Exploiting Language Models to Recognize Unseen Actions

Dieu-Thu Le
DISI, University of Trento
dle@disi.unitn.it

Raffaella Bernardi
DISI, University of Trento
bernardi@disi.unitn.it

Jasper Uijlings
DISI, University of Trento
jrr@disi.unitn.it

ABSTRACT

This paper addresses the problem of human action recognition. Typically, visual action recognition systems need visual training examples for all actions that one wants to recognize. However, the total number of possible actions is staggering as not only are there many types of actions but also many possible objects for each action type. Normally, visual training examples are needed for all actions of this combinatorial explosion of possibilities. To address this problem, this paper is a first attempt to propose a general framework for *unseen* action recognition in still images by exploiting both visual and language models. Based on objects recognized in images by means of visual features, the system suggests the most plausible actions exploiting off-the-shelf language models. All components in the framework are trained on universal datasets, hence the system is general, flexible, and able to recognize actions for which no visual training example has been provided. This paper shows that our model yields good performance on unseen action recognition. It even outperforms a state-of-the-art Bag-of-Words model in a realistic scenario where few visual training examples are available.

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*Language Models, Text Analysis*; I.4.8 [**Image Processing and Computer Vision**]: Scene analysis—*Object Recognition*

General Terms

Theory, Experimentation

Keywords

human action recognition, object recognition, language models

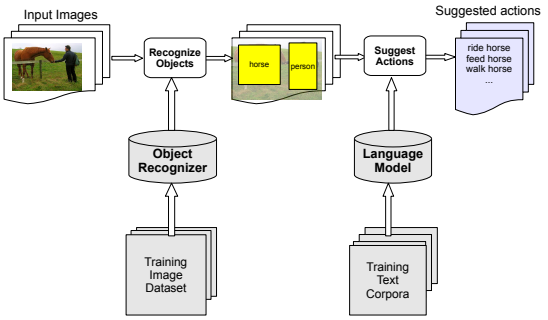

Figure 1: Human action suggestion framework: Object recognizers and language models are learned from general datasets. Actions are suggested based on objects recognized in images.

1. INTRODUCTION

The problem of action recognition has challenged the Computer Vision community for quite a long time. Currently, research on action recognition in still-images focuses on data sets of around 40 human actions defined by "verb-object" relations, like "playing violin" or "riding a bike", where each action has a good number of training examples. However, the combinatorial explosion of verb-object relations makes the task of learning human actions directly from their visual appearance computationally prohibitive and makes the collection of proper-sized image datasets infeasible. Furthermore, actions are a rather complex semantic concept, since an action is expressed by the combination of a verb with an agent and a patient, as well as other possible elements of what, in computational linguistics and artificial intelligence, is known as a "frame". We assume that one can know an action by knowing the frame it belongs to.

Therefore, we aim to develop an action recognizer that can recognize *unseen* actions based on their frames, where unseen means that no visual training examples with action labels are available. Having such a system enables us to handle much more actions than currently considered within the Computer Vision community and will guarantee the scalability and stability of results. To this end, we propose a framework in which the knowledge extracted from language models is learned from an open domain and very large text corpora.

Like other action recognition work, we consider only images that contain human actions. We focus on identifying these actions based on objects which are recognized in the images. In brief, this paper addresses the following research questions: (1) Can language models built from general text corpora

suggest good actions given the objects in the image? (2) How can we integrate a language model with an object recognition model to recognize unseen actions? (3) How does our resulting framework compare to a state-of-the-art Bag-of-Words model on action recognition in a realistic scenario where only few examples are available for training?

2. RELATED WORK

Visual features.

Several researchers noted that actions are highly semantic and can therefore best be recognized through their components rather than global appearance. [6] proposes a model of person-object interaction features based on spatial co-occurrences of body parts and objects, expressing the position in terms of scale-space *coordinates*. [13, 26] shows the importance of exploiting human *poses* too, while [11] investigates the interaction with *spatial information*. Recently, [28] integrated recognized objects, scenes, and human poses into one model: An action is represented by a sparse, weighted sum of action bases, consisting of attributes (verbs related to the action) and parts (objects and poses). All these methods, while successful, need many visual training examples. Our work aims to reduce the reliance on visual training data; we exploit language models to provide probabilities on the relation between those entities for which good detectors exist.

Linguistic features.

Language models have been successfully used in computer vision. In [8], the meaning of images is represented through object-verb-scene triples. A triple works as an intermediate representation of images and descriptive sentences and is used to match the two. [24] also attempt to generate sentences for images by using an online learning method for multi-keyphrase estimation using a grammar model. Similarly, [27] take an image description to consist of a noun, a verb, a scene, and a preposition and aim to generate a never seen descriptive sentence for a given image. To this end, they combine object and scene detectors from computer vision with language models extracted from a dependency parsed corpus to compute the probability of the action and of the preposition to be associated with the image. In particular, they define their vocabulary to consist of verbs, nouns, locations, and prepositions. They select the most likely description by calculating probabilities from co-occurrence statistics from a subset of the Gigaword corpus [10]. Similarly to [27] we extract co-occurrence statistics from a text corpus and transform them into probability scores. Differently from them, we exploit language models which are not tailored to the specific action detection task but which are built independently. This might effect our system performance, but it makes our results more general and stable. Moreover, we perform action recognition rather than generating descriptive sentences.

Unseen action/event recognition.

