YouTube2Text: Recognizing and Describing Arbitrary Activities Using Semantic Hierarchies and Zero-shot Recognition

Sergio Guadarrama
UC Berkeley

Niveda Krishnamoorthy
UT Austin

Girish Malkarnenkar
UT Austin

Subhashini Venugopalan
UT Austin

Raymond Mooney
UT Austin

Trevor Darrell
ICSI, UC Berkeley

Kate Saenko
UMass Lowell

Abstract

Despite a recent push towards large-scale object recognition, activity recognition remains limited to narrow domains and small vocabularies of actions. In this paper, we tackle the challenge of recognizing and describing activities "in-the-wild". We present a solution that takes a short video clip and outputs a brief sentence that sums up the main activity in the video, such as the actor, the action and its object. Unlike previous work, our approach works on out-of-domain actions: it does not require training videos of the exact activity. If it cannot find an accurate prediction for a pre-trained model, it finds a less specific answer that is also plausible from a pragmatic standpoint. We use semantic hierarchies learned from the data to help to choose an appropriate level of generalization, and priors learned from web-scale natural language corpora to penalize unlikely combinations of actors/actions/objects; we also use a web-scale language model to "fill in" novel verbs, i.e. when the verb does not appear in the training set. We evaluate our method on a large YouTube corpus and demonstrate it is able to generate short sentence descriptions of video clips better than baseline approaches.

1. Introduction

Automatic understanding and description of activities "in-the-wild" remains challenging despite recent advances in activity recognition [27, 24, 2, 5]. While object recognition methods have gone large-scale, with datasets such as ImageNet[7] and LabelMe [25], activity datasets have lagged behind. Most activity recognition methods focus on narrow domains with a handful of actions (e.g., Figure 1(a-b)), in which labelled example videos of all actions (as well as actors and objects) are available for training. Contrast these with YouTube videos (Figure 1(c)), where the range of both activities and objects is broad and training data for each label is scarce or unavailable.

Figure 1: Unlike conventional approaches which recognize a limited set of actions in datasets like (a-b), we recognize a broad range of activities in short YouTube video clips, and generate brief text summaries similar to the human-generated ones shown in (c).

Furthermore, methods for generating human-understandable natural language descriptions, such as those shown in Figure 1(c), have yet to scale to such broad domains. Potential applications of automatic video description include summarization, text-based retrieval, or auto-captioning for the visually impaired. Recent results on activity description in video have been restricted to a small set of actions and objects [12, 2]. Work on large-vocabulary description has focused mostly on nouns/adjectives and on still imagery [19]. Large-vocabulary video activity description present unique challenges, including modeling dynamics and actor-action-object relationships from limited training data, as well as dealing with polysemy and ambiguity.

In this paper, we take steps towards scalable "in-the-wild" description of short videos by making two observations: 1) that there are several valid ways to describe the same activity, and 2) the description does not need to be

very specific to be useful. This is illustrated by Figure 1(c), where the top-left video was described by human annotators (who were asked to describe the main activity) as either *"a person cooking"*, *"a woman chopping vegetables"*, or *"a cook preparing a meal"*, etc. We see that not only do people use the different verbs ("chop", "slice") and nouns ("woman", "cook") interchangeably for descriptions, but also use words with varying specificity, i.e. "cooking" is less specific than "slicing", but still useful in conveying the gist of the activity.

Inspired by these insights, we propose a novel language-driven approach to describe short YouTube videos. First, we mine the natural sentence descriptions provided by humans to learn semantic relationships. For example, "beach" and "rocks" are similar in that they are both correlated with "play", as in "playing on the beach", or "playing on the rocks". Then, based on these relationships, we build semantic hierarchies and develop a model that is able to trade off between more or less general descriptions. Our model is related to Deng et al.[6], but formed over Subject/Verb/Object (S,V,O) *relationships*, and optimized with respect to *semantic* similarity (Section 4). Figure 2 illustrates the advantage of our approach: while the conventional method attempts (unsuccessfully) to predict the most accurate basic category classifiers (leaf nodes), our model "backs off" to the more general phrase that is more accurate visually *and* with respect to the human annotations.

Another contribution of our paper is to use a web-scale language model to "fill in" novel verbs, i.e. when the verb does not appear in the training set. We call this "zero-shot" verb recognition. Intuitively, it works by hypothesising that, e.g., given the subject "person", object "car" and the model prediction "move", the most likely verb is "drive". Our final contribution is to provide an end-to-end generation system, complete with surface realization of the best predicted subject/verb/object triple as a grammatical sentence.

