Vision-Language Model for Soccer Broadcast Commentary Generation

Nathaniel Chien
Department of Computer Science
Stanford University
nchien2@stanford.edu

Jing Gong
NVIDIA
Stanford University
jing9@stanford.edu

Ruiming Wang
Apple Inc.
Stanford University
ruimingw@stanford.edu

Abstract

In our research on automatic sports captioning for soccer broadcasts, we employ two approaches utilizing an end-to-end Transformer model. Firstly, we use a pre-existing action-spotting embedding, bypassing the need for training an encoder. We then fine-tune this model for generating precise commentary-style captions. Secondly, we construct our own attention-based embedding from scratch, ensuring our model's focus on the most relevant parts of the input data. The encoder, either pre-trained or self-developed, is integrated with a new decoder for final output generation. In both cases, we maintain the same model architecture, highlighting the potential of machine learning in enriching the sports broadcasting industry with engaging, automated soccer commentary.

1 Introduction

Detecting events in video streams, then interpreting and finally relaying important information for human consumption is very useful, given a lot of events are captured in the form of video. In our project, we aim to leverage language model that can generate captions in the specific style of sports commentary. This would help pave the way for the use of automatic captioning in a wider set of applications. In addition, being able to generate rich semantic captioning is a crucial part of the sports industry especially for fan engagement and it is the perfect playground for improving commentary generation.

The vision language model takes in a video clip of a soccer broadcast as input and generates a caption commenting on interesting events in the video. Our goal is to generate captions that should fit the style of a sports commentator. In our research, we investigate specific pre-existing embedding[8] that has been generated from a model designed for a different generic purpose. We explored this sets of embedding to uncover its potential applications and the diverse patterns it can represent. Simultaneously, we intend to develop our own attention embedding from scratch, leveraging attention mechanisms for optimal focus on the most relevant parts of input data. A fresh embedding decoder will be trained in both cases, including the pre-existing embedding and our novel attention embedding. We also exlored the pre-trained models in video captioning area, and fine tunning pre-trained models seem to be generating more smooth languages in our case.

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

2 Related work

Models that are able to convert videos from text first began to gain popularity with a paper by Venugopalan, et al. [5] Building off of existing LSTMs that achieved state of the art performance on image captioning, they created a sequence-to-sequence model that was able to achieve good performance on a variety of tasks including movie description and annotation of Youtube clips. Video to text models have developed rapidly since then, with new architectures and training schemes. Lei, et al.[1] found that end-to-end models that employ sparse sampling of the input videos perform better than previous methods which trained dense feature extractors. Tang, et al. [4] utilized a more advanced transformer architecture as well as a video-text matching pre-training stage to achieve state of the art results. Most recently, Wang et al. [6] did away with any external object detectors or taggers, and simplified architecture to a single encoder and decoder. With the new architecture and more pre-training, their GIT model was able to surpass human performance.

While great strides have been made in the task of video understanding and captioning, there has been less research in the specific domain of video commentary. Most models are trained to generate short descriptive captions like classification tasks and the generation performance is not very promising as well. Yu et al. [7] addressed these limitations by collecting a dataset of sports videos, and training a model with a novel performance evaluation metric to encourage the desired long-form narrative captions. Qi et al. [2] built on this work by adding attentive motion representation and group relationship modeling to improve the model's understanding of sports.

Given the extensive body of work, including datasets like SoccerNet [3] and innovative approaches like the two-stage paradigm by Zhou et al. (2021) [8], we anticipate leveraging these findings for our research. The goal of our project is to develop a model capable of generating football commentary with an emphasis on understanding and improving audience engagement.

3 Dataset and Features

For our project, we use the Soccernet Dense Video Captioning Dataset[1], which consists of 471 videos of soccer broadcast games in 224p and 720p along with their corresponding captions. These captions contain both general commentary and descriptions of the actions depicted in the broadcast. The authors also included anonymized and identified versions of the caption to make the task generalizable (Figure 1) and ensure that the model is not required to learn the names of specific teams or players. In addition, 8576-dimensional feature embeddings have already been provided at 2fps. These embeddings were generated from an action-spotting model made by Baidu research[8] which was the winner of a 2021 challenge.

Figure 1: An example of a video frame and anonymized caption from the soccernet dataset.

