Early Embedding and Late Reranking for Video Captioning

Jianfeng Dong[†] Xirong Li[‡*] Weiyu Lan[‡] Yujia Huo[‡] Cees G. M. Snoek[§]
[†]College of Computer Science and Technology, Zhejiang University
[‡]Key Lab of Data Engineering and Knowledge Engineering, Renmin University of China
[§]Intelligent Systems Lab Amsterdam, University of Amsterdam

ABSTRACT

This paper describes our solution for the MSR Video to Language Challenge. We start from the popular ConvNet + LSTM model, which we extend with two novel modules. One is *early embedding*, which enriches the current low-level input to LSTM by tag embeddings. The other is *late reranking*, for re-scoring generated sentences in terms of their relevance to a specific video. The modules are inspired by recent works on image captioning, repurposed and redesigned for video. As experiments on the MSR-VTT validation set show, the joint use of these two modules add a clear improvement over a non-trivial ConvNet + LSTM baseline under four performance metrics. The viability of the proposed solution is further confirmed by the blind test by the organizers. Our system is ranked at the 4th place in terms of overall performance, while scoring the best CIDEr-D, which measures the human-likeness of generated captions.

Keywords

Video captioning; MSR Video to Language Challenge; Tag embedding; Sentence reranking

1. INTRODUCTION

The goal of this paper is to automatically assign a caption to a web video, such as 'A teenage couple perform in an amateur musical' or 'Cars racing on a road surrounded by lots of people'. State-of-the-art approaches rely on a deep convolutional network with a recurrent neural network [13, 15], and emphasize on innovating inside the network architecture [9, 16]. We focus on enhancing video captioning, without the need to change internal structures of the networks.

We are inspired by [2]. The authors employ a bi-modal semantic embedding model to project image and text into a common subspace, wherein image-text similarity is computed and later used for sentence reranking to refine captions for images. We also learn from [4], which annotates a test image with Flickr tags by neighbor voting [5], and reranks sentences generated by an LSTM according to their match with the tags. We adopt the two ideas, but repurpose and redesign the captioning architecture for video.

Contributions of this work are as follows. We answer the MSR Video to Language Challenge by proposing an *Early Embedding* and *Late Reranking* solution. Given the importance of the initial input to LSTM, early embedding is introduced to enrich the network input by tag embeddings, based on a novel re-use of video tagging results. Late reranking is to promote relevant captions by re-scoring a list of candidate sentences. Extensive experiments on the MSR-VTT-10k validation set shows that our solution brings clear improvements to a ConvNet + LSTM baseline under varied metrics including BLEU4, METEOR, CIDEr-D and ROUGE-L. The effectiveness is also verified by the blind test conducted by the organizers.

2. PROPOSED SOLUTION

Our solution is inspired by the popular ConvNet + LSTM architecture [14]. We introduce two new modules, i.e., early embedding and late reranking, designed to enhance the input and the output of the underlying sentence generation model. The proposed solution is illustrated in Fig. 1. Our system is able to cope with novel sentence generation models as long as they accept a real-valued feature vector as input and produce a number of sentences.

2.1 Video Representation

Following the common practice of applying pre-trained ConvNets for video content analysis [7, 9, 15], we extract ConvNet features for a given video clip. Frames are uniformly sampled from the clip with an interval of 10 frames. A video level representation is obtained by mean pooling on ConvNet features extracted from the frames. Concerning the choice of the ConvNet, we employ Googlenet-bu4k, a variant of Googlenet [10] trained by Mettes *et al.* [7] using a bottom-up reorganization of the ImageNet hierarchy. Our experiments confirm that this model, though originally aiming for video event recognition, is better than its standard counterpart for video captioning as well. The pool5 layer after ReLU is used, resulting in a feature vector of 1,024 dimensions. Observing the increasing popularity of 3-D ConvNets (C3D) for video captioning [9, 15], we have also experimented with a C3D model trained by Tran *et al.* on one million sports videos [11]. Though being longer (4,096-dim), the C3D feature is inferior to the Googlenet-bu4k feature according to our experiments. Moreover, in

