Leveraging Convolutional Models as Backbone for Medical Visual Question Answering

Lei Liu, Xiangdong Su*, Guanglai Gao
College of Computer Science
Inner Mongolia University
National & Local Joint Engineering Research Center of Intelligent Information Processing Technology for Mongolian
Hohhot, China
liulei@mail.imu.edu.cn, cssxd@imu.edu.cn, csggl@imu.edu.cn

2024 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) | 979-8-3503-8622-6/24/$31.00 ©2024 IEEE | DOI: 10.1109/BIBM62325.2024.10821918

***Abstract*—Convolutional neural networks (CNNs) have made significant contributions to computer vision and offer the advantages of higher training efficiency and lower model complexity. However, their application as the backbone in medical visual question answering (MedVQA) remains an open question. To address this issue, we employ popular convolutional models, including ResNet, DenseNet, and ShuffleNet, as the foundation for MedVQA, achieving outstanding performance. Different backbones can be tailored to diverse real-world scenarios. The central challenge in utilizing CNNs for visual question answering is effectively managing textual features and integrating multi-modal information. To overcome this challenge, we design a novel global interaction attention (GIA) that facilitates efficient interactions between text and image features. Additionally, we utilize the dot product before the classifier output to enhance visual and textual modal fusions. To further enhance model performance, we propose a novel multi-modal hidden mixup (MHidMix) technique for data augmentation, which involves interpolating hidden states during model training. This data augmentation technique smoothes the decision boundary without the need for complex sample selection, further improving model performance. Experimental results underscore the versatility of our proposed framework across various convolutional models, leading to outstanding performance on four MedVQA datasets. Notably, we achieved an accuracy increase of 9.4% on the PathVQA dataset and 4.5% on the OVQA dataset.**

***Index Terms*—Medical visual question answering, Convolutional neural networks, Multi-modal hidden mixup**

I. Introduction

Medical Visual Question Answering (MedVQA) involves answering questions about medical images and relevant textual information, such as radiology reports. The distinctive features of medical images pose unique challenges in MedVQA.

Many existing MedVQA research follows traditional approaches used in the VQA task, where a convolutional neural network (CNN) is employed to extract image features, an LSTM [1] to extract text features, and a fusion module [2]–[4] to combine multimodal information for prediction. Some studies also explore transformer [5], [6] models as backbone networks to handle feature extraction and fusion of the two modalities. Despite the significant role of convolutional networks in computer vision, their inherent limitations in processing text content present challenges for their direct application as backbone networks in the VQA task, raising doubts about their effectiveness in achieving exceptional performance.

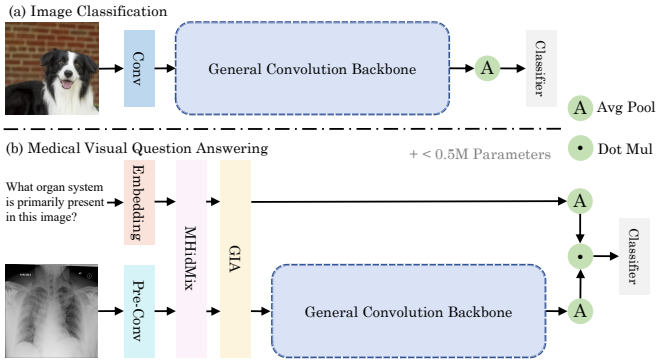

Fig. 1: (a) Convolutional models for image classification. (b) Convolutional models are used as backbone for MedVQA.

MedVQA involves responding to concise questions concerning medical images, often comprising just around ten words. In this context, the brevity of these text descriptions allows for streamlined feature extraction methods, obviating the need for resource-intensive and parameter-heavy models like transformers [7]. The convolutional model demonstrates its versatility across numerous scenarios and excels in extracting essential features from medical images. If convolutional models can effectively apply the MedVQA task, these trained models can also be applied to various scenarios. For instance, in scenarios involving mobile and embedded devices, the MedVQA model becomes a valuable tool for aiding physicians in medical diagnoses. This setting is particularly well-suited for small convolutional models like ShuffleNet [8] and MobileNet [9].

