Multi-Scale Progressive Attention Network for Video Question Answering

Zhicheng Guo Jiaxuan Zhao Licheng Jiao Xu Liu Lingling Li
School of Artificial Intelligence, Xidian University, Xi'an, Shaanxi Province, 710071, China
{zchguo, jiaxuanzhao}@stu.xidian.edu.cn
lchjiao@mail.xidian.edu.cn {xuliu, llli}@xidian.edu.cn

Abstract

Understanding the multi-scale visual information in a video is essential for Video Question Answering (VideoQA). Therefore, we propose a novel Multi-Scale Progressive Attention Network (MSPAN) to achieve relational reasoning between cross-scale video information. We construct clips of different lengths to represent different scales of the video. Then, the clip-level features are aggregated into node features by using max-pool, and a graph is generated for each scale of clips. For cross-scale feature interaction, we design a message passing strategy between adjacent scale graphs, i.e., top-down scale interaction and bottom-up scale interaction. Under the question's guidance of progressive attention, we realize the fusion of all-scale video features. Experimental evaluations on three benchmarks: TGIF-QA, MSVD-QA and MSRVTT-QA show our method has achieved state-of-the-art performance.

1 Introduction

Video Question Answering (VideoQA) is a popular vision-language task, which focuses on predicting the correct answer to a given natural language question based on the corresponding video. VideoQA task entails representing video features in both spatial and temporal dimensions. Compared with the visual features of a picture in Visual Question Answering, it requires a more complex attention.

Therefore, (Jang et al., 2017) employed appearance features and motion features as video representation, and designed a dual-LSTM network based on spatio-temporal attention to fuse visual and text information. Next, memory networks are widely used to capture long-term dependencies. For example, (Cai et al., 2020) applied feature augmented memory to strengthen the information augmentation of video and text. Complex relational reasoning is important for VideoQA task. Consequently, a conditional relationship network (Le et al., 2020) was designed in previous work, which can support high-order relationships and multi-step reasoning.

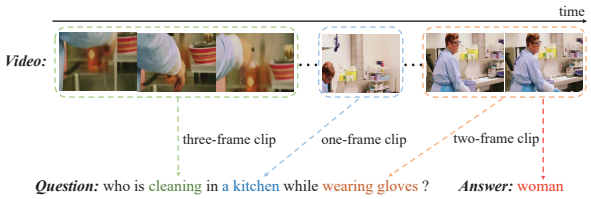

Figure 1: Understanding the video and answering the question require different levels of clips.

Many methods complete this task from a certain aspect, however, none of them have a fine-grained understanding of video information. When looking for the answer in a question-based video, the video frames corresponding to different objects in the question are of different lengths. As shown in Fig. 1, when asked "who is cleaning in a kitchen while wearing gloves?", we need to find the keywords "cleaning", "a kitchen" and "wearing gloves" from different levels of clips. Previous methods searched for the answer on the same level of clips in a video, leading to insufficient or redundant information.

Firstly, we construct clips of different lengths from the frame sequence, and regard the length of a clip as its scale information. Then, multi-scale graphs are generated separately for clips of different scales. The nodes in the multi-scale graphs indicate video features corresponding to different clips. For implementing relational reasoning, the nodes in each scale graph are first updated by using graph convolution. Most importantly, under the guidance of the question, progressive attention has been utilized to enable the fusion of multi-scale features during cross-scale graph interaction. In detail, each graph is gradually updated in top-down scale order, followed by updating each graph in bottom-up scale order. Finally, node features of a graph are fused with question embedding, and a classifier is employed to find the answer.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Short Papers), pages 973–978
August 1–6, 2021.