Several other studies have been able to do unseen event or action recognition. Both [16] and [14] learn attributes of an image. Unseen events can be retrieved by a manual definition of such event in terms of attributes. [22] use a manually defined ontology of events in terms of objects to recognise previously unseen events. In contrast, we learn relations between objects and actions from language.

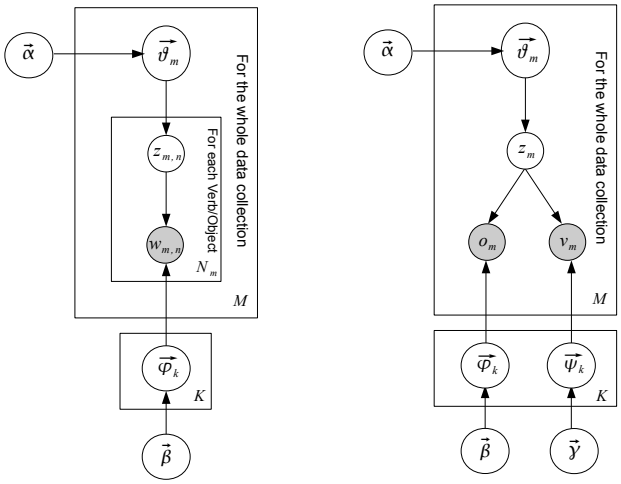

Figure 2: Generative graphical model of LDA (left) vs. ROOTH-LDA (right)

3. RECOGNIZING UNSEEN ACTIONS

We propose a framework (see Figure 1) able to recognize human actions in still images (e.g., "a person is riding a bike") without having previously seen their image representations; we do not rely on the standard visual learning paradigm where many training images are needed to learn a specific action. Instead, we only learn the visual appearance of *objects*. Then we exploit linguistic corpora to learn which verbs can relate a human actor with the visually detected objects. The object appearance models and language models are learned from unrelated datasets, which makes our system highly flexible and ensures stability and scalability.

Specifically, we first learn the appearance models for a set of objects O using standard object localisation systems [9, 25]. Then, given an image I, we use the localization systems to predict the probability of the presence of an object $o^i \in O$. From our previously built *universal* language model, harvested from general text corpora, we predict the probability that a verb v^j is associated with an object: $P(v^j|o^i)$. Hence the object recognizer suggests objects for the language model which in turn finds the most probable actions. We use a weighted linear combination to combine these two probability models for the prediction of the action $a^{ij} \equiv \{o^i, v^j\}$:

$$P(a^{ij}|I) = \alpha \times P(o^i|I) + (1-\alpha) \times P(v^j|o^i, \phi). \quad (1)$$

3.1 Language Model

We exploit a language model to select plausible verbs for a recognized object in an image ($P(V|O)$). Computational linguists have already tackled an analogous task, known as "selectional preferences": compute the plausibility of a noun to be the object of a given verb. Such systems have obtained high correlation with human judgements. In this paper, we compare two language models: We take an off-the-shelf Distributional Semantics Model (DSM) called Type Distributional Memory (TypeDM) [1], and implement ROOTH-LDA [21] which is a variant on Latent Dirichlet Allocation.

TypeDM.

DSMs are based on the hypothesis that the meaning of a word is given by the contexts in which it occurs. Distribu-

tional Memory (DM) [1] is a DSM built as a multitask semantic model, viz. distributional information are extracted once and for all from the corpus in the form of a set of weighted $\langle \text{word}_1, \text{link}, \text{word}_2 \rangle$ tuples; the weights are assigned by Local Mutual Information (LMI); the links could be of different levels of lexicalization giving rise to different DM models. TypeDM has shown to perform best on different tasks. We have used TypeDM directly as a pre-computed semantic resource of weighted tuples: we extracted all tuples in which a verb is linked to a noun $\langle \text{word-v}, \text{link}, \text{word-n} \rangle$, where we ignored *link* for simplicity; we ranked all these tuples based on the noun (the object) and compute the probability of the verb given the object as follows:

$$P(v^j|o^i) = \frac{P(v^j, o^i)}{P(o^i)} = \frac{LMI_{ij}}{\sum_{j=1}^{L} LMI_{ij}} \quad (2)$$

where L is the number of tuples with $w_2 = o^i$, and the LMI are the weights provided by the TypeDM tuples.

Since we took directly the weights of tuples from TypeDM, this model can only predict verb-object pairs that have occurred in the corpora.[1] Every association (V, O) that has not been seen in the corpora will be assigned 0 to its probability.

ROOTH-LDA.

A topic model (e.g., LDA [2]) is a generative model that discovers the abstract "topics" in a collection of documents. LDA was used successfully for preference selection [19, 21].

The most straightforward way of applying LDA (Figure 2, left) provides us with semantic clusters of verbs/objects, but does not jointly model both of them. Therefore it does not provide the conditional probability of a verb given an object. This joint probability is instead obtained by the ROOTH-LDA model (Figure 2, right) proposed in [21] inspired by [20]. We follow this method and adapt it to our goal.

