Learning Event Representations by Encoding the Temporal Context

Catarina Dias[1] and Mariella Dimiccoli[2,3](✉)

[1] Faculty of Engineering, University of Porto,
Rua Doutor Roberto Frias, 4200-465 Porto, Portugal
catfdias@gmail.com

[2] Department of Mathematics and Computer Science, University of Barcelona,
Gran via de les Corts Catalanes 585, 08007 Barcelona, Spain
dimiccolimariella@gmail.com

[3] Computer Vision Center, Campus UAB,
Edifici O, 08193 Cerdanyola del Valles, Barcelona, Spain

Abstract. This work aims at learning image representations suitable for event segmentation, a largely unexplored problem in the computer vision literature. The proposed approach is a self-supervised neural network that captures patterns of temporal overlap by learning to predict the feature vector of neighbor frames, given the one of the current frame. The model is inspired to recent experimental findings in neuroscience, showing that stimuli associated with similar temporal contexts are grouped together in the representational space. Experiments performed on image sequences captured at regular intervals have shown that a representation able to encode the temporal context provides very promising results on the task of temporal segmentation.

Keywords: Representation learning · Event learning · LSTM · Neural networks

1 Introduction

As our sensory system is inherently continuous, we experience the world as an uninterrupted stream of perceptual stimuli. However, sensory information is automatically segmented by our brain into discrete *events* that can be understood, remembered and retrieved from the memory. How these event representations are generated at neural level is a very active area of research in neuroscience [1–5]. Although firstly questioned more than fifty years ago [1], it was only in 2007 that the seminal work of Zacks [2] revealed the key role of uncertainty and surprise in determining event boundaries. Later on, Kurby and Zacks [3] hypothesized that event segmentation might arise as a side effect of integrating information over the recent past to improve predictions about the near future. More recently, Shapiro et al. [4] have shown that neural representation

L. Leal-Taixé and S. Roth (Eds.): ECCV 2018 Workshops, LNCS 11131, pp. 587–596, 2019.
https://doi.org/10.1007/978-3-030-11015-4_44

of events are not tied to predictive uncertainty, but arise from temporal community structures: items that share the temporal context are grouped together in a representational space. Focusing more on a higher level processing, DuBrow and Davachi [5] have argued that event boundaries are generated by changes in our goals and these goals determine how information is stored and retrieved in our brain.

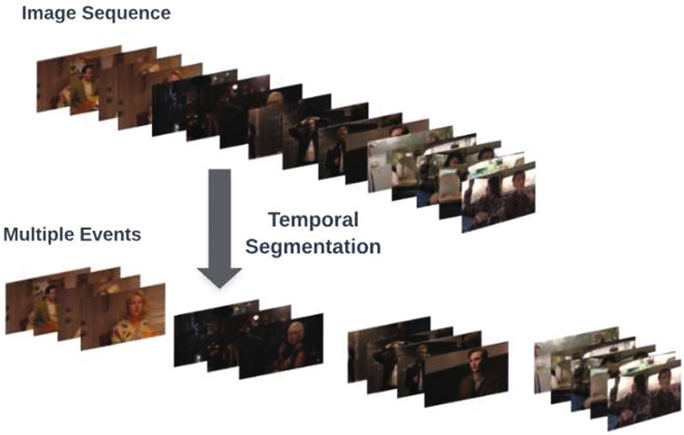

Fig. 1. Example of temporal segmentation of a sequence of images.

Besides neuroscience, event segmentation is also receiving an increasing attention in computer vision [6–10]. Indeed, temporal video segmentation can be considered the first step towards automatic annotation and recognition of digital video sequences (see Fig. 1). The growing size of today's available unconstrained videos on internet raises the need of automatically detecting, recognizing and retrieving the type of complex events occurring in them. Typically, state of the art algorithms for event or action recognition rely on event boundary detection and visual features extracted at image level or at event level such as semantic features or optical flow. Most recent and promising approaches for event detection [11,12] use concept scores as intermediate representation, which are the confidence of the occurrence of the concepts in the video. For example, the event *Having a dinner with friends* can be described as the occurrence of *food*, *laughing*, *people*, *bottles* ... etc. However, the resulting concept-based event representation is highly noisy due to the high variability of concept's appearance on the one hand and to the high variability of the concept representation for a complex event on the other hand. Therefore it is difficult to segment temporally a video based on semantic features.