This embedding is generated by concatenating the features generated by five different fine-tuned video action recognition models. Specifically, they combined results from a temporal pyramid network, a global temporal attention model, a video transformer network, an interaction-reduced channel-separated CNNs, and an I3D-Slow model. [8]

[1]https://www.soccer-net.org/publications

The original task outlined by the authors was for dense video captioning, in which the entire video would be passed as input, and the model would be tasked with generating captions over the entire length. For our project, we have reframed this task into clip captioning, where we will be passing in short clips centered around the captioning time, and generating individual captions for each clip. In processing our dataset to be passed into the model, we gathered clips by finding the frames at which commentary occurred and taking a 15-second window centered at the frame.

4 Methods and Novelty

Firstly we proposal a novel application of the GIT model architecture for the Soccernet datset: a vision encoder + text decoder as shown in Figure 3. For the visual encoder part, each frame has its own embedding generated from image encoder and then projected as visual features with linear layers before feeding into text decoder. For text decoders, Multi-head attentions perform the learning between visual and text captions. Detailed novelty are listed below in 4.1.1 as using action spotting embedding and 4.1.2 as leveraging pre-trained weights

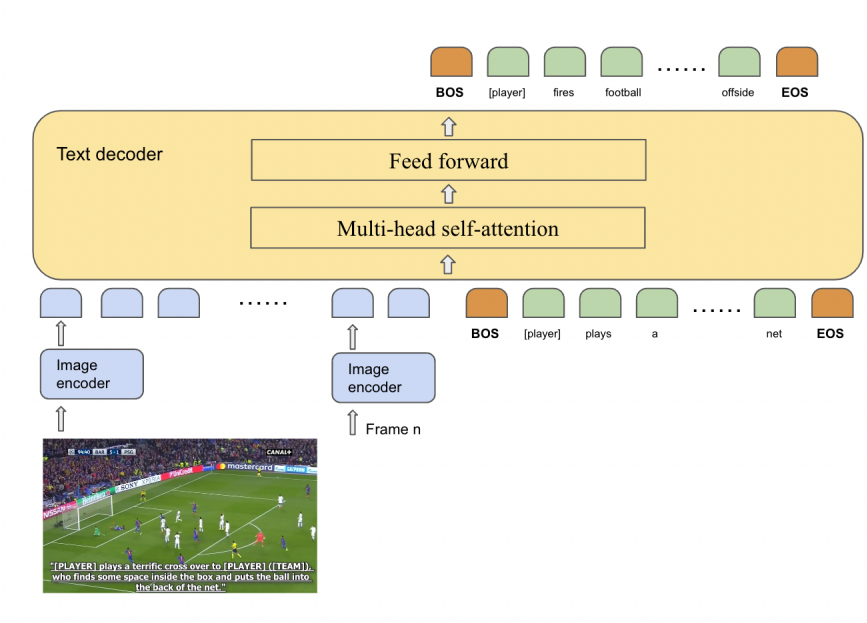


Figure 2: GIT Model Architecture with frame visual embeddings

4.1 Leverage Existing action spotting embeddings

For our first attempt, we decided to utilize the action-spotting embeddings that were provided with the dataset. This approach has the advantage of allowing us to avoid working directly with large video data, speeding up training and data processing. To improve upon the baseline model from the original SoccerNet-caption paper, we decided to use a more modern transformer decoder architecture rather than an LSTM. Specifically, we adapted the GIT architecture, a sequence to sequence architecture that takes in temporal embeddings and decodes it using multi-headed self-attention to generate an output sequence. In order to use the GIT architecture, we removed the image encoder and instead directly passed in the action-spotting embeddings as input. To convert the embeddings to the right dimensionality for the model, we added a fully connected and batchnorm layer.

For training, we use a causal training objective where the model's goal is to minimize the loss when performing next-token prediction over an input sequence. We constructed our training input sequences by concatenating the action-spotting embeddings with the embeddings of the tokenized caption. For each position in the sequence, we mask all future tokens and ask the model to predict the next token. Loss is evaluated as the cross entropy loss between the predicted sequence distribution and the input sequence shifted to the right by one. For the purposes of calculating loss, we mask out

the embeddings that are a part of the input, and only consider the predicted values for the ground-truth caption.

When generating, we pass in only the action-spotting embeddings, and sample from the predicted distribution at each timestep until and end of sequence token is generated.

4.2 Attention-based Video embedding

Our second attempt maintains the decoder architecture and training objective of the above model, but uses an attention-based video encoder instead of the provided action-spotting vectors. We also decided to utilized pretrained weights and fine-tune the model on our data, rather than training the decoder from scratch. In addition, since the action-spotting labels are a concatenation of 1-dimensional features that we think it potentially loses some spatial feature information about images. By training our own vision transformer based encoder we provide more information and guide to the decoder to better understand the video content.

