Action Recognition in Still Images Using Word Embeddings from Natural Language Descriptions

Karan Sharma Arun CS Kumar Suchendra M. Bhandarkar
Department of Computer Science, The University of Georgia, Athens, GA 30602-7404, USA
E-mails: {karan@uga.edu, aruncs@uga.edu, suchi@cs.uga.edu}

Abstract

Detecting actions or verbs in still images is a challenging problem for a variety of reasons such as the absence of temporal information and polysemy of verbs which lead to difficulty in generating large verb datasets. In this paper, we propose to first detect the prominent objects in the image and then infer the relevant actions or verbs using Natural Language Processing (NLP)-based techniques. The proposed scheme obviates the need for training and using visual action detectors on images, an approach which tends to be error-prone and computationally intensive. This paper provides a valuable insight in that the detection of a few significant (i.e., top) objects in an image allows one to extract or infer the relevant actions or verbs in the image. To this end, we propose NLP-based approaches relying on the word2vec and the Object-Verb-Object triplet models for predicting the actions from top-object detections and also analyze their nuances. Our experimental results show that verbs can be reliably and efficiently detected by exploiting the top object detections in an image.

Keywords: action recognition, natural language processing, word2vec model, Object-Verb-Object triplet model

1. Introduction

Action recognition in still or static images is a challenging problem for a variety of reasons such as the absence of temporal information and the polysemy of verbs which leads to difficulty in generating large verb datasets. In addition, learning high quality verb detectors or classifiers can be difficult because the corresponding decision boundaries are often non-linear and unclear. For example, in the case of the verb *playing*, one would need to train the corresponding detector on a very large dataset comprising of tennis images, baseball images, images of a child playing with a toy and so on. The underlying visual patterns in the images that contain the action of interest may be very hard to capture reliably, thus making it extremely difficult to learn an accurate action classifier. However, if one can reliably determine that there is a *person* and a *tennis racket* in an image, then one can also infer that the corresponding verb is *playing* or *hitting* with high certainty. In this paper, we draw upon the above intuition to detect verbs in a still image without having to explicitly train and employ visual verb detectors. We propose to identify the significant activity (i.e., verb) in an image by taking advantage of the highly plausible object-pair detections in an image.

By drawing upon the recent advances in Natural Language Processing (NLP), we provide a valuable insight in that with reliable detection of the significant or prominent (i.e., top) objects in an image, we can predict the corresponding verb in an image without having to employ visual verb detectors explicitly on the input image.

We believe that the proposed approach could be potentially useful in various scenarios, especially:
(1) Situations where annotated verb datasets are not available. In this paper, as a proof of concept, we use the sentences in the *Microsoft Common Object in Context* (MS COCO) dataset [19] for training NLP-based models to enable action recognition in still images. However, we believe our results could be extrapolated to situations where ground truth verbs for images are not available and the models need to be learned from a general real-world text corpus such as Wikipedia.
(2) Design of software applications (i.e., apps) for mobile user devices such as smartphones and tablets that are typically resource constrained. Our approach has the advantage of obviating the use of computationally intensive visual activity (i.e., verb) detectors on the input image. In contrast, the detection of significant object-pairs and the corresponding verb assignments in an input image can be made both, reliably and efficiently using the proposed NLP approaches based on word embeddings. The extraction of deep learning features from an input image is, computationally, the most expensive stage in deep learning architectures. As a result, substantial current research effort has focused on speeding up the deep learning feature extraction stage without loss in accuracy with the goal of making the deep learning architecture deployable on resource-constrained mobile

978-1-5090-4941-7/17 $31.00 © 2017 IEEE
DOI 10.1109/WACVW.2017.17

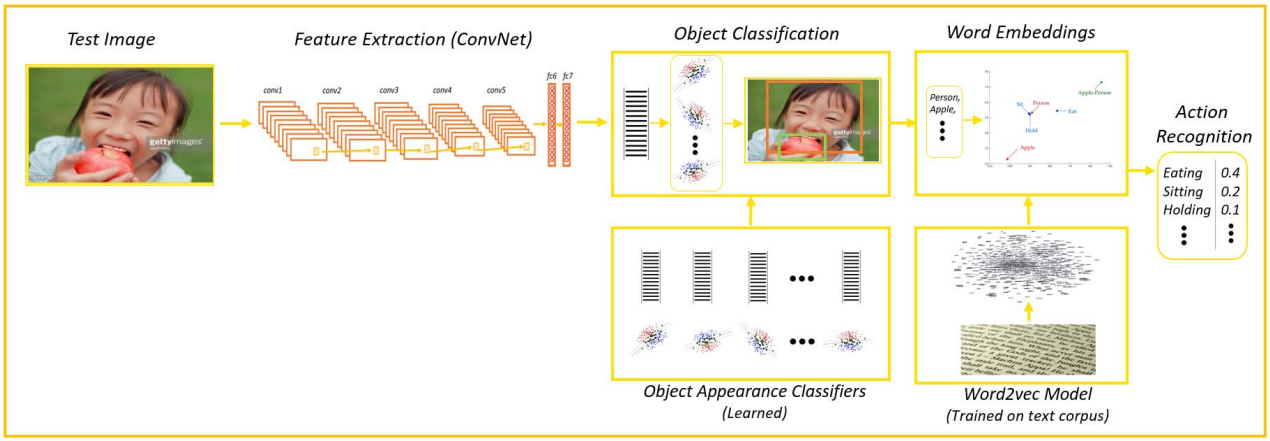

Figure 1. Deep learning features are extracted for an input test image using a Convolutional Neural Network (ConvNet), followed by SVM-based classification for each object category. The top-objects are mapped into the word embedding space. The action is inferred from the top-objects in the word2vec space.