Given the relevance of the problem, in this paper, we propose to learn a representation for image sequences suitable for the task of temporal segmentation. Inspired by the theory Shapiro et al. [4], our model aims at embedding the temporal context of images via a self-supervised *pretext task* consisting in predicting the feature vector of neighbor frames given the concept vector of the current frame.

The reminder of this paper is as follows. In Sect. 2, we present the proposed approach, while in Sect. 3 we detail and discuss experimental results. We conclude the paper with Sect. 4, summarizing its main contributions and findings.

2 Learning Event Representations

Here, we first introduce the concepts underlying the proposed model. Second, we describe the *pretext task* that we use to learn the temporal representations and subsequently present the *validation task* which is employed to evaluate the quality of the learned representations (see Fig. 2).

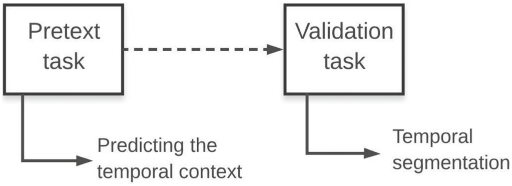

Fig. 2. Modules of the proposed method.

2.1 Underlying Model

Hereafter, we consider a toy example to illustrate the intuition underlying our model. Let us suppose that we are given a sequence of N images and each image is represented by a feature vector as shown in Fig. 3(a).

Relying on the feature vector representation, we built a directed graph, where each vertex corresponds to a feature vector and each direct edge going from, says, A to B indicates that feature vector B temporally follows feature vector A in the image sequence. The resulting graph, shown in Fig. 3(b) indicates that the underlying representation of the image sequence presents two community structures, with many edges joining vertices of the same community and comparatively few edges joining vertices of different communities. Therefore, the directed graph could be regarded as a more intuitive simplified model that represents the temporal context of sequence data.

In the next section, we focus on how to automatically learn a feature space, where temporally nearby frames belonging to the same event lie close to each other.

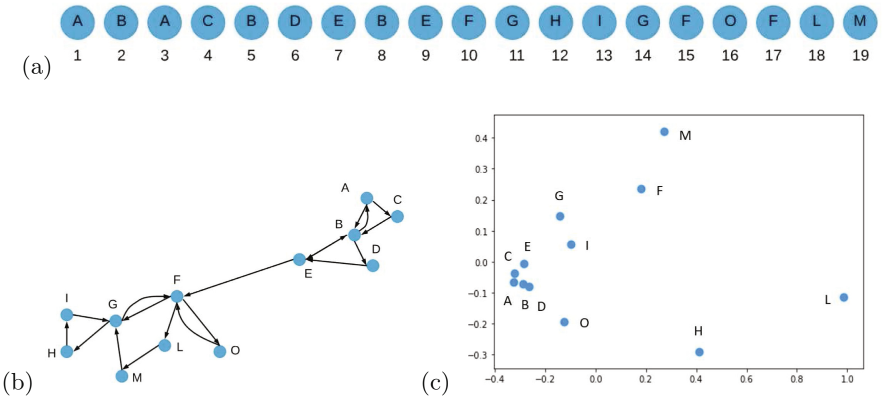

Fig. 3. (a) Image sequence representation: the letters indicate the feature vectors identifier and the numbers represent the temporal order. (b) Directed graph encoding the temporal relations between feature vectors. (c) Non-classical and non-metric multidimensional scaling of the activation vectors.

2.2 Pretext Task

Representation learning aims at building intermediate representations of data useful to solve machine learning tasks. In self-supervised learning, one trains a model to solve a so-called *pretext task* on a dataset without the need for human annotation by exploiting labeling that comes for "free" with the data.