A relation m is a pair of $< v_m, o_m >$, which is generated by picking up a distribution over topics $\overrightarrow{\vartheta}_m$ from a Dirichlet distribution ($Dir(\overrightarrow{\alpha})$). Then the topic assignment z_m for both v_m and o_m is sampled from a multinomial distribution $Mult(\overrightarrow{\vartheta}_m)$. Finally, a particular verb v_m is generated by sampling from multinomial distribution $Mult(\overrightarrow{\psi}_{z_m})$ and a particular object o_m is generated from $Mult(\overrightarrow{\varphi}_{z_m})$ (Figure 2, right). In this way, we keep two different verb-topic and object-topic distributions that share the same topic indicators. We have estimated the model by Gibbs Sampling with relatively simple algorithms following the sampling method for LDA described in [12]. In particular, the topic z_i of a particular verb v_i and object o_i is sampled from the following multinomial distribution:

$$p(z_i = k|\overrightarrow{z}_{\neg i}, \overrightarrow{v}, \overrightarrow{o}) = \frac{n^{(o)}_{k,\neg i} + \beta}{\sum_{o=1}^{V_o} n^{(o)}_{k,\neg i} + V_o \times \beta} \times \frac{n^{(v)}_{k,\neg i} + \gamma}{\sum_{v=1}^{V_v} n^{(v)}_{k,\neg i} + V_v \times \gamma} \times \frac{n^{(k)}_{m,\neg i} + \alpha}{\sum_{k=1}^{K} n^{(k)}_{m,\neg i} + K \times \alpha} \quad (3)$$

where $\overrightarrow{v}$, $\overrightarrow{o}$ and $\overrightarrow{z}$ are the vectors of all verbs, objects and their topic assignment of the whole data collection; α, β, γ are Dirichlet parameters. $n^{(o)}_{k,\neg i}$, $n^{(v)}_{k,\neg i}$ is the number of times object o and verb v is assigned to topic k accept the current one. Let V_o and V_v be the number of objects and verbs in the dataset and K be the number of topics, the two verb-topic and object-topic distributions are computed as:

$$\varphi_{k,o} = \frac{n^{(o)}_k + \beta}{\sum_{o=1}^{V_o} n^{(o)}_k + \beta}; \qquad \psi_{k,v} = \frac{n^{(v)}_k + \gamma}{\sum_{v=1}^{V_v} n^{(v)}_k + \gamma} \quad (4)$$

Finally, to get the conditional probability of a verb v^j given an object o^i, we calculated it through the topic indicator z by summing up over z all products of the conditional probability of the corresponding verb and object given the same topic.

$$\begin{aligned} P(v^j|o^i) &= \frac{P(v^j, o^i)}{P(o^i)} \propto \frac{\sum_{k=1}^{K} P(v^j|z=k) \times P(o^i|z=k)}{\sum_{k=1}^{K} P(o^i, z=k)} \\ &= \frac{\sum_{k=1}^{K} \psi_{v^j,k} \times \varphi_{o^i,k}}{\sum_{k=1}^{K} \varphi_{o^i,k}} \end{aligned} \quad (5)$$

As LDA-ROOTH is a generative model, it also predicts the probability of (V,O) pairs that did not occur in the corpus.

3.2 Object Localisation System

In this paper we use two different object localisation systems [25, 9]. We do not want to base our object recognition on a global image impression, such as the common image-based BoW representation, as an action is really between a human and an object and less dependent on its surroundings.

The two object localisation systems [25, 9] differ in visual features but share similarities in training: Both need training images where objects are annotated using bounding boxes. In both methods, negative examples are automatically obtained from the training data by finding so-called hard examples: image windows that yield high object probabilities but do not correspond to the object. Given an image, both systems predict the most likely bounding boxes where a specific object o^i is present, together with its probability $P(o^i|I)$.

The part-based method of Felzenszwalb et al. [9] is based on a sliding window approach and Histogram of Oriented Gradient (HOG) [5]. For each object class the method automatically determines several poses. For each pose HOG-templates are learned for the complete object and for object-parts, the latter which are automatically determined using a latent, linear SVM. During testing, the HOG-templates are applied to a dense, regular search grid within the image. Locations with the highest template response for both parts and the complete object yield a predicted location with corresponding probability. The framework is widely used and this paper uses their publicly available code (see [9]).

The method of [25] is based on the BoW paradigm [4]. In common BoW, SIFT-descriptors [17] or variants are extracted on a densely sampled grid. Using a previously learned visual vocabulary (e.g. created by kmeans) each SIFT desciptor is assigned to a specific visual word. The BoW representation is given by a histogram of visual word counts within the image, often using the Spatial Pyramid [15] which regularly divides the image to introduce a rough form of spatial consistency.

In [25], the authors propose to represent not a complete image but only the object using BoW. However, such representation is computationally too expensive for a sliding window approach which visits over 100,000 locations. Therefore the authors propose Selective Search which uses multiple hierarchical segmentations to generate around 1500 high-quality,

[1] TypeDM could also be used to compute the plausibility of verb-object pairs never occurred in the corpus.

class independent, object locations. The BoW representation for these 1500 locations can be generated within reasonable time. In this paper, we model the BoW based localisation method after [25], using the publicly available selective search code. The BoW implementation itself is modelled after the fast implementation proposed by [23]. In experiments we denote this BoW localisation method by BoWL.

The details of our implementation are as follows. First, we extract SIFT descriptors [17] and two colour variants, RGB-SIFT and Opponent SIFT [25] at every single pixel in the image (ultra-dense). We use a single scale of 16 by 16 pixels and a Gaussian derivative filter with sigma = 0.667. Principal Component Analysis is used on the descriptors to reduce their dimensionality by a factor 3. Then each descriptor is assigned to a visual word using a Random Forest based visual vocabulary [18, 23], which is as accurate as the usual k-means clustering yet is much more computationally efficient. Specifically, we use four trees of depth ten, resulting in 4096 visual words per SIFT variant. The trees are learned beforehand on a random subset of all descriptors in the training set using the global image labels. The visual words and their locations are stored to be able to quickly compute visual word histograms from subregions within an image.

