INSIGHT@DCU TRECVID 2019: VIDEO TO TEXT

Luis Lebron
Insight@DCU
Dublin City University, Whitehall, Dublin
luis.lebroncasas@insight-centre.org

Kevin McGuinness
Insight@DCU
Dublin City University, Whitehall, Dublin
kevin.mcguinness@insight-centre.org

Noel E. O'Connor
Insight@DCU
Dublin City University, Whitehall, Dublin
noel.oconnor@insight-centre.org

ABSTRACT

In this paper we describe the approach we developed for the TRECVID video to text task, specifically the free-text generation sub-task. This sub-task consists of generating a textual description using only the information that can be extracted from the videos. We tackle the problem using a commonly used BLSTM network with an alternate enhance mechanism. To improve the model we study the effect of using different datasets and features. One of the main problems of the video captioning challenge is the size of the vocabulary, which adds another level of complexity, as the model needs to produce a rich vocabulary without previous knowledge of the scene. Therefore, we also discuss the use of an image captioning module to guide the initial text obtained from the video.

***K*eywords** Video captioning · C3D · Attention BLSTM · Image captioning

1 Introduction

Video captioning, which consists of extracting a sentence or paragraph to describe a video, is a very challenging task in the computer vision field that has witnessed renewed interest in the community since the arrival of deep learning. From this new set of techniques, it has become possible to generate more complex text.

Although the most used techniques nowadays are from deep learning, the task of video captioning started with more traditional methods.The first works in the topic were based on templates where the aim was to predict a triplet of "words' representing the subject, verb and the object [1]. These methods are very limited in the complexity of the expression and the detail that they can capture. The next round of algorithms used embedding spaces. For instance [2] takes images from internet pairs with sentences that are embedded into an embedding space that can then be subsequently used to obtain the sentences associated with the original images. Again these methods can only describe a limited amount of scenes and cannot build new sentences to describe new videos. Deep learning provided a new range of techniques to solve these problems. RNNs have been proved to be useful in tasks related to text and sequences, such as text translation or handwriting recognition. They have the capacity to generate free text without copying the exact sentences from the training datasets. Initial approaches of this kind use the encoder-decoder structure to generate single sentences which describe the whole video. Datasets like the ones generated for TRECVID's video-to-text task [3] have helped to provide enough data to train these models, which are well known for requiring a huge amount of training data. With the emergence of more complex datasets like ActivityNet, these methods became the start of a new trend of algorithms that focus on not only generating a single sentence but on producing multiple sentences to explain multiple clips in longer videos.

We decided to explore the encoder-decoder strategy for TRECVID-VTT. The model used is based on [4], which uses LSTMs and an attention model to encode the frames and decode the textual description. From there we explore the use of different techniques to improve the baseline results. We replace the BLSTM with the popular Transformer model also we try to increase the amount of training data with an additional dataset. Our main idea was to include text as input to provide a base description of the model.

The structure of this paper is the following. Section 2 describe the basic model and the changes introduced. Section 3 presents the datasets used. Section 4 concludes with the results and the policy used to train the models.

2 Model

This section presents the baseline model and the improvements that we explored. As mentioned above we use as a baseline the work of [4].

The model consists of a BLSTM for encoding the input features. We use the C3D [5] as input to this BLSTM which are well known for capturing the temporal information of a video. This is followed by soft attention, which is fed again to a final LSTM. The sentences are then produced from a softmax function on top of the last LSTM. A beam-search [6] method is used to find the sentences with the highest probability.

A variation we explore is to try to predict a caption from the middle frame and use it as an additional input for the model. The intuition behind this idea is to obtain an initial sentence from which the model can generate a more complex one.

To do so we use the model from [7] to generate this caption and then this caption is embedded to a words spaces. This goes through an LSTM to encode the sentences and it is appended to the input vector of the soft-attention model. One noteworthy aspect is that we share the same embedding between the input and the output words to help the model identify the same words.