4.3 Hyper-parameters

We ran embedding models over a grid search of the following hyper-parameter values: 1) learning rate: [.0001, .0003, .0005, .001], batch size: [16, 32, 64], dropout rate: [0.1, 0.2, 0.3] . The results presented below represent the highest performing hyper-parameter combination for this model.

For fine-tuning GIT models, we ran the model training with the following hyper-parameter values: 1) learning rate: [.0001, .0003, .0005, .001], dropout rate: [0.1, 0.2, 0.3]. Due to GPU memory restrictions and number of frames per example, when feeding the model with large batch size, we got GPU memory errors and had to compromise with smaller batch size. Regarding the window size(s) and framerate we use for each clip, we have to compromise that to window size=3 to make sure all frames can be fit into GPU memory.

5 Evaluation

5.1 Performance Comparison

We utilized the model introduced in the SoccerNet paper as our baseline for comparison purposes. This model employs feature embedding vectors and NetVLAD poolings to enhance its performance in soccer commentary generation. To establish a foundation, we initially ran this model with preliminary parameters (framerate=2, LR=0.001, batch size=256, etc), which served as our baseline model. To evaluate our models, we calculated the standard NLP metrics of Bleu, ROUGE, and METEOR. The results of these evaluations for the baseline model as well as our proposed models are listed in Table 1. Among all models, Leveraging pre-trained GIT model and fine tuning with SoccerNet data gives the best performance over all three metrics. For all experiments, we use framerate=2.

Model	Window(s)	METEOR	Bleu@4	ROUGE
Pretrained embedding+NetVLAD (baseline)	45	23.6	6.1	23.8
Pretrained embedding+Attention Decoder	15	10.9	1.1	15.3
Pretrained Video Model	3	6.3	0.0	12.8
Pretrained+Fine Tuned Video Model	3	**31.9**	**6.4**	**29.5**

Table 1: Captioning performance on test set

5.2 Caption Generation

Our initial model that utilized the provided action-spotting embeddings had relatively poor performance. Although it was able to train and minimize training and validation loss to a value of around 0.4, it did so without a proper understanding of language. By examining specific examples, we

were able to see that the model was simply predicting tokens that had a generally high likelihood of occuring in any position, such as 'and' and 'in'. While this is not ideal, it is to be expected given the difficulty of our task. Learning proper grammar and language structure is a challenge especially given the relatively low amount of text data in our task. Another potential issue we identified was that the action-spotting embeddings were 1-dimensional, and thus lacked the important spatial data that might be important for predicting

In order to solve the issues we observed with our embedding-only model, we moved on to using a pretrained GIT model. Initially, the model achieved worse performance than our embedding-only model. Although it output grammatically correct sentences, they often had little to do with the soccer games themselves. But after fine-tuning the model performance improved drastically, achieving a loss of around 0.02 and performing better than the baseline model on all metrics we evaluated. We can see from example generations (Table 2) that it is able to formulate grammatically correct sentences that are related to what is happening in the input clip.

6 Conclusion & Future Work

This work explored the viability of predicting captioning of multi-player sports game events. While there are some prior work on general video captioning, bit there is little predictive works particularly at the sports field, especially with the recent blossom of Transformers and LLMs(large language models). What work does exist relies on traditional RNN models such as LSTM which cannot natively ingest text sequences and produce text like real languages. Therefore, we investigated whether we could leverage transformer architectures to automate text predictions on videos. To do so, we explored two ways of vision modeling to help extract visual features for language models. 1) Using pre-trained action-spotting features based on CNN architecture that contains action information that can help target key actions when generating captions. 2) Fine-tune pre-trained Vision language captioning models that using attentions to learn visual embedding. We found that training decoder with action spotting embeddings from scratch is a little bit challenging to generate text like real smooth language. While fine-tuning pre-trained models which contains pretrained weights and visual attention encoders that can easily produce good languages that has grammar sense and it can be even more beneficial to produce domain professional contents by further fine-tuning.

