Video Summarization through Combination of Audio & Frame interweaving and

Rafe Kruse, Raymond Yu, Jeffrey Liang, Richard Liu

CEN4914 Spring 2021
Department of CISE
University of Florida

Advisor: Dr. C. Silva, *email*: catiaspsilva@ece.ufl.edu
Department of ECE
University of Florida, Gainesville, FL 32611

Abstract

With an ever increasing amount of video content available online the need for effective content acquisition, sorting, and discovery becomes more important. The ability to accurately and efficiently summarize video content into a text format can facilitate users and other content discovery algorithms to more efficiently identify relevant content. This paper proposes and implements a pipeline for video text summarization that relies upon a model that utilizes frame captioning and abstract text summarization techniques on short form video content, with mixed results dependent upon video subject matter.

1. Introduction.

With a growing amount of video content available being able to leverage these videos is reliant upon users ability to gain access to relevant content. However, the interpretation of video content is a complicated problem both in terms of ability to capture relevant semantic meaning of videos and the amount of compute required for the massive amount of data that exists. Being able to effectively and efficiently summarize video content would open the doors for improving a users ability to identify relevant long form content or for automated systems that rely upon natural language processing (NLP) to sort video to more effectively categorize content.

Our pipeline relies upon well established abstract text summarization(ATS) and visual processing methods in order to make a cohesive video summarization model that can capture both audio and visual semantic meaning for more accurate summarization. Our model can be split in two two separate models that work in tandem. The first being an autoencoder-decoder image captioning model, and the second being an ATS text summarization model. The primary focus of this paper will be on the first image captioning model as this is the portion of the project I (Rafe Kruse) was responsible for. The pipeline also relies upon supplementary scripts and preprocessing operations, however, those implementations were the responsibility of another project collaborator and are out of scope for this paper.

1.1. Problem Domain.

Video summarization is a newer problem that is very similar to the more popular image recognition challenge. For our pipeline we opted to leave the summarization to the more established NLP ATS models that have already been proven within the field to be effective at capturing semantic meaning. However, these models are reliant upon text input so the focus of this paper is on obtaining meaningful and relevant information from the visuals of the videos that can be fed into the final ATS model. Our chosen method for going about this was to apply a frame captioning model at regular intervals to gain important visual and temporal information from images.

1.2. Literature Overview.

The problem of image captioning is very close to two other machine learning applications namely image segmentation[6] and object recognition[5,7]. Image captioning started out being approached using the same techniques as the aforementioned problems, namely applying convolutional neural networks (CNNs) [5]. CNNs have been the chosen solution for approaching many image based tasks for their innate ability to efficiently process images and extract spatially relevant information. Convolution operations are also suitable for being parallelized which makes them ideal for being computed on GPUs allowing for far faster training than their predecessors.

In recent years autoencoder-decoder architectures have been showing improved results over previous CNN methods[4]. These networks are constructed as feedforward non-recurrent networks very similarly to the original basic MLP networks[2]. The power of these networks comes from their central layer, or bottleneck layer, that is of a reduced dimensionality. These networks are trained complete, then utilized with the decoder portion, the layers after the bottleneck layer, removed. The model then is able to take input and convert it into a compressed representation.

The combination of autoencoders with a new model feature, attention have added to their improved labeling ability over previous methods[1,3]. Image based data are feature rich and very dense which makes the problem of captioning them difficult for traditional models, however, with the addition of attention autoencoders can more efficiently train with a greater accuracy[3]. Attention works by focusing the model on the relevant feature

vectors of an image so the model spends the majority of its compute learning from parts of images that carry the most meaning.

2. Technical Approach

The total pipeline for the project was separated into two distinct teams as previously alluded to: The text team and the video team. This paper's following section will be covering the approach of the video team's frame captioning model. However, the pipeline in its entirety can be seen below in figure 1 for reference as to how the two parts interact.

