EURECOM participation in TrecVid VTT 2018

Danny Francis, Benoit Huet, Bernard Merialdo

October 30, 2018

Abstract

This paper describes the submissions of the EURECOM team to the TrecVid 2018 VTT task. We participated in the Sentence Matching subtask. Our approach is to project both descriptive texts and videos in the same vector space through a deep neural network, and to compare them using a cosine similarity. In particular, we compare several variants of sentence embeddings.

1 Introduction

EURECOM participated in the Sentence Matching subtask of the TrecVid 2018 [1] Video-to-Text (VTT) task for the first time. The approach we chose to follow was to improve the approach of the best team of 2017 [8]. The Sentence Matching subtask of the VTT task requires to link videos and sentences describing these videos. Testing data is composed of 1,000 videos, and five datasets of 1,000 sentences, each sentence corresponding to one video. For each video and for each dataset of sentences, teams are asked to rank sentences from the closest to the most dissimilar. Evaluation is performed on each sentence dataset using the Mean Inverse Rank measure.

2 Our Model

2.1 Definition of our model

As stated in Section 1, our model aims at improving the model of [8]. In [8], video embeddings are derived as follows:

- frames are extracted every 0.5 second for each video;
- features vectors $(v_1, ..., v_n)$ are derived from these frames using the penultimate layer of a ResNet-152 [11];
- these features vectors are then fed sequentially into a GRU [6], whose hidden states $(h_1, ..., h_n)$ are concatenated to corresponding features vectors, to obtain contextualized features vectors $(s_1, ..., s_n) = (v_1 \| h_1, ..., v_n \| h_n)$;
- these contextualized features vectors are combined through a soft attention mechanism to form a vector v, which is actually a weighted sum of $s_1, ..., s_n$;
- this vector v is then projected into a vector space after two fully-connected layers with ReLU activations, where each activation is preceded by a batch normalization.

In our model, the same process is applied for computing video embeddings. However, before feeding v into the two fully-connected layers, we concatenated it with a vector that we derived from the video using the last layer of an RGB-I3D [4]. Moreover, [8] used a ResNet-152 trained on ImageNet [7] whereas we used the ResNet-152 trained on ImageNet and finetuned on MSCOCO [13] proposed by [9].

In [8], text-embeddings are derived as follows:

- three text representations are derived (one using an average of Word2Vec [14] embeddings, a second one is a BoW representation and a third one is derived by taking the last hidden state of a GRU) and concatenated;

- the resulting vector is then fed into two consecutive fully-connected layers following the same process as for videos.

Regarding the text-embeddings part of our model, we tried to replace the GRU by GRCs [10] or a bidirectional GRU. GRCs are extensions of GRUs that we proposed in [10], where we showed that they could improve results of GRUs on multimodal matching tasks. Our model is summarized by Figure 1.

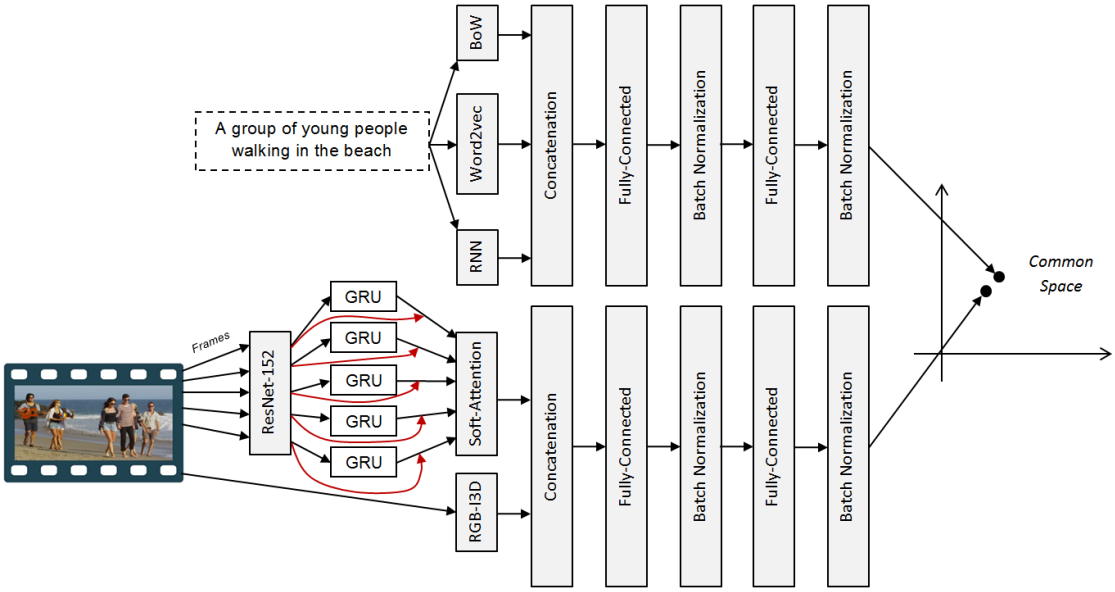

Figure 1: Our model. RNN can be a GRU, a GRC or a bidirectional GRU.