This paper delves into the exploration of convolutional models and designs a novel framework for effectively employing general convolutional models as backbone networks. The primary challenge addressed in this work is the direct integration of textual information into convolutional networks as backbone structures for MedVQA. To overcome this limitation, we propose a global interaction attention (GIA), as illustrated in Figure 1, which facilitates an initial interaction between text and image before passing the image into the backbone network. This early interaction ensures the ability of the convolutional network to effectively extract text-based image features, distinct from the conventional fusion methods [2]–

[4] of image and text features. Additionally, we employ global pooling separately on final visual and textual features and then dot-multiply these pooled features, which are subsequently fed into the classifier. Drawing inspiration from Manfoid Mixup [10], we propose the MHidMix data augmentation technique to interpolate and mix hidden states in the network, reducing the risk of model overfitting. This technique of hidden state interpolation smoothes the decision boundary, eliminating the necessity for intricate sample selection, and enables the model to achieve exceptional performance through direct adjustments to the hidden state. The experimental results on VQA-RAD [11], SLAKE [12], PathVQA [13], and OVQA [14] datasets demonstrate that the framework proposed in this paper can be applied to any general-purpose convolution model, resulting in an excellent performance. Additionally, the inclusion of MHidMix further improves the effectiveness of the model, thereby validating the effectiveness of the data augmentation.

II. Method

We present a comprehensive framework that utilizes a convolutional model as the backbone for the MedVQA task. Figure 2 illustrates the complete structure of our framework.

Preprocessing. In our designed framework, when presented with a medical image $\boldsymbol{I} \in \mathbb{R}^{H\times W\times 3}$ and its corresponding text question $\boldsymbol{T} \in \mathbb{R}^{L}$, we employ a Pre-Conv to downsample the image and extract shallow visual features. We utilize Glove [15] to initialize the text embedding.

A. Pre-Conv

The pre-convolution module (Pre-Conv) serves the purpose of downsampling the image to facilitate subsequent global interaction attention (GIA) for both text and image. To achieve this, we employ two layers of 3×3 convolutions, incorporating batch normalization [16], and GELU activation functions. This process helps extract shallow visual features necessary for subsequent steps in our framework. Given an input image tensor $\boldsymbol{I} \in \mathbb{R}^{H\times W\times 3}$, the step is formulated as:

$$\hat{\boldsymbol{I}} = \phi(BN(W_3(W_3(\boldsymbol{I})))), \tag{1}$$

where ϕ represents the GELU activation function, BN represents batch mormalization, and W_3 denotes 3×3 convolution. This downsampling process results in an image represented as $\hat{\boldsymbol{I}} \in \mathbb{R}^{\hat{H}\times\hat{W}\times C}$, where C typically has a value of 64. The height ($\hat{H}$) and width ($\hat{W}$) of the downsampled image are reduced to one-fourth of the original image size.

Simultaneously, we perform text embedding and mapping, resulting in $\hat{\boldsymbol{T}} \in \mathbb{R}^{L\times C}$. When utilizing MHidMix, the text embedding and image features are mixed, as described in the MHidMix subsection. Afterward, the mixed text embedding and shallow image features are inputted into the GIA module to enable attention-based interaction.

B. MHidMix

In the VQA dataset, each example is represented as (V, Q, A), where V denotes an input medical image, Q corresponds to a given question, and A corresponds to the associated answer, represented as a one-hot vector. MHidMix stands out among other multi-modal data augmentation techniques [17]–[19] by employing hidden state interpolation to effectively smooth decision boundaries. This distinctive approach eliminates the necessity for complex sample selection processes, rendering it a straightforward and highly efficient method for data augmentation. Moreover, MHidMix is a plug-and-play data augmentation technique, ensuring that no additional computing resources are required. By incorporating MHidMix into the training process, the model learns from a diverse set of interconnected samples, leading to more robust and generalized feature representations. As a result, MHidMix helps mitigate the risk of overfitting and significantly enhances the performance of the model in the MedVQA task.

Given two VQA training sample hidden states (V_i, Q_i, A_i) and (V_j, Q_j, A_j), MHidMix is used to generate new training sample hidden states $(V_\varphi, Q_\varphi, A_\varphi)$, and it is generated via

$$\begin{aligned} V_\varphi &= \lambda V_i + (1-\lambda)V_j, \\ Q_\varphi &= \lambda Q_i + (1-\lambda)Q_j, \\ A_\varphi &= \lambda A_i + (1-\lambda)A_j, \end{aligned} \tag{2}$$

where the mixing coefficient λ is introduced to control the degree of mixing, following the $Beta(\alpha, \beta)$ distribution, where α and β are hyperparameters that determine the sampling of the distribution. Notably, we have observed that the most prominent effects occur when λ is sampled from $Beta(3, 1)$ in our experiments. By adjusting the values of α and β, we can effectively control the degree of interpolation during the mixing process, allowing for fine-grained control over the interpolation of visual and textual information.

Recognizing that the VQA task involves two modalities, namely image, and text, we specifically designed MHidMix to enhance the capability of the convolutional model to extract multimodal features. By incorporating MHidMix into the training process, we aim to optimize the performance of the model by effectively interpolating both visual and textual information, leading to improved VQA results.

