Using Multimodal Video and Language Representation for Video Question Answering

Akshata Bhat
akshatabhat@cs.wisc.edu

Rohit Kumar Sharma
rsharma@cs.wisc.edu

Abstract

Video Question Answering is the task of answering questions given videos and natural language (subtitles or audio) as the context. In this task, the challenge is to incorporate information from both videos and language in a way that makes it easier for spatio-temporal reasoning which is important for video question answering. In this work, we explore a technique of multimodal representation learning to jointly learn representations using various video and language based features such as action recognition and BERT in the realm of video question answering.

1 Introduction

Visual Question Answering (VQA) is a well known Computer Vision problem. It is a problem of building a system that can answer open-ended natural language questions by just looking at the image. The questions imposed on the system could be something like the following: How many people are in the image? Who is wearing the glasses? What is the color of the child's hat? This task was initially considered ambitious. But with the advent of specialized datasets for the task such as VQA [1] and Microsoft COCO [11], and with the advancement in Deep Learning, the task achieved significant performance improvement (as of today, state of the art accuracy on VQA Challenge is 75.26%) over the recent years.

Compared to image based question answering, there has been much less work on video based question answering. Video question answering is more challenging because of the following reasons: challenges arising from individual video frame understanding (variations in viewpoints, illumination, occlusion, scale, background clutter to name a few), challenges arising from natural language data such as subtitles or audio (word ambiguity, unstructured text understanding, semantic meaning, rich representation learning, co-reference resolution, etc.,), aligning visual and language cues, unavailability of large scale video question answering datasets and complexity arising from the temporal dependency of question on various frames to arrive at an answer. The latter challenge of spatio-temporal reasoning (understanding relationship among different entities and their evolution in temporal domain) is applicable to the task of video question answering more than other video based computer vision tasks like activity recognition, activity localization, etc., where the scope of the activities is limited to a local temporal region. Effective multimodal representation learning to jointly learn representations using multiple modalities (here video and language) can help uncover some of the complications arising, and can be seen as a step towards true AI.

Building intelligent systems that solve visual tasks such as image captioning, video summarizing, object detection, action recognition, etc., can help provide assistance to the visually impaired people or in the domain of cognitive robotics by providing an overall understanding of the visual world. A system which can answer questions posed by a user based on the real visual world can greatly assist us in our day to day tasks. One example could be to quickly determine the temporal timestamp of a particular fact being discussed in a video. A video question answering system can thus make doing research faster by quickly getting answers about specific questions based on the video.

Humans are very good at understanding questions and inferring answers based on visual experiences and videos. Having such a capability in a Machine Learning model is now becoming close to reality with the introduction of specialized datasets for these tasks and more advanced deep learning techniques which can model the semantic understanding from these videos and their captions.

More recently, the availability of specialized datasets for video question answering such as TVQA [9] has led to an increased interest in the task of Video Question Answering. In this work, we explore some techniques of using joint model representation for video and language and use those representations to solve video question answering. Specifically, we use features for video frames from pre-trained models such as Faster R-CNN for visual concepts and action detection models. For language, we explore using BiLSTM model to get sentence-level features from word embeddings and using BERT model to obtain sentence embeddings. We combine the video and language features to build a classifier for answering the questions. In the end, we evaluate our approach on the TVQA dataset and discuss the results.

2 Related Work

There has been a lot of work in the past on visual question answering datasets and models [1, 8, 15, 13, 21, 23, 6, 17, 22]. Several datasets and models have been proposed for video question answering in the past [5, 19, 14]. But these datasets and models don't solve the actual multimodal scenario for QA, as they are either only vision-focused or language-focused.

In TVQA [9] they propose a large scale Video QA dataset where the questions are compositional in nature. They also propose a multi-stream end-to-end trainable neural network, using regional visual features, visual concept features and ImageNet features as video features, BiLSTM encoders for video and text, followed by joint modelling of context and query.

In the project, we aim to exploit the recent work of generating multimodal representations using both the videos and language [18] and apply the learned representations in the task of Video Question Answering.

BERT [2], a language representation model with an unsupervised pre-training objective, which can be fine tuned on several downstream tasks, obtains state-of-the-art results on several natural language understanding tasks. Since this model is pre-trained on huge corpus - BooksCorpus and English Wikipedia, it has a rich representation and understanding of language and can be used to encode the questions and answers in our task, to extract initial set of features for textual data.