Let h_t^{text} be the output of the LSTM that encodes the text input, such as:

$$h_t^{\text{word}} = f(h_{t-1}^{\text{word}}, x_{t-1}), \text{for} \quad t = 1...N, \tag{2.1}$$

where N is the length of the sentences, x_t is the t word of the sentences embed in the Glove's spaces. Then we can define all the sentences as:

$$\mathbf{h}^{\text{sentence}} = [h_1^{\text{word}}, h_2^{\text{word}}, \ldots, h_N^{\text{word}}]^\top. \tag{2.2}$$

Then the new attention became:

$$\alpha_i^j = \mathbf{v}^T \tanh\left(\mathbf{W}_1 \mathbf{h}^{att,j} + \mathbf{W}_2(\mathbf{z}_i \parallel \mathbf{h}^{\text{sentence}})\right), \tag{2.3}$$

where $\mathbf{v} \in \mathbb{R}^V$, $\mathbf{W}_1 \in \mathbb{R}^{V \times N}$ and $\mathbf{W}_2 \in \mathbb{R}^{V \times T}$ are learnable parameters of the model and $\tanh$ function operates element-wise. $\mathbf{z}_i$ is the i vector by the image encoder. Let $h_i^{att,j}$, $j = 1, \ldots, M$, denote the i-th $LSTM^{att}$ output component at time t and $\mathbf{h}^{att,j} = [h_1^{att,j}, h_2^{att,j}, \ldots, h_N^{att,j}]^\top \in \mathbb{R}^N$.

Attention mask coefficients β_i^j are computed as a softmax function of α_j^i, namely,

$$\beta_i^j = \frac{\exp(\alpha_i^j)}{\sum_{j=1}^M \exp(\alpha_i^j)}, \qquad \sum_{j=1}^M \beta_i^j = 1. \tag{2.4}$$

The attention vector $\mathbf{o}_i \in \mathbb{R}^M$ is computed as a weighted sum of the $LSTM^{att}$ output vectors $\mathbf{h}_i^{att} = [h_i^{att,1}, h_i^{att,2}, \ldots, h_i^{att,M}]^\top \in \mathbb{R}^M$, that is,

$$\mathbf{o}_i = f_{att}\left(\mathcal{H}^{att}, \mathbf{z}_i \parallel \mathbf{h}^{\text{sentence}}\right) = \sum_{j=1}^M \beta_i^j \mathbf{h}_t^{att}, \tag{2.5}$$

where $\mathcal{H}^{att} = \{\mathbf{h}_i^{att}\}_{i=1}^N$.

We investigate improving this technique by replacing the LSTM in the encode-decode architecture with two Transformers [8]. These are well-known for improving results and better at capturing the temporal dependency between frames, however they are also very demanding in terms of the resources needed for training.

3 Dataset

The main dataset used in our experiments is the TRECVID-VTT dataset. This year we used the annotations from the previous years (2016, 2017 and 2018) and the videos from 2019 without the sentences. In the three previous years, the datasets follow the same pattern: the videos come from Vine and have a length of no more than 10 seconds. Each year they annotated around 2000 videos. This year the only difference is that the videos come from Vine and Flickr but the

other characteristics are more or less the same. There are five or fewer annotations per video and they try to reflect the *"who, what, where and when"* as depicted in the video in a short sentence.

TGIF [9] consists of 100k animated GIFs with 120k sentences describing their content. The GIFs comes from Tumblr from May to June of 2015. On average each GIF last 3.65 seconds and contains 40.62 frames. They provided one sentence per clip in the training and validation sets and three from the test set.

4 Experimental Results

This section presents an analysis of the result of our different variations.

We report three of the widely use metrics: BLEU,CIDEr and METEOR. BLEU computes the geometric mean of n-gram matching words count between the references and the candidate sentence. In our case, we report the BLEU-4. The main aim of CIDEr is to capture the consensus between the annotations and the reference sentence. Finally, METEOR computes the mean of weighted unigrams' precision and recall. It also includes functions to evaluate stemming and synonyms.

We train using stochastic gradient decent (SGD) with ADAM optimizer [10], and use beam search of 6. The C3D model that we use to extract the deep features for the input was pretrained in the sport 1M dataset.

During the first phase where we train the models and compare them, we split the datasets in the following way: TRECVID-VTT 2016 and a 50% of TRECVID-VTT 2017 are used for train, then the rest of TRECVID-VTT 2017 is used for validation and finally TRECVID-VTT 2018. When using the TGIF we use the whole dataset for training.

Table 1. Base model and transform model trained without TGIF dataset.

	BLEU-4	CIDEr	METEOR
IVTT-BLSTM	0.0014	0.0730	0.1257
IVTT-Transfrom	0.0010	0.0660	0.1160

For the first round of experiments, we don't use the TGIF dataset only the data from TRECVID. However, this does not provide enough data to train the models. In table 1 we can see the baseline results for the baseline model and the variation with the transform. IVTT-Transform is an expensive model to train and the result are not as good as expected at first. Taking as reference other works it seems that these initial results can be improved with more data, but even adding the TGIF dataset will not be sufficient. As such, we decided to discard further experiments with this model and focused on other variation of the baseline.