devices [1, 15, 16, 17, 18]. In this paper, we offer an alternative perspective that obviates the need to train and deploy separate visual activity detectors. In future, the computationally efficient extraction of deep learning features in conjunction with the computational savings of NLP model-based action detection could greatly enhance automated image captioning and image annotation systems.

Most previous work has focused on action recognition in video frames [3], with some work on zero-shot action recognition in videos [2, 13, 14, 33]. Most papers that deal with activity recognition in static images, rely on either pose estimation [20], parts-based detection [4], detection of human-object interactions [28] or weakly supervised learning in multi-modal settings [6, 7]. We contend that the need to learn the computationally complex verb detectors described in [4, 20, 28] even in the case of static images can be obviated by just predicting the verb using the top-object detections and NLP-based word embeddings. In addition to computational efficiency and accuracy, we show that the proposed approach has the crucial advantage of conceptual simplicity. Xu et al. [33] exploit word embeddings in a zero-shot framework for action recognition in videos using computationally intensive transductive manifold learning techniques. In contrast, the proposed approach uses word embeddings directly and efficiently to infer the underlying action in static images via detection and recognition of the top objects in the image.

2. Motivation

We surmise that the top-object detections in an image have enough information for one to reliably infer the relevant actions in the image. First, given the current state-of the art there are pragmatic reasons why such an approach would be practically useful. Second, there are certain inherent properties of the visual and linguistic worlds that make action prediction based on the detection of top objects intuitively appealing. We list the following reasons for why we believe that action recognition models in static images should be driven by top-object detections:

(1) Objects are cohesive structures, in that, concrete objects, even when malleable, are held together by percepts or parts which gives an object a property of wholeness [10]. This property of wholeness is not a characteristic of relational categories such as verbs. It is this cohesion that makes it fundamentally easy to recognize objects using visual features. On the other hand, correctly recognizing a relational category is often very challenging because it involves recognizing a relationship between disjointed elements [10].

(2) Objects map directly to concrete entities in the world [10]. This direct translation between nouns and objects makes possible a stable assignment of a word to an object in an image. On the other hand, relational categories such as verbs tend to describe relations between disjoint entities. These relational categories may not map directly to the various object configurations present in an image.

(3) Verbs are more polysemous than nouns [8] in that verbs have more senses of meaning than nouns. The verb *sitting* can be used in a variety settings. For example, in the sentences "John is sitting on a chair.", and "The apple is sitting on a table.", the word *sitting* is used in different senses. This polysemy amongst verbs can make the generation of training datasets for verb detectors very hard.

(4) Verbs are mutable in that verbs adjust themselves in meaning to the context of sentence [9]. For example, in the sentence, "A person is jumping on pizza.", the verb *jumping* adjusts its meaning to the nouns in the sentence. This makes it even harder to automatically create large verb training datasets for verb detectors.

(5) On the more pragmatic side, object classification has reached a stage where one can reliably detect objects in an image. Object classification systems have shown impressive results on several challenging datasets in the computer vision research community. As of yet, comparable success has not been met by systems for automatic detection and classification of actions, visual phrases and attributes in still images.

3. Related Work

3.1. Action (verb) recognition

Action recognition in videos is a relatively easier problem and has been addressed by using various spatio-temporal descriptors followed by a classifier [5, 24, 32]. However, in static images, unlike videos, there is no temporal information. Also, the problem is further exacerbated by the lack of reliable annotation data. Hence, several existing works in action recognition on static images use supervised learning of visual information integrated with linguistic information mined from a text corpus [11, 14, 34]. More recently, Gao et al. [6, 7] used word embeddings in conjunction with deep learning features for action recognition in still images; however, their technique still entails the learning of action classifiers in a multimodal setting. In the proposed approach, we argue that even in the case of static images, with reliable recognition of objects in the image, one can successfully use NLP-based word embeddings to describe the underlying action with a reasonable degree of accuracy. More recently, Jain et al. [13] have proposed an approach that explores the efficacy of word embeddings for action recognition in videos using knowledge of the objects in the video. In contrast, the strength of our approach lies in the fact that we use word embeddings in a relatively simple manner in *static* images and yet obtain good results on a challenging set of *static* images. In their recent work, Xu et al. [33] use word embeddings in a zero-shot framework for action recognition in videos. Their approach is based on learning a mapping between the visual features of the action and a semantic descriptor (i.e., word vector). Determination of the mapping involves a computationally intensive semi-supervised transductive learning procedure which calls for access to testing data in the training phase. We believe that semi-supervised transductive learning may be appropriate for videos because of the complexity of mapping spatio-temporal features to semantic meaning, however, for static images, word embeddings could be used more directly and efficiently to infer the underlying action from the objects. Alexiou et al. [2] have proposed another interesting approach that employs word embeddings for zero-shot action recognition in videos. Their approach shows that the mining and alignment of synonyms from general text data can enrich action word vector embeddings via the introduction of more robust semantic context from a wider range of text domains. Their approach to action recognition, based on directly learning the mapping from the visual action features to the word2vec space, is well suited for video data. In contrast, the proposed approach relies on the detection and recognition of objects which provides more concrete information that can be used to infer the underlying action. We contend that the proposed approach is better suited for action recognition in still images for reasons discussed in the Section 2.