Several models have been proposed to address Computer Vision and Natural Language tasks that incorporate BERT: VisualBERT [10], ViLBERT [12], and VideoBERT [18]. In VisualBERT, the transformer architecture on which BERT is based on is used to align parts of image and text with self-attention for addressing a variety of vision and language tasks. ViLBERT introduces a way to learn joint embeddings of image and language content using BERT's co-attention transformer layers. The learned embeddings are used in downstream tasks such as visual question answering (image based), caption-based image retrieval, etc. VideoBERT also proposes a joint visual-linguistic model to learn high level semantic representations in a self-supervised manner using unlabeled data from sources such as YouTube, and use those representations in action classification and video captioning tasks. VisualBERT and ViLBert only address image based downstream tasks (including question answering), whereas VideoBERT addresses video based downstream tasks except video based visual question answering.

3 Methodology

According to our initial plan, we wanted to leverage the VideoBERT model to obtain the multimodal embeddings for our dataset and use these embeddings in the downstream task of Video Question Answering. But due to (i) unavailabilty of pretrained model and unavailabilty of implementation for VideoBERT, (ii) resource limitation and engineering challenges in implementing a similar model from scratch, we decided to instead explore pre-trained video action recognition models.

The high-level architecture of our model which is an extension of the approach provided in TVQA[1] is shown in Figure 1. Below, we describe some key points:

- The model consists of multiple streams - one stream per context (subtitles, visual concept features and video action features)
- To encode textual data, we use Glove Embeddings (300d). Questions and answers are encoded using BiLSTM and the hidden states are stacked to obtain the feature representation.
- Subtitles contain multiple sentences, hence these are flattend into a long sentence. The sentence is encoded using BiLSTM and the hidden states are stacked to obtain the feature representation.
- Visual concept features are extracted using Faster R-CNN model [16]. Objects are detected for each frame and unique objects are extracted across all these frames for the video clip.
- We use video action recognition features from R(2+1)D model [20] and something-something baseline model [4].
- Each of the above given contextual inputs are jointly modeled with question-answer pair to obtain context-aware query (video-aware question representation and video-aware answer representation) using a Bidirectional Attention Flow model, followed by fusion of question-answer using element-wise dot product, temporal max-pooling and fully connected layer to obtain scores for each answer.
- The above mentioned steps are performed separately for each context (input stream) and the scores from these streams are added to obtain a final score.

[1] https://github.com/jayleicn/TVQA

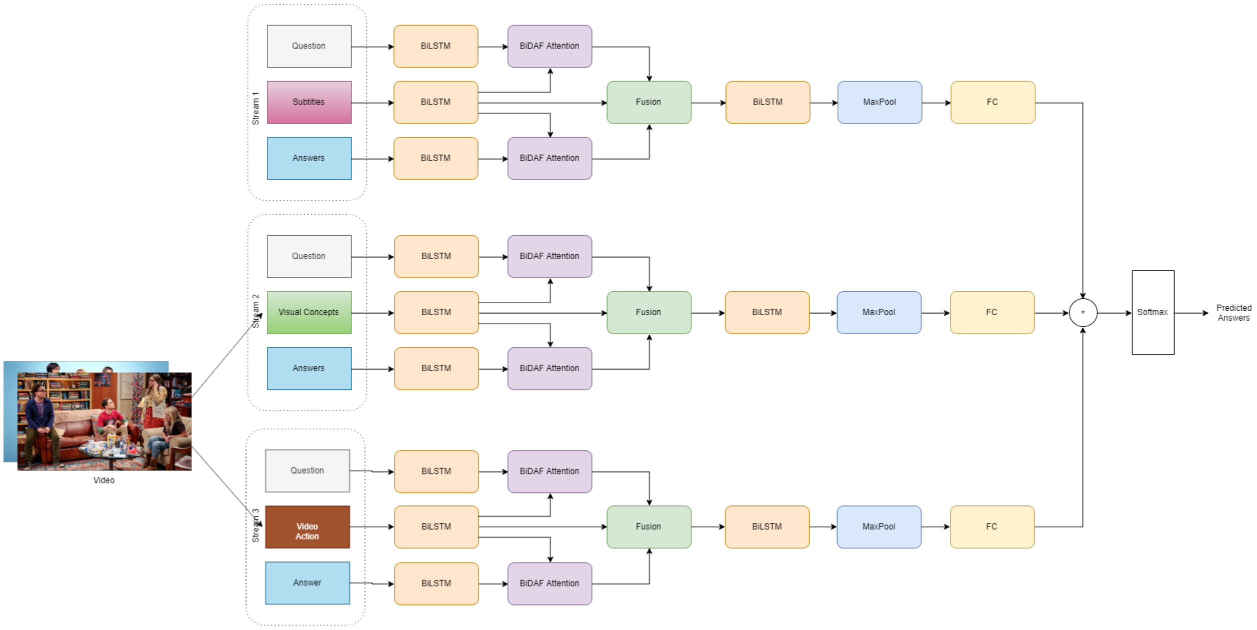

Figure 1: Model Architecture

4 Experiments

4.1 Dataset

We use TVQA [9] dataset for this project. It is a large scale localised, compositional video QA dataset based on 6 TV shows. It consists of 152.5K QA pairs from 21.8K video clips (60-90 seconds per clip), spanning over 460 hours of video. The QAs are multiple choice questions with 5 candidate answers, of which only one is correct. The dataset also contains dialogue (character name + subtitles) for each video clip. The distribution of questions and answers in the dataset is shown in the Figure 2.