4. DATASETS

In this section, we will describe all datasets that we use in our experiments: a new action dataset, which contains 89 actions annotated by us to evaluate the performance of the action suggestion system; the image datasets used for training our object recognizers and the corpora from which we have built our language models.

4.1 89 action dataset

Most available datasets used for evaluating action recognizers are restricted to specific domains (e.g., playing musical instruments, sport activities, etc.) or consider a limited number of actions (7 everyday action, Stanford 40 action dataset). Moreover, all these data sets contain many learning examples well distributed over all actions, but this distribution does not reflect the reality where many more possible actions exist for which few examples are available. To overcome these limitations, we have collected a new dataset from 11.5 thousand images of the PASCAL 2012 VOC trainval set [7] selecting all those images representing a human action, obtaining 2,038 images. In PASCAL 2012 VOC there are in total 20 objects. Figure 3 reports for each object the number of images in total and the number of images that contain human actions.

As the images in this dataset were not collected for any specific kind of actions, we believe it gives a general overview of the possible human actions, involving the PASCAL objects. Starting from the object label assigned to each image in the PASCAL data set, we manually annotated the 2,038 images with a verb to obtain the label of the human action (verb-object). The data sets is annotated with 19 objects and 36 verbs, that combine into 89 actions. Considering the training vs. validation split used in the PASCAL competition, our human action data set consists of 1,104 images in the training set and 934 images in the validation set[2].

In the data set, there are objects, such as aeroplane, bird, potted plant, which are associated with only few actions

[2]We made the dataset available at http://disi.unitn.it/ dle/pascalaction.php

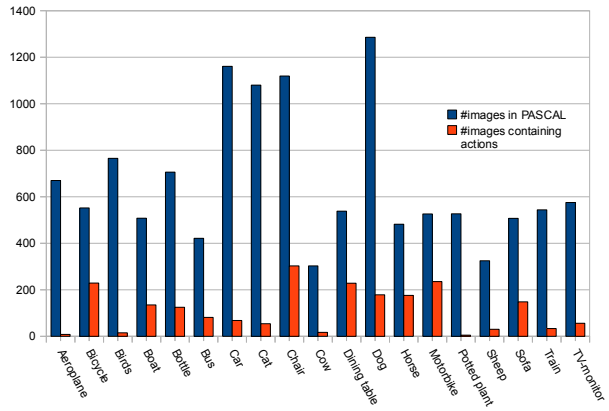

Figure 3: Images containing actions in the PASCAL VOC 2012 trainval dataset

(e.g., 8 images with actions related to aeroplanes, 15 images with birds). Objects that are involved in more actions are bicycle (ride, fix), chair (sit), motorbike (ride), bottle (drink). In many pictures, the action is simply a person touching or holding an object.

4.2 Language datasets

The language models are built from large open domain corpora. TypeDM [1], has been harvested from a concatenation of three corpora: Web-derived ukWac; a mid-2009 dump of the English Wikipedia; and the British National Corpus (BNC).[3] The model contains 2.83 billion tokens: 20,410 nouns and 5,026 verbs.

We have built the LDA-ROOTH model using our implementation (Section 3) estimating it on the BNC, which was PoS-tagged, and lemmatized with TreeTagger[4] and dependency parsed with MaltParser.[5] We have not estimated the LDA-ROOTH model on the whole corpus used to built the TypeDM, since building the LDA-ROOTH model is computationally expensive. The chosen number of topics is 200, the hyper-parameters α, β, γ were set to 0.5, 0.1 and 0.1 respectively and the number of iterations is set to 1,000.

Suggested verb-object combinations look quite interpretable and satisfactory as shown in Table 1. For example, verbs like "wear", "buy", "hang", "design", "dress" have a high weights in the cluster (Topic 46) in which the most probable object is "clothes" ; "spend", "take", "enjoy" are matching with nouns like "time", "day", "hour" (Topic 0); "carry", "conduct" with "research", "interview" (Topic 138) and so on. Totally, there are 33,258 objects and 8,888 verbs.

4.3 Object recognizer dataset

To train the object recognizer, [9, 25] used the trainval set of PASCAL VOC 2007 [7], which contains 20 objects: person, bird, cat, cow, dog, horse, sheep, aeroplane, bicycle, boat, bus, car, motorbike, train, bottle, chair, dining table, potted plant, sofa, TV/monitor. The training set consists of 5,011 images and 12,608 objects. Note that there is no overlap between this dataset and the 2012 VOC dataset from which we created our 89 action dataset.