3.2. Word2vec model

The motivation for *word2vec*-based approaches emanates from the distributional hypothesis proposed by Harris [12]. The central idea in the distributional hypothesis is that the words that occur in similar contexts tend to have similar meanings. For example, given the sentences, *"A person is eating pizza."* and *"A person is eating chocolate."*, we know from the distributional hypothesis that the words *pizza* and *chocolate* are similar in that they occur in similar contexts. Word2vec [21] is a word embedding scheme that converts a word into a low-dimensional vector. Each word is mapped to a point in a hypothetical space such that words that have similar meanings tend to be closer in this hypothetical space. These word embeddings are used in a variety of applications and have had a significant impact in the fields of NLP, computer vision and information retrieval.

4. Top-object Detection-driven Verb Prediction Model

In this paper, we present the insight that the determination of top-nouns is enough to predict the relevant verb in an image, hence, separate verb detectors are not required for describing the action in most static images. In support of the above insight, we analyze various NLP techniques where we predict the verb from the top-nouns in an image without explicitly learning a verb detector for that image. Figure 1 depicts the computational pipeline for the proposed approach. In the proposed approach, we detect the top objects in an image, identify the most plausible two objects (i.e., object pair) in the image, and then assign the most meaningful action (verb) to this object pair (Figure 2). This approach could prove practically useful in two potential scenarios:

(1) Real world situations where automatic generation of training data for verb detectors is very hard. Objects in an image directly map to concrete entities (i.e., nouns) in the real world [8, 10]. This direct translation between nouns and objects enables the stable assignment of a word to an object in an image. In contrast, relational categories such as verbs tend to describe relations between disjoint entities. Moreover, verbs are also more *polysemous* than nouns, in that verbs have more senses of meaning than do nouns [8].

For example, the verb *running* can be used in a variety of settings, for example, "John is *running* for public office.", and "John is *running* on the field.", use the verb *running* in different senses. Likewise, verbs are also *mutable* in that their meaning can be adjusted based on the context of sentence [9]. For example, in the sentence, "John is *jumping* to a conclusion", the verb *jumping* adjusts its meaning to the nouns in the sentence. The properties of polysemy and mutability make automated generation of training datasets for verbs a very difficult task.

(2) With the recent proliferation of resource-constrained mobile devices that constitute the Internet-of-things (IoT), it is important to have image analysis and retrieval techniques that could provide significant algorithmic time gains. Hence, by recognizing the *object-pair* and associated *verb* in a time-efficient manner, one could describe the crux of the story underlying an image even in the most resource-constrained environments. Whereas feature extraction and object detection and classification are unavoidable in an automated image annotation or captioning system, we believe that when inferring a relational category, such as the action or verb, significant algorithmic time gains can be achieved if we can reliably infer a verb from its associated objects in constant (i.e., $O(1)$) time. Although deep learning feature extraction is computationally intensive, substantial research efforts are underway to make deep learning architectures deployable in resource-constrained environments [1, 15, 16, 17, 18]. However, in addition to speeding up feature extraction and other various aspects of deep learning, we believe these research efforts could be greatly assisted by reducing the number of Support Vector Machines (SVMs) (or other classifier functions) employed at test time. Although the computational expense associated with deep learning feature extraction is significantly higher than that associated with SVM-based classification (or with other classifier functions), we believe, with the research efforts currently underway, deep learning feature extraction will get significantly faster in due course. In that case, we contend that the proposed approach, which reduces the number of SVMs (or other classifier functions) needed for action recognition/classification will effectively complement the computationally efficient deep learning feature extraction techniques in the very near future.

In this paper, we analyze and propose two models to predict verbs from top-nouns - the *Object-Verb-Object (OVO) triplet* model and the *word2vec* model. Both models have their advantages and shortcomings depending on the underlying dataset and application scenario.

4.1. Object-Verb-Object (OVO) triplet model

The Object-Verb-Object (OVO) triplet model explicitly models the probability distribution of words based on their co-occurrences. However, there are situations in real world datasets wherein such co-occurrences may not cover all situations. Hence, we need techniques that learn the dependencies between verbs and nouns in a more implicit manner. For example, if we have verb *eating* as predicted by the OVO triplet model using nouns *person* and *pizza*, then the OVO triplet model cannot extrapolate to predicting the verb *eating* for the nouns *person* and *chocolate*. However, the word2vec model explained in the following subsection is capable of handling such situations.