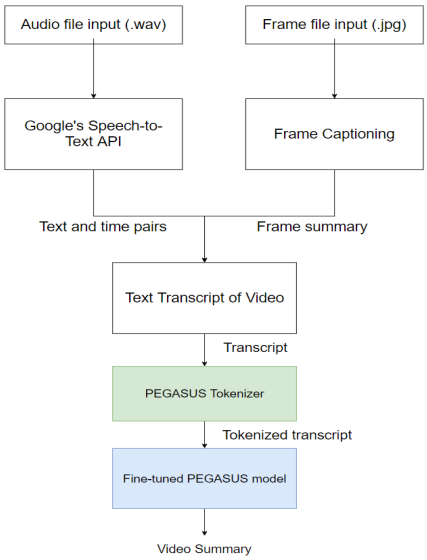

Figure 1. Full Pipeline for video text summarization model that takes in component .wav and .jpg files for a video and returns a text summary.

2.1. Data

The dataset chosen for training the frame captioning model within this paper was the MS-COCO (Common Objects in Context) dataset. This dataset consists of over 82,000 images with 5 separate caption annotations resulting in 410,000 labeled image caption pairs. This dataset was an ideal choice because of the size and the wide variance in images within the dataset that would help allow our model to generalize to video frames.

We chose to preprocess the image data using the InceptionV3 model[9]. The inceptionV3 is a very deep convolutional network for feature extraction trained on the Imagenet dataset for optimal frame feature compression. This model is what facilitates us

to convert the raw image data into a feature set that our encoder-decoder network can process. Finally the text of the captioning and our model outputs are tokenized with a vocabulary size of 5000 meaning that the text is converted into length 5000 vectors that our model can manipulate.

2.2. Frame Captioning Model

Attention is a model function that works by mapping queries to a set of key-value pairs. This function then outputs the sum of weighted compatibilities of the query and the various keys. In simpler terms this effectively identifies the most important parts of the image based upon the resulting caption and focuses the model to those portions of the image by weighing them more heavily. Attention can be implemented in two different ways additive and via dot product. For our model we chose to use an additive implementation known as Bahdanau attention for its Tensorflow support.

The bulk of our model is made up of a series of encoder decoder layers. The encoder is constructed as a set of identical layers with two components: the attention function mentioned previously, followed by a dense feed-forward neural network. Additionally the final output is stacked and normalized to avoid exploding gradient values before being passed into the decoder. The decoder follows a similar structure to the encoder, starting with an attention function. The resulting attention weights and context vectors are then combined with the encoder output after it has been embedded using the tokenizer. This resulting feature stack is finally fed through a dense fully connected feed forward neural network.

2.3. Loss and Optimization

For our frame model during training the predicted output of the decoder is compared to the dataset ground truth and a loss value is calculated using a variant of the well known cross-entropy loss function, sparse categorical cross entropy loss. Cross-entropy is a loss metric that quantifies the spread of two probability distributions and is the ideal metric for multi-class classification problems. Frame captioning falls under this category as the model works by predicting the next token of the sentence from a set of probabilities for each word vector within the embedded training vocabulary. Sparse categorical cross

entropy loss is a minor variant that is ideal for sparse data such as the embedding of frames where the data dimension is large, but a small amount of the data is relevant.

Encoder decoder networks are built upon the same basic perceptron layer network as MLPs which lends them to being trained as most other common artificial neural networks. All these networks are made up of independently connected layers of nodes that have to be optimized, thus a concise optimization problem cannot be written for these networks. Instead the backpropagation algorithm is carried out across the network making use of the above cross-entropy gradients in order to iteratively train the network over a series of epochs. For this project the well known ADAM[8] optimization algorithm was chosen for its proven ability to efficiently and effectively train a diverse set of model architectures.

2.4. Results

The results for our paper are presented using the BiLingual Evaluation Understudy (BLEU), the Metric for Evaluation of Translation with Explicit ORdering(METEOR) metric, and the Recall-Oriented Understudy for Gisting Evaluation(ROUGE). The BLEU and ROUGE metric has been implemented with different variants such as BLEU/ROUGE-1, the quantity of overlap of 1-grams, or BLEU/ROUGE-2, the quantity of overlap of 2-grams. For our frame model results we opted for the BLEU-1, BLEU-2, BLEU-3, and METEOR to give a wide spread of values to compare to similar literature. Our full pipeline video text summarizer is quantified using ROUGE-1, ROUGE-2, ROUGE-L and METEOR to give a wide spread of values to get a better idea of how our model operates in different contexts.