In our case, similarly to *word2vec* models in natural language processing [13], the pretext task is a prediction task, that given a frame, aims at predicting the temporally neighbor frames corresponding to the temporal context. This leads to learn a function from a given frame to the frames surrounding it. We considered two different implementations. The first one is a simple neural network with a single hidden layer that takes as input the feature vector x_i of the frame i and is trained to output the concept vector $x_{i\pm n}$ of the frame $i \pm n$, with $n \in \{1, .., m\}$ by minimizing the Mean Squared Error (MSE) between x_i and the estimation of x_{i+1}, say $\hat{x}_{i+1}$. After training, the new feature vector $\tilde{x}_i$ embedding the temporal context for the image i, is obtained by multiplying x_i to the learned weight matrix W. This procedure is illustrated on Fig. 4.

The second one is a many-to-many encoder-decoder long short term memory (LSTM) recurrent neural network. The model is trained to predict the feature vector of the next frame in a sequence based on the n previous ones. This process is illustrated in Fig. 5. The network is trained by feeding batches of sequential frames of size $n < N$ randomly extracted, where N is the full length of the video sequence. The new family of feature vectors $\tilde{\mathbf{x}}$ are obtained after training, as output of the encoder by feeding it the original feature vectors $\mathbf{x}$.

In the case of the toy example illustrated in the previous section, we used one-hot encoding for the 12 feature vectors and we trained a neural network to predict the feature vector of the image $i + 1$ given the concept vector i.

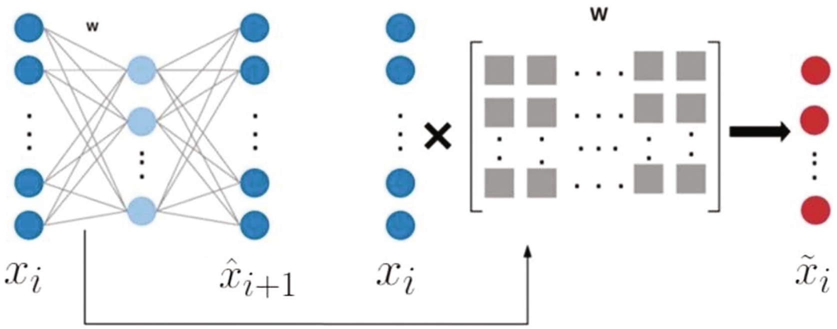

Fig. 4. Left: Training a single layer neural network to predict the concept vector of the image $i+1$ given the concept vector of the image i as input. Right: Extraction of the new feature vector for the image i by multiplying the original concept vector by the weights of the trained neural network.

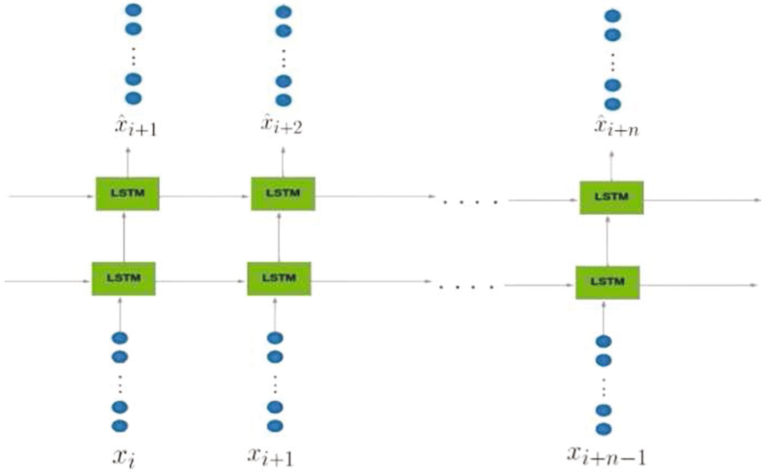

Fig. 5. Encoder-decoder LSTM layer neural network. The new feature vector is obtained as output of the decoder (first layer LSTM) after training.

In Fig. 3(c) is shown the multidimensional scaling of the activation vectors obtained multiplying the original feature vectors by the weight matrix learned with the prediction task. From this figure we can clearly see how the feature vectors that on Fig. 3(b) were linked by numerous edges lie close to each other on this new representational space (A, B, C, D, E on the one side, and G, I, F, M, H, L, O on the other side).