Figure 2. Outline of the proposed top-object detection-driven verb prediction model

In the OVO triplet model, we predict the relevant verb using the following equation:

$$p(verb|noun_1, noun_2) = \arg\max_i P(verb_i|noun_1, noun_2) \quad (1)$$

The probability values in equation (1) are computed using the textual data in the training set. At test time, given two objects (i.e., nouns), the most highly probable verbs are determined using equation (1) and assigned to the image.

4.2. Word2vec model

The word2vec verb prediction model uses the word2vec representation scheme [22] which is based on the embedding of a word in a hypothetical low-dimensional vector space. In the word2vec representation scheme, each word is mapped to a point in the hypothetical vector space such that words that have similar meanings tend to be proximal in this vector space. In our case, we intend to capture the relationships between nouns and verbs, for example *pizza* and *eat*, and noun-pairs and verbs, for example, *Person-dog* and *walk*.

It is interesting to examine why the word2vec representation is able to learn the word embeddings that capture the relations between nouns and verbs, or between noun-pairs and verbs. Note that the word2vec representation groups semantically similar words into proximal regions in the hypothetical vector space, i.e., words that are similar in meaning such as *beautiful* and *pretty* are mapped to proximal points in the hypothetical vector space. In this sense, the word2vec representation treats synonymy, not as a binary concept, but rather one that spans a continuum. However, we hypothesize that even when the words are not obviously synonymous or similar in meaning, the distance between

their corresponding points in the vector space can still convey something significant about their relationship.

For the sake of clarification, consider the following example: Assume that we are given a collection of the following four sentences:
"A person is driving a car on the road". "A car is passing a truck on the road". "A car is parked on the road". "A person is driving a truck".

In the above sentences, the context of the noun *car* is {*person, road, truck, driving*} whereas the context of the verb *driving* is {*person, road, truck, car*}. As the contexts of *car* and *driving* are very similar, word2vec will place the embeddings of *car* and *driving* in close proximity in the vector space although *car* and *driving* are strictly not synonymous words. Based on context, among all verbs, the verb *driving* will tend to be closer to the noun *car* based on their respective embeddings in the vector space. For a more rigorous treatment of why the word2vec embedding tends to capture such linguistic regularities the interested reader is referred to [25].

The problem of determining the closest verb to two given top nouns can be stated as follows. Given a set of verbs V and top nouns n_1 and n_2, the closest verb from the set V to the top nouns n_1 and n_2 is given by:

$$\arg\max_i \ \{SIM(v_i, n_1) + SIM(v_i, n_2)\} \qquad (2)$$

where $SIM(v_i, n_1)$ and $SIM(v_i, n_2)$ are the cosine similarities of the vector representation of verb v_i to the vector representations of nouns n_1 and n_2 respectively.

One of the problems with above formulation is that certain nouns such as *person* and *apple*, when considered independently, may have multiple verbs that are proximal in vector space. For example, *person* and *apple*, when considered independently, may be proximal to multiple verbs such as *sit*, *hold*, *sleep* and so on. Simple addition of the cosine similarities as shown in equation (2) does not bias the verb prediction towards *eat* when **both** the nouns *person* and *apple* are present in the same sentence. To circumvent the above problem, in the sentence database accompanying the MS COCO training dataset [19], we append each sentence with all object-pairs occurring in that sentence. In other words, we identify all the nouns in a sentence and form all pairs of these nouns before appending them to the sentence. For example, given a sentence *"Person is eating an apple sitting on the table."*, we convert the sentence into following three sentences:
"Person is eating an apple sitting on the table apple-person."
"Person is eating an apple sitting on the table person-table."
"Person is eating an apple sitting on the table apple-table."

This simple preprocessing step potentially captures the dependences between all possible noun-pairs and all verbs

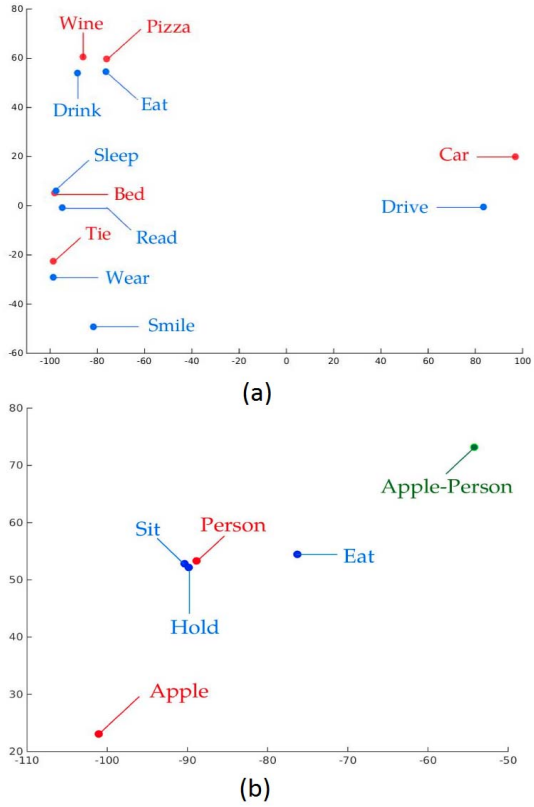

Figure 3. Visualization of word embeddings is 2D space using *t-SNE* dimensionality reduction [30]. (a): Most verbs tend to occur closer to their attached nouns. (b): Appended nouns (*apple-person*) occur nearer to verb *eat* than individual nouns *apple* and *person*.

in the sentence. Next, we train the word2vec model on the modified sentence database. After the model is trained, it will have learned the verb that defines a relationship between a pair of objects. Figure 3 clarifies the above argument using the projection of these word embeddings in a 2D space using the *t-SNE* dimensionality reduction technique [30].