2.4.1. Frame Captioning

The frame captioning model in this paper was trained using a single NVIDIA 1080Ti GPU for 50 epochs with each 16 image batch taking approximately 0.26 seconds to evaluate. A common 80/10/10 training/validation/test split was applied and our model obtained the following results compared to similar papers in table 1. Two human readable samples are provided in figure 2 for reader reference.

Metric	This Paper	Xu et al., 2015	Vinyals et al., 2014

BLEU-1	49.4	71.8	66.6
BLEU-2	34.6	50.4	46.1
BLEU-3	19.1	35.7	32.9
METEOR	16.8	25.0	24.6

Table 1. Score comparison of our frame captioning model with similar encoder-decoder models within current literature.

P. A group sits together on the street at a table GT1. a big family seated on a table and having drinks GT2. A large group of people is setting around a table on the street. GT3. A group of people sitting and standing around a wooden table. GT4. A group of adults dine outdoors next to a busy street. GT5. a bunch of people sit around a table of food	P. A man in back on a surfboard on the ocean water GT1. A kid on a surfboard is giving it all he's got. GT2. A boy on a surfboard surfing the waves at the ocean. GT3. A young boy is surfing through the water. GT4. A young child is surfboarding in the ocean waves and having some fun. GT5. A young boy in the water on a surfboard

Figure 2. Sample of two frames with the predicted caption P, and the five provided ground truths GT1-5 from the MS-COCO dataset.

2.4.2. Video Summarization

Our complete video summarization model is an aggregate of the frame captioning model explained above and fellow team members ATS model. The below results in Table 2 are for our custom dataset and for this reason have no similar comparisons within literature, so we also opted to include ours and others results on the ActivityNet dataset. It is also important to note our model is unique in that it aims to summarize and similar papers focus on video captioning.

Metric	This Paper	Wang et al., 2018	Iashin et al., 2020
ROUGE-L	36.87/11.42	-/19.29	-/-

BLEU-3	7.42/1.99	-/4.41	-/2.6
BLEU-4	6.51/1.71	-/2.30	-/1.31
METEOR	14.86/4.33	-/9.65	-/7.31

Table 1. Score comparison of our video text summarization model on Custom dataset/ActivityNet with similar models within current literature.

3. Technical Challenges and Solutions

Supervised training models are reliant upon label data to be performant. This is where our team ran into our first challenge, despite there being a large amount of publicly available video data online. Our model requires pairs of videos with their corresponding text summaries, this form of labelled data is not prevalent and the majority of existing video summary datasets are sparse and often lack any form of audio or are very short form. This made finding usable data for training and validation a challenge for our team. In order to remedy this we chose to construct our own dataset as an aggregate of videos from the preexisting TVSum dataset[12] and different publicly available YouTube videos. We also within our team created summary labels for the videos within the dataset. This dataset allowed us to have a benchmark for model testing that was well fit for our pipeline. However, given the dataset was handmade it is quite small which limits its generalizability and there is bias because our team were responsible for creating labels.

Given that we are working with videos and our model operates on still frame captures our team had to decide on a frame rate at which to sample frames from the videos. This posed a problem because should the frame rate be too high there would likely be duplicate captions and the ATS model input would be heavily biased towards visual data drowning out any transcript information. If the sampling rate is set to low there is a greater risk for losing out on important visual information. In order to remedy this problem we put an emphasis on the ATS models ability to summarize with less of a focus on frequency of recurring information. This allows the frame sampling to remain high so as to not miss any visual information while mitigating the issue of too much repetitive data because the ATS model has been modified to handle this.

Finally our team had the challenge of how we would integrate the separate data produced by the frame model and the transcripts that were generated by the Google

speech-to-text API. Within the problem of text summarization a very important feature of data is temporal consistency of the features that are fed into the model for summarization. For our model it was important that the frame caption and transcript data lined up time wise while not interfering with the semantics of each form's text. We explored a few methods of integration and found that the ideal solution that produced the best results in our case was sentence wise splicing of frame captioning into the audio transcript.