2. Background and Related Work

Most prior work on natural-language description of visual data has focused on static images [30, 15, 19, 10, 29]. S. Li et al. in [19] generate sentences given visual detections of objects, visual attributes and spatial relationships, but do not consider actions. A. Farhadi et al. proposed a system [10] that maps images and the corresponding textual descriptions to a "meaning" space with an object, action and scene triplet, but deal with a fixed small set of training triplets. Y. Yang et al. in [29] used text-mined knowledge to generate descriptions of static images after performing object and scene detection, but do not perform activity recognition.

The existing work on describing videos with sentences [12, 17, 13, 8, 14, 5] deals with constrained domains with a limited set of actions or objects, and does not exploit text

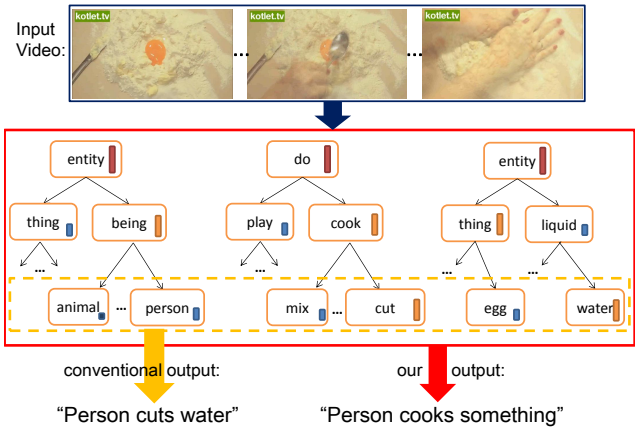


Figure 2: Conventional methods try to predict a caption composed of the most visually likely objects and actions (leaf nodes), whereas our method can predict a less specific phrase that is nonetheless visually plausible and informative. The bars inside nodes indicate the posterior probability of the node given the input video (more red and taller indicates higher probability).

mining or semantic hierarchies. T.S. Motwani in [20] explored how object detection and text mining can aid activity recognition in videos; however, they do not determine a complete SVO triple for describing a video nor generate a full sentential description. Our work also differs from all previous description work in that we reason over hierarchical phrases, allowing us to predict more general but more semantically correct sentences when the visual detectors are unsure. A similar recent approach [6] also trades off accuracy and specificity, but does not deal with video or optimize over SVO triplets. Ours is also the first approach to evaluate on generic, large and diverse set of challenging YouTube videos that cover an unprecedented range of activities. There has been existing work on "in-the-wild" video tagging on YouTube (e.g., [1]) but their focus is on mining text tags, and not on describing actions.

3. Overview of our Approach

Our overall activity description approach consists of the following main steps. We first mine (S,V,O) triplets from the natural language descriptions of the videos, and build a separate semantic hierarchy for each part of the triplet (H_S, H_V, and H_O). Second, we learn a visual model for each leaf of H_S, H_V, and H_O. Third, we learn to predict node triplets over the learned hierarchies by maximizing the semantic similarity to the training data. These first three steps are described in Section 4. Fourth, we learn a language model from web-scale text corpora and use it as a prior on triplets, to infer verbs missing from our vocabulary. Finally,

we generate sentences based on the best triplets. These last two steps are described in Section 5.

We demonstrate our method on a diverse activity dataset. Previously published activity recognition methods that work on datasets such as KTH [26], Drinking and Smoking [16] and UCF50 [24] have a very limited recognition vocabulary of activity classes, ranging from 6 to 12 action classes. Our dataset on the contrary contains more than 218 different verbs in the human descriptions (see Section 4), and over 241 different objects. We use the English portion of the YouTube data collected by Chen et al. [4], consisting of 1,970 short video clips, each with an average of 16 natural-language descriptions provided by Amazon Mechanical Turk workers. Subsets of this data were previously used by [20] and [14], however, contrary to these works, we use all the videos, and not only the 20 objects included in the PASCAL dataset [9].

4. Hierarchical Semantic Model

Building the Semantic Hierarchies: We capitalize on the rich linguistic variation in the corpus to learn semantic hierarchies suitable for video activity description. We followed Motwani and Mooney's [20] approach to automatically extract semantic SVO triplets from the human generated sentences. We then filtered those labels that don't appear at least in the description of 5 videos, obtaining 45 Subjects, 218 Verbs, and 241 Objects. During this filtering, we allow synonyms of nouns by including all words with a Lesk similarity (as implemented in [21]) of at least 0.5. For instance, this groups together "person, he, she, man, woman, someone, ..." into "person", and "car, auto, automobile, motorcar" into "car".

We tested different ways to build the hierarchies for subjects, verbs and objects using the idea of distributional clustering [22] and co-occurrence of the labels. Using 5 fold cross-validation on the training data, we found that the best way was to use the sample Spearman rank correlation between their number of mentions in the descriptions of the videos to compute the distance between labels, and the average distance between all pairs of labels to compute the distance between clusters (see Figure 3 for examples).