*Corresponding author (xirong@ruc.edu.cn)

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MM '16, October 15 - 19, 2016, Amsterdam, Netherlands

© 2016 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ISBN 978-1-4503-3603-1/16/10…$15.00

DOI: http://dx.doi.org/10.1145/2964284.2984064

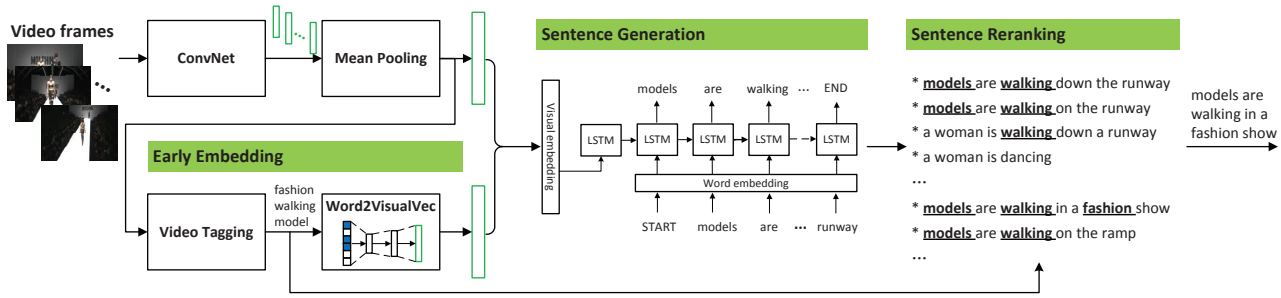

Figure 1: Proposed video captioning system. Given a test video, we extract a video-level ConvNet feature, which is concatenated with a tag embedding by the *Early Embedding* module. The enriched input is fed into the *Sentence Generation* module, which outputs a list of 20 sentences. By *reranking* these sentences either by tag matching or video-text similarity, our system chooses the top ranked sentence as the final video caption.

contrast to previous works [9, 15], the concatenation of the two features does not yield any improvement. Hence, we rely on the Googlenet-bu4k feature as the visual input to the sentence generation module.

2.2 Early Embedding

Instead of directly feeding the visual feature to the sentence generation module as done in previous works, we enrich the input using tags detected to be relevant with respect to the given video. Next, we describe how the tags are predicted, and consequently two strategies for tag embedding.

2.2.1 Video Tagging

Tags are predicted through multiple channels. First, since we have employed two ConvNets, it is natural to leverage their semantic output, i.e., a set of 4k ImageNet labels predicted by Googlenet-bu4k and a set of 487 sport-related concepts from C3D. The 4k labels are down-sampled from the complete ImageNet dataset by using a bottom-up reorganization of the ImageNet hierarchy, excluding over-specific classes and classes with few images and thus making the final classes have balanced positive images [7]. Besides, we train linear SVM classifiers on the public FCVID dataset [3], giving us a video tagging system that predicts 237 video categories. Notice that these three concept sets were constructed independently in advance to this challenge. In order to predict tags more relevant to the challenge, we further exploit the MSR-VTT training set [15] by the neighbor voting algorithm [5]. Despite its simplicity, this algorithm remains on par with more complex alternatives [6]. Given a test video, ten nearest neighbor videos are retrieved from the training set, and tags with the highest occurrence in captions of the neighbors are selected.

Although the C3D feature is less effective than the Googlenet-bu4k feature, it appears to better capture strong motion-related patterns, say in sports. Hence, we employ both features for neighbor voting. This results in tags predicted by five channels, summarized as

1. A pretrained Googlenet-bu4k [7], predicting 4,437 ImageNet classes;
2. A pretrained C3D [11], predicting 487 sport-related concepts;
3. Linear SVMs trained by us on FCVID using the Googlenet-bu4k feature, predicting 237 video categories;
4. Neighbor voting with the MSR-VTT training set as the source set and the Googlenet-bu4k feature for similarity computation;
5. Neighbor voting with the MSR-VTT training set as the source set but using the C3D feature instead.