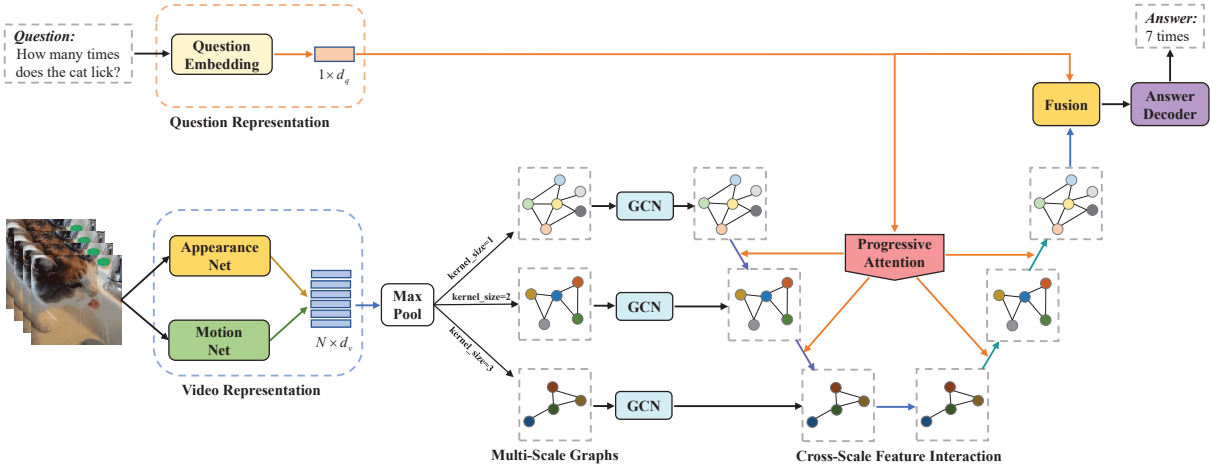

Figure 2: The model architecture of Multi-Scale Progressive Attention Network for VideoQA. Major contributions focus on the construction of multi-scale graphs and the progressive attention for cross-scale feature interaction.

2 Method

An overview of the proposed MSPAN is shown in Fig. 2. The input is a short video and a question sentence, while the output is the produced answer.

2.1 Video and Question Representation

Video representation N frames are uniformly sampled to represent the video. Then we use the pre-trained ResNet-152 (He et al., 2016) to extract video appearance features for each frame. And, we apply the 3D ResNet-152 (Hara et al., 2018) pre-trained on Kinetics-700 (Carreira et al., 2019) dataset to extract video motion features. Specifically, 16 frames around each frame are placed into the 3D ResNet-152 to obtain the motion features around this frame. Finally, we get a joint video representation by concatenating appearance features and motion features. By using a fully-connected layer to reduce feature dimension, we obtain video representation as $V = \{v_i : i \leq N, v_i \in R^{2048}\}$.

Question representation All words in question are represented as 300-dimensional embeddings initialized with pre-trained GloVe vectors (Pennington et al., 2014). And a 512-dimensional question embedding is generated from the last hidden state of a three-layer BiLSTM, i.e., $q \in R^{512}$.

2.2 Multi-Scale Graphs Generation

Each object in the video corresponds to a different number of frames, but previous methods (Seo et al., 2020; Lei et al., 2021) cannot treat various levels of visual information separately. Therefore, we construct clips of different lengths to express the visual information in the video delicately, and regard the length attribute as a scale.

We use max-pools of different kernel-sizes to aggregate frame-level visual features, and kernel-size is the scale attribute of these clips. In this way, clip-level visual features are obtained, as follows:

$$P = \{pool_i | 1 \leq i \leq K, kernel\_size_i = i\} \quad (1)$$

$$V_i = P_i(v_1, v_2, \cdots, v_N) \quad (2)$$

Where K is the range of scales, and $K \leq N$. Thus, we construct $M_i = N - i + 1$ clips at scale i:

$$V_i = \{v_j^i : 1 \leq j \leq M_i, v_j^i \in R^{2048}\} \quad (3)$$

In order to reason the relationships between different objects in a video, we separately build a graph for each scale. Each node in a graph represents the clip-level visual features. Only when two nodes contain overlapping or adjacent frames, an edge will be connected between them. Frame interval of the j-th clip at scale i is $[j, j + i - 1]$, so all edges in the K graphs can be expressed as:

$$E_i = \{(x, y) | x - i \leq y \leq x + i\} \quad (4)$$

Finally, these multi-scale graphs constructed in this paper can be denoted as $G_i = \{V_i, E_i\}$.

