DEEP-CAM: Attention based Multi-modal Deep Learning Models for Medical Instructional Question Generation

Shaswati Saha
University of Maryland Baltimore County
MD, USA 21250
ssaha3@umbc.edu

Sanjay Purushotham
University of Maryland Baltimore County
MD, USA 21250
psanjay@umbc.edu

Abstract

This paper describes the participation of the UMBCVQA team in the Medical Instructional Question Generation (MIQG) task of the MedVidQA challenge at TREC Video Retrieval Evaluation (TRECVID 2023). The goal of the MIQG task is to generate instructional questions for which the given medical video segment serves as the visual answer. We propose DEEP-CAM, a deep spatio-temporal, cross-modality, and cross-attention encoder-decoder model that takes a medical video segment and its corresponding subtitle text as input and generates a natural language question as output. DEEP-CAM first extracts visual features from the videos and textual embeddings from the subtitles corresponding to the video frames, simultaneously learning the attention for both the text and video frames. Furthermore, these jointly attended features are passed through an LSTM-based decoder to generate instructional questions based on the provided video frames.

- *Training data: We used 800 videos with 2710 questions from the MedVidQA dataset [8]. In addition, we extracted and used time-stamped subtitles for either the entire video or video segments.*
- *Our approach: We proposed DEEP-CAM, a deep spatio-temporal, cross-modality, and cross-attention encoder-decoder model that takes a medical video segment and subtitle text to generate an instructional question.*
- *Runs: We submitted two runs to the challenge. The key difference between our submitted runs is that in Run 1, we utilized the timed subtitles, while in Run 2, we provided the entire subtitle of a video to our model.*
- *Results: We found that the first iteration outperforms the second on all metrics, including ROUGE-2 [16], ROUGE-L [16], and BERTScore [24].*

1. Introduction

Visual Question Answering (VQA) and Visual Question Generation (VQG) from images are emerging research areas [4, 7, 11–15, 20] at the intersection of natural language processing and computer vision. A Visual query Answering (VQA) system receives an image and a query in natural language as input, and generates a response in natural language as output. A VQG mechanism's purpose is to generate natural language questions based on images. Both Visual Question Answering (VQA) and Visual Question Generation (VQG) integrate natural language processing to comprehend the question and generate the response, along with computer vision techniques to comprehend the image's content.

This paper describes the participation of the University of Maryland Baltimore County (UMBC) (team name UMBCVQA) in the Medical Instructional Question Generation (MIQG) task of the MedVidQA challenge at TREC Video Retrieval Evaluation (TRECVID 2023) [2]. Figure 1 shows an example of a visual answer (temporal segment) and corresponding health-related question for the MIQG task from the MedVidQA datasest [8].

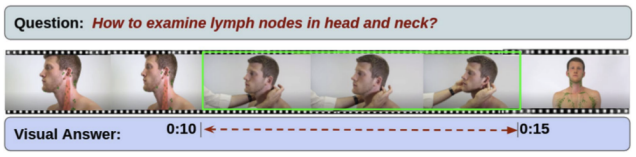

Figure 1. An example of a health-related question and its visual answer (temporal segment) from the video sampled from MedVidQA dataset [8].

2. Methodology

Figure 2 shows the overview of our proposed model DEEP-CAM. DEEP-CAM is a deep learning based cross-attention encoder-decoder model that operates across different dimensions including spatio-temporal aspects and

multiple modalities. It encompasses several key components: a BERT-based embedding module, an I3D-based spatio-temporal representation model, a Video attention module, and Cross-Attention Multi-modal encoder and decoder modules. In the following sections, we will delve into the dataset and preprocessing steps, detail the training pipelines, and provide a comprehensive description of all these modules.