We fuse the multiple tagging results by majority voting, empirically preserving the top three tags for the subsequent tag embedding.

2.2.2 Tag Embedding

For deriving a vectorized representation from the predicted tags, Bag-of-Words (BoW) is a straightforward choice. We construct the BoW vocabulary by sorting tags in descending order in terms of their frequency in the training set and preserve the top 1,024 tags, a number equaling to the size of the Googlenet-bu4k feature. Entries of the BoW vector are zero except for those dimensions corresponding to the predicted tags.

As BoW is known to have difficulties in describing inter-tag relationships, we further consider a deeper tag embedding using a very recent Word2VisualVec model [1], which is capable of predicting visual ConvNet features from text. Word2VisualVec adds a multi-layer perceptron on top of Word2Vec [8], with the network optimized such that the output of an input sentence is close to the ConvNet feature of an image the sentence is describing. Hence, Word2VisualVec captures visual and semantic similarities. We empirically choose a three layer structure of 500-100-1024 for predicting Googlenet-bu4k features and 500-1000-2000-4096 for predicting C3D features. By default the model embeds tags into the Googlenet-bu4k feature space for its good performance, unless stated otherwise.

Given either BoW or Word2VisualVec embedding, the tag vector is concatenated with the Googlenet-bu4k feature extracted from the video, and fed into the sentence generation module described next.

2.3 Sentence Generation

We train the image captioning model of Vinyals *et al.* [14] for sentence generation. At the heart of the model is an LSTM network which generates a sentence given the visual input, with the goal of maximizing the sentence's posterior probability. Let θ be the network parameters. The probability is expressed as $p(S|x;\theta)$, where x is the video feature, S is a sentence of n words, $S = \{w_1, ..., w_n\}$. Applying the

chain rule together with a log function, the probability is computed via

$$\log p(S|x;\theta) = \sum_{t=0}^{n+1} \log p(w_t|x, w_0, ..., w_{t-1}; \theta), \qquad (1)$$

where w_0 = START and w_{n+1} = END are two special tokens indicting the beginning and the end of the sentence. Conditional probabilities in Eq. 1 are computed in a greedy manner, based on the current chosen word and the LSTM memory. The size of the memory cell in LSTM is empirically set to 512. Beam search is applied to generate a list of 20 sentences most likely to describe the test video.

Next, we aim to improve the quality of video captioning by sentence reranking.

2.4 Late Reranking

We are inspired by concept-based [4] and semantic embedding based [2] sentence ranking, but we repurpose and redesign them for video.

For concept-based sentence reranking, instead of averaging over tags as described in [4], we use the sum function to favor sentences that maximize the matches. For instance, given prediction 'dog' (0.6) and 'playing' (0.3), we prefer 'a dog is playing' to 'a dog is running on grass'. Given the predicted tag set P, the matched score of a sentence S is computed as

$$TagMatch(S;P) = \sum_{w \in P \cap S} score(w), \qquad (2)$$

where $score(w)$ is the tag relevance score provided by the video tagging system described in Section 2.2. As our experiments show, using sum instead of averaging gives a clear boost to the performance.

For semantic embedding based sentence reranking, we reuse the previously trained Word2VisualVec to project each sentence into the same visual feature space as the video. Consequently, we compute the video-text relevance score in terms of the cosine similarity between the corresponding ConvNet features.

The sentences are reranked in descending order either by the TagMatch scores or by the cross-media scores, and the top positioned one is chosen as the final video caption.

3. EVALUATION

3.1 Experimental Setup

Datasets. Our video captioning system is trained using the MSR-VTT-10K [15] training set, which consists of 6,513 video clips. Each clip is associated with 20 English sentences generated by crowd sourcing and one of 20 high-level YouTube categories such as "music", "people"and "gaming". The system and its various settings are evaluated using the MSR-VTT-10K validation set of 497 video clips.

Performance metric. Following the evaluation protocol, we report BLEU@4, METEOR, CIDEr-D and ROUGE-L. CIDEr-D is specifically designed to measure the extent to which automatically generated sentences appearing to be written by humans [12], while the other three metrics are originally meant for evaluating machine translation. Averaged value of these metrics is used for overall comparison in the Challenge.

