Verbs On Action:
Zero-Shot Activity Recognition with Videos

Evin Pinar Ornek
Technical University of Munich
evin.oernek@tum.de

Marie-Francine Moens
Katholieke Universiteit Leuven
sien.moens@cs.kuleuven.be

Abstract

In this paper, we examined the zero-shot activity recognition task with the usage of sequential visual frames, i.e. videos. We introduce an auto-encoder model to construct a multimodal joint embedding space between the visual and textual manifolds. On the visual side, we used activity videos and a state-of-the-art 3D convolutional action recognition network to extract their features. On the textual side, we worked with GloVe word embeddings. The zero-shot recognition results are evaluated by top-n accuracy. Then, the manifold learning ability is measured by mean Nearest Neighbor Overlap score. In the end, we discuss the results of our empirical studies and how zero-shot action recognition can be improved.

1. Introduction

In this work, we explore the possibilities of learning **verbs** from multimodal cues in a similar way to humans and propose a neural network model that aims to jointly capture the visual and textual representation. The problem is to build a cross-modal joint space which will help retrieving a textual modal given a visual modal, or vice versa. If it is possible to create such a joint space that connects different modalities and types of inputs, it will be helpful in many arising areas such as video retrieval through natural language, event detection, video captioning, text generation from video, visual hallucination or synthesis from language... Furthermore, a human-like behavior on learning and capturing the multimodal inputs can be defined as a weakly supervised learning problem such as unsupervised learning, few-shot or zero-shot learning.

Specifically, we focus on zero-shot learning of action describing verbs. On visual side, we use video inputs that reflect temporally rich spatial features of each action. On the language side, we use distributional word embeddings of verbs to reflect the semantic features of such verbs through their co-occurences with other verbs. We build a joint embedding space by pairing up the visual and textual embeddings and using two separate auto-encoders for each of them, whose bottleneck layers are shared. The idea of using shared-embeddings between different feature domains exist and already tried out in many works. However, in this work, we question whether such shared spaces will work better than direct mappings, and we evaluate this with Nearest Neighbor Overlap score.

It is shown that the current state-of-the-art zero-shot object recognition is achieved through learnable linear mapping functions with a selection of different losses [7, 6]. On activity recognition side, Zellers et al. [10] used linguistic attributes for activity recognition, whereas Guadarrama et al. [2] focused on subject/verb/object triplets. Xu et al. [9] tried transductive approaches where test labels are observed during training. In this work, we use only video and verb pairs, and we furthermore hypothesize that an auto-encoder will be able to better capture the modal representations than the compatibility mappers in [8] because it will learn to reconstruct the modalities both separately and in a cross modal fashion.

2. Method

Video Understanding Temporal visual features can be captured best through sequences of frames (videos). We used a state-of-the-art action classification network to extract the visual features of videos. Specifically, the penultimate layer of I3D network are used. We sparsely select t frames out of the original number T, and use the embedding of dimension 1024.

Textual Representation The distributional word embeddings have proven to be successful for the natural language processing tasks. In this work, we used Glove word embeddings for simplicity. It is possible to use Word2Vec, or other universal language models. Currently, our question is whether a multimodal space is meaningful for zero-shot learning or not.

Joint Embedding Space We have, (a) a video vector with C features that represent the most important spatio-temporal

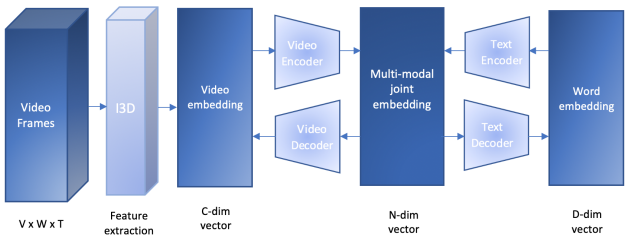

Figure 1. The multi-modal joint embedding architecture. Video input of size *VxWxT* is extracted through i3D model to a video embedding of size *C*. For textual side, relevant word's Glove vector embedding of size *D* is used. Video and text encoder extracts the video and word embeddings to represent common features in a shared joint embedding space. Decoders work on the other direction, to resolve the video and text back to original input. Model is inspired from [5]. Different from them, we aim to build a minimalist zero-shot activity learning model by using short video clips and matching words instead of sentences.

features of the activity, (b) a word vector with D dimensional embeddings. There are several approaches to find the relationship between them. One way is to find a direct mapping from video input to a word embedding. There, each video will be reduced to the number of dimensions of the word embeddings through either a fixed linear map or a neural network. However, as Collell et al. showed in their work, such mappings conserve the semantics of the input vectors rather than learning the common features along with the paired targets or mappings [1]. In other words, they work biased towards the input vectors and will not be able to capture a symmetric relationship between different modalities.