As can be seen in Figure 3, the learned hierarchies group labels that would be separated in the WordNet hierarchy. For example for the group of verbs "cut, chop, dice, piece, slice", WordNet would group "cut" with "tear, trim, drill" instead of with "chop, dice, piece, slice"; and the group of objects "food, dish, noodle, pasta, plate, spaghetti" would be split into "dish, plate", "food" and "pasta, noodle and spaghetti". Thus our learned hierarchies capture better the similarity of nouns and verbs in terms of how they are used to describe activities.

Defining Semantic Accuracy over Hierarchies: We construct a hierarchy $H = (V, E)$ of labels for each category

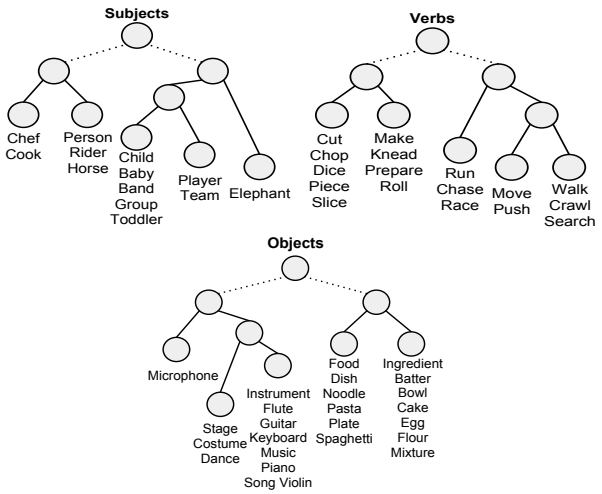


Figure 3: Small portions of the Hierarchies learned over Subjects, Verbs and Objects

(S,V, and O), as tree with an unique root $\hat{v} \in V$, and where each node $v \in V$ represents the set of labels under it (see Figure 3). We use two measures of accuracy, the first is a 0-1 loss defined over the leaf or internal nodes (similar to the accuracy defined in [6]) and the second is based on a similarity function between pairs of nodes in the hierarchy as described below. A matching function $\mu_{L_t} : V \rightarrow [0, 1]$ is then defined over a hierarchy H with respect to a ground truth set leaf nodes $L_t \subset L$ (L is the set of all leafs in the tree) as:

$$\mu_{L_t}(v) = \underset{l \in L_t}{max} \{s_t(v, l)\} \tag{1}$$

where s_t is the similarity between any two nodes in the hierarchy.

We define a binary accuracy s_{01} at the leaf nodes of the hierarchy as the 0-1 loss:

$$s_{01}(v, l) = \mathbb{I}[v{==}l] \tag{2}$$

The 0-1 loss can be unduly harsh, for instance if we incorrectly choose "pasta" instead of a "spaghetti" as the object, it should be considered better than choosing "guitar". Similarly, predicting "cut" instead of "make" is better than choosing "run". In order to account for such similarities, we use WUP similarity [28] between the predicted and correct items (other similarity functions defined over semantic hierarchies could also be used):

$$s_{\text{WUP}}(v, l) = \frac{2 \cdot \text{depth}(lcs)}{\text{depth}(v) + \text{depth}(l)} \tag{3}$$

where lcs is the least common ancestor of v and l, and depth is the depth in the hierarchy, starting at 1 for the root. For example, s_{WUP}(motorbike, dog)=0.10,

s_{WUP}(motorbike, bicycle)=0.78, s_{WUP}(slice, go)=0.28, s_{WUP}(slice, chop)=0.80.

The accuracy $\phi_H(f)$ of the classifier f with respect to a hierarchy H is then:

$$\phi_H(f) = \mathbb{E}[\mu_{L_t}(f(X))]$$

Visual Leaf Node Classifiers: To train visual classifiers for the leaf node concepts we use the following steps: First, for each video we extract Dense Trajectories [27] and encode each activity descriptor using a previously learned codebook. Second, for each video we extract 2 frames per second, and for each frame, apply the object detectors proposed in [11] and [18], and select the maximum score assigned to each object in any frame. Then we combine activity and object descriptors using a multi-channel approach [31] and pass it to a non-linear SVM [3] (see Section 7).

There are many objects for which we lack a pre-trained model, and we have models for several objects that don't appear in the dataset. As in [18], we use the outputs of the object detections as features and learn leaf node classifiers for each individual Subject, Verb, and Object.

Optimizing Phrase Specificity for Maximum Semantic Similarity: Once the leaf visual classifiers are trained, we use the hierarchies constructed in Section 4 to predict nodes by trading off specificity with semantic similarity, i.e. how semantically close the predicted triplet is to the true action.