2.1. Dataset and Preprocessing

The MedVidQA dataset for the MIQG task, as described in [8], comprises a training dataset with 800 videos and 2710 questions, a validation dataset with 49 videos and 145 questions, and a test dataset with 50 videos and 155 questions. Notably, the videos within this dataset do not include any associated subtitles. To augment the provided datasets, we extract subtitles for all the videos in the dataset using the corresponding URLs, employing a library known as the *youtube-transcript-api* [9]. Furthermore, we map the video segments (visual answers) to subtitles, along with their corresponding timestamps (e.g., as shown in Fig 1). To facilitate this process, we utilize GloVe embeddings [21] to convert the tokenized questions into vocabulary indices, with a vocabulary size of 2069.

The visual answers, i.e., video segments, in the above dataset have varying lengths and different frame rates (fps). To facilitate batch-based training of our proposed DEEP-CAM model, we aim to standardize the video lengths by introducing a Video Frame Sampling (VFS) algorithm. VFS selectively removes frames from some videos while using specific frames more frequently in others, ensuring that each data sample attains an identical frame length. We refrain from employing extrapolation between frames along the time-axis (i.e., generating intermediate frames by interpolating between adjacent frames) to create synthetic frames, as this could distort the original video content and hinder learning. Similarly, we avoid frame interpolation along the time-axis (i.e., replacing multiple frames with interpolated frames) to create synthetic frames. Instead, we employ oversampling when there are insufficient frames and undersampling when there are surplus frames. Algorithm 1 provides a detailed description of our proposed VFS algorithm's underlying procedure.

The choice of parameter L in VFS, which represents the number of desired frames, is crucial. Selecting a higher periodicity for frame removal can disrupt temporal consistency between objects, limiting the applicability of computer vision algorithms like normalizing flows [1]. On the other hand, using the same frame for training at a higher periodicity can cause spatio-temporal learning models to underperform by biasing them towards spatial attention over temporal. According to our experiment, we find that the best choice of $L = 1000$.

Algorithm 1 Video Frame Sampling

Input: 2D video frames, $\boldsymbol{G} = \{\boldsymbol{g_n}\}_{n=1}^{N} \in \mathcal{R}^{3\times N\times H\times W}$
Parameter: number of desired frames, L
Output: 2D video frames, $\boldsymbol{F} = \{\boldsymbol{f_l}\}_{l=1}^{L} \in \mathcal{R}^{3\times L\times H\times W}$

depletion factor $d = \lfloor \frac{L}{N} \rfloor$
remainder factor $r = L \ (mod\ N)$
if $N = L$ **then**
  return $\boldsymbol{F}$
if $N < L$ **then**
  $f[1 : r - \lfloor \frac{r}{d} \rfloor] = g[1]$
  $f[L - \lfloor \frac{r}{d} \rfloor : L] = g[N]$
  $j = r - \lfloor \frac{r}{d} \rfloor + 1$
  for $n \in \{0, 1, .., N\}$ **do**
    $f[j : j + d] = g[n]$
    $j := j + d$
  return $\boldsymbol{F}$
if $N > L$ **then**
  $j = r - \lfloor \frac{r}{d} \rfloor + 1$
  for $l \in \{0, 1, .., L\}$ **do**
    $f[l] = g[j]$
    $j := j + d$
  return $\boldsymbol{F}$

2.2. Our proposed DEEP-CAM model

As shown in Figure 2, DEEP-CAM takes in multimodal inputs: 1) visual data: video frames (extracted from videos) and 2) natural language text: subtitle descriptions.

Considering a single batch of input, the video data is denoted as $F \in \mathcal{R}^{3\times L\times H\times W}$ where 3 represents RGB color channels, $L = 1000$ represents the number of frames per input, (H, W) represents the spatial dimension of each frame. Since, video data is more expensive than still images (i.e., the number of frames), we first exploit a pre-trained feature extractor to (a) spatially downsize each frame, (b) extract meaningful features both spatially and temporally. More specifically, we use I3D model [3] to extract spatio-temporal features which is originally trained on video action classification: $I3D(F) \in \mathcal{R}^{832\times L\times \frac{H}{16}\times \frac{W}{16}}$. I3D is pre-trained using both 2D CNN, 3D CNN, and normalizing flows. To achieve further compute efficiency, we apply a 3D Average Pooling with stride $= (4, 1, 1)$ to downsample the temporal dimension (e.g., $\mathcal{R}^{832\times \frac{L}{4}\times \frac{H}{16}\times \frac{W}{16}}$). We denote the pooled features as $F' = AvgPool(I3D(F)) \in \mathcal{R}^{832\times \frac{L}{4}\times h'\times w'}$ where $h' = \frac{H}{16}, w' = \frac{W}{16}$. Sequentially, we pass these pooled features through a Video Attention ($\mathcal{VA}(.)$) module that contains 3D convolutions to learn spatio-temporal attention to obtain $\Psi = \mathcal{VA}(F') \in \mathcal{R}^{832\times \frac{L}{4}\times h'w'}$.