More formally, given the set of verbs V, and noun-pair n_{12}, the closest verb in V to the noun-pair is $np_{j,k} = (n_j, j_k)$ is given by:

$$\arg\max_i \ SIM(v_i, np_{j,k}) \qquad (3)$$

where $SIM(v_i, np_{j,k})$ is the cosine similarity between the vector representations of the verb v_i and noun-pair $np_{j,k} = (n_j, j_k)$.

In the above model, once the necessary steps of feature extraction and object detection are performed, verb prediction for a given noun-pair can be achieved in $O(1)$ time. During testing, the top verbs can be easily retrieved in $O(1)$ time using an appropriate hash data structure once the top-

2 nouns (objects) in the test image are detected. Since the verb is detected in a zero-shot manner (i.e., without requiring visual training examples of the action underlying the verb), the computational expense of training verb detectors and running them on the image is obviated. After the word2vec model is trained on the modified sentence dataset, we choose plausible verbs that are closest in distance to each object-pair and store the verbs in the database along with the object-pair.

4.3. Model training

4.3.1 Training of object detectors

We train an SVM-based object detector/classifier for each of the 80 annotated object categories in the MS COCO dataset [19]. The inputs to the SVM-based classifiers are the VGG-16 *fc-7* image features [29] extracted using the Matconvnet package [31].

4.3.2 Training of the OVO triplet model

The OVO triplet model is trained on the sentences corresponding to the training images (in the MS COCO dataset) using equation (1). For each pair of objects that occur together in a particular sentence, we learn the probability value of the corresponding verb in that sentence.

4.3.3 Training of the word2vec model

The word2vec model is trained with a window size of 10 using the implementation of Rehurek and Sojka [27]. This results in a 300-dimensional vector for each word using the skip-gram model for word2vec [21]. In the skip-gram approach, the input to the deep-learning neural network (DLNN) is the word, and the context is predicted from the word. For example, given the contextual input *eat*, the model will predict {*person*, *pizza*}. To train the skip-gram model, given a sequence of words $w_1, w_2, w_3, ...w_T$, we maximize the following objective function:

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0}\log p(w_{t+j}|w_t) \tag{4}$$

where c is the context parameter that specifies the number of words to be predicted from a given word [23]. The term $p(w_{t+j}|w_t)$ signifies the prediction probability of the context given the word. The stochastic gradient descent algorithm is used for training the skip-gram model. More details on the skip-gram architecture can be found in [23]. After the skip-gram model is trained, we obtain the word embeddings corresponding to each word in the dataset.

The nouns within each sentence are converted to noun-pairs and appended to the end of the sentence, as explained previously. We also perform a couple of additional preprocessing steps on the entire data. First, we stem each word using Porter's stemmer; for instance, *driving* and *drive* are both converted to *driv*. Additionally, words synonymous to *person* such as *human*, *woman*, *boy*, *girl*, *people* etc. are converted to *person*. Currently, we try to infer only the most frequently occurring verbs in the MS COCO dataset, i.e., we select the top-n (where $n = 51$) most frequently occurring verbs in the MS COCO dataset. The top verbs in the MS COCO dataset are obtained by parsing the training captions using the Stanford parser.

The 40,000 images in the MS COCO validation set are split into two subsets, each containing 20,000 images; one subset is used for validation of the hyperparameter tuning procedure and the other subset for testing. The validation subset is used to learn the hyperparameters and also the other required parameters for the skip-gram model.

4.4. Model testing

Given a test image, we run all the object detectors on it and select the *top-2* highly probable object detections as candidate objects. For this object-pair, we use the word2vec model to obtain the closest verbs. For each test image, we recognize an object-pair, and predict the plausible verbs in the image. Among the comparison measures introduced below, all except one, predict two plausible verbs in an image. If any one of the predicted verbs occurs in any of the ground truth captions of an image, we regard the prediction as accurate. In addition, even if there are multiple verbs in the ground truth captions for a particular image, intuitively, we just need to infer one verb accurately to describe the crux of the story underlying the image. For example, if the two ground truth captions for a particular image are *Person is riding a motorcycle* and *Person is driving a motorcycle*, it would suffice to just get one of the two verbs *riding* or *driving* correct. Hence, when computing the prediction accuracy, we aim to get just one verb correct in the ground truth captions.