Since the main objective is to transfer the learned representation to target task, in the next section we detail how we validate the goodness of the new learned representation.

2.3 Validation Task

To validate the goodness of the learned representation we applied a classical temporal segmentation algorithm to both, the concept vector representation and the new learned representation. In particular, we used a Binary Partition Tree (BPT) based approach [14]. BPT is a hierarchical clustering method that iteratively merges the most similar temporally adjacent neighbor frames until a single event is obtained. The initial partition of events is given by all frames. Each time two neighboring events are merged, the resulting event is modeled as the mean feature vector of the two original events. The temporal segmentation is obtained by cutting the binary tree, using as criterion the number of events or a more complex function that takes into account the similarity of all the merging. Since the goal of this work is to show that encoding the temporal context is beneficial for temporal segmentation, we used the most simple criterion.

3 Experimental Results

In the following, we first detail our experimental setup, including the dataset used in the experiments, the metrics employed in the validation task, how the initial feature vectors where computed and which type of experiments were performed. Then, we present and discuss the experimental results.

3.1 Experimental Setup

Dataset. We used a subset of the EDUB-Seg dataset [15,16], consisting of ten image sequences belonging to five different users and captured by a wearable photo-camera that takes pictures at regular intervals of 30 s, with an average of 662 images per sequence. The subset we considered, comes together with the ground truth event segmentation and concept vectors describing the probability of each concept in the images.

Evaluation Measure. To evaluate the performance of the temporal segmentation we used the F-measure. In particular, we considered as true positives (TPs) the images that the BPT detects as boundaries of an event and that were manually defined in the GT. The false positives (FPs) are the images detected as events delimiters, but that were not marked in the GT, and the false negatives (FNs) the boundaries not detected but present in the GT. In all cases we considered a tolerance of 5 frames as in [16]. Good event segmentations correspond to F-measure values close to 1.

Feature Extraction. In our experiments, the original feature vectors are obtained as in [16] by firstly detecting concepts independently on each image by means of a concept detector, and then by clustering them in a semantic space by relying on WordNet [17] (see Fig. 6). The number of clusters found determines the size of the vocabulary of concepts. Each element of the feature vector corresponds to the probability of finding a given concept in the image.

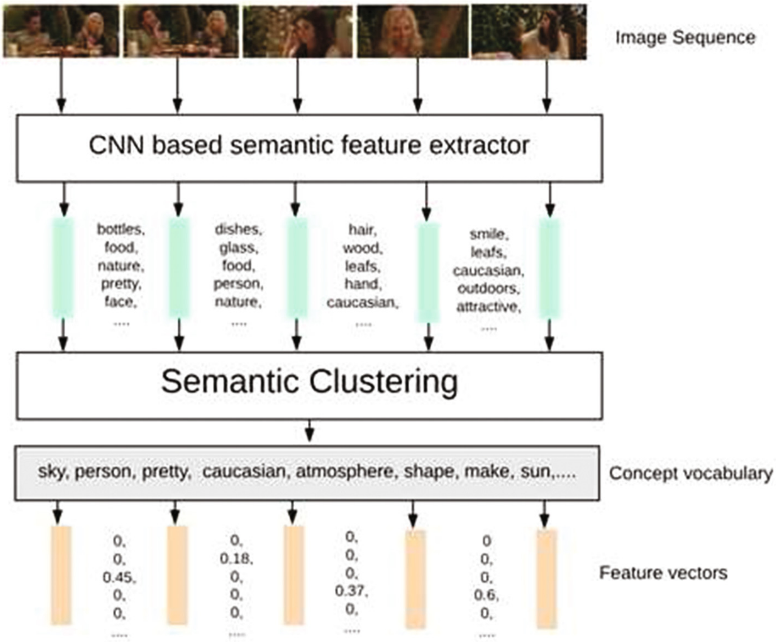
Image Sequence
CNN based semantic feature extractor
bottles, food, nature, pretty, face, ...
dishes, glass, food, person, nature, ...
hair, wood, leafs, hand, caucasian, ...
smile, leafs, caucasian, outdoors, attractive, ...
Semantic Clustering
sky, person, pretty, caucasian, atmosphere, shape, make, sun,....
Concept vocabulary
0, 0, 0.45, 0, 0, ...
0, 0.18, 0, 0, 0, ...
0, 0, 0, 0.37, 0, ...
0 0, 0, 0.6, 0, ...
Feature vectors

Fig. 6. Semantic feature extraction and generation of concept-based event representation as in [16].