An approach to overcome the inherent limitations of the direct mapping between multimodal vectors might be to build up an embedding space between the videos and words and incorporate a sophisticated loss function to capture the model. Our model is illustrated in the Fig. 1. Inspired from [5], we used two auto-encoders to create an embedding space which are described as:

$$\textbf{Video Encoder } E_V : v \mapsto z_v$$
$$\textbf{Video Decoder } G_V : z \mapsto v$$
$$\textbf{Text Encoder } E_T : t \mapsto z_t$$
$$\textbf{Text Decoder } G_T : z \mapsto t$$

The *"G"* stands for "Generator", whereas *"E"* for "Encoder". Their model is trained with a mixture of these loss functions:

$$\mathcal{L}_{recons}(v,t) = ||G_V(E_V(v)) - v||_2 + ||G_T(E_T(t)) - t||_2 \quad (1)$$

$$\mathcal{L}_{joint}(v,t) = ||E_V(v) - E_T(t)||_2 \quad (2)$$

$$\mathcal{L}_{cross}(v,t) = ||G_T(E_V(v)) - t||_2 + ||G_V(E_T(t)) - v||_2 \quad (3)$$

$$\mathcal{L}(v,t) = \alpha_1\mathcal{L}_{recons}(v,t) + \alpha_2\mathcal{L}_{joint}(v,t) + \alpha_3\mathcal{L}_{cross}(v,t) \quad (4)$$

Our auto-encoder is trained and tested with the loss functions described above (L_{recons}, L_{joint} and L_{cross}). In addition to these losses, the model will be able to learn better with the negative samples. We wish a non-related class vector to be far from the true class vector in the shared space. Hence, we used a margin ranking loss:

$$\mathcal{L}_{rank}(s_1, s_2) = max(0, margin - (s_1 - s_2)) \quad (5)$$

Here, $s_1 = cos(E_V(v), E_T(t))$ is the similarity between the constructed vector from a paired text and video. Whereas $s_2 = cos(E_V(v_n), E_T(t_n))$ is the similarity between the unpaired text and video. We wish the s_1 be higher than s_2. The margin ranking loss enforces this with a predefined margin.

3. Experiments

3.1. Datasets

The Kinetics Human Action Video Dataset consists of 400 different human activity classes with 10 seconds of videos [4]. These activities cover a broad range of movements, such as sports(playing basketball, snowboarding), basic body motions(jumping, clapping), eating or cooking related activities, hobbies, communicative motions etc. As a textual input, the Glove word embeddings of each activity class is used. Some activities include multiple words such as "playing piano". We followed Iyyer et al's approach [3] and averaged the Glove vectors over these words [1]. The Kinetics Dataset has 400 videos for each 400 action classes. In this work, a smaller subset is generated by randomly selecting 300 classes and 40 videos for each of the class. In total, there are 10K train, 1K validation and 1K test data.

[1] https://nlp.stanford.edu/projects/glove/

			Seen Classes			Unseen Classes	
L_{recons}	L_{rank}	L_{cross}	top-5	top-10	top-30	top-5	top-10
1	0	0	12	20	30	0	12
1	1	0	7	10	18	10	14
0.1	1	0	3	4	9	5	7
1	1	1	6	8	21	8	9
0.1	1	1	6	10	21	8	9
0.1	1	1	11	16	29	12	18
0.1	1	1	8	12	23	9	11

Table 1. Results on the action class prediction averaged over the test sets (in %). The top-5, top-10 and top-30 accuracies are given for the seen class prediction on the test dataset, whereas the top-5 and top-10 accuracies are given for the unseen class prediction task.