2.2 Training

We trained our model using a common hard-negative triplet ranking loss [9]. More formally, if v is the embedding computed for a video, s the embedding computed for a sentence corresponding to that video, then the loss $l(v, s)$ corresponding to the couple (v, s) is defined as follows:

$$l(v,s) = \max_{\overline{s} \neq s}(\max(0, \alpha - \cos(v,s) + \cos(v,\overline{s}))), \quad (1)$$

where α is a hyperparameter that we set to 0.2.
We used several datasets for training and validation:

- MSVD [5];

- MSR-VTT [16];

- TGIF [12] (for computer memory problems, we only used 60,000 sentence-video pairs from TGIF);

- TrecVid VTT 2016 test data [3];

- TrecVid VTT 2017 test data [2].

Table 1: Our results in terms of Mean Inverse Rank

Runs	Subset A	Subset B	Subset C	Subset D	Subset E
Run 1	0.194	0.190	0.194	0.193	0.199
Run 2	0.197	0.197	0.197	0.184	0.204
Run 3	0.202	0.209	0.206	0.186	0.212
Run 4	0.231	0.240	0.234	0.224	0.241

Our validation set was composed of 200 videos from TrecVid VTT 2016 test data and 200 videos from TrecVid VTT 2017 test data with corresponding sentences. Therefore, the validation set contained 400 different videos. We used MSVD, MSR-VTT and TGIF for training, for a total of 65,782 different videos with all corresponding sentences. Eventually, we used the remaining data from TrecVid VTT 2016 and TrecVid VTT 2017 to form a finetuning dataset of 3,088 videos.

We trained our models using the RMSProp method [15] with TensorFlow default parameters and gradient clipping between -5 and 5. We first trained our model on the training set, applying a learning rate of 0.00003 during 20 epochs (dividing the learning rate by two if validation loss did not improve during three consecutive epochs), with mini-batches of size 25. Then, we set the learning rate to 0.00002, and finetuned our model on the finetuning dataset during 60 epochs, dividing the learning rate by two if validation loss did not improve during three consecutive epochs.

3 Our runs

EURECOM submitted four different runs to the VTT Sentence Matching subtask. The runs are numbered from 1 to 4, with the expected best runs having the highest numbers. For each run and for each video, sentences are ranked by decreasing cosine similarity.

RUN 1 : We apply the model we described in Section 2. The RNN we used for computing sentence embeddings was a simple GRU.

RUN 2 : This run was similar to RUN 1, but we replaced the GRU by a GRC.

RUN 3 : In this run, the GRU of RUN 1 is replaced by a bidirectional GRU.

RUN 4 : This final run is a merge of previous runs. The merge is performed by summing the cosine similarities of the three previous runs, to obtain a new score for each sentence.

4 Results

We reported our results in Table 1. Our runs are ranked as we expected on subsets A, B, C and E. It is not the case for subset D, as the simple GRU obtained better results than the GRC and the bidirectional GRU.

In Figures 2-6, all results on different sentences subsets are presented. Our results are in red. As one can see, our ensemble method did better than other methods. Our future work will be dedicated to finding finer ensemble methods to see how results can be further improved.