4. Conclusions

With a combination of well developed frame captioning and ATS models we were able to develop a video summarization pipeline for short form video. However, deep learning models are entirely dependent upon having large well labelled dataset, something that is lacking for the task of video summarization. This discrepancy is what led to our team creating a custom labelled video dataset to validate our model on. This proved to be beneficial but wasn't sufficient for adequate training and as such our models performance was subpar when compared to human labels. In the future we would like to revisit the problem when more applicable datasets have been generated at which time we would likely approach the problem with a more integrated audio visual model that could make full use of the richer dataset. This newer pipeline would likely be built upon a deep reinforcement learning model that is dependent on very large amounts of data to leverage a reinforcement policy to converge to a more accurate summarizer.

5. Standards and Constraints

Standards: All development for this project was done using Python 3 and relevant publicly available NLP/ML libraries and datasets. Our team chose to utilize Jupyter Notebooks for prototyping and our final release version of software is a python library and set of scripts.

Constraints: Training the deep learning models requires a large amount of compute. In our case a Nvidia GTX 1080 TI and 16 Gigs of Ram over a time frame of around 80 hours was used for training. Our project could be reproduced with similar hardware or better given adequate training time.

6. Acknowledgements

Our team would like to recognize the advisor to our project Dr. Catia Silva who has been instrumental in our project's success. She has been a great resource for advice, assistance, and knowledge on the subject matter while also offering her time for managing weekly meetings to facilitate continued progress and development on our project.

7. References

[1] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., . . . Bengio, Y. (2016, April 19). Show, attend and tell: Neural image caption generation with visual attention. Retrieved January 14, 2021, from https://arxiv.org/abs/1502.03044v3

[2] Rosenblatt, Frank. x. Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books, Washington DC, 1961

[3] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., . . . Polosukhin, I. (2017, December 06). Attention is all you need. Retrieved March 14, 2021, from https://arxiv.org/abs/1706.03762

[4] Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015, April 20). Show and tell: A neural image caption generator. Retrieved February 4, 2021, from https://arxiv.org/abs/1411.4555v2

[5] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016, May 09). You only look once: Unified, real-time object detection. Retrieved March 3, 2021, from https://arxiv.org/abs/1506.02640

[6] Linda G. Shapiro and George C. Stockman (2001): “Computer Vision”, pp 279-325, New Jersey, Prentice-Hall, ISBN 0-13-03-0796-3

[7] Dasiopoulou, Stamatia, et al. “Knowledge-assisted semantic video object detection” IEEE Transactions on Circuits and Systems for Video Technology 15.10 (2005): 1210-1224

[8] Kingma, Diederik P. and Jimmy Ba. “Adam: A Method for Stochastic Optimization” Corr abs/1412.6980 (2015): n. pag.

[9] Szegedy, Christian & Vanhoucke, Vincent & Ioffe, Sergey & Shlens, Jon & Wojna, ZB. (2016). Rethinking the Inception Architecture for Computer Vision. 10.1109/CVPR.2016.308.

[10] Iashin, V. (2020, March 17). *Multi-modal Dense Video Captioning*. ArXiv.Org. https://arxiv.org/abs/2003.07758v2

[11] Wang, J. (2018, March 31). *Bidirectional Attentive Fusion with Context Gating for Dense Video Captioning*. ArXiv.Org. https://arxiv.org/abs/1804.00100v2

[12] Yale, S., Vallmitjana, J., Stent, A., & Jaimes, A. (2015). TVSum: Summarizing Web Videos Using Titles. Cv-foundation. Retrieved February 10, 2021, from https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Song_TVSum_Summarizing_Web_2015_CVPR_paper.pdf

8. Biography

Rafe Kruse is a Dr. Phillips high school graduate and soon to be graduate of the University of Florida in the Spring of 2021 with a degree in computer science and a minor in electrical engineering. He has previously interned for Lockheed Martin, Amazon AWS, and will be returning to Amazon AWS in the summer of 2021 for a second internship. After graduation he looks to pursue a career as a software developer with an emphasis on machine learning.

Appendix A – Abstract Text Summarization

Note to reader: This task was not done by this author, so it received little attention in the paper. In summary, the final model output is reliant upon a pre trained PEGASUS text summarization model that is responsible for generating the final video summary from the joined text input.

Appendix B – Audio to Text (Google's Speech to Text API)

Note to reader: This task was not done by this author, so it received little attention in the paper. In summary, google's public audio transcription model was integrated into the project pipeline as a means of getting a text transcript of audio converted from videos extracted .wav files.