In addition, we tested the model on the IAPR TC-12 dataset which consists of 200K images and their captions. Visual features are extracted through VGG-128, resulting 128 dimensional embedding. Textual features are extracted by bidirectional gated recurrent unit(bi-GRU) and has 64 dimensions. We used the pre-trained feature vectors from [2]. Even though the main purpose of our model is to learn the temporal visual inputs, static inputs will help to improve and test the model further. IAPR TC-12 Dataset is split to 16K train, 2K validation and 2K test.

3.2. Model and Implementation Details

Both auto-encoders consist of 3 feed-forward layers of decreasing sizes, with the ReLU non-linearity and drop-out between each FF layer. The joint space has a smaller number of dimensions: for the Kinetics Dataset of 1024-d vision and 300-d text inputs. The bottleneck in between is tested with sizes of 300-d, 200-d and 150-d. Likewise, for the IAPR TC-12 Dataset, the joint space size for 128-d vision and 64-d vision is tested with 64-d and 48-d. The auto-encoders are trained with the Adam optimizer, started with a learning rate of $1e-3$ with a weight decay of $1e-5$. The drop-out rate of 0.5 is used, though there was no major effect of different rates. The best models during training are selected by the lowest validation error.

3.3. Evaluation and Results

First, we measured the action class prediction accuracy with **top-N accuracy**. For that, we decoded the test video vectors into textual vectors, and retrieved the most similar N word vectors according to cosine similarity metric. If the actual class is in the set of most similar vectors, it is counted towards a hit, hence contributing for the **top-N**. It is important to note that the tests are not in the "Generalized Zero-Shot Learning(GZSL)" setting, and the similarity is applied in the global word embedding space, not restricted by only verb classes.

Second, the zero-shot action recognition is measured to evaluate the generalization capability of the model over the unseen classes. In this case, the test set consists of randomly selected 10 unseen classes with 40 instances each. We did not include the seen class instances in the test set in order to evaluate the zero-shot accuracy explicitly. Again, we calculated the top-N accuracies.

In Table 1, the prediction accuracy for seen and unseen classes are reported. The hyper-parameters of each loss function is adjusted to examine the effects of different combinations. Each model is trained for 300 epochs. The first five lines show the results for the test setting where there are 300 classes with 40 video instances. In addition, we have extended the tests with two more settings where the results are shown at the bottom two rows of Table 1. The first setting included 100 training classes with 100 videos for each instance. The number of unseen classes is increased to 30. In the second setting, the number of training classes is increased to 200 with 100 instances each, and again with 30 unseen classes.

Here, since there is no other works that use only videos and word embeddings, it is not possible to compare these results with another model. Zellers *et al*. [10] tackles to solve the activity recognition problem, but they use verbs' linguistic properties, which are more informative for classification task than word embeddings.

Nearest Neighbor Overlap

Furthermore, in order to compare the effect of each input modality on the representation space, we have used the mean nearest neighbor overlap measure (mNNO) [1]. The mNNO is defined as:

$$\boldsymbol{mNNO}^K(V,Z) = \frac{1}{KN}\sum_{i=1}^{N} \mathrm{NNO}^K(v_i, z_i) \qquad (6)$$

where $V = \{v_i\}_{i=1}^N$ and $Z = \{z_i\}_{i=1}^N$ are two sets of N paired vectors. The $\mathbf{NNO}^K(v_i, z_i)$ indicates the number of common vectors in the K nearest neighborhoods of v_i and z_i. For instance, let the nearest 3 neighbors of v_{cat} be $\{v_{dog}, v_{tiger}, v_{lion}\}$ and z_{cat} be $\{z_{mouse}, z_{tiger}, z_{lion}\}$. The intersection of their neigborhood is $\{tiger, lion\}$ for K=3, and the mNNO score is $2/3$.

[2] https://github.com/gcollell/neural_cross_modal_maps

In our setting, when we decode a textual vector from a visual vector, the number of neighbor overlaps for these two vectors, and the neighbor overlaps between the ground truth word vector and the decoded word vector should be ideally close to each other. This will mean that the model can learn equal amount of information from different modalities instead of reflecting the topology of only one or the other.