Experiments. We performed three different class of experiments, with slightly different prediction tasks. The first one is the most simple forward prediction (NN forward): given the i^{th} feature vector as input to predict the $(i+1)^{\text{th}}$ feature vector. The second one is a forward prediction that takes into account the previous n feature vectors to predict the next feature vector (LSTM forward). In our experiments we used $n = 50$. The third one is a forward-backward prediction (NN forward-backward): given the i^{th} feature vector as input to predict the neighbor feature vectors inside a window of length n centered at i. We considered $n = \{2, 3, 4, 5\}$. All models were trained by using the MSE of the prediction as loss function and by using Stochastic Gradient Descent (SGD) as optimizer. The results reported on Tables 2 and 3 were achieved by setting the learning rate to 0.1 for the encoder-decoder LSTM model, with momentum to 0.9 for the NN models. The number of neurons used in the hidden layer was different accordingly to the prediction task as shown on Table 1. The size of the original feature vector was 50, extracted following the procedure described in Sect. 2.2. For each sequence the weights that ensured a better performance were saved for later generation of the new learned representation. The BPT based approach described in Sect. 2.3 for event segmentation was applied by relying on both, the original feature vectors and the new learned representation.

Table 1. Number of neurons in the hidden layer used in the different prediction tasks.

Prediction task	#neurons
NN F	30
LSTM F	20
NN FB	30

Table 2. Results obtained in the different experiments for each sequence.

	Baseline	NNF n = 1	NNFB n = 2	NNFB n = 3	NNFB n = 4	NNFB n = 5	LSTM F n = 1
User1-1	0.32	0.39	0.51	0.57	**0.71**	0.47	0.35
User1-2	0.29	0.27	0.36	0.38	0.35	**0.39**	**0.39**
User1-3	0.57	0.58	0.56	**0.70**	0.67	0.53	0.53
User2-1	0.51	0.53	0.56	**0.59**	0.56	0.50	0.50
User2-2	0.38	0.60	0.61	0.64	**0.69**	0.67	0.65
User2-3	0.56	0.71	**0.79**	0.75	0.71	0.75	0.75
User3-1	0.30	0.25	0.33	0.35	0.34	0.27	**0.41**
User3-2	0.39	**0.42**	0.33	0.39	0.37	0.37	0.34
User4	0.33	0.42	0.35	0.42	0.34	0.31	**0.43**
User5	0.38	0.44	0.39	0.42	0.43	0.43	**0.47**

3.2 Results and Discussion

Table 2 shows the results obtained on each sequence and for each prediction task. On Table 3, is reported the average over all sequences for each experiment. As it can be observed on Table 3, the F-measure of the temporal segmentation obtained relying on a representation that encode the temporal context, outperforms the baseline, obtained using the original feature vectors, for all prediction tasks. In particular, the representation learned through the forward-backward prediction achieves the best performance. Furthermore, the performance increases with the size of the temporal window achieving the maximum value for a window of size 3 and then decreases again. These results have shown that, although its simplicity, the proposed approach is very effective to learn event representations, and suggest that encoding the temporal context is crucial for event learning.

Key issues to be investigated, is what features are most suited as basis for the temporal embedding in videos, if they can be learned in an end-to-end fashion and which prediction task would be more effective in the video domain.

Table 3. Average results for each prediction task and baseline performance.

Validation task	F-measure
NN F, n = 1	0.4610
LSTM F	0.4820
NN FB, n = 2	0.4790
NN FB, n = 3	**0.5210**
NN FB, n = 4	0.5170
NN FB, n = 5	0.4690
Baseline	0.4030

4 Conclusions

To the best of our knowledge, this work has presented the first attempt to learn image representations suitable for event segmentation. The proposed approach is inspired to recent experimental findings in neuroscience showing that neural representations of events arise from temporal community structures. To learn the temporal embedding, we proposed a pretext task consisting of predicting the feature vector of neighboring images in a temporal window of fixed size, by using two different approaches: a simple neural network and an encoder-decoder LSTM. Experimental results performed on a dataset of image sequences captured at regular intervals have shown that the new learned representation outperforms the original feature-based representation on the task of temporal segmentation. The generalization of the approach to temporal segmentation of video, would have an important impact in the processing of untrimmed videos.