There are several limitations of the current work which can be addressed in future work: 1) Pretrained models have proven to be advantageous at the language level. Therefore, we intend to dive deeper into the application of prompt-based tuning techniques to optimize our models for this targeted downstream tasks efficiently. 2) Actions are not always captured the generated texts from our model, and we believe combining domain specific knowledge such as action spot labels can provide more guidance to this generation task. We plan to add action spotting detection layers to this architecture as well. 3) While the SoccerNet dataset has been valuable for our research, its limited size, comprising only hundreds of games, poses certain limitations. To overcome this, we plan to expand our dataset by downloading additional soccer videos from YouTube. 4) Training models on video data in the form of frames can be computationally demanding, particularly in terms of GPU memory usage. To address this challenge, we will explore the utilization of GPUs with higher memory bandwidth, such as the A100, to accelerate the training feedback loop.

7 Contributions

Jing: Video dataset exploration and evaluation. Baseline model training, video loader and fine tuning pre-trained model.
Nathaniel: AWS EC2 management and data handling. Model training and development with embedding-only architecture. Ruiming: Background research, debug model. Local Data download and model training.

References

[1] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. *CoRR*, abs/2102.06183, 2021.

[2] Mengshi Qi, Yunhong Wang, Annan Li, and Jiebo Luo. Sports video captioning via attentive motion representation and group relationship modeling. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(8):2617–2633, Aug 2020.

[3] SoccerNet. Osoccernet.

[4] Mingkang Tang, Zhanyu Wang, Zhenhua Liu, Fengyun Rao, Dian Li, and Xiu Li. Clip4caption: CLIP for video caption. *CoRR*, abs/2110.06615, 2021.

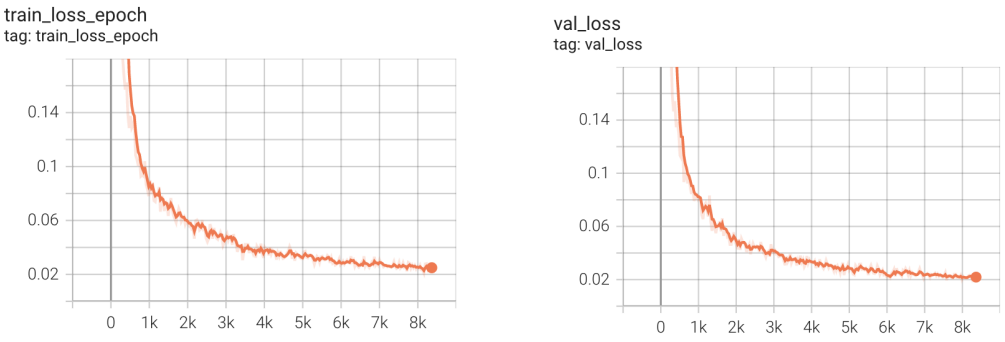
[5] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*, pages 4534–4542, 2015.

[6] Jianfeng Wang, Zhengyuan Yang, Xiaowei Hu, Linjie Li, Kevin Lin, Zhe Gan, Zicheng Liu, Ce Liu, and Lijuan Wang. Git: A generative image-to-text transformer for vision and language. *arXiv preprint arXiv:2205.14100*, 2022.

[7] Huanyu Yu, Shuo Cheng, Bingbing Ni, Minsi Wang, Jian Zhang, and Xiaokang Yang. Fine-grained video captioning for sports narrative. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6006–6015, 2018.

[8] Xin Zhou, Le Kang, Zhiyu Cheng, Bo He, and Jingyu Xin. Feature combination meets attention: Baidu soccer embeddings and transformer based temporal detection. *Baidu Research*, 2023.

8 Appendix



(a) train loss (b) validation loss

Figure 3: Fine tuned GIT Model Train/validation loss versus number of iterations

(a) Cross into the box (b) Referee blows a foul

Figure 4: Frames from the clips corresponding to the captions in Table 2

True Caption	[PLAYER] ([TEAM]) sends a cross into the box , but [PLAYER] comes off his line to gather the ball .
	'[PLAYER] ([TEAM]) makes a strong challenge and [REFEREE] blows for a foul .'
Pre-trained embedding+Attention Decoder	play – entertaining player and , area with injury this . pass at area teams wonderful defending at . with , with . . . . . at have the around and the the . . . low at . . . . . onto . . . .'
	play foul ! at in now . today by the . . . of their , , . and the [PLAYER] . . and the the . 's in in in in and in and . and . and possession long . . and the . and so in as
Pre-trained Video Model	a person is playing with a computer game and then a video of the game.
	a person is talking about the music.
Pre-trained+Fine Tuned Video Model	[player] ([team]) attempts to send over a cross in order to find one of his teammates, but an opposition defender averts the danger.
	the referee signals a substitution has been made. [player] is replaced by [player] ([team]).

Table 2: Example Generations with the ground-truth captions for our three models