3.2 Experiments

Our baseline is a standard ConvNet + LSTM architecture, without early embedding and late reranking. It scores BLEU4 of 37.9, METEOR of 25.0, CIDEr-D of 35.7 and ROUGE-L of 57.0 on the validation set, see Table 1. In order to investigate the effect of early embedding, late reranking and their joint use on the performance, we incrementally add and evaluate the individual components with varied settings. This results in fifteen runs in total.

Early Embedding. For the ease of comparison, the runs have been sorted in descending order according to average value of the four metrics. As we see from Table 1, the runs with early embedding only, i.e., #2, #6, #7 and #8, are ranked ahead of the baseline, showing the effectiveness of early embedding. Given the same tagging result, either from KNN or from MajorityVote, Word2VisualVec gives comparable or better performance than BoW.

Late Reranking. We compare the four runs using late reranking alone, i.e., #3, #11, #13 and #15, against the baseline. The fact that two runs are positioned after the baseline suggests that the reranking module needs to be designed more carefully. Compared to the baseline, CIDEr-D and METEOR of run#13 in fact increase from 35.7 to 38.1 and from 25.0 to 25.9, respectively. Nonetheless, there is a significant drop on BLEU4, from 37.9 to 31.5. This is probably because BLEU4 (and similarly ROUGE-L) relies on precise matches between n-grams, while such an order is discarded in the Word2VisualVec embedding. Interestingly, video-text similarity computed in the C3D feature space outperforms the baseline, with its BLEU4 remaining lower. The result suggests that utilizing a visual feature different from the feature that is already used for sentence generation is more effective for sentence reranking. TagMatch is the most effective implementation of the reranking module.

Early Embedding and Late Reranking. As Table 1 shows, the runs that jointly use early embedding and late reranking beat the baseline except for run#14 which uses the less effective TagMatch-mean strategy. The best solution is to perform video tagging by MajorityVote, encode the predicted tags via Word2VisualVec, and later rerank the sentences by TagMatch.

Blind test result. For blind test on the test set, we are allowed to submit three runs at the maximum. Their configurations correspond to run #1, #2 and #6 in Table 1. An overview of their performance on the test set is presented in Table 2. Performance rank is consistent with the validation set, showing that our solution generalizes to previously unseen data. Some video tagging and captioning results are given in Table 3. Our system scores the best CIDEr-D.

4. CONCLUSIONS

Experiments on the MSR-VTT validation set and the blind test on the test set support conclusions as follows. The proposed early embedding and late reranking solution can effectively improve the state-of-the-art ConvNet + LSTM method for video captioning. We demonstrate a successful re-use of image classification results from ConvNets for refining video captions, even though they were not meant for this purpose. Our recommendation is to perform video tagging by MajorityVote, encode the predicted tags via Word2VisualVec, and later rerank the sentences by TagMatch.

Table 1: Performance of our solution with varied settings on the validation set, sorted by averaged value of the four metrics in descending order. BoW means representing predicted tags by bag-of-words, while Word2VisualVec is to encode these tags using Word2VisualVec. TagMatch-mean denotes the strategy described in [4].

Run	Module Configuration		Performance Metrics			
	Early Embedding	*Late Reranking*	*BLEU4*	*METEOR*	*CIDEr-D*	*ROUGE-L*
1	MajorityVote + Word2VisualVec	TagMatch	39.4	**27.5**	**48.0**	**60.0**
2	MajorityVote + Word2VisualVec	–	**40.5**	26.0	44.1	58.7
3	–	TagMatch	38.4	27.0	44.1	59.1
4	KNN + BoW	TagMatch	37.0	26.5	45.2	58.3
5	MajorityVote + Bow	TagMatch	37.9	26.4	44.4	58.1
6	KNN + Word2VisualVec	–	40.3	25.4	40.6	57.8
7	KNN + BoW	–	38.6	25.5	41.3	58.7
8	MajorityVote + BoW	–	39.0	25.2	41.5	58.0
9	MajorityVote + Word2VisualVec	video-text similarity (C3D)	34.9	26.4	41.1	57.0
10	MajorityVote + Word2VisualVec	video-text similarity	33.8	26.4	40.9	56.7
11	–	video-text similarity (C3D)	33.9	26.2	40.1	56.5
baseline	–	–	37.9	25.0	35.7	57.0
13	–	video-text similarity	31.5	25.9	38.1	55.4
14	MajorityVote + Word2VisualVec	TagMatch-mean	32.5	23.8	37.2	55.0
15	–	TagMatch-mean	31.6	23.4	31.0	54.0