Acknowledgments. This work was partially founded by TIN2015-66951-C2, SGR 1742, ICREA Academia 2014, Marató TV3 (20141510), Nestore Horizon2020 SC1-PM-15-2017 (769643) and CERCA. The funders had no role in the study design, data collection, analysis, and preparation of the manuscript. The authors gratefully acknowledge NVIDIA Corporation for the donation of the GPU used in this work.

References

1. Newtson, D., Engquist, G.A., Bois, J.: The objective basis of behavior units. J. Pers. Soc. Psychol. **35**(12), 847 (1977)
2. Zacks, J.M., Speer, N.K., Swallow, K.M., Braver, T.S., Reynolds, J.R.: Event perception: a mind-brain perspective. Psychol. Bull. **133**(2), 273 (2007)
3. Kurby, C.A., Zacks, J.M.: Segmentation in the perception and memory of events. Trends Cogn. Sci. **12**(2), 72–79 (2008)
4. Schapiro, A.C., Rogers, T.T., Cordova, N.I., Turk-Browne, N.B., Botvinick, M.M.: Neural representations of events arise from temporal community structure. Nature Neurosci. **16**(4), 486 (2013)
5. DuBrow, S., Davachi, L.: Temporal binding within and across events. Neurobiol. Learn. Memory **134**, 107–114 (2016)

6. Koprinska, I., Carrato, S.: Temporal video segmentation: a survey. Signal Process. Image Commun. **16**(5), 477–500 (2001)
7. Krishna, M.V., Bodesheim, P., Körner, M., Denzler, J.: Temporal video segmentation by event detection: a novelty detection approach. Pattern Recogn. Image Anal. **24**(2), 243–255 (2014)
8. Liwicki, S., Zafeiriou, S.P., Pantic, M.: Online kernel slow feature analysis for temporal video segmentation and tracking. IEEE Trans. Image Process. **24**(10), 2955–2970 (2015)
9. Theodoridis, T., Tefas, A., Pitas, I.: Multi-view semantic temporal video segmentation. In: 2016 IEEE International Conference on Image Processing (ICIP), pp. 3947–3951. IEEE (2016)
10. Iwan, L.H., Thom, J.A.: Temporal video segmentation: detecting the end-of-act in circus performance videos. Multimed. Tools Appl. **76**(1), 1379–1401 (2017)
11. Xu, Z., Yang, Y., Hauptmann, A.G.: A discriminative CNN video representation for event detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1798–1807 (2015)
12. Chang, X., Yang, Y., Hauptmann, A.G., Xing, E.P., Yu, Y.L.: Semantic concept discovery for large-scale zero-shot event detection. In: International Joint Conference on Artificial Intelligence (IJCAI) (2015)
13. Mikolov, T., Yih, W.t., Zweig, G.: Linguistic regularities in continuous space word representations. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 746–751 (2013)
14. Salembier, P., Garrido, L.: Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval. IEEE Trans. Image Process. **9**(4), 561–576 (2000)
15. Talavera, E., Dimiccoli, M., Bolaños, M., Aghaei, M., Radeva, P.: R-clustering for egocentric video segmentation. In: Paredes, R., Cardoso, J.S., Pardo, X.M. (eds.) IbPRIA 2015. LNCS, vol. 9117, pp. 327–336. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19390-8_37
16. Dimiccoli, M., Bolaños, M., Talavera, E., Aghaei, M., Nikolov, S.G., Radeva, P.: SR-clustering: semantic regularized clustering for egocentric photo streams segmentation. Comput. Vis. Image Underst. **155**, 55–69 (2017)
17. Miller, G.A.: Wordnet: a lexical database for english. Commun. ACM **38**(11), 39–41 (1995)