[3]http://www.natcorp.ox.ac.uk/

[4]http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

[5]http://www.maltparser.org/

Topic 26:				Topic 46:				Topic 0:				Topic 88:			
Verb		Object		Verb		Object		Verb		Object		Verb		Object	
keep	0.56	pace	0.02	wear	0.48	clothes	0.04	spend	0.53	time	0.13	reduce	0.3	cost	0.06
maintain	0.04	record	0.02	remove	0.03	hat	0.03	take	0.02	day	0.06	increase	0.1	risk	0.04
gather	0.03	watch	0.01	buy	0.03	dress	0.02	enjoy	0.01	hour	0.06	cut	0.07	amount	0.02
check	0.02	secret	0.01	take	0.02	jacket	0.02	leave	0.01	year	0.05	incur	0.02	loss	0.01
stand	0.01	distance	0.01	pull	0.02	suit	0.02	last	0.01	night	0.03	control	0.02	emission	0.01
take	0.01	company	0.01	hang	0.01	shirt	0.02	work	0.01	lot	0.02	limit	0.02	number	0.01
mean	0.01	momentum	0.01	don	0.01	coat	0.01	devote	0.01	life	0.02	minimise	0.02	time	0.01
pick	0.01	control	0.01	sport	0.01	shoe	0.01	ask	0.01	evening	0.01	avoid	0.01	pollution	0.01
allow	0.01	child	0.01	put	0.01	uniform	0.01	use	0.01	month	0.01	eliminate	0.01	unemployment	0.01
force	0.01	peace	0.01	design	0.01	trouser	0.01	kill	0.01	week	0.01	involve	0.01	liability	0.01
remain	0.01	house	0.01	get	0.01	cap	0.01	talk	0.01	rest	0.01	reflect	0.01	intake	0.01
stay	0.01	diary	0.01	match	0.01	boot	0.01	visit	0.01	minute	0.01	impose	0.01	expenditure	0.01
steal	0	pressure	0.01	like	0.01	skirt	0.01	mean	0.01	deal	0.01	create	0.01	power	0.01
send	0	mind	0.01	knit	0.01	glass	0.01	read	0.01	part	0.01	assess	0.01	dependence	0.01
step	0	level	0.01	tear	0.01	jean	0.01	waste	0.01	weekend	0.01	curb	0.01	use	0.01

Table 1: Random ROOTH-LDA topics with their most probable verbs and objects

5. EXPERIMENTS

We test the performance of our framework in two settings: *categorization* and *retrieval*. In the *categorization* setting we test how well our framework can predict an action given a specific image and estimate the usefulness of the language model. For evaluation, for each image i we measure the position of the correct action p_i, and report both the average and median position (AvgPos, MedPos) over all N images, where: $AvgPos = \frac{\sum_{i=0}^{N} p_i}{N}$, and $MedPos$ is the median of the set $\{p_1, \cdots, p_n\}$. In the *retrieval* setting we test how good our system is in retrieving images for a particular action. We determine its performance in ranking the images for each action, and measure the Average Precision. We compare our system with a state-of-the-art BoW retrieval framework. Like in most work on human action recognition, we assume all images contain human actions.

5.1 Categorization experiments

We run three kinds of categorization experiments: first we evaluate the language model on its own – hence we take the correct object in the image as given by the gold standard; then we optimize our integration of the language model with an object recognizer. Finally, we evaluate our integrated framework on unseen action recognition.

5.1.1 Language model with object gold standard

In this experiment, we want to determine how well the respective language models can suggest the correct action in an image *given that we know the correct objects that appear in the image*. For the TypeDM, we extract all tuples associated with these objects. Totally, there are 14.2 thousand possible actions related to the 19 objects (viz., the PASCAL 20 objects without "person"). For the LDA-ROOTH model, we generate all possible combinations between these 19 objects and the 8.888 verbs in its vocabulary, obtaining 169 thousand combinations. We use the two models to suggest actions for each image given the correct objects. Remark: there are 5 actions in 36 images that do not occur in TypeDM: clean aeroplane, touch aeroplane, touch bus, touch motorbike, and touch sheep. As we cannot predict the correct action for these images using TypeDM, we cannot measure their position. Therefore we exclude these 36 images from our evaluation.

Figure 4 and Table 2 report the results. First of all, we observe that the average position is 28.8 for TypeDM model and only 73.6 for LDA-ROOTH. Furthermore, the boxplots in Figure 4 show that the average position is significantly affected by a few images for which the position number for the correct action is high. For the median position, which is unaffected by outliers, we see a position of 1 for TypeDM and 45 for LDA-ROOTH. In fact, TypeDM puts the correct action at the first position in 65% of the images(!).

	TypeDM	LDA-ROOTH
Average Position	28.8	73.6
Median Position	1	45

Table 2: AvgPos and MedPos within 2,038 images of the 89 action dataset given object gold standard

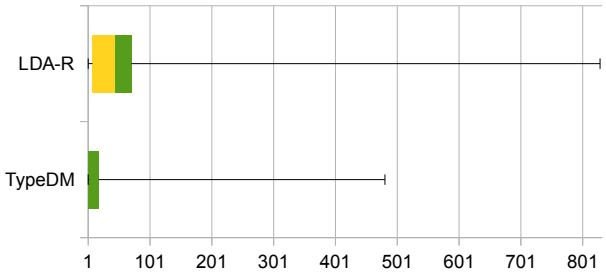

Figure 4: Correct action positions of 2,038 images in the 89 action dataset given the object gold standard. Boxplots show the smallest position, lower quartile, median, upper quartile and highest position

We conclude that TypeDM performs much better than LDA-ROOTH. There is one caveat: TypeDM was learned on more data. But this is made possible because TypeDM is computationally less expensive to learn. Hence, from a practical perspective, TypeDM is the model of choice. In the experiments below, we will evaluate the integration of the object recognizer considering only TypeDM.

5.1.2 Parameter optimization for the integration of the visual and language models

Our aim for this experiment is to find an optimal way to combine TypeDM with the object recognizer to suggest actions for images. We use a weighed linear combination as defined in Equation 1. We experiment with weight values α: $\{0.1, 0.2, ..., 0.9\}$. To avoid overfitting, we report results on three repetitions of two-fold cross-validation (Figure 5). The optimal values are: 0.4 for BoWL and 0.6 for the part-based method. We will use these alphas in the experiments below.