The specificity $\psi_H(v)$ of node in a hierarchy is defined by the decrease in entropy:

$$\psi_H(v) = \log_2 |L| - \log_2 \sum_{l \in L} [v \in \pi(l)]$$

Given the specificity of a node and the posterior probability $p_{Y|X}(v|x)$ over the set of nodes, we define a classifier that trades-off between accuracy and specificity using a parameter λ as follows:

$$f_\lambda(x) = \underset{v \in V}{\text{argmax}} (\psi_H(v) + \lambda) p_{Y|X}(v|x)$$

In a way similar to that proposed in [6], we obtain the posterior probabilities by learning one-vs-all SVM classifiers for the leaf nodes, obtain probability estimates via Platt scaling [23], and sum them to get internal node probabilities. The difference in our case is that, instead of using the WordNet hierarchy, we use the hierarchy learned from the data (as explained in section 4).

Another crucial difference between from [6] is the way we estimate λ. Instead of fixing it by specifying a desired accuracy (for example 90%) for each hierarchy, we set λ to maximize the WUP similarity between the predicted *triplet* and the set of correct triplets using cross-validation on the training set. This allows our model to trade off specificity by exploiting the relationships between valid combinations of subjects, verbs and objects, whereas simply fixing a high accuracy can lead to over-generalization (see results in Section 7).

Since our internal nodes are set of labels, to generate a sentence we must pick a representative word for them. Choose the node in WordNet with highest cumulative WUP similarity to all the labels in the set. For example, for the prediction ('person', 'cut', 'carrot, tomato, potato, onion, bread') it chooses (person, cut, vegetable).

5. Zero-shot Language Model

For zero-shot activity recognition, knowledge mined from web-scale textual corpora can help determine unseen verbs for describing the video. In order to discover activities that were unseen during training, we expand the top detected verbs with their most similar verbs to generate a larger set of potential verbs for describing the action. Text-mined likelihoods are then used to determine the activity that best fits the detected objects. For example, if "person" and "car" are the top subject and object detections and "move" is the top verb detection, we can expand "move" with similar verbs like "ride" and "drive" to describe the video as "A person is driving a car" without needing any training videos for "ride" or "drive". This idea can be used to expand "coarse" activity detections, obtained by training classifiers on available (possibly limited) activity training data, with "finer" activities unseen at training time.

We employ language models trained on four large text corpora (English Gigaword - 1200 million words, British National Corpus – 100 million words, ukWac – 2000 million words, and WaCkypedia_EN – 800 million words) for obtaining S-V-O triplet likelihoods.

For zero-shot detection we follow the method suggested in [14] and expand the detections of observed verbs with their most similar verbs from the set of unseen verbs. To combine the vision detection scores with their real-world likelihood and determine the best overall SVO, we use simple linear interpolation. When computing the overall vision score, we make a conditional independence assumption and multiply the probabilities of the subject, activity and object. To account for expanded verbs, we additionally multiply by the WUP similarity between the original (V_{orig}) and expanded (V_{exp}) verbs. The resulting SVO triplets are then scored using Equation 4 to select the best triplet.

$$\begin{aligned} score = P(S|vid) * P(V_{exp}|vid) * Sim(V_{exp}, V_{orig}) \\ * P(O|vid) * svo\_likelihood \end{aligned} \quad (4)$$

A template-based approach is utilized for surface realization such that each sentence is of the form:

"Determiner (A,An,The) - Subject - Verb (Present, Present Continuous) - Preposition - Determiner - Object."

where the subject, verb and object are obtained from the

content planning stage. Determiners and prepositions are allowed to be optional. To get a list of appropriate prepositions, we mine the text corpora for "prep_" dependencies and for every verb-object combination we find the most frequently occurring prepositions. The candidate sentences generated using the template above, are ranked for plausibilty using a language model trained on the GoogleNgram corpus and the top ranked sentence is used to describe the video.

6. Experimental Setup

We split the 1,970 videos in the YouTube corpus into two: (1,300) for training and validation, and (670) for test; splits were contiguous groups of videos by index number to reduce any locality effect in the dataset.

Activity Descriptors: We used the motion descriptors developed by Wang et al [27] as they achieve state-of-the-art activity recognition performance; their approach extracts dense trajectories and computes HoG (Histograms of Oriented Gradients), HoF (Histograms of Optical Flow) and MBH (Motion Boundary Histogram) features over space-time volumes around the trajectories. We used the default parameters proposed in their paper and their code ($N = 32$, $n_\sigma = 2$, $n_r = 3$), and adopted a standard bag-of-features representation. We construct a codebook for each descriptor (Trajectory, HoG, HoF, MBH) separately. For each descriptor we randomly sampled 100K points and clustered them using K-means into a codebook of 4000 words. Descriptors are assigned to their closest vocabulary word using Euclidean distance. Each video is then represented as a histogram over these clusters.