We report results on the subsets S_1 and S_2 of the validation set of the MS COCO dataset, which we use for testing purposes. S_1 is a subset of the validation dataset wherein the ground truth captions have at least two objects from the annotated noun set, and at least one verb from top-51 most frequently occurring verbs in the MS COCO dataset. S_2 is a subset of S_1 wherein the top-2 objects have been correctly detected in an image. These results are used to show how effectively a verb is inferred after the object-pair is correctly detected in an image. We compare the results of the proposed scheme under the following evaluation scenarios:
Random Baseline (Rand): where the two verbs are generated randomly for the top-noun detections in an image.
1-Obj Baseline (1-Obj): where the top-most object (object with highest probability) is used to predict top-2 verbs in

Table 1. Comparison of verb prediction accuracy results of the word2vec model (VD_1, VD_2, VD_3) and the OVO model (*OVO*) with a random baseline (*Rand*), the 1-object baseline (*1-Obj*) and the visual action classifier baseline (*Vis*). DS denotes the data subset. Accuracy is measured based on whether one of the two predicted verbs matches one of the ground truth verbs.

DS	*Rand*	*1-Obj*	*Vis*	VD_1	VD_2	VD_3	*OVO*
S_1	9%	35.2%	37.7%	36.9%	31.4%	32.8%	55%
S_2	10%	45.81%	41.4%	53.43%	57.74 %	52.35 %	79%

an image using word embeddings. The top-2 verbs that are closest in distance to this top-most object are selected.

Vis: where visual action classifiers (such as *walking*, *swimming*, etc.) are explicitly trained using deep learning features followed by SVM-based classification.

VD_1: where the top-2 closest verbs are the ones with the lowest mean distance from the top-2 object detections measured using equation (2).

VD_2: where the top-2 closest verbs are the ones with the lowest distance from the appended noun-pair measured using equation (3).

VD_3: where the top-2 verbs are assigned as follows: if the closest verb determined using equations (2) and (3) is the same, we assign this verb to an image, and the second closest verb is assigned using equation (2). Otherwise, one of the top-2 verbs is assigned using equation (2) and the other using equation (3).

VD_4: where the verbs are assigned using set union between the top three verbs determined using equations (2) and (3).

OVO: where the verbs are assigned using the OVO triplet model using equation (1).

5. Experimental Results and Discussion

The verb prediction accuracy results under the evaluation scenarios described in Section 4.4 are compared in Table 1. These results lend support to our claim that top-object detections could be used to infer other information in an image such as verbs. Once we know the plausible object pairs in a static image, we can infer or predict the corresponding verb in $O(1)$ time with reasonable accuracy using the OVO triplet model or the word2vec model as shown in Table 1. Also, when either one or both of the top-object detections are incorrect, the word2vec model is observed to underperform the visual action classifiers. However, when both the top-object detections are correct, the word2vec model outperforms the visual action classifiers. This suggests that getting top-object detections correct is important for improving the performance of the word2vec model.

For the OVO triplet model, we see that we are able to predict the verb accurately given that the detection of the top-2 objects is accurate. In the MS COCO dataset, the co-occurence patterns of nouns and verbs are similar for both, the training set and the test set, hence the OVO triplet model works well for the MS COCO dataset. However, in real world datasets, the training set and test set may exhibit some dissimilarities, and hence we may have to resort to models such as word2vec for predicting verbs. The results of the word2vec-based approach are analyzed in the following paragraphs.

Figure 4. Qualitative results for the verb prediction model. (a) VD_1: Bad; (b) VD_1: Good; (c) VD_2: Bad; (d) VD_2: Good

In the case of the word2vec-based approach, if the object-pair is correctly recognized in an image, the results in the case of VD_2 are observed to be slightly better than those in the case of VD_1 and VD_3. Hence the proposed technique of appending the object-pair to the end of the sentence does provide a non-trivial benefit for verb prediction. However, if the object-pair is not correctly identified in an image, then the results in the case of VD_1 are observed to outperform those in the case of VD_2 and VD_3. In other words, finding the closest verb by computing the mean distance to the top-2 nouns (equation (2)) is better than using the minimum distance to the object-pair (equation (3)) when at least one of the objects is incorrectly detected. The qualitative results for VD_1 and VD_2 are shown in Figure 4.

In the case of the word2vec-based approach, the results of VD_3, where we try to get the combined benefits of VD_1 and VD_2, were inferior to those of both VD_1 and VD_2. This could be attributed to the fact that results in the case of VD_1 and VD_2 had only a marginal quantitative difference. This appears to suggest that to get actual benefits of both VD_1 and VD_2, we may need to predict more than 2 verbs. Hence, we conducted additional experiments with VD_4 obtaining an accuracy of 56.63% on S_1 and 73.09% on S_2. Therefore, in the case of VD_4, where we predict multiple verbs using both VD_1 and VD_2 we are far more successful in getting at least one verb correct. We believe that besides predicting multiple verbs, there are a couple of other reasons for the relative success of VD_4. There are

situations where VD_1 will be successful, and there are situations where VD_2 will be successful; VD_4 denotes the best of both worlds where VD_1 corrects and compensates for weakness of VD_2 and vice versa. Also, the high accuracy of VD_4 suggests if we try to predict a few more verbs (of the order of 3-6), than there is a very high probability of getting at least one of them correct.

Overall, from the results it is clear that just detecting the most prominent objects in an image is enough to predict the underlying action (verb) in an image with competitive accuracy. The proposed NLP approaches based on the OVO triplet and word2vec models successfully beat the baseline results, thus lending support to our claim.