2.3 Cross-Scale Feature Interaction

Before cross-scale feature interaction, the original node features of K graphs are copied as $V_i^o = V_i$.

Interaction at the same scale. For all nodes with the same scale, we apply a two-layer graph convolutional network (GCN) (Kipf and Welling, 2017) to perform relational reasoning over the K

graphs. The process of graph convolution is represented as:

$$X_{l+1} = \hat{D}^{-\frac{1}{2}}\hat{A}\hat{D}^{-\frac{1}{2}}X_l W_l \quad (5)$$

Where $\hat{A}$ is the input adjacency matrix, X_l is the node feature matrix of layer l, and W_l is the learnable weight matrix. The diagonal node degree matrix $\hat{D}$ is used to normalize $\hat{A}$. Due to the small number of nodes in each graph, we decide to share the weight matrix W_l when K graphs are updated.

Interaction at top-down scale. We realize the interaction of adjacent scale graphs from small scale to large scale. Therefore, visual information is understood step by step from details to the whole through the interaction of top-down scale. Guided by the question, the nodes in graph G_i are used to update the nodes in graph G_{i+1}. Visual features at different scales show hierarchical attention to the question, so we call it progressive attention.

If the clip corresponding to node x in graph G_i has the same frames as the clip corresponding to node y in graph G_{i+1}, there will exist a directed edge from x to y. Therefore, we can use the edge to fuse the cross-scale features of these same frames.

Firstly, visual features and question embedding are fused to capture the joint features of each node in graph G_i. Then, the process of message passing from graph G_i to graph G_{i+1} can be expressed as:

$$m_{xy} = (W_1 v_y^{i+1}) \otimes ((W_2 v_x^i) \odot (W_3 q))^T \quad (6)$$

Where $\otimes$ is the outer product, $\odot$ is the hadamard product. After receiving the delivery messages, the attention weights of these messages are calculated:

$$w_{xy} = \underset{x \in \mathcal{N}_y}{soft\max}(m_{xy}) \quad (7)$$

Where $\mathcal{N}_y$ is the set of all neighbor nodes in graph G_i through cross-scale edges. Consequently, all the messages passed into node y are summed to derive the update of node y, as follows:

$$\tilde{v}_y^{i+1} = \sum_{x \in N_y} w_{xy} \cdot ((W_4 v_x^i) \odot (W_5 q)) \quad (8)$$

$$V_{i+1}^u = \{\tilde{v}_y^{i+1} : y \leq M_{i+1}, \tilde{v}_y^{i+1} \in R^{2048}\} \quad (9)$$

When updating all nodes in graph G_{i+1}, we consider the new features V_{i+1}^u and the original features V_{i+1}^o. Therefore, we use the residual connection to preserve original information of the video:

$$V_{i+1} = W_6[V_{i+1}; V_{i+1}^u] + V_{i+1}^o \quad (10)$$

Where $[;]$ is the concatenation operator. Above $W_1 \sim W_6$ are learnable weights, and they are shared in the update of graphs $G_2 \sim G_K$. To summarize, the update of $K-1$ graphs is a progressive process from small scale to large scale, hence it is referred to as top-down scale interaction.

Interaction at bottom-up scale. After an overall understanding of the video, people can accurately find all details related to the question at the second time they watch the video. Therefore, we achieve an understanding of the video from global to local through bottom-up scale interaction. After the previous interaction, we realize the interaction of adjacent graphs from large scale to small scale.

Following the same method as top-down scale interaction from Eq. 6 to Eq. 10, we apply graph G_i to update graph G_{i-1} in this interaction. But the weights $W_1 \sim W_6$ are another group in the update of graphs $G_{K-1} \sim G_1$. After this interaction, graph G_1 can grasp the all-scale video features related to the question by progressive attention.