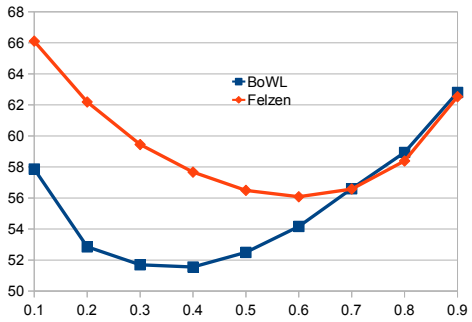

Figure 5: Alpha values and the corresponding average position over three runs

Model	BoWL		Felzen	
	AvgPos	MedPos	AvgPos	MedPos
General model	52.6	5	57.1	6
Tailored model	48.6	4	53.7	5
89 action model	10.3	3	12.3	4

Table 3: Average and median position obtained by two object recognizers integrated with TypeDM

5.1.3 *Integrated visual and language models*

In these experiments, we evaluate how a real system, consisting of a language model and an object recognizer, performs in three different scenarios.

Unbounded Action Prediction.

In this experiment, for each image the model assigns a ranked list of all 14.2 thousand actions in TypeDM, viz., the same scenario considered in the previous experiments. See Table 3 for the results achieved by integrating TypeDM with the BoWL and Felzen object recognizers.

First of all, one can see that the integration of TypeDM with BoWL performs better than the Felzenszwalb object recognizer (Felzen) in terms of AvgPos (52.6 and 57.1, respectively). Moreover, the median for both methods is pretty low: 5 for BoWL and 6 for the part-based object detector. This shows that our combination of TypeDM with the object recogniser yields an accurate action recognition system.

Tailored Action Prediction.

The verb and object co-occurrence frequency in texts may be different from the one in images. Therefore, in this experiment we want to adapt TypeDM to reflect the use of actions in the image dataset so to improve the model predictions.

To do this, we first define *general verbs* as those that go with many different objects. In images, the more objects a verb goes with, the more general it is (Figure 6). The top 5 general verbs based on this definition are: touch, sit, hold, feed, look. Similarly, general verbs in text are those whose probability distributions over objects do not vary much. That means, if a verb goes most of the time with a small number of objects, it is more specific; if a verb occurs with many different objects with similar probability, this verb is more general. Given this definition, we count within 90% of the probability distribution of a verb, how many objects a verb is associated with (Figure 7). The top 5 most general verbs according to this definition are: use, take, get, see, stay.

We first make some qualitative observations: Most of the specific verbs in images are also quite specific in text (≈70% verbs with 1 object in images have $\leq$ 8 objects in text). Most of the general verbs in images are also quite general in text (≈80% verbs with more than 4 objects in images have $\geq$11 objects in text). However, there are some verbs (e.g., push, follow, stay, use) that are general in text but more specific in images. Some specific verbs in text (e.g., ride, feed) are general in the image dataset, this is due to the fact that our image dataset has several objects like sheep, horse, motorbike, bike that often go with these verbs.

To tailor the language model to further improve the performance of the system, we adjust the probability of each verb by exploiting the analysis of verbs in the image dataset. Our tailoring technique is rather soft, since we require to know only the number of the objects that go with each verb in the dataset. Theoretically, the specific objects used here do not need to coincide with the ones from the particular image dataset on which we do action prediction. Therefore, we could also obtain this information from another image dataset. In this paper, we do not, hence there is some bias.

The main idea is to lower down the probability of verbs general in text but specific in images and vice versa. We do this as follows. Let $NO(V)$ be the number of objects a verb V goes with in our image dataset, we tailor the probability as: $P_{\text{tailored}}(V,O) = P(V,O) \times NO(V)$.

The results in Table 3 show that this tailored model achieves better average position than the not-tailored one (from 52.6 to 48.6 and from 57.1 to 53.7 for BoWL and Felzen object recognizer, respectively). In Table 3, the median of BoWL is 4 and Felzen is 5, one position better than the general model. The results show the effectiveness of our tailoring method based on the generality of verbs.

Bounded Action Prediction.

In this last scenario, we assume that we want to predict the presence of an action out of the 89 actions in the image dataset. This setting corresponds to the standard scenario used in the action recognition literature, since most state-of-the-art methods are unable to recognise unseen actions.

As shown in Table 3 the AvgPos for BoWL is 10.3 and for Felzen is 12.3. The AvgPos improvement is mostly due to the fact that the lowest possible position in this evaluation scenario is only 89, it is mostly in the difficult images as it is highlighted by the median: 3 (BoWL) and 4 (Felzen), viz., only one position higher than the results we achieved for the tailored model and only two positions higher than the general model.

We conclude that our framework yields accurate action predictions, even in the more difficult and realistic scenario where the possible actions are not known beforehand.

Human Evaluation.

Finally, we briefly evaluate to which extent the results of the unbounded action scenario are underestimated because of an incomplete annotation. In particular, we randomly selected 100 images, where the gold standard is found within top 40 actions. A human annotator went through the ranked lists proposed by the tailored model using BoWL as object recognizer. Then the annotator manually marked the first correct action in the ranked list by looking at each image.