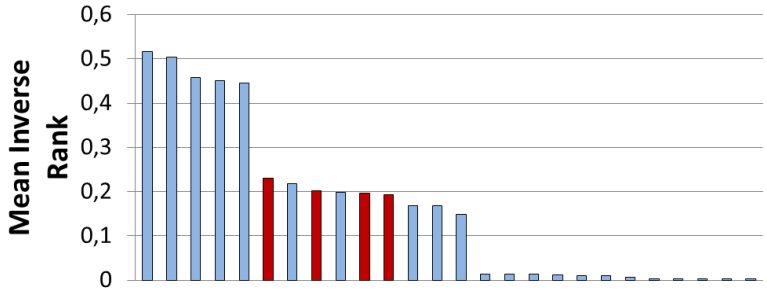

Figure 2: Results on subset A

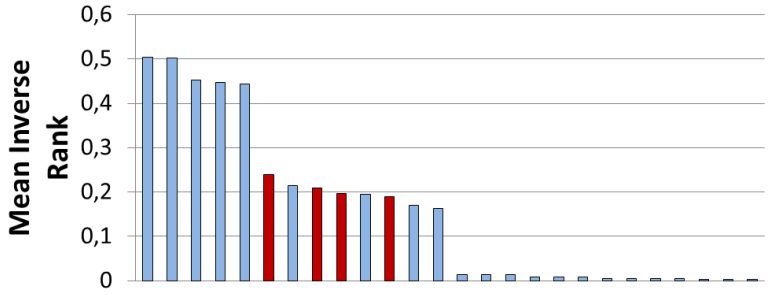

Figure 3: Results on subset B

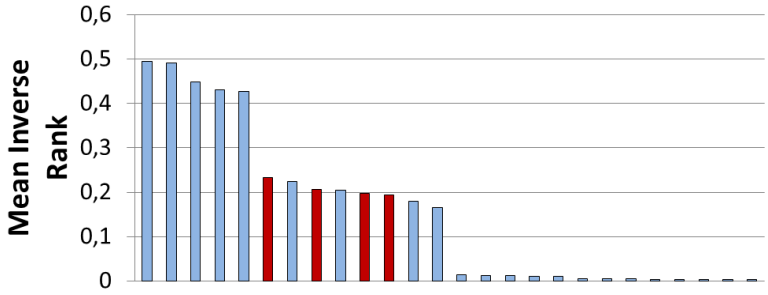

Figure 4: Results on subset C

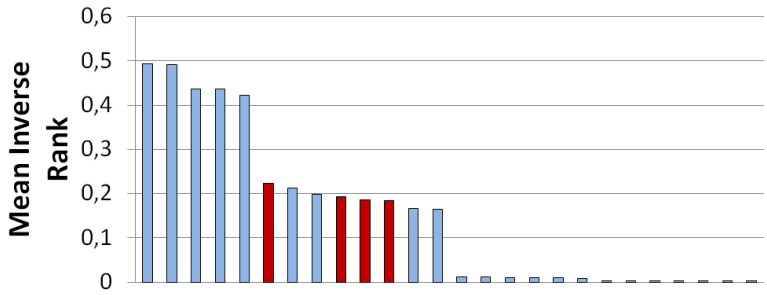

Figure 5: Results on subset D

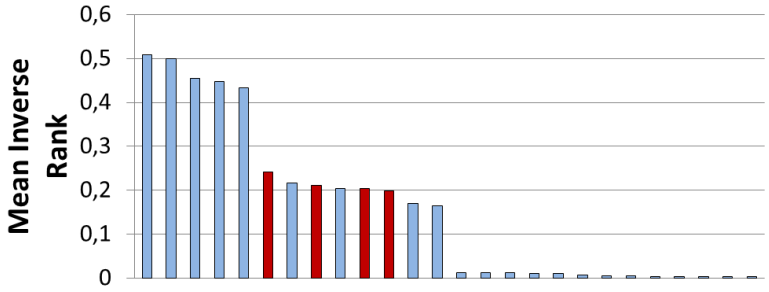

Figure 6: Results on subset E

Acknowledgments

This work has been funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 780069 Methods for Managing Audiovisual Data: Combining Automatic Efficiency with Human Accuracy (MeMAD). We gratefully acknowledge the support of NVIDIA Corporation with the donation of one of the Titan Xp GPU used for this research

References

[1] Awad, G., Butt, A., Curtis, K., Lee, Y., Fiscus, J., Godil, A., ... & Blasi, S. (2018). TRECVID 2018: Benchmarking Video Activity Detection, Video Captioning and Matching, Video Storytelling Linking and Video Search. In Proceedings of TRECVID 2018.

[2] Awad, G., Butt, A., Fiscus, J., Joy, D., Delgado, A., Michel, M., ... & Eskevich, M. (2017, November). Trecvid 2017: Evaluating ad-hoc and instance video search, events detection, video captioning and hyperlinking. In Proceedings of TRECVID (Vol. 2017).

[3] Awad, G., Fiscus, J., Michel, M., Joy, D., Kraaij, W., Smeaton, A. F., ... & Ordelman, R. (2016). TRECVID 2016. Evaluating Video Search, Video Event Detection, Localization and Hyperlinking.

[4] Carreira, J., & Zisserman, A. (2017, July). Quo vadis, action recognition? a new model and the kinetics dataset. In Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on (pp. 4724-4733). IEEE.

[5] Chen, D. L., & Dolan, W. B. (2011, June). Collecting highly parallel data for paraphrase evaluation. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1 (pp. 190-200). Association for Computational Linguistics.

[6] Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 1724-1734).

[7] Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on (pp. 248-255). Ieee.

[8] Dong, J., Huang, S., Xu, D., & Tao, D. DL-61-86 at TRECVID 2017: Video-to-Text Description.

[9] Faghri, F., Fleet, D. J., Kiros, J. R., & Fidler, S. (2017). Vse++: Improved visual-semantic embeddings. arXiv preprint arXiv:1707.05612, 2(7), 8.

[10] Francis, D., Huet, B., & Merialdo, B. (2019, January). Gated Recurrent Capsules for Visual Word Embeddings. In International Conference on Multimedia Modeling. Springer, Cham.

[11] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 770-778).

[12] Li, Y., Song, Y., Cao, L., Tetreault, J., Goldberg, L., Jaimes, A., & Luo, J. (2016). Tgif: A new dataset and benchmark on animated gif description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4641-4650).

[13] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. In European conference on computer vision (pp. 740-755). Springer, Cham.

[14] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).

[15] Tieleman, T., & Hinton, G. (2012). Lecture 6.5 — RmsProp: Divide the gradient by a running average of its recent magnitude. In COURSERA: Neural Networks for Machine Learning.

[16] Xu, J., Mei, T., Yao, T., & Rui, Y. (2016). Msr-vtt: A large video description dataset for bridging video and language. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 5288-5296).