2.4 Multimodal Fusion and Answer Decoder

After T iterations of cross-scale feature interaction, we read out all the nodes in graph G_1. Then, a simple attention is used to aggregate the N nodes. And, final multi-modal representation is given as:

$$\tilde{w}_j = soft\max(W_7(W_8 v_j^1) \odot (W_9 q)) \quad (11)$$

$$\tilde{F} = \sum_{j=1}^{N} \tilde{w}_j \cdot v_j^1 \quad (12)$$

$$F = ELU(W_{10}\tilde{F} \odot W_{11} q + b) \quad (13)$$

Where ELU is activation function, above $W_7 \sim W_{11}$ are learnable weights and b is learnable bias. We can find the answer by applying a classifier (two fully-connected layers) on multi-modal representation F. Multi-label classifier is applied to open-ended tasks, and cross-entropy loss function is used to train the model. Due to repetition count is a regression task, we use the MSE loss function.

For the multi-choice task, each question corresponds to R answer sentences. We first get the embedding of each answer in the same way as the question embedding. Then we use the multi-modal fusion method in Eq. 11∼13 to fuse the answer embedding with node features. After using two fully-connected layers, the answer scores $\{s_i\}_{i=1}^R$ have appeared. This model is trained by minimizing the hinge loss (Jang et al., 2017) of pairwise comparisons between answer scores $\{s_i\}_{i=1}^R$.

3 Experiments

3.1 Datasets

TGIF-QA (Jang et al., 2017) is a widely used large-scale benchmark dataset for VideoQA. And four task types are covered in this dataset: repeating action (Action), repetition count (Count), video frame QA (FrameQA) and state transition (Trans.). **MSVD-QA** (Xu et al., 2017) and **MSRVTT-QA** (Xu et al., 2016) are open-ended tasks which are generated from video descriptions. In both datasets, questions can be divided into 5 types according to question words: what, who, how, when and where.

3.2 Implementation Details

We evenly sample $N = 16$ frames for each video in the three datasets. The hyperparameters we set in experiments are as follows: $T = 3$, $K = 8$. When training the network, Adam is used with an initial learning rate of 10^{-4}. For TGIF-QA dataset, the batch size is 64. While the batch size is set to 128 for both MSVD-QA and MSRVTT-QA datasets.

3.3 Results

We compare our MSPAN with the state-of-the-art methods: PSAC (Li et al., 2019), HME (Fan et al., 2019), FAM (Cai et al., 2020), LGCN (Huang et al., 2020), HGA (Jiang and Han, 2020), QueST (Jiang et al., 2020) and HCRN (Le et al., 2020).

Table 1: Comparison on TGIF-QA dataset.

Method	Action	Count	FrameQA	Trans.
PSAC	70.4	4.27	55.7	76.9
HME	73.9	4.02	53.8	77.8
FAM	75.4	3.79	56.9	79.2
LGCN	74.3	3.95	56.3	81.1
HGA	75.4	4.09	55.1	81.0
QueST	75.9	4.19	59.7	81.0
HCRN	75.0	3.82	55.9	81.4
MSPAN	**78.4**	**3.57**	**59.7**	**83.3**

Results on TGIF-QA. As shown in Table 1, our method outperforms the state-of-the-art methods by **2.5%** and **1.9%** of accuracy on Action and Transition tasks. For the Count task, our method also achieves the best Mean Square Error (MSE) of **3.57** among all methods. Due to QueST used multi-dimension visual features containing more appearance information, our method can only get the same accuracy **59.7%** as QueST on the FrameQA task.

Table 2: Comparison on MSVD-QA dataset.

Method	What	Who	How	When	Where	All
	62.7%	33.9%	2.8%	0.4%	0.2%	100%
HME	22.4	50.1	73.0	70.7	42.9	33.7
QueST	24.5	52.9	79.1	72.4	50.0	36.1
HGA	23.5	50.4	**83.0**	72.4	46.4	34.7
FAM	23.1	51.6	82.2	71.4	51.9	34.5
MSPAN	**31.0**	**53.8**	77.0	**72.4**	**53.6**	**40.3**

Table 3: Comparison on MSRVTT-QA dataset.