On the other hand, natural language text (t) description of the video, in the form of subtitles is provided with corresponding frames of the video as well. We first use the tokenizer from BERT [5] and then feed the text tokens to a pre-trained BERT to obtain an embedding, $\Gamma = BERT(t) \in \mathcal{R}^{T\times E}$ where max token length $T = 512$ and embedding size $E = 1024$.

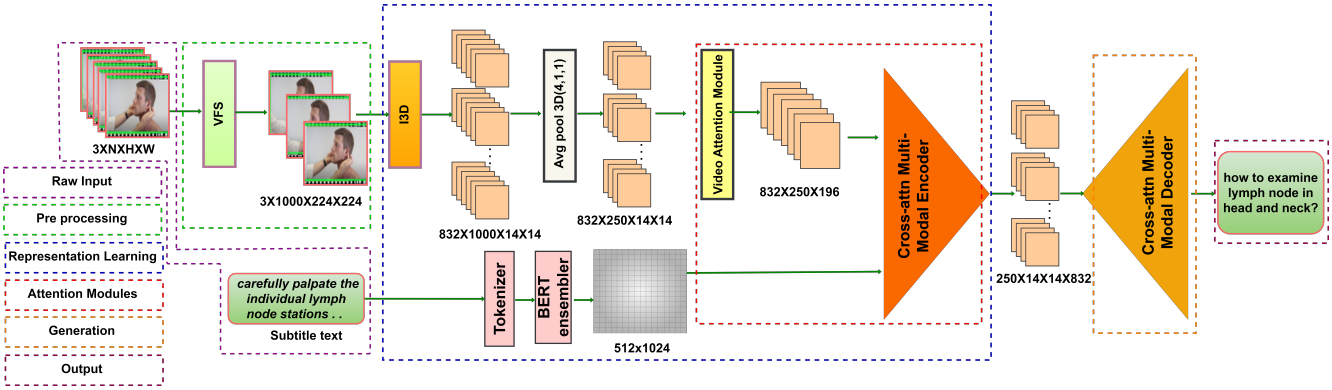

Figure 2. Overview for our proposed DEEP-CAM model for Question Generation given video and subtitle pair.

To generate questions both from the text embedding, Γ and video features, Ψ, we then use a Cross Attention Multi-Modal Encoder, $\mathcal{EN}(.)$ to learn a joint embedding and then use a Cross Attention Multi-Modal Decoder, $\mathcal{DE}(.)$ to generate instructional questions Q. In the subsequent sections, we discuss the following building blocks: Video Attention ($\mathcal{VA}(.)$), Cross Attention Multi-Modal Encoder, $\mathcal{EN}$, Cross Attention Multi-Modal Decoder, $\mathcal{DE}$.

2.2.1 Video Attention Module

We first learn three projections of the input video features called Key $\mathcal{K}$, Query $\mathcal{Q}$, Value $\mathcal{V}$ using 3D point-wise convolution operators, $\mathcal{W}_*$ defined in Eqn 1, 2, 3.

$$\mathcal{K} = F' \circledast \mathcal{W}_k \tag{1}$$

$$\mathcal{Q} = F' \circledast \mathcal{W}_q \tag{2}$$

$$\mathcal{V} = F' \circledast \mathcal{W}_v \tag{3}$$

We then compute a scaled Hadamard-attention weight metric $\mathcal{A} = \mathcal{Q}\mathcal{K}^T/\sqrt{d_q}$ where an attention score is computed among each spatial position and each temporal (each pixel in every frame). We then, use the attention scores and compute self-attention scores $\mathcal{S} = softmax(\mathcal{A})\mathcal{V}$ as shown in Eqn 4 and Fig 3. Moreover, the attention weight matrix, $\mathcal{A}$ is a $n \times N$ matrix where $N = 250 \times 14 \times 14$.