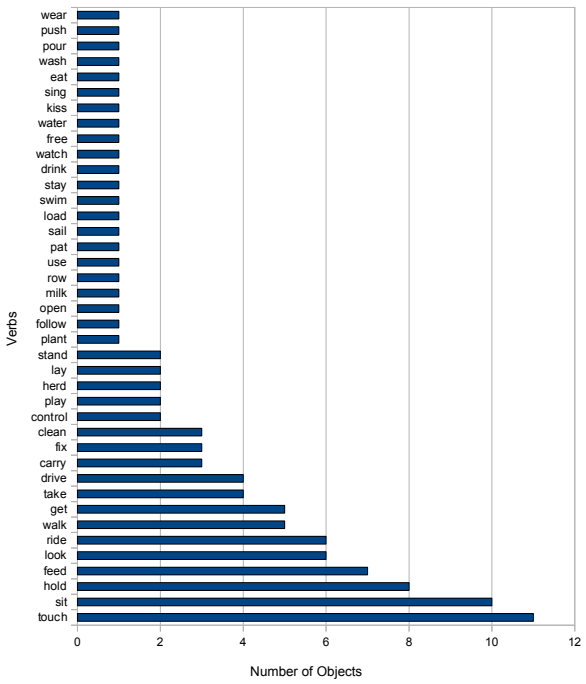

Figure 6: General verbs in images: based on verb-object associations (nr. of objects they go with)

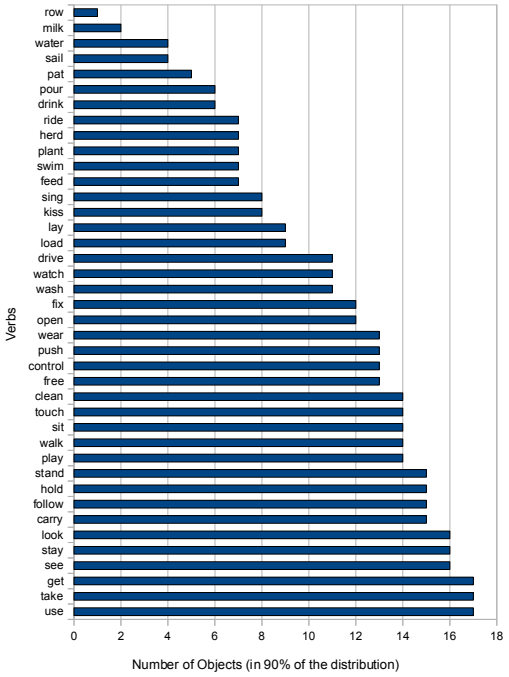

Figure 7: General verbs in text: based on verb-object associations (nr. of objects accounting for 90% of the probability distribution over the total objects of a given verb)

The AvgPos of these 100 images according to the gold standard is 17.2 and the AvgPos according to the human annotation is 5.6. This shows that the action performance of the system could be higher than that based on our current annotation. The reason is that there is usually more than one way of describing the same action in an image and that sometimes there are also different actions presented in the same image. This qualitative analysis suggests that our system for unseen action recognition works even better than is suggested by the experiments above.

5.2 Retrieval experiments

In this section we carry out an image retrieval experiment. We compare our system with a state-of-the-art BoW implementation. This BoW implementation uses the same features as BoWL (see Section 3.2), yet it represents the complete image using a Spatial Pyramid [15] of 1x1 and 1x3. Results on the Pascal VOC 2007 classification challenge are 60.4 MAP (mean average precision), sufficiently close to the 61.7 MAP reported by Chatfield et al. [3].

As the BoW method needs training examples, we split our action dataset into two by using the predefined Pascal 2012 training and validation split. To be able to optimize the parameters of the SVM using cross-validation we demand that an action has at least two training examples. For evaluation, an action should have at least one test example. These constraints results in a data set with 44 actions (whereas our model can retrieve all 84 actions found in the language model (89 minus the 5 not present in TypeDM).

Results on the action retrieval task for the BoW approach and our proposed model are reported in Table 4. Surprisingly, our model with BoWL object recognizer outperforms the BoW approach: 0.22 vs. 0.19 MAP, respectively. The BoW method suffers, of course, from a lack of training examples. Yet our method has only seen the objects itself but never how an action looks like. Still, it gives results slightly better than the BoW system.

We conclude that our system is able to achieve good performance in image retrieval on unseen actions. In a real-world scenario, where training data is limited, our system even outperforms a state-of-the-art BoW implementation.

6. CONCLUSIONS

This paper has presented a framework for unseen action prediction in still images based on visual and language models. Particularly, we used a visual model to detect the appearance and locations of objects, and a language model for inferring the possible relations between these objects. We combine these to recognize unseen actions for which no visual training examples have been provided. All components of the system rely on general datasets and hence can be used to predict actions in any image dataset.

Empirical results on a real image dataset have shown that the system achieved good performance in predicting unseen actions: the median ranking of correct actions of a general model and of a model tailored to the image dataset is 5 and 4, respectively. In a realistic scenario where few training examples are available, our model outperforms with 0.22 MAP, a state-of-the-art Bag-of-Words approach that achieves 0.19 MAP.