Table 2. Base model, image caption model and model with image caption text as input model trained with TGIF dataset.

	BLEU-4	CIDEr	Meteor
Image captioning	0.0025	0.0420	0.1111
IVTT-BLSTM	0.0020	0.1012	0.1333
IVTT-BLSTM-Text	0.0013	0.1090	0.1358

In the next experiment, we included the TGIF dataset to increase the amount of data. We also explored the addition of the caption extracted from the Image Captioning module into the model as input. The results can be found in 2. The first thing to notice is that, as expected, more data improves the baseline. Apart from this we also can see that the base Image captioning model performs well even if we do not retrain it to adapt to the videos and sentences of the dataset. However, if we use these sentences in our model the performance decreases, thus it can be argued that the additional inputs increase the feature spaces and the model already struggles to learn from the TRECVID data.

Table 3. Resutls on the TRECVID 2019.

	BLEU-4	CIDEr	METEOR
IVTT-BLSTM	0.0045	0.0350	0.1415

We only submitted our baseline run. This model has to be trained in the whole TRECVID-VTT 2016 and TRECVID-VTT 2018 plus 50% of TRECVID-VTT 2017. For validation, we use the rest of the TRECVID-VTT 2017 as in the other experiments and predict all the test dataset. Another noteworthy aspect is that we use Glove embedding for the output words. The result can be seen in table 3.

As discussed previously, the change to a transform architecture does not provide an improvement in our experiments. This is likely due to the limited number of samples and the fact that we couldn't do many experiments as this is a time-consuming model. With respect to the sentences from the image caption module, as standalone they get the best performance to compare to our model but they are limited by the pretrain vocabulary. Adding them to our model does not yield as good results as we expected. This is likely because the approach we follow to include them is quite simple. Concatenating the whole sentences to the image input vector proves to add too much redundancy. It also includes some level of confusion as it can describe objects that at the moment may not be in the scene.

5 Conclusions

We propose a model to learn captions from short videos. We train from multiple topics to make the model independent of the category of the video. We explore the use of image captioning models to establish a base sentence from the model. We also try to replace the LSTM architecture for a Transform one. None of these approaches results in an improvement on the metrics, which is why we decided to only submit the baseline with a new training methodology.

Following our initial experiments, our next work will focus on exploring new kinds of input data to help the model learn in small-medium datasets. We believe that adding information like the objects/actors in the scene and their relationship may help. Also, it is important to find ways to generate more data from this kind of datasets.

6 Acknowledgement

This research was supported by the Irish Research Council Enterprise Partnership Scheme together with United Technologies Research Center Ireland and Insight@DCU.

References

[1] Jesse Thomason, Subhashini Venugopalan, Sergio Guadarrama, Kate Saenko, and Raymond Mooney. Integrating language and vision to generate natural language descriptions of videos in the wild. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1218–1227, 2014.

[2] Mayu Otani, Yuta Nakashima, Esa Rahtu, Janne Heikkilä, and Naokazu Yokoya. Learning joint representations of videos and sentences with web image search. In *European Conference on Computer Vision*, pages 651–667. Springer, 2016.

[3] George Awad, Asad Butt, Keith Curtis, Yooyoung Lee, Jonathan Fiscus, Afzal Godil, Andrew Delgado, Alan F. Smeaton, Yvette Graham, Wessel Kraaij, and Georges Quénot. Trecvid 2019: An evaluation campaign to benchmark video activity detection, video captioning and matching, and video search & retrieval. In *Proceedings of TRECVID 2019*. NIST, USA, 2019.

[4] Álvaro Peris, Marc Bolaños, Petia Radeva, and Francisco Casacuberta. Video description using bidirectional recurrent neural networks. In *International Conference on Artificial Neural Networks*, pages 3–11. Springer, 2016.

[5] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. C3D: generic features for video analysis. *CoRR*, abs/1412.0767, 2014.

[6] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[7] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: Lessons learned from the 2015 mscoco image captioning challenge. *IEEE transactions on pattern analysis and machine intelligence*, 39(4):652–663, 2016.

[8] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[9] Yuncheng Li, Yale Song, Liangliang Cao, Joel Tetreault, Larry Goldberg, Alejandro Jaimes, and Jiebo Luo. Tgif: A new dataset and benchmark on animated gif description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4641–4650, 2016.

[10] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 12 2014.