dodge, A. Goyal, K. Yamaguchi, K. Stratos, X. Han, A. Mensch, A. C. Berg, T. L. Berg, and H. D. III. Midge: Generating image descriptions from computer vision detections. In *EACL*, 2012.

[113] W. C. B. P. A. E. Mohamed Elhoseiny, Scott Cohen. Automatic annotation of structured facts in images. In *Arxiv*, 2016.

[114] M. Muja and D. Lowe. Flann-fast library for approximate nearest neighbors user manual. *Computer Science Department, University of British Columbia, Vancouver, BC, Canada*, 2009.

[115] G. K. Myers, R. C. Bolles, Q.-T. Luong, J. A. Herson, and H. B. Aradhye. Rectification and recognition of text in 3-d scenes. *IJDAR*, 7, 2005.

[116] M.-E. Nilsback and A. Zisserman. Automated flower classification over large number of classes. In *ICVGIP*, 2008.

[117] 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 *ICLR*, 2014.

[118] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, 2011.

[119] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In *NIPS*, 2009.

[120] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In *NIPS*, 2009.

[121] D. Parikh and K. Grauman. Interactively building a discriminative vocabulary of nameable attributes. In *CVPR*, 2011.

[122] D. Parikh and K. Grauman. Interactively building a discriminative vocabulary of nameable attributes. In *CVPR*, 2011.

[123] G. Patterson and J. Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *CVPR*, 2012.

[124] J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. *EMNLP*, 2014.

[125] B. Plummer, L. Wang, C. Cervantes, J. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015.

[126] R. Qiao, L. Liu, C. Shen, and A. v. d. Hengel. Less is more: zero-shot learning from online textual documents with noise suppression. In *CVPR*, 2016.

[127] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005.

[128] M. Ren, R. Kiros, and R. Zemel. Exploring models and data for image question answering. In *NIPS*, 2015.

[129] M. Rohrbach. Combining visual recognition and computational linguistics: linguistic knowledge for visual recognition and natural language descriptions of visual content. 2014.

[130] M. Rohrbach, S. Ebert, and B. Schiele. Transfer learning in a transductive setting. In *NIPS*, 2013.

[131] M. Rohrbach, S. Ebert, and B. Schiele. Transfer learning in a transductive setting. In *NIPS*. 2013.

[132] M. Rohrbach, M. Stark, and B. Schiele. Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In *CVPR*, 2011.

[133] M. Rohrbach, M. Stark, and B. Schiele. Evaluating knowledge transfer and zero-shot learning in a large-scale setting. In *CVPR*, 2011.

[134] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele. What helps where–and why? semantic relatedness for knowledge transfer. In *CVPR*, 2010.

[135] M. Rohrbach, M. Stark, G. Szarvas, and B. Schiele. Combining language sources and robust semantic relatedness for attribute-based knowledge transfer. In *Parts and Attributes Workshop at ECCV*, 2010.

[136] B. Romera-Paredes and P. Torr. An embarrassingly simple approach to zero-shot learning. In *Proceedings of The 32nd International Conference on Machine Learning*, pages 2152–2161, 2015.

[137] D. Rusu, L. Dali, B. Fortuna, M. Grobelnik, and D. Mladenic. Triplet extraction from sentences. In *International Multiconference" Information Society-IS*, 2007.

[138] M. A. Sadeghi and A. Farhadi. Recognition using visual phrases. In *CVPR*, 2011.

[139] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*. 2010.

[140] R. Salakhutdinov, A. Torralba, and J. Tenenbaum. Learning to share visual appearance for multiclass object detection. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1481–1488. IEEE, 2011.

[141] R. Salakhutdinov, A. Torralba, and J. B. Tenenbaum. Learning to share visual appearance for multiclass object detection. In *CVPR*, 2011.

[142] B. Saleh, A. Farhadi, and A. Elgammal. Object-centric anomaly detection by attribute-based reasoning. In *CVPR*, 2013.

[143] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *IPM*, 1988.

[144] B. Schölkopf, R. Herbrich, and A. J. Smola. A generalized representer theorem. In *COLT*, 2001.

[145] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *ICLR*, 2014.

[146] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil. A convolutional latent semantic model for web search. Technical report, Technical Report MSR-TR-2014-55, Microsoft Research, 2014.

[147] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[148] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[149] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng. Zero-shot learning through cross-modal transfer. In *NIPS*, 2013.

[150] R. Socher, M. Ganjoo, H. Sridhar, O. Bastani, C. D. Manning, and A. Y. Ng. Zero shot learning through cross-modal transfer. In *NIPS*, 2013.

[151] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. June 2015.

[152] M. Torki and A. Elgammal. Putting local features on a manifold. In *CVPR*, 2010.

[153] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *PAMI*, 2008.

[154] L. Torresani, M. Szummer, and A. Fitzgibbon. Efficient object category recognition using classemes. In *ECCV*, 2010.

[155] L. Torresani, M. Szummer, and A. Fitzgibbon. Efficient object category recognition using classemes. In *ECCV*. 2010.

[156] J. van Hout, M. Akbacak, D. Castan, E. Yeh, and M. Sanchez. Extracting spoken and acoustic concepts for multimedia event detection. In *ICASSP*, 2013.

[157] V. Vapnik and A. Vashist. A new learning paradigm: Learning using privileged information. *Neural Networks*, 2009.

[158] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. 2015.

[159] C. Wah and S. Belongie. Attribute-based detection of unfamiliar classes with humans in the loop. In *CVPR*, 2013.

[160] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical report, California Institute of Technology, 2010.

[161] S. Wu, S. Bondugula, F. Luisier, X. Zhuang, and P. Natarajan. Zero-shot event detection using multi-modal fusion of weakly supervised concepts. In *CVPR*, 2014.

[162] J. Xiao, J. Hays, K. Ehinger, A. Oliva, A. Torralba, et al. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010.

[163] K. Xu, J. Ba, R. Kiros, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*. 2015.

[164] J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive svms. In *MULTIMEDIA*, 2007.

[165] L. Yang and A. Hanjalic. Supervised reranking for web image search. In *ACM Multimedia*, 2010.

[166] Y. Yang, C. L. Teo, H. Daumé III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *EMNLP*, 2011.

[167] B. Yao and L. Fei-Fei. Grouplet: A structured image representation for recognizing human and object interactions. In *CVPR*, 2010.

[168] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. Guibas, and L. Fei-Fei. Human action recognition by learning bases of action attributes and parts. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1331–1338. IEEE, 2011.

[169] D. Zeimpekis and E. Gallopoulos. Clsi: A flexible approximation scheme from clustered term-document matrices. In *In SDM*, 2005.

[170] H. Zhang, T. Xu, M. Elhoseiny, X. Huang, S. Zhang, A. Elgammal, and D. Metaxas. Spda-cnn: Unifying semantic part detection and abstraction for fine-grained recognition. In *CVPR*, 2016.

[171] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev. Panda: Pose aligned networks for deep attribute modeling. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1637–1644. IEEE, 2014.

[172] Z. Zhong and H. T. Ng. It makes sense: A wide-coverage word sense disambiguation system for free text. In *ACL*, 2010.

[173] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *NIPS*, 2014.

[174] G. K. Zipf. The psycho-biology of language. 1935.