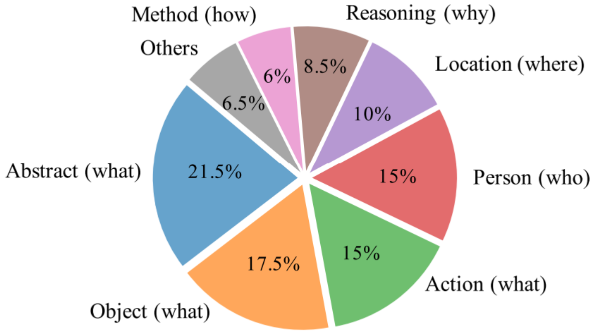

Figure 2: TVQA question and answer type distribution

4.2 Implementation Details

We implemented the model on top of the TVQA repository[2] using PyTorch framework. For training the model, we deployed a VM on Google Cloud Platform with 2 vCPUs, 13 GB memory and 1 12 GB NVIDIA Tesla K80 GPU. Later, when we tried incorporating BERT embeddings for the language objects like subtitles, questions and answers, the Tesla K80 GPU was running out of memory. Hence, we deployed another VM with 1 32 GB NVIDIA V100 GPU to run the experiments with BERT features.

For video features, TVQA uses Faster R-CNN model pre-trained on Visual Genome [7], to detect objects. For each clip, GloVe word embeddings for top-k detections are used as visual concept features.

To generate video action recognition features for video clips, we used the R(2+1)D model pre-trained on large scale 65M Instagram videos [3] from Facebook VMZ repository[3]. We used the PyTorch version of the model[4]. We also experimented with human-object interaction features. We used the baseline model[5] pre-trained on 20BN dataset to generate the video clip features. Because of the large size of TVQA dataset, pre-processing the video clips and generating the action recognition features was taking a long time. In the best interest of time, we decided to only use one TV series (BBT) from the

[2] https://github.com/akshatabhat/TVQA

[3] https://github.com/facebookresearch/VMZ

[4] https://github.com/akshatabhat/ig65m-pytorch

[5] https://github.com/Rohit–Sharma/smth-smth-v2-baseline-with-models

Model	Streams	Accuracy
Baseline	S	63.15%
	S + V_C	65.26%
Our Model	S + V_R	63.71%
	S + V_S	64.47%
	S + V_C + V_R	65.23%
	S + V_C + V_S	**65.36%**
	S + V_C + V_S + V_R	**65.47%**

Table 1: Performance of all the methods. S: Subtitle, V_C: Visual Concept features, V_R: R(2+1)D features, V_S: Something-something features. The models performing better than the baseline are shown in boldface.

dataset for the rest of the experiments. Extraction of R(2+1)D features took about 18 hrs and 20BN feature extraction took around 12 hrs for the BBT frames.

For generating BERT embeddings for questions, answers and subtitles, we used the pre-trained BERT models implemented in PyTorch framework in Huggingface Transformers repository[6].

We used 80%-10%-10% train, dev and test split for training the model. The test split we used is the public test dataset provided in the TVQA dataset. We train the model for 100 epochs with early-stopping. We have not modified the hyper-parameters provided in the repository as the model performed the best with them.

4.3 Results

In this section we present the experimental results we obtained. We experimented with various combinations of features as streams for question answering. The results are shown in Table 1.

In Figures 3 we plot training and validation loss achieved by the model while training the model for all combinations of experiments. The training loss keeps increasing indicating that the model learns during training. The validation loss starts increasing after about 2.5k steps indicating that the model starts overfitting. The minimum validation loss is achieved by the model with all streams combinations as shown (subtitles, visual concept features, R(2+1)D features, and something-something features). We are still in the process of training with BERT embeddings for language components and hence do not present the results for the same.

The baseline model with just visual concept features performs better than the model with R(2+1)D features. This is because of the following reasons: (i) the proportion of questions that rely on actions is fewer than the proportion of questions that rely on objects (See Figure 2), (ii) there might be a domain mismatch in the Kinetics dataset and the TVQA dataset, (iii) some complex questions might rely on actions across the temporal scope of the video clip. But R(2+1)D model uses only temporally local scopes to generate features instead of looking at the larger scope.