$$\mathcal{S} = softmax(\frac{\mathcal{Q}\mathcal{K}^T}{\sqrt{d_q}})\mathcal{V} \tag{4}$$

2.2.2 Cross Attention Multi-Modal Encoder

Image attention model, capable of anticipating the significance of each spatial grid in relation to the question, is advantageous for accurately predicting the answer. Based on the findings presented in [6], it is evident that including an attention mechanism enables the model to accurately

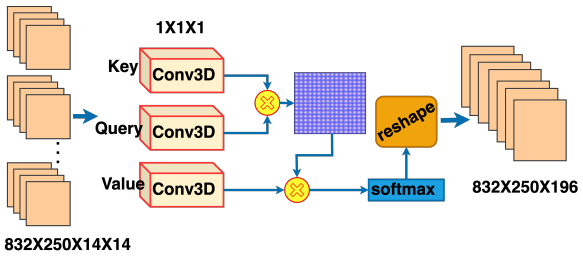

Figure 3. Video Attention (*spatio-temporal*) Module.

determine the significant region for the question, resulting in improved performance compared to the model without attention. Nevertheless, the attention model described in reference [6] solely concentrates on acquiring image attention and completely disregards question attention. Due to the natural language interpretation of the questions, the impact of each word varies dramatically. Thus, we propose a coattention learning method (refer to Figure 3) that simultaneously learns the attentions for both the query and image. The key distinction between the network architecture of our co-attention model and the attention model in [6] lies in the inclusion of a question attention module following the LSTM networks. This module enables the learning of attention weights for each word in the question. In contrast to prior co-attention models used for Visual Question Answering (VQA) [17, 18], our model has a loosely connected design for the image and question modules. This means that we do not utilize the image features during the learning process of the question attention module. This assumption is based on the belief that the network has the ability to deduce the question's focus, namely the keywords, without having access to the image, similar to how people do. To learn quantized features fused from two different modalities we use Multi-modal Factorized Bilinear (MFB) Pooling introduced in [23].

2.2.3 Cross Attention Multi-Modal Decoder

The learned joint encoding from the encoder that contains the fused attention weights of visual features and subtitles is passed through a Cross Attention Multi-Modal Decoder to produce instructional questions. This strategy is inspired from [22]. Before we generate the questions, the combined co-attended features are fed into a Cross Attention Multi-Modal Decoder that employs an LSTM-based structure. The fused features are utilized to initialize the states of the LSTM block. (during training time) As we go through the decoder, at each step i, we fuse the output of the attention module from the previous step $(i-1)$ with the current word in the ground truth question. The combined vector is subsequently fed into the LSTM module to acquire the ith word in the predicted question.

3. Experiments and Results

In this section, we briefly describe the experiments, including the implementation details, evaluation metrics, and our submission and results.

3.1. Implementation Details

Our DEEP-CAM model is implemented in PyTorch. We use the pre-trained I3D model, from which the 'Mixed_4f' output (without the final average pooling) is used as the visual features. We tokenize the questions and extract embeddings using Glove [21], while for the subtitles corresponding to the video frames, we use BERT embeddings [5]. We use the Adam optimizer [10] with a fixed learning rate of 0.0001 to update the parameters. The training batch size is set to 1, and all models are trained for 10 epochs.

3.2. Evaluation Metrics

As the standard practice and specified by the challenge, we employ the BLEU [19], Rouge [16], and BertScore [24] to assess the effectiveness of question generation.

3.3. Submissions

UMBCVQA team participated in the TRECVID 2023 MIQG challenge and submitted two separate runs.

- Run-1: We employ timed subtitles that correspond to the start and finish time stamps of the video segments (visual answers) provided in the dataset to train our Cross-Attention MultiModal Encoder-Decoder Network.