Object Descriptors: Our object descriptors are based on the well-known Deformable Parts Model (DPM) [11] as it achieves state-of-the-art performance on the PASCAL (VOC) Challenge, and also forms the substrate for the ObjectBank representation [18], which has itself demonstrated strong performance as a concept representation in contemporary challenges. For each extracted keyframe, we computed DPM-based representations using ObjectBank and the standard PASCAL object classes. We defined a feature vector corresponding to the PASCAL classes based on the max score of each category per image, and returned both the PASCAL scores and the ObjectBank scores with max-pooling over the set of frames, as the object descriptors for a video clip.

Multi-channel SVM: For classification we use a non-linear SVM [3] and combine the information from both object and activity features using a multi-channel approach as proposed in [31], with a RBF-kernel over the pairwise distances:

$$K(x_i, x_j) = \exp\Big(- \sum_c \frac{1}{A^c} D_c(x_i^c, x_j^c) \Big) \qquad (5)$$

where $D_c(x_i^c, x_j^c)$ is the distance between video x_i and x_j with respect to the c-th channel, and A^c is the mean value of the distances between the training samples for the c-th channel (as proposed in [31]). We use χ^2 distance for the activity descriptors $Trajectory, HoG, HoF, MBH$, and $Correlation$ distance for the objects descriptors $DPM, ObjectBank$. To set the C hype-parameters of the SVMs we used grid search and 5 fold cross-validation over the training set. (We have also tested other distances but they were performing worse during the cross-validation).

Defining Ground Truth: Since every video can have multiple SVO-triplets we define two ways to account for ground truth, one based on the most common triplet (denoted 'Most Common') and another one based on the set of valid triplets ('Valid Set'), defined as triplets where each component S, V and O is mentioned in at least two descriptions. For example, for the video shown in Figure 2 the set of triplets extracted from the descriptions with their associated frequencies are: {(person, mix, egg):10, (person, mix, yolk):2, (person, prepare, dough):1, (person, mix, maida):1, (person, mix, ingredient):1, (person,mix,gel):1, (person, mix, flour):1, (person, mix, cream):1, (person, make, dough):1, (person, cut, yolk):1, (person, cut, egg):1, (person, add, paste):1 (egg, mix, flour):1, (cook, prepare, dough):1}

From this set of triplets the 'Most Common' answer is (person,mix,egg), and the 'Valid Set' {(person, mix, egg), (person, mix, yolk), (person, prepare, dough), (person, mix, flour)}.

7. Experimental Results

Binary Accuracy. First we evaluate our learned hierarchy on predicting subject, verb and object labels in terms of binary (0-1 loss) accuracy. Since the words at higher-level nodes of our trees do not tend to appear in the human descriptions, here we use the first level of our hierarchies, which group the flat raw labels (see Section 4). The number of first-level nodes is 8 for Subjects, 100 for Verbs and 100 for Objects. Table 1 compares: a baseline (Prior) that uses the prior distribution of subject, verbs and objects in the training set and simply predicts the triplet composed of the most frequent items (person, play, guitar); the SVM classifiers for flat, raw labels (FL) and for the first level of our hierarchies (OU). We see that visual classifiers do significantly better than a triplet-prior baseline (except for subject, for which simply guessing person does very well), and semantic grouping improves performance.

WUP Accuracy. Next, in Table 2 we compare our full hierarchical method with two baselines. To evaluate higher-level node predictions, we use the WUP similarity score. The flat (FL) baseline predicts the most confident output for each SVM trained over the whole set of labels without any hierarchy or grouping. The hierarchical (HE) baseline is an implementation of the "Hedging Your Bets" method [6],

Method	0-1 Loss		
	S%	V%	O%
Prior	78.36	13.43	6.12
FL	78.51	22.09	12.84
OU	**80.90**	**29.10**	**17.01**

Table 1: Binary 0-1 accuracy of predicting subject, verb and object labels with Prior:most frequent triplet, FL:flat visual classifiers, OU:first level of our semantic hierarchies.

Alg	WUP Similarity					
	Most Common			Valid Answer		
	S%	V%	O%	S%	V%	O%
FL	88.94	43.56	36.77	93.28	59.52	51.91
HE	78.13	31.29	23.37	81.03	45.71	28.45
OU	**92.57**	**46.83**	**46.66**	**93.72**	**61.19**	**58.41**

Table 2: Comparison of WUP Similarity

which combines the outputs of the SVMs according to the WordNet hierarchy and chooses the appropriate level of generalization by setting the accuracy to some prespecified value (0.9 in our experiments). Our method (OU) computes a probability distribution over the learned semantic hierarchies and chooses the appropriate level of generalization by optimizing the WUP similarity of the predictions using cross-validation on the training set. The table measures the similarity to both the 'Most Common' gold-standard answer (single triplet for each video), and the 'Valid Answer' (any of the Subjects, Verbs and Objects mentioned by humans). Since the WUP similarity depends on the hierarchy used, to do a fair comparison in Table 2 we use the WUP similarity defined over the WordNet hierarchy for all the methods. Our approach predicts words that are more similar on average to the human triplets than either baseline, especially for the most common answer.