C. GIA

Given a preprocessed image tensor $\boldsymbol{I} \in \mathbb{R}^{\hat{H}\times\hat{W}\times C}$ and a text tensor $\boldsymbol{T} \in \mathbb{R}^{L\times C}$, the GIA module generates query ($\boldsymbol{Q}$), key ($\boldsymbol{K}$), and value ($\boldsymbol{V}$) components. Since $\boldsymbol{I}$ is obtained through a two-layer convolutional network, and the $\boldsymbol{K}$ generated from the text $\boldsymbol{T}$ is obtained through embedding, the $\boldsymbol{Q}$ and $\boldsymbol{K}$ do not require complex transformations. We perform reshape and matrix multiplication directly for $\boldsymbol{Q}$ and $\boldsymbol{K}$. To address the insufficient extraction of text information in the model, we enhance the importance of text information by incorporating additional linear mapping and non-linear interaction for the $\boldsymbol{V}$ component. Specifically, we calculate $\boldsymbol{V} = \tanh(W\boldsymbol{T}) \odot \sigma(W\boldsymbol{T})$, where W denotes linear mapping. The entire calculation process is as follows:

$$\begin{aligned} \hat{\boldsymbol{I}} &= W_3 Attention(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) + \boldsymbol{I}, \\ Attention(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) &= \boldsymbol{V} \cdot Softmax(\frac{\boldsymbol{Q}\cdot\boldsymbol{K}^T}{\sqrt{C}}), \end{aligned} \tag{3}$$

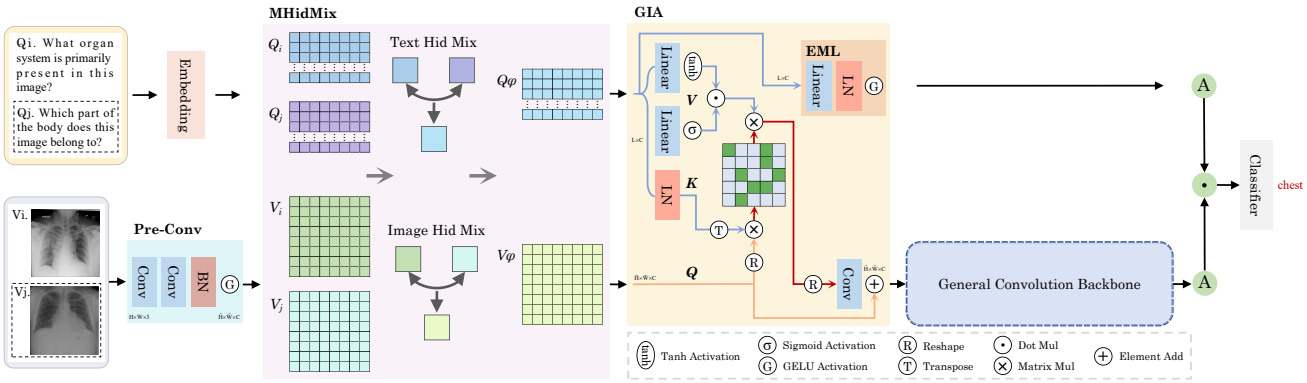

Fig. 2: The MedVQA framework utilizes a general convolutional model as the backbone. "Pre-Conv" denotes the pre-convolution module, "MHidMix" denotes the multi-modal hidden mixup, and "GIA" denotes the global interaction attention.

where W_3 represents the application of a 3×3 convolutional operation to aggregate features after the interaction between the $\boldsymbol{V}$ and the attention map $\boldsymbol{A} \in \mathbb{R}^{\hat{H}\hat{W} \times L}$. In traditional Self-Attention (SA) [7], [20], the time and memory demands of the key-query dot-product interaction increase quadratically with the spatial resolution of the input, specifically O(H^2W^2) for images with dimensions H $\times$ W pixels. The innovation in our GIA lies in its application of SA across channels, rather than spatial dimensions. As a result, the computational complexity of GIA is only linear. This interaction facilitates the fusion of image features with the text embedding, resulting in a more comprehensive representation. The image features that have been influenced by the text are passed into the convolutional backbone for further processing and analysis. Since the convolutional backbone involves various normalization and GELU activation operations to enhance model convergence, we apply similar normalization and activation functions to the $\boldsymbol{T}$. This ensures consistent treatment of both image and text features, promoting better convergence and model performance. The EML shown in Figure 2 signifies extended text embedding mapping. The entire processes are as follows:

$$\begin{aligned} \hat{\boldsymbol{I}} &= ConvolutionalBackbone(\boldsymbol{I}), \\ \hat{\boldsymbol{T}} &= \phi(LN(W(\boldsymbol{T}))), \end{aligned} \tag{4}$$

D. Final Dot-project

In the final phase of our framework, we employ average pooling and dot multiplication to combine the visual features extracted by the convolutional backbone with the text features obtained from the EML. This dot-multiplication operation enables meaningful interaction between the visual and textual modalities, and these interaction features are then passed through a classifier consisting of a single layer of Linear and LayerNorm, resulting in precise results. This highlights the importance of the dot-multiplication of the final features.