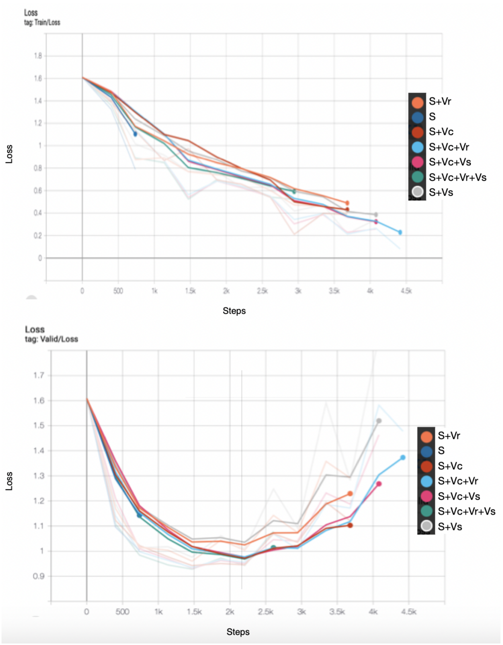

Figure 3: Training and Validation Loss

Our model with visual concept features and Something-something features as streams performs better than the baseline model (shown in boldface). This indicates that the human-object interaction video features are more relevant to the BBT dataset. Intuitively, the number of questions that involve human-object interactions are more common than the questions that rely on other actions.

5 Conclusion

We present a technique for learning joint model representations for video and language components for the downstream task of video question answering. We explored a multi-stream architecture and used action recognition video features (both generic actions and human-object interactions) as visual concepts.

For future work, we wish to explore the usage of sophisticated techniques presented in ViLBERT and VisualBERT to perform contextual matching and jointly learn visual and language features. For obtaining subtitle-aware question and answer representation, we propose using BERT as an alternative to BiDAF model.

[6] https://github.com/huggingface/transformers

References

[1] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. In *International Conference on Computer Vision (ICCV)*, 2015.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

[3] D. Ghadiyaram, M. Feiszli, D. Tran, X. Yan, H. Wang, and D. Mahajan. Large-scale weakly-supervised pre-training for video action recognition, 2019.

[4] R. Goyal, S. E. Kahou, V. Michalski, J. Materzyńska, S. Westphal, H. Kim, V. Haenel, I. Fruend, P. Yianilos, M. Mueller-Freitag, F. Hoppe, C. Thurau, I. Bax, and R. Memisevic. The ”something something” video database for learning and evaluating visual common sense, 2017.

[5] Y. Jang, Y. Song, Y. Yu, Y. Kim, and G. Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering, 2017.

[6] J.-H. Kim, S.-W. Lee, D.-H. Kwak, M.-O. Heo, J. Kim, J.-W. Ha, and B.-T. Zhang. Multimodal residual learning for visual qa, 2016.

[7] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. C. Niebles. Dense-captioning events in videos, 2017.

[8] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. S. Bernstein, and F.-F. Li. Visual genome: Connecting language and vision using crowdsourced dense image annotations, 2016.

[9] J. Lei, L. Yu, M. Bansal, and T. L. Berg. Tvqa: Localized, compositional video question answering, 2018.

[10] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang. VisualBERT: A Simple and Performant Baseline for Vision and Language. aug 2019.

[11] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár. Microsoft coco: Common objects in context, 2014.

[12] J. Lu, D. Batra, D. Parikh, and S. Lee. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. aug 2019.

[13] J. Lu, J. Yang, D. Batra, and D. Parikh. Hierarchical question-image co-attention for visual question answering, 2016.

[14] J. Mun, P. H. Seo, I. Jung, and B. Han. Marioqa: Answering questions by watching gameplay videos, 2016.

[15] M. Ren, R. Kiros, and R. Zemel. Exploring models and data for image question answering, 2015.

[16] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149, Jun 2017.

[17] K. J. Shih, S. Singh, and D. Hoiem. Where to look: Focus regions for visual question answering, 2015.

[18] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid. VideoBERT: A Joint Model for Video and Language Representation Learning. apr 2019.

[19] M. Tapaswi, Y. Zhu, R. Stiefelhagen, A. Torralba, R. Urtasun, and S. Fidler. Movieqa: Understanding stories in movies through question-answering, 2015.

[20] D. Tran, H. Wang, L. Torresani, J. Ray, Y. LeCun, and M. Paluri. A closer look at spatiotemporal convolutions for action recognition. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun 2018.

[21] C. Xiong, S. Merity, and R. Socher. Dynamic memory networks for visual and textual question answering, 2016.

[22] H. Xu and K. Saenko. Ask, attend and answer: Exploring question-guided spatial attention for visual question answering, 2015.

[23] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola. Stacked attention networks for image question answering, 2015.