Generation. Table 4 shows example sentences generated based on the triplets predicted by the three methods, as well as (the most common) human annotation. The top examples are ones where our model does better than FL and HE, and the bottom three are examples where it does similar or worse. We see that HE tends to predict very general words most of the time, whereas FL predicts specific nodes but makes a lot of mistakes. In contrast, OU method outputs more general descriptions that are nonetheless informative about the content of the video, what make it more suitable for video retrieval.

Zero-shot Activity Recognition. We also conducted an experiment to see if our method can use a language model to learn to describe activities involving verbs for which *no* training videos are provided. We held out a random fraction of verbs during training and judged the system's ability to still predict them during testing based on subject and object context. Figure 4 reports the percentage of unseen verbs

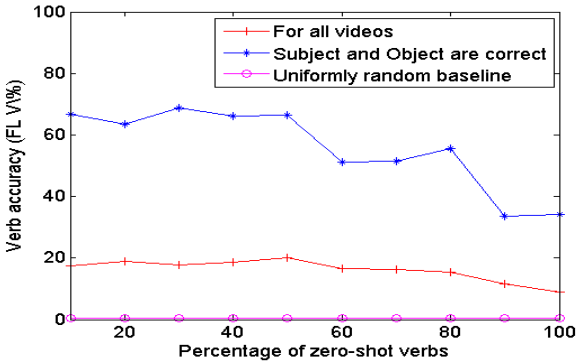


Figure 4: Zero-shot Activity Recognition

correctly predicted (0-1 loss) using our model. We measure the accuracy over all videos that contain unseen verbs as well as the subset of these videos where the subject and object were correctly identified. The results in Figure 4 were averaged across 8 runs, removing a different random set of verbs for zero-shot recognition in each run. Since we have no information on the test verbs during training, we cannot assume any priors about their likelihoods. So, we compare to a baseline where the system picks a verb uniformly at random (0.459%). Even with a large portion of verb models missing, in a reasonable fraction of cases, our language model is still able to "fill in" the correct verb from context.

7.1. Human Evaluation

We use Amazon Mechanical Turk (AMT) to compare the methods by evaluating them on a video retrieval task. We use each of the methods proposed in this paper to build separate video retrieval systems. We then ask humans to judge how well each system does and use the human judgements to compare the methods. The FL, HE and OU algorithms are used to predict SVO triplets for all 670 test videos. For each test video, we measure the similarity of its SVO triplet with the triplets of all the test videos to retrieve the 3 most similar videos. Similarity between triplets is measured using WUP scores. The retrieval is done independently for the FL, HE, OU, and ground truth triplets. We then ask AMT workers to rate, on a scale of 1 to 5, how relevant the retrieved videos are with respect to the given video. The average ratings of the videos retrieved by each method is presented in Table 3. The Kruskal-Wallis test (H=77.27, 2 d.f, $p<0.0001$) and ANOVA test ($F(2,1947)=43.92$, $p<0.0001$) confirm that the differences in the ratings of the three sys-

Retrieval Method	FL	HE	**OU**	Ground Truth
Average Rating	1.81	1.54	**1.99**	3.90

Table 3: Amazon mechanical turker ratings for videos retrieved by FL, HE, OU and ground truth triplets.

tems are statistically significant. The human evaluation results are consistent with the WUP similarity based evaluation (see table 2). This also indicates that WUP is a good intrinsic measure of evaluation.

8. Conclusions

Broad-coverage activity recognition has wide application in surveillance and retrieval applications, yet few existing methods work outside limited verb vocabularies. We presented a system that takes a short video clip and outputs a brief sentence that sums up the main activity in the video, such as the actor, the action and its object. We addressed the challenge of recognizing and describing activities "in-the-wild". Unlike previous work, our approach has broad verb and object coverage and works on out-of-domain actions: it does not require training videos of the exact activity. If it cannot find an accurate prediction for a pre-trained model, it finds a less specific answer that is also plausible. The semantic hierarchies learned from the data help to choose an appropriate level of generalization, and a prior learned from web-scale natural language corpora penalizes unlikely combinations of actors/actions/objects and allows zero-shot activity recognition. We evaluated our method on a large YouTube corpus and demonstrated it was able to generate short sentence descriptions of video clips better than baseline approaches.

Acknowledgments This research was partially supported by NSF Awards IIS-1212928, IIS-1016312, IIS-1116411, DARPA's MSEE Program, U.S. ARO Award W911NF-10-2-0059 and Toyota.