III. Experiment

We evaluate the effectiveness of our proposed framework on four diverse datasets: VQA-RAD [11], the English version of SLAKE [12], PathVQA [13], and OVQA [14].

A. Experimental Setups

The experiments were conducted on a single NVIDIA Tesla P100 16GB GPU. We utilized the AdamW optimizer with hyperparameters $\beta 1 = 0.9$ and $\beta 2 = 0.999$, along with an initial learning rate of 1e-4. Accuracy was chosen as the evaluation metric, consistent with other MedVQA evaluation methods, to assess the performance of the model. When incorporating MHidMix, the model was trained using a mixing coefficient λ sampled from a $Beta(3, 1)$ distribution.

B. Performance Comparison

As shown in Table I, we begin by presenting the state-of-the-art baselines achieved in the field. The results include MFB, SAN, and BAN, which are obtained through direct training on the MedVQA dataset. M-Mixup and VQAMix outperform traditional methods by leveraging multimodal data augmentation. Other models [7], [20]–[22], [24] undergo pre-training on large-scale external medical datasets and subsequently evaluate their performance on the MedVQA task.

General Convolution. Table II presents the comprehensive experimental results for various general convolutional models, including ResNet-50, and others. Our designed framework showcases remarkable effectiveness even without utilizing MHidMix. For example, RegNet achieves an impressive performance of 56.9% on PathVQA, while GoogleNet achieves an outstanding accuracy of 69.7% on OVQA, surpassing the performance of previous baseline models. With the incorporation of MHidMix data augmentation, ResNet-50 achieves a performance of 83.4% on the SLAKE dataset, outperforming the previous large-scale pre-training model M3AE by 0.2%.

Lightweight Convolution. In our experiments, we assessed the performance of lightweight convolutional models, specifically MobileNetV2, ShuffleNetV2, and GhostNet. The results displayed in Table II highlight the promising performance of these lightweight models in the MedVQA task. Although they may not surpass the general convolutional models, they outperform traditional VQA models like MFB, SAN, and BAN. When integrated with MHidMix, the performance of the lightweight models is substantially enhanced, underscoring the effectiveness of our framework in improving their capabilities.

TABLE I: Results of other baselines on different datasets, with "OP" indicating the open category, "CL" for the closed category, and "ALL" representing overall performance. #Time signifies the time (second) needed for the model to make an inference on one sample. "↑" denotes performance based on the transformer model with a very large number of parameters.

Method	#Params (M)	#Time (s/sample)	VQA-RAD [11] OP(%)	VQA-RAD [11] CL(%)	VQA-RAD [11] ALL(%)	SLAKE [12] OP(%)	SLAKE [12] CL(%)	SLAKE [12] ALL(%)	PathVQA [13] OP(%)	PathVQA [13] CL(%)	PathVQA [13] ALL(%)	OVQA [14] OP(%)	OVQA [14] CL(%)	OVQA [14] ALL(%)
ResNet50 + MFB [2]	50.0	0.141	14.5	74.3	50.6	72.2	75.0	73.0	-	-	-	-	-	-
ResNet50 + SAN [3]	42.4	0.134	31.3	69.5	54.3	74.0	79.1	76.0	1.6	59.4	30.5	-	-	-
ResNet50 + BAN [4]	48.5	0.143	37.4	72.1	58.3	74.6	79.1	76.3	2.9	68.2	35.6	-	-	-
M-Mixup [18]	28.6	0.132	53.1	81.3	70.2	79.9	87.5	82.9	-	-	-	-	-	-
VQAMix [17]	27.9	0.123	56.6	79.6	70.4	-	-	-	**13.4**	83.5	48.6	-	-	-
MEVF + BAN [21]	27.9	0.123	49.2	77.2	66.1	77.8	79.8	78.6	8.1	81.4	44.8	36.3	76.3	60.4
MMQ [22]	57.9	0.148	53.7	75.8	67.0	-	-	-	**13.4**	**84.0**	**48.8**	**56.9**	76.2	**68.5**
BAN + CR [23]	53.8	0.178	60.0	79.3	71.6	77.8	79.8	78.6	-	-	-	52.6	77.7	67.7
CPRD + BAN [24]	25.8	-	52.5	77.9	67.8	79.5	83.4	81.1	-	-	-	-	-	-
MQAT [6] ↑	94.5	0.884	49.8	76.3	65.7	79.7	87.7	82.8	-	-	-	-	-	-
MMBert [7] ↑	109.2	0.941	63.1	77.9	72.0	-	-	-	-	-	-	37.9	**80.2**	63.3
M3AE [20] ↑	148.6	1.031	**67.2**	**83.4**	**77.0**	**80.3**	**87.8**	**83.2**	-	-	-	-	-	-