In future work we want to investigate other visual information, such as relative positions between objects, scene recognition and exploit language models to find relations between them for a more accurate action prediction. For

Action	Classic BoW	Unseen Felzen	Unseen BoWL	Action	Classic BoW	Unseen Felzen	Unseen BoWL	Action	Classic BoW	Unseen Felzen	Unseen BoWL
drive bus (25)	0.717	0.816	0.814	pat dog (10)	0.083	0.050	0.220	watch TV (8)	0.032	0.114	0.243
sail boat (23)	0.822	0.444	0.657	hold bird (3)	0.015	0.013	0.207	feed bird (2)	0.009	0.005	0.068
sit table (111)	0.678	0.352	0.652	walk horse (8)	0.226	0.064	0.201	touch horse (8)	0.040	0.027	0.062
ride motorbike (85)	0.553	0.448	0.609	hold dog (35)	0.210	0.140	0.191	walk dog (16)	0.144	0.088	0.061
ride horse (75)	0.594	0.669	0.607	get bus (6)	0.118	0.122	0.183	take bus (2)	0.362	0.049	0.054
feed sheep (7)	0.040	0.096	0.540	row boat (24)	0.473	0.105	0.182	stay boat (8)	0.024	0.019	0.032
sit chair (148)	0.410	0.406	0.468	touch cat (7)	0.071	0.041	0.173	sit car (7)	0.354	0.068	0.028
sit sofa (59)	0.371	0.299	0.458	touch dog (6)	0.236	0.028	0.164	play dog (11)	0.020	0.011	0.021
hold cat (19)	0.123	0.060	0.395	lay sofa (11)	0.086	0.034	0.160	touch motorbike (14)	0.100	0.020	0.020
ride bike (84)	0.440	0.489	0.378	drive train (4)	0.074	0.417	0.130	drink bottle (15)	0.024	0.013	0.019
drive car (23)	0.204	0.612	0.367	hold bottle (40)	0.158	0.160	0.126	feed cat (4)	0.007	0.172	0.018
take train (8)	0.108	0.149	0.356	sit motorbike (18)	0.132	0.162	0.118	carry dog (2)	0.003	0.008	0.007
get train (3)	0.031	0.181	0.339	hold bike (10)	0.045	0.056	0.087	push chair (2)	0.009	0.001	0.006
walk bike (14)	0.127	0.192	0.280	herd sheep (2)	0.002	0.006	0.077	feed bottle (5)	0.033	0.008	0.004
milk cow (2)	0.006	0.003	0.003	touch sheep (5)	0.032	0.002	0.002	**MAP**	**0.19**	**0.16**	**0.22**

Table 4: (Mean) Average Precision of Classical BoW and our approach which integrates a Felzen/BoWL object recogniser with TypeDM. The number of training examples for Classical BoW are in brackets.

example, the positions between objects might correlate with prepositions used in language models (e.g., position "on" often goes with "ride horse") and some actions might appear more often in some specific scenes in images as well as in language models. Furthermore, we also want to consider the harder, rarely considered scenario where images may not contain any human action at all.

7. REFERENCES

[1] M. Baroni and A. Lenci. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721, 2010.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

[3] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman. The devil is in the details: an evaluation of recent feature encoding methods. In *BMVC*, 2011.

[4] G. Csurka, C. R. Dance, L. Fan, J. Willamowski, and C. Bray. Visual Categorization with Bags of Keypoints. In *ECCV International Workshop on Statistical Learning in Computer Vision*, Prague, 2004.

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

[6] V. Delaitre, J. Sivic, and I. Laptev. Learning person-object interactions for action recognition in still images. In *NIPS*, 2011.

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. IJCV, 2010.

[8] 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*. Springer, 2010.

[9] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *TPAMI*, 2010.

[10] D. Graff and C. Cieri. English gigaword. In *Linguistic Data Consortium*. 2003.

[11] A. Gupta, A. Kembhavi, and L. S. Davis. Observing human-object interactions: Using spatial and functional compatibility for recognition. In *TPAMI*, volume 31, pages 1775–1789, 2009.

[12] G. Heinrich. Parameter estimation for text analysis. Technical report, 2004.

[13] N. Ikizler, R. G. Cinbis, S. Pehlivan, and P. Duygulu. Recognizing actions in still images. In *ICPR*, 2008.

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

[15] S. Lazebnik, C. Schmid, and J. Ponce. Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories. In *CVPR*, New York, 2006.

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

[17] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 60:91–110, 2004.

[18] F. Moosmann, B. Triggs, and F. Jurie. Fast discriminative visual codebooks using randomized clustering forests. In *NIPS*, pages 985–992, 2006.

[19] A. Ritter, Mausam, and O. Eztioni. A latent dirichlet allocation method for selectional preferences. In *ACL*, 2010.

[20] M. Rooth, S. Riezler, D. Prescher, G. Carroll, and F. Beil. Inducing a semantically annotated lexicon via em-based clustering. In *ACL*, 1999.

[21] D. O. Séaghdha. Latent variable models of selection preference. In *ACL*, 2010.

[22] J. Stottinger, J. Uijlings, A. Pandey, N. Sebe, and F. Giunchiglia. (unseen) event recognition via semantic compositionality. In *CVPR*, 2012.

[23] J. R. R. Uijlings, A. W. M. Smeulders, and R. J. H. Scha. Real-time Visual Concept Classification. *IEEE Transactions on Multimedia*, 12, 2010.

[24] Y. Ushiku, T. Harada, and Y. Kuniyoshi. Efficient image annotation for automatic sentence generation. In *ACM MM*, 2012.

[25] K. E. A. van de Sande, J. Uijlings, T. Gevers, and A. Smeulders. Segmentation as Selective Search for Object Recognition. In *ICCV*, 2011.

[26] Y. Wang, H. Jiang, M. S. Drew, Z.-N. Li, and G. Mori. Unsupervised discovery of action classes. In *CVPR*, 2006.

[27] Y. Yang, C. L. Teo, H. Daumé, III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. EMNLP, Stroudsburg, PA, USA, 2011.

[28] B. Yao, X. Jiang, A. Khosla, A. L. Lin, L. J. Guibas, and L. Fei-Fei. Action recognition by learning bases of action attributes and parts. In *ICCV*, 2011.