6. Limitations of the Proposed Approach

One of the limitations the proposed approach is that currently we use only two objects for predicting the verb. There are reasons why two objects may yield good results for verb prediction in many real-world situations. A verb is a natural connector between the subject and object in many real-world situations. However, there are situations where multiple objects in an image are needed for predicting the verb accurately. For example, consider a situation where "*A person is baking pizza in an oven.*" In this situation, knowing the nouns *person* and *pizza* would most likely lead us to infer the verb as *eating*; however, knowing the three objects *person*, *pizza*, and *oven*, would most likely lead us to infer that the most appropriate verb is *baking*. In addition, we have not addressed the effect of the relative spatial positions of the objects on the accuracy of verb prediction in our current work. Considering the example above, the relative spatial positions of the objects *person* and *pizza* could be used to disambiguate between the predicted verbs *eating* and *baking*; if the *pizza* is spatially close to the mouth of the *person* then *eating* would be the more likely verb, otherwise *baking* would be more appropriate.

Another limitation of the current work is that global scene context is not incorporated in the verb prediction model. Knowledge of the global scene context in conjunction with knowledge of the top objects could potentially enhance the accuracy of verb prediction. In the previous example, if the global scene context is *dining room*, then detecting the objects *person* and *pizza* would lead us to infer the verb *eating* over the verb *baking*. Alternately, if the global scene context is *kitchen*, then the objects *person* and *pizza* would lead us to infer the verb *baking* over the verb *eating*.

Also, in our experiments, the OVO triplet model yielded better results than the word2vec model. This can be attributed to the fact that the training and testing datasets for the MS COCO benchmark are not too different. However, real world situations do not exhibit this characteristic. In the real world, unlike the MS COCO training set, two objects and a verb may not co-occur with one another. For example, we may have never seen instances of *tennis racket* and *person* in single image. In such situations, the OVO triplet model will not be able to assign a probability value to any verb that is associated with the nouns *tennis racket* and *person*. The word2vec model would be more appropriate in this situation. In the word2vec space, *tennis racket* will appear close to other sports-related entities such as *ball* and *baseball bat*, thus facilitating the assignment of an appropriate verb such as *hit*.

7. Conclusions and Future Work

In this paper we have proposed a scheme to detect the actions (verbs) in a still image by first detecting the prominent objects in the image and then using Natural Language Processing (NLP)-based OVO triplet and word2vec models to infer the relevant verbs. Our approach obviated the need for training and using visual action detectors which tend to be error-prone and computationally intensive. This paper also provided a valuable insight in that the detection of a few significant (i.e., top) objects in an image allows one to extract the relevant actions or verbs in the image without entailing the learning of an action or verb from visual training data. For this purpose, we proposed NLP approaches based on the word2vec and the OVO triplet models for predicting the actions from top-object detections and also analyzed their nuances. Our experimental results showed that verbs can be reliably and efficiently detected by exploiting the top object detections in an image.

With regard to future work, we plan to extend our work in various directions. We plan to take relative spatial positions of objects in predicting verbs. We also plan to account for situations where more than two objects occur in an image. And in addition to objects, we also plan to incorporate entire scene context in addition to the knowledge of the prominent objects for predicting the verbs. Also, we plan to conduct more robust studies where the efficacy of word2vec approaches is evident, leading to use of word embeddings to resolve all the quirks and nuances of action (verb) recognition based on top-n (where $n \geq 2$) object detection.

Acknowledgment: The authors wish to thank Devi Parikh and Hao Wu for their invaluable suggestions during this research.

References

[1] S. Anwar, K. Hwang & W. Sung (2015). Structured pruning of deep convolutional neural networks. *arXiv preprint arXiv:1512.08571*. 2, 4

[2] I. Alexiou, T. Xiang & S. Gong. (2016). Exploring synonyms as context in zero-shot action recognition. *Proc. IEEE Intl. Conf. Image Processing (ICIP 2016)*, pp. 4190-4194. 2, 3

[3] G. Cheng, Y. Wan, A.N. Saudagar, K. Namuduri & B.P. Buckles (2015). Advances in human action recognition: A survey. Available online *https://arxiv.org/abs/1501.05964*. 2

[4] V. Delaitre, J. Sivic, & I. Laptev (2011). Learning person-object interactions for action recognition in still images. *Proc. NIPS 2011*, pp. 1503-1511. 2

[5] I. Everts, J.C. van Gemert & T. Gevers (2014). Evaluation of color spatio-temporal interest points for human action recognition. *IEEE Trans. Image Processing*, Vol. 23(4), pp. 1569-1580. 3

[6] J. Gao & R. Nevatia (2016). Learning action concept trees and semantic alignment networks from image-description data. *arXiv preprint arXiv:1609.02284*. 2, 3

[7] J. Gao, C. Sun & R. Nevatia (2016). ACD: Action concept discovery from image-sentence corpora. *ACM Intl. Conf. Multimedia Retrieval (ICMR 2016)* , New York, NY. 2, 3

[8] D. Gentner (1981). Some interesting differences between verbs and nouns. *Cognition and Brain Theory*, Vol. 4(2), pp. 161-178. 2, 3

[9] D. Gentner & I. M. France (1988). The verb mutability effect: Studies of the combinatorial semantics of nouns and verbs. *Lexical Ambiguity Resolution: Perspectives from Psycholinguistics, Neuropsychology, and Artificial Intelligence*, pp. 343-382. 2, 4