We compared the mNNO scores from text-to-video and from video-to-text, with both the auto-encoder and the linear mapping settings. We have first measured the scores for IAPR TC-12 dataset for clear explanations. Then, we calculated for the Kinetics actions dataset. The results are reported in Table 2.

			$X, f(X)$	$Y, f(X)$
IAPR TC	$V \rightarrow T$	*ff*	**0.428**	0.162
		AE	0.097	**0.257**
	$T \rightarrow V$	*ff*	0.049	0.049
		AE	0.1	**0.236**
Kinetics	$V \rightarrow T$	*ff*	**0.349**	0.109
		AE	**0.144**	0.079
	$T \rightarrow V$	*ff*	0.056	**0.059**
		AE	0.081	**0.149**

Table 2. Mean nearest neighbor overlap scores for video-to-text and text-to-video transfers for the test dataset. *ff* indicates the neural feed forward layer, whereas AE is our two-way auto-encoder. *X* represents the input, *f(X)* the mapped output, and *Y* the ground truth.

It can be seen that for the IAPR TC-12 dataset, the auto-encoder learns the modalities similar to the output modality without depending whether it is a video or a text, though the neural mapper sustains the features of the input modality. For the Kinetics dataset, no matter if it is a linear mapping or an auto-encoder, the transferred features are more similar to video modality. The reason behind this might be the high difference in the number of visual and textual dimensions (1024 versus 300).

4. Discussion and Conclusion

1. During training with L_{joint} loss, the model stopped learning after several iterations. Moreover, it gave a single fixed vector output for any test sample without depending on the input class. Hence, we believe that such a loss used with the combination of others finds a fixed solution and loses the ability to generalize and vary on different inputs.
2. The cross loss taught the model to retrieve the related item across the encoders. The effect of this has been tested in the preliminary experiments. Without the cross loss, the model cannot bridge the modalities.
3. The ranking loss better enhanced the ability to differ between unrelated items and added an extra cue for "sharedness".
4. The drop-out made the model better generalizable and relieved the effect of *hubness* to some extent. Without drop-out, classification results did not have a large variety and pointed to same set of words.
5. Non-linearity increased the accuracy on the preliminary experiments, which is related with increasing the learning capacity of the model.

The existing zero-shot **object** classification problem is shown to have higher accuracy with compatibility learning models that learn the mapping between the distributions rather than the attribute classifiers [8]. However, zero-shot **activity** recognition state-of-the-art accuracy is achieved by the help of manually defined linguistic attributes of verbs [10]. In this work, to remove the need of such manual dependencies, we attempted to use a compatibility learning model through a minimalistic cross-modal network. Future work includes extending this with discriminative loss, using a different language model or a video recognition model.

References

[1] Guillem Collell and Marie-Francine Moens. Do neural network cross-modal mappings really bridge modalities? In *Association for Computational Linguistics(ACL)*, 2018.

[2] Sergio Guadarrama, Niveda Krishnamoorthy, Girish Malkarnenkar, Subhashini Venugopalan, Raymond Mooney, Trevor Darrell, and Kate Saenko. Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition. In *ICCV, 2013.*

[3] Mohit Iyyer, Varun Manjunatha, Jordan L. Boyd-Graber, and Hal Daumé III. Deep unordered composition rivals syntactic methods for text classification. In *ACL*, pages 1681–1691, 2015.

[4] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017.

[5] A. Piergiovanni and M. S. Ryoo. Unseen Action Recognition with Multimodal Learning. *ArXiv e-prints*, June 2018.

[6] Richard Socher and Milind Ganjoo and Christopher D. Manning and Andrew Y. Ng. Zero Shot Learning Through Cross-Modal Transfer. In *NIPS*. 2013.

[7] Carina Silberer and Mirella Lapata. Learning grounded meaning representations with autoencoders. In *ACL*, 2014.

[8] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning - the good, the bad and the ugly. In *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2017.

[9] Xu Xun, Hospedales Timothy, and Gong Shaogang. Transductive zero-shot action recognition by word-vector embedding. *International Journal of Computer Vision*, 123(3), July 2017.

[10] Rowan Zellers and Yejin Choi. Zero-shot activity recognition with verb attribute induction. In *EMNLP*, 2017.