Table 2: Performance on the test set. The top rows correspond to the top three performers.

Official rank	Submission	BLEU4	METEOR	CIDEr-D	ROUGE-L
1	v2t_navigator	**40.8**	**28.2**	44.8	**60.9**
2	Aalto	39.8	26.9	45.7	59.8
3	VideoLAB	39.1	27.7	44.1	60.6
This work:					
4	*run#1*	38.7	26.9	**45.9**	58.7
–	*run#2*	39.2	25.4	41.0	58.3
–	*run#6*	39.3	24.7	39.6	57.6

Table 3: Examples showing good (top four rows) and bad (bottom two rows) video tags and captions.

Test video	Tagging	Captioning
	fashion, walking, model	**baseline:** a woman is dancing **run2:** models are walking down the runway **run1:** models are walking in a fashion show
	woman, kitchen, cooking	**baseline:** a woman is cooking **run2:** a woman is cooking **run1:** a woman is cooking in a kitchen
	cat, baby, dog	**baseline:** a man and a woman are eating food **run2:** a baby is playing with a cat **run1:** a baby is playing with a cat
	horse, racing, man	**baseline:** gameplay footage of someone playing a game **run2:** people are riding horses **run1:** people are racing horses in a race
	water, people, walking	**baseline:** person is recording the beautiful waterfalls **run2:** a person is explaining something **run1:** there is a man is walking in the water
	girl, singing, show	**baseline:** a woman is singing **run2:** a boy is singing **run1:** a girl is singing

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities and the Research Funds of Renmin University of China (No. 14XNLQ01), Natural Science Foundation of China (No. 61303184), and Social Science Foundation of China (No. 12&ZD141).

5. REFERENCES

[1] J. Dong, X. Li, and C. Snoek. Word2VisualVec: Cross-media retrieval by visual feature prediction. *CoRR*, abs/1604.06838, 2016.

[2] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, et al. From captions to visual concepts and back. In *CVPR*, 2015.

[3] Y.-G. Jiang, Z. Wu, J. Wang, X. Xue, and S.-F. Chang. Exploiting feature and class relationships in video categorization with regularized deep neural networks. *CoRR*, abs/1502.07209, 2015.

[4] X. Li and Q. Jin. Improving image captioning by

concept-based sentence reranking. In *PCM*, 2016.

[5] X. Li, C. Snoek, and M. Worring. Learning social tag relevance by neighbor voting. *IEEE Trans. Multimedia*, 11(7):1310–1322, 2009.

[6] X. Li, T. Uricchio, L. Ballan, M. Bertini, C. Snoek, and A. D. Bimbo. Socializing the semantic gap: A comparative survey on image tag assignment, refinement, and retrieval. *ACM Computing Surveys*, 49(1):14:1–14:39, 2016.

[7] P. Mettes, D. Koelma, and C. Snoek. The ImageNet shuffle: Reorganized pre-training for video event detection. In *ICMR*, 2016.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. In *ICLR*, 2013.

[9] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui. Jointly modeling embedding and translation to bridge video and language. In *CVPR*, 2016.

[10] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[11] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015.

[12] R. Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015.

[13] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence-video to text. In *ICCV*, 2015.

[14] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015.

[15] J. Xu, T. Mei, T. Yao, and Y. Rui. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, 2016.

[16] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu. Video paragraph captioning using hierarchical recurrent neural networks. In *CVPR*, 2016.