Method	What	Who	How	When	Where	All
	68.5%	27.7%	2.5%	1.0%	0.3%	100%
HME	26.5	43.6	82.4	76.0	28.6	33.0
QueST	27.9	45.6	83.0	75.7	31.6	34.6
HGA	29.2	45.7	**83.5**	75.2	34.0	35.5
FAM	26.9	43.9	82.8	70.6	31.1	33.2
MSPAN	**31.9**	**47.2**	83.2	**77.5**	**38.4**	**37.8**

All in all, our method makes sense of the multi-scale information of the video, so that the effect on tasks related to action recognition, temporal relationship and object count are very noticeable.

Results on MSVD-QA. As shown in Table 2, our method improves the overall accuracy by **4.2%** compared to recent methods. We have achieved the best accuracy on questions whose question words are "What" , "Who", "When" and "Where". Due to a small proportion, the accuracy on the question word "How" is lower than other methods.

Results on MSRVTT-QA. As shown in Table 3, our method achieves the best overall accuracy of **37.8%**. What's more, Our method could obtain excellent accuracy on different question words.

4 Ablation Studies

To explore the potential of our network, ablation experiments are performed on TGIF-QA dataset. Default hyperparameters are: $T = 3$ and $K = 8$. We study the effectiveness of our network in the next two aspects, as shown in Table 4 and Fig. 4.

4.1 Different Structures

Considering the interaction of cross-scale graphs, three structures are designed, as shown in Fig. 3. For the dense scale in Fig. 3 (a), we apply graphs $G_1 \sim G_K$ to update each graph G_i. The other two structures have been introduced in Sec 2.3, and we will not use a graph to update itself for the three

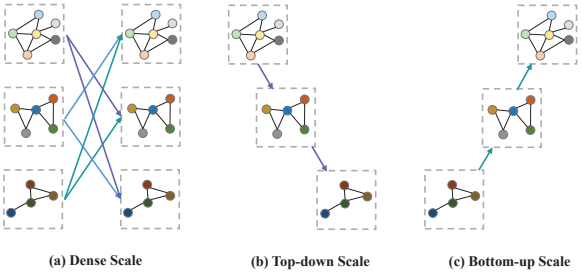

Figure 3: Three methods of cross-scale feature interaction, where the dense connection is not adopted.

structures. The readout of top-down scale interaction is graph G_K, and the readout of bottom-up scale interaction is G_1. However, the readout of dense scale interaction is all K graphs. Our network is a combination of top-down scale interaction and bottom-up scale interaction, but we will use these two structures separately for comparison.

4.2 Network structure

When choosing the pooling function to aggregate these frames in a clip, we find that max-pool is more effective than avg-pool. In reverse gradient propagation of max-pool, only the maximum of features in the previous layer receive the gradient. So, max-pool facilitates the fusion of appearance features and motion features in the previous layer.

Our experiments show that GCN is beneficial to the stable training of models. If there is no GCN, the gradient will gradually disappear as the number of interactions between the graphs increases. The role of GCN is to re-recover the features of these nodes which have lost their visual features.

As shown in Table 4, the performances of the three structures in Fig. 3 are poorer than that of our entire network. Due to dense connections between all scale graphs, the dense scale interaction will add much unnecessary computation, and make it difficult to accurately find the visual information related to the question. Although both the top-down scale interaction and the bottom-up scale interaction can achieve good performance. However, the combination of these two structures will obtain a more detailed understanding of the video.

4.3 Hyperparameters T and K

As the number of iterations T increases, the model will achieve better performance. But when $T = 4$, the effect of the model decreases, as shown in Table 4. Because too many modules will produce noise for answer generation. The improvement

Table 4: Ablation experiments of four types: (1)Replacing max-pool with avg-pool. (2)Without GCN. (3)Different structures in Fig. 3. (4)Different iterations T.