References

[1] H. Aradhye, G. Toderici, and J. Yagnik. Video2text: Learning to annotate video content. In Data Mining Workshops, 2009. ICDMW '09. IEEE International Conference on, pages 144–151, 2009. 2

[2] A. Barbu, A. Bridge, Z. Burchill, D. Coroian, S. Dickinson, S. Fidler, A. Michaux, S. Mussman, S. Narayanaswamy, D. Salvi, et al. Video in sentences out. In Proceedings of the 28th Conference on Uncertainty in Artificial Intelligence (UAI), pages 102–12, 2012. 1

[3] C. Chang and C. Lin. Libsvm: a library for support vector machines. ACM Trans. on Intelligent Systems and Technology (TIST), 2(3):27, 2011. 4, 5

[4] D. L. Chen and W. B. Dolan. Collecting highly parallel data for paraphrase evaluation. In Proceddings of ACL, 2013, pages 190–200, Portland, Oregon, 2011. 3

[5] P. Das, C. Xu, R. F. Doell, and J. J. Corso. A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In Computer Vision and Pattern Recognition (CVPR), 2013., pages 2634–2641. IEEE Computer Society, 2013. 1, 2

[6] J. Deng, J. Krause, A. C. Berg, and L. Fei-Fei. Hedging your bets: Optimizing accuracy-specificity trade-offs in large scale visual recognition. In Computer Vision and Pattern Recognition (CVPR), 2012., pages 3450–3457. IEEE, 2012. 2, 3, 4, 5, 8

[7] J. Deng, K. Li, M. Do, H. Su, and L. Fei-Fei. Construction and Analysis of a Large Scale Image Ontology. In Vision Sciences Society, 2009. 1

[8] D. Ding, F. Metze, S. Rawat, P. Schulam, S. Burger, E. Younessian, L. Bao, M. Christel, and A. Hauptmann. Beyond audio and video retrieval: towards multimedia summarization. In Proceedings of the 2nd ACM International Conference on Multimedia Retrieval, page 2. ACM, 2012. 2

[9] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. International Journal of Computer Vision, 88(2):303–338, June 2010. 3

[10] A. Farhadi, M. Hejrati, M. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. Computer Vision–ECCV 2010, pages 15–29, 2010. 2

[11] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. IEEE Transactions on Pattern Analysis and Machine Intelligence., 32(9):1627–1645, 2010. 4, 5

[12] M. Khan and Y. Gotoh. Describing video contents in natural language. Proceedings of the EACL Workshop on Innovative Hybrid Approaches to the Processing of Textual Data, pages 27–35, 2012. 1, 2

[13] A. Kojima, T. Tamura, and K. Fukunaga. Natural language description of human activities from video images based on concept hierarchy of actions. International Journal of Computer Vision, 50(2):171–184, 2002. 2

[14] N. Krishnamoorthy, G. Malkarnenkar, R. J. Mooney, K. Saenko, and S. Guadarrama. Generating natural-language video descriptions using text-mined knowledge. In Procedings of AAAI, 2013, 2013. 2, 3, 4

[15] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. Berg, and T. Berg. Baby talk: Understanding and generating simple image descriptions. In Computer Vision and Pattern Recognition (CVPR), 2011., pages 1601–1608. IEEE, 2011. 2

[16] I. Laptev and P. Perez. Retrieving actions in movies. In International Conference on Computer Vision, 2007. ICCV 2007., pages 1–8. IEEE, 2007. 3

[17] M. Lee, A. Hakeem, N. Haering, and S. Zhu. Save: A framework for semantic annotation of visual events. In Computer Vision and Pattern Recognition Workshops, 2008. CVPRW'08., pages 1–8. IEEE, 2008. 2

[18] L. Li, H. Su, E. Xing, and L. Fei-Fei. Object bank: A high-level image representation for scene classification and semantic feature sparsification. Advances in Neural Information Processing Systems, 24, 2010. 4, 5

[19] S. Li, G. Kulkarni, T. Berg, A. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. In Proceedings of the Fifteenth Conference on Computational Natural Language Learning, pages 220–228. Association for Computational Linguistics, 2011. 1, 2

[20] T. Motwani and R. Mooney. Improving video activity recognition using object recognition and text mining. In European Conference on Artificial Intelligence. ECAI, 2012. 2, 3

[21] T. Pedersen, S. Patwardhan, and J. Michelizzi. Wordnet:: Similarity: measuring the relatedness of concepts. In Demonstration Papers at HLT-NAACL 2004, pages 38–41. Association for Computational Linguistics, 2004. 3

[22] F. Pereira, N. Tishby, and L. Lee. Distributional clustering of english words. In Proceedings of the 31st annual meeting on Association for Computational Linguistics, pages 183–190. Association for Computational Linguistics, 1993. 3