[10] D. Gentner (2006). Why verbs are hard to learn. *Action meets word: How children learn verbs*, pp. 544-564. 2, 3

[11] A. Farhadi, M. Hejrati, M.A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier & D. Forsyth (2010). Every picture tells a story: Generating sentences from images. *Proc. Eur. Conf. Computer Vision (ECCV 2010)*, pp. 15-29. 3

[12] Z. S. Harris (1954). Distributional structure. *Word*, Vol. 10(23), pp.146-162. 3

[13] M. Jain, J.C. van Gemert, T. Mensink & C.G.M. Snoek (2015). Objects2action: Classifying and localizing actions without any video example. *Proc. IEEE Intl. Conf. Computer Vision (ICCV 2015)*. 2, 3

[14] N. Krishnamoorthy, G. Malkarnenkar & R. Mooney (2013). Generating natural-language video descriptions using text-mined knowledge. *Proc. AAAI*, Vol. 1, pp. 541-547. 2, 3

[15] N. D. Lane, S. Bhattacharya, P, Georgiev, C. Forlivesi & F. Kawsar (2015). An early resource characterization of deep learning on wearables, smartphones and internet-of-things devices. *Proc. ACM Intl. Workshop on Internet of Things towards Applications*, November, pp. 7-12. 2, 4

[16] N. D. Lane & P. Georgiev (2015). Can deep learning revolutionize mobile sensing?. *Proc. ACM Intl. Workshop on Mobile Computing Systems and Applications*, February, pp. 117-122. 2, 4

[17] N. D. Lane, S. Bhattacharya, P. Georgiev, C. Forlivesi, L. Jiao, L. Qendro & F. Kawsar (2016). Deepx: A software accelerator for low-power deep learning inference on mobile devices. *Proc. IEEE Intl. Conf. Information Processing in Sensor Networks (IPSN 2016)*, April, pp. 1-12. 2, 4

[18] Y. LeCun, J.S. Denker, S.A. Solla, R.E. Howard & L.D. Jackel (1989). Optimal brain damage. *Proc. NIPS 1989*, Vol. 2, November, pp. 598-605. 2, 4

[19] T-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar & C.L. Zitnick (2014). Microsoft COCO: Common objects in context. *Proc. Eur. Conf. Computer Vision (ECCV 2014)*, pp. 740-755. 1, 5, 6

[20] S. Maji, L. Bourdev & J. Malik (2011). Action recognition from a distributed representation of pose and appearance. *Proc. IEEE Intl. Conf. CVPR*, pp. 3177- 3184. 2

[21] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado & J. Dean (2013). Distributed representations of words and phrases and their compositionality. *Proc. NIPS 2013*, pp. 3111-3119. 3, 6

[22] T. Mikolov, K. Chen, G.S. Corrado & J. Dean (2013). Efficient estimation of word representations in vector space. *Proc. Intl. Conf. Learn. Rep. (ICLR 2013)*. 4

[23] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado & J. Dean (2013). Distributed representations of words and phrases and their compositionality. *Proc. NIPS 2013*, pp. 3111-3119. 6

[24] X. Peng, C. Zou, Y. Qiao & Q. Peng (2014). Action recognition with stacked Fisher vectors. *Proc. Eur. Conf. Computer Vision (ECCV 2014)*, pp. 581-595. 3

[25] J. Pennington, R. Socher & C.D. Manning (2014). Glove: Global Vectors for Word Representation. *Proc. Conf. Empirical Methods on Natural Language Processing (EMNLP)*, Vol. 14. 5

[26] J. Platt (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, Vol.10(3), pp. 61-74.

[27] R. Rehurek & P. Sojka (2010). Software framework for topic modeling with large corpora. *Proc. LREC 2010 Wkshp. New Challenges for NLP Frameworks*. Valletta, Malta, pp. 46-50. 6

[28] G. Sharma, F. Jurie & C. Schmid, (2017). Expanded parts model for semantic description of humans in still images. *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 39(1), pp. 87-101. 2

[29] K. Simonyan & A. Zisserman (2014). Very deep convolutional networks for large-scale image recognition. *Proc. Intl. Conf. Learn. Rep. (ICLR 2014)*. 6

[30] L. Van der Maaten & G. Hinton (2008). Visualizing data using *t-SNE. Jour. Mach. Learn. Res.*, Vol. 9, pp. 2579-2605. 5

[31] A. Vedaldi & K. Lenc (2015). MatConvNet-convolutional neural networks for MATLAB. *Proc. ACM Conf. Multimedia Systems (MMSys 2015)*. 6

[32] H. Wang & C. Schmid (2013). Action recognition with improved trajectories. *Proc. IEEE Intl. Conf. Comp. Vis. (ICCV 2013)*. 3

[33] X. Xu, T. Hospedales & S. Gong (2017). Transductive zero-shot action recognition by word-vector embedding. *Intl. Jour. Computer Vision*, pp. 1-25. 2, 3

[34] Y. Yang, C.L. Teo, H. Daume III & Y. Aloimonos (2011). Corpus-guided sentence generation of natural images. *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, pp. 444-454. 3