Parameters	Action	Count	FrameQA	Trans.
Avg-pool	78.0	3.56	59.5	83.3
w/o GCN	77.5	3.64	59.1	82.7
Dense scale	77.2	3.74	59.2	82.0
Top-down scale	78.1	3.62	59.6	82.8
Bottom-up scale	78.1	3.60	59.3	82.6
$T = 0$	75.2	3.86	56.7	79.9
$T = 1$	77.1	3.69	58.6	82.5
$T = 2$	77.7	3.61	**59.7**	82.9
$T = 4$	77.6	3.63	59.4	82.5
Full MSPAN	**78.4**	**3.57**	59.7	**83.3**

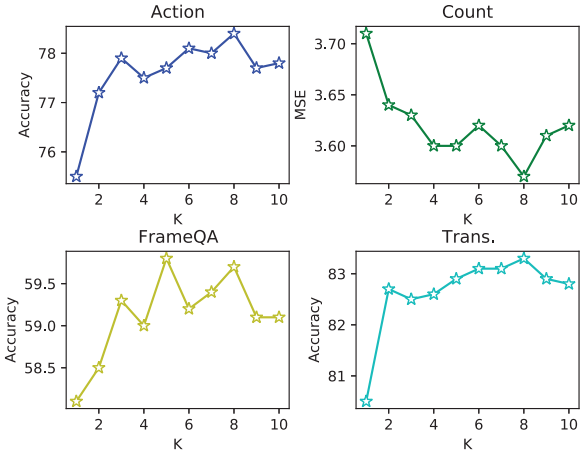

Figure 4: Ablation experiments for different scales K.

of models with the increase of K is very obvious, and best performance is obtained when $K = 8$, as shown in Fig. 4. However, the larger K also means that many multi-scale graphs, which will lead to network instability.

5 Conclusion

We introduce a multi-scale learning method to achieve a fine-grained understanding of the video. Compared with existing spatio-temporal attention, we use progressive attention to realize cross-scale feature interaction. The top-down and bottom-up structures we have designed are conducive to learning all-scale visual information of the video. For longer videos, we plan to use dilated max-pools with different strides to reduce the size of graphs. In general, we consider the VideoQA task from the perspective of multi-scale information interaction, and the proposed network is instructive.

References

Jiayin Cai, Chun Yuan, Cheng Shi, Lei Li, Yangyang Cheng, and Ying Shan. 2020. Feature augmented memory with global attention network for videoqa. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 998–1004.

Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. 2019. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*.

Chenyou Fan, Xiaofan Zhang, Shu Zhang, Wensheng Wang, Chi Zhang, and Heng Huang. 2019. Heterogeneous memory enhanced multimodal attention model for video question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1999–2007.

Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. 2018. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6546–6555.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Deng Huang, Peihao Chen, Runhao Zeng, Qing Du, Mingkui Tan, and Chuang Gan. 2020. Location-aware graph convolutional networks for video question answering. In *AAAI*, pages 11021–11028.

Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. 2017. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2758–2766.

Jianwen Jiang, Ziqiang Chen, Haojie Lin, Xibin Zhao, and Yue Gao. 2020. Divide and conquer: Question-guided spatio-temporal contextual attention for video question answering. In *AAAI*, pages 11101–11108.

Pin Jiang and Yahong Han. 2020. Reasoning with heterogeneous graph alignment for video question answering. In *AAAI*, pages 11109–11116.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. 2020. Hierarchical conditional relation networks for video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9972–9981.

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

Xiangpeng Li, Jingkuan Song, Lianli Gao, Xianglong Liu, Wenbing Huang, Xiangnan He, and Chuang Gan. 2019. Beyond rnns: Positional self-attention with co-attention for video question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8658–8665.

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

Paul Hongsuck Seo, Arsha Nagrani, and Cordelia Schmid. 2020. Look before you speak: Visually contextualized utterances. *arXiv preprint arXiv:2012.05710*.

Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. 2017. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1645–1653.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296.