- Run-2: We employ the same procedure as in the first run, but here we utilize the complete subtitle of a video, disregarding the starting and ending times of the video segments.

3.4. Results

UMBCVQA team participated in the TRECVID 2023 MIQG challenge and obtained scores for two separate runs, as indicated in Table 1. We note that, based on these two iterations, the first iteration outperforms the second on all metrics except the BLEU score. This discrepancy occurs due to the presence of textual information that is unrelated to the visual features of a specific frame.

RunID	BLEU	BLEU-4	ROUGE-2	ROUGE-L	BERTScore
Run-1	0	0	0.13167	0.3154	0.87683
Run-2	0	0	0.12262	0.26083	0.85332

Table 1. Evaluation of our question generation method based on the submitted runs in the TRECVID 2023 MIQG challenge.

References

[1] Abdelrahman Abdelhamed, Marcus A Brubaker, and Michael S Brown. Noise flow: Noise modeling with conditional normalizing flows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3165–3173, 2019. 2

[2] George Awad, Keith Curtis, Asad A. Butt, Jonathan Fiscus, Afzal Godil, Yooyoung Lee, Andrew Delgado, Eliot Godard, Lukas Diduch, Yvette Graham, , and Georges Quénot. Trecvid 2023 - a series of evaluation tracks in video understanding. In *Proceedings of TRECVID 2023*. NIST, USA, 2023. 1

[3] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 2

[4] Santiago Castro, Mahmoud Azab, Jonathan Stroud, Cristina Noujaim, Ruoyao Wang, Jia Deng, and Rada Mihalcea. Lifeqa: A real-life dataset for video question answering. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 4352–4358, 2020. 1

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2, 4

[6] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 457–468, 2016. 3

[7] Noa Garcia, Mayu Otani, Chenhui Chu, and Yuta Nakashima. Knowit vqa: Answering knowledge-based questions about videos. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 10826–10834, 2020. 1

[8] Deepak Gupta, Kush Attal, and Dina Demner-Fushman. A dataset for medical instructional video classification and question answering. *Scientific Data*, 10(1):158, 2023. 1, 2

[9] Jdepoix. Jdepoix/youtube-transcript-api. 2

[10] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[11] Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Information maximizing visual question generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2008–2018, 2019. 1

[12] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara Berg. Tvqa: Localized, compositional video question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, page 1369. Association for Computational Linguistics, 2018. 1

[13] Jie Lei, Licheng Yu, Tamara Berg, and Mohit Bansal. Tvqa+: Spatio-temporal grounding for video question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8211–8225, 2020. 1

[14] Jie Lei, Licheng Yu, Tamara L Berg, and Mohit Bansal. Tvr: A large-scale dataset for video-subtitle moment retrieval. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16*, pages 447–463. Springer, 2020. 1

[15] Yikang Li, Nan Duan, Bolei Zhou, Xiao Chu, Wanli Ouyang, Xiaogang Wang, and Ming Zhou. Visual question generation as dual task of visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6116–6124, 2018. 1

[16] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004. 1, 4

[17] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. *Advances in neural information processing systems*, 29, 2016. 3

[18] Hyeonseob Nam, Jung-Woo Ha, and Jeonghee Kim. Dual attention networks for multimodal reasoning and matching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 299–307, 2017. 3

[19] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002. 4

[20] Charulata Patil and Manasi Patwardhan. Visual question generation: The state of the art. *ACM Computing Surveys (CSUR)*, 53(3):1–22, 2020. 1

[21] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014. 2, 4

[22] Dhruv Sharma, Sanjay Purushotham, and Chandan K Reddy. Medfusenet: An attention-based multimodal deep learning model for visual question answering in the medical domain. *Scientific Reports*, 11(1):19826, 2021. 4

[23] Zhou Yu, Jun Yu, Jianping Fan, and Dacheng Tao. Multi-modal factorized bilinear pooling with co-attention learning for visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 1821–1830, 2017. 3

[24] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019. 1, 4