[23] J. C. Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In ADVANCES IN LARGE MARGIN CLASSIFIERS, pages 61–74. MIT Press, 1999. 4

[24] K. Reddy and M. Shah. Recognizing 50 human action categories of web videos. Machine Vision and Applications, pages 1–11, 2012. 1, 3

[25] B. Russell, A. Torralba, K. Murphy, and W. T. Freeman. Labelme: a database and web-based tool for image annotation. In International Journal of Computer Vision, 2007. 1

[26] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In Pattern Recognition, 2004. ICPR 2004., volume 3, pages 32–36. IEEE, 2004. 3

[27] H. Wang, A. Klaser, C. Schmid, and C.-L. Liu. Action recognition by dense trajectories. In Computer Vision and Pattern Recognition (CVPR), 2011., pages 3169–3176. IEEE, 2011. 1, 4, 5

[28] Z. Wu and M. Palmer. Verbs semantics and lexical selection. In Proceedings of the 32nd annual meeting on Association for Computational Linguistics, pages 133–138. Association for Computational Linguistics, 1994. 3

[29] Y. Yang, C. L. Teo, H. Daumé, III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In Proc. of the Conference on Empirical Methods in Natural Language Processing, EMNLP '11, pages 444–454, 2011. 2

[30] B. Yao, X. Yang, L. Lin, M. Lee, and S. Zhu. I2t: Image parsing to text description. Proceedings of the IEEE, 98(8):1485–1508, 2010. 2

[31] J. Zhang, M. Marszałek, S. Lazebnik, and C. Schmid. Local features and kernels for classification of texture and object categories: A comprehensive study. International Journal of Computer Vision, 73(2):213–238, 2007. 4, 5

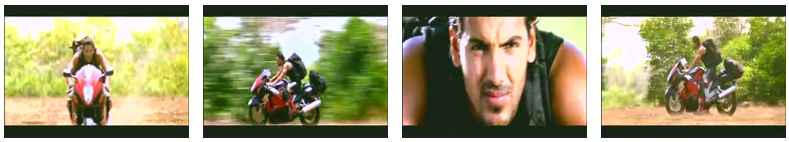
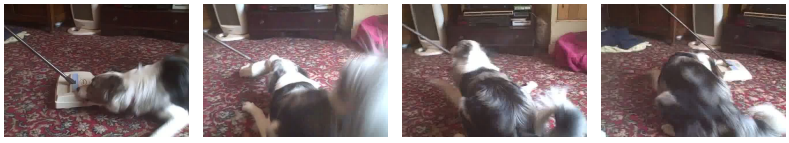
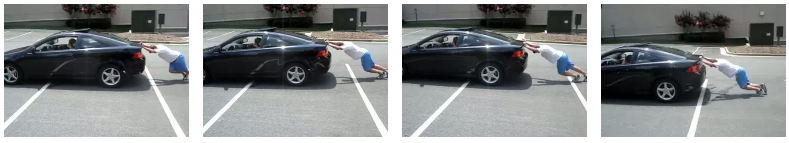
	GT: *A woman is mixing some egg with flour.* FL: A person cuts the water. OU: **A person cooks something.** HE: A person does something.
	GT: *A cat is playing with a ferret.* FL: A person plays a water. OU: **An animal plays something.** HE: An animal does something.
	GT: *A man is riding a motorcycle.* FL: A person rides a person. OU: **A person rides a vehicle.** HE: The person does something.
	GT: *A toy train runs into a toy car.* FL: A car rides the motorbike. OU: **A car rides the vehicle.** HE: Someone does something.
	GT: *A dog is attacking a vacuum.* FL: A dog plays a water. OU: **An animal does something with the instrument** HE: An animal does something.
	GT: *A man is pushing a car.* FL: A person plays a motorbike. OU: **A person rides a vehicle.** HE: A person does something.
	GT: *A baby panda is climbing a step.* FL: The cat plays with the water. OU: **An animal plays an instrument.** HE: An animal does something.
	GT: *A man is doing exercise.* FL: A person plays a person. OU: **A person dances a device.** HE: A person does something.
	GT: *A man is playing a guitar.* FL: **A person plays a guitar.** OU: **A person plays a guitar.** HE: **A person plays a guitar.**
	GT: *A man is pouring noodles into a bowl.* FL: A person plays a guitar. OU: A person plays a guitar. HE: **A person does something.**
	GT: *A train passes by Mount Fuji.* FL: A person plays a motorbike. OU: The person moves something. HE: Someone does something.

Table 4: Some examples of videos and: (GT) most common human sentence; (FL) the flat classifiers; (OU) our semantic hierarchical method; (HE) the method in [6]. The top examples are ones where our model does better than FL and HE, and the bottom three are examples where it does similar or worse.