TABLE II: Performance of our framework on general and lightweight convolutional models. White rows represent results with the base convolutional backbone, gray rows show results with MHidMix applied. SOTA models without MHidMix are highlighted in green, while those with MHidMix are in blue.

Model	#Params (M)	#Time (s/sample)	VQA-RAD OP(%)	VQA-RAD CL(%)	VQA-RAD ALL(%)	SLAKE OP(%)	SLAKE CL(%)	SLAKE ALL(%)	PathVQA OP(%)	PathVQA CL(%)	PathVQA ALL(%)	OVQA OP(%)	OVQA CL(%)	OVQA ALL(%)
VGG-19 [25]	40.1	0.042	34.7	75.2	59.2	77.8	84.7	80.5	26.1	84.8	55.6	41.1	79.0	63.9
			52.6 + *17.9*	78.5 + *3.3*	68.3 + *9.1*	80.5 + *2.7*	86.8 + *2.1*	83.0 + *2.5*	27.5 + *1.4*	85.1 + *0.3*	56.4 + *0.8*	53.1 + *12.0*	83.5 + *4.5*	71.4 + *7.5*
GoogleNet [26]	6.6	0.102	52.0	75.5	66.2	78.5	83.7	80.6	28.2	85.0	56.7	52.6	81.0	69.7
			57.6 + *5.6*	77.0 + *1.5*	69.3 + *3.1*	79.3 + *0.8*	88.5 + *4.8*	82.9 + *2.3*	31.3 + *3.1*	85.0 + *0*	58.2 + *1.5*	55.5 + *2.9*	82.5 + *1.5*	71.7 + *2.0*
ResNet-50 [27]	24.2	0.093	44.2	78.5	65.0	79.3	85.6	81.8	27.6	85.3	56.5	47.7	80.4	67.3
			58.7 + *14.5*	78.5 + *0*	70.7 + *5.7*	80.4 + *1.1*	88.1 + *2.5*	83.4 + *1.6*	29.9 + *2.3*	85.6 + *0.3*	57.8 + *1.3*	55.7 + *8.0*	80.7 + *0.3*	70.8 + *3.5*
DenseNet-169 [28]	13.1	0.096	46.4	77.4	65.2	77.9	85.4	80.8	26.2	85.2	55.8	48.1	80.6	67.6
			59.8 + *13.4*	76.0 - *1.4*	69.6 + *4.4*	80.4 + *2.5*	86.1 + *0.7*	82.6 + *1.8*	29.3 + *3.1*	85.2 + *0*	57.3 + *1.5*	57.4 + *9.3*	83.2 + *2.6*	73.0 + *5.4*
Regnet [29]	38.4	0.080	44.7	77.3	64.4	77.4	84.9	80.4	28.5	85.1	56.9	47.4	81.1	67.7
			48.1 + *3.4*	78.4 + *1.1*	66.4 + *2.0*	78.1 + *0.7*	88.5 + *3.6*	82.2 + *1.8*	27.9 - *0.6*	86.1 + *1.0*	57.1 + *0.2*	51.8 + *4.4*	80.6 - *0.5*	69.1 + *1.4*
EfficientNetV2 [30]	13.4	0.081	53.7	77.8	68.3	77.1	86.8	80.9	21.1	83.2	52.2	37.9	77.0	61.4
			59.3 + *5.6*	80.7 + *2.9*	72.2 + *3.9*	78.8 + *1.7*	87.1 + *0.3*	82.1 + *1.2*	27.4 + *6.3*	84.6 + *1.4*	56.0 + *3.8*	47.0 + *9.1*	79.0 + *2.0*	66.2 + *4.8*
EdgeNeXt [31]	18.3	0.061	44.7	78.9	65.4	78.1	86.6	81.4	26.0	83.6	54.9	51.1	80.0	68.5
			56.5 + *11.8*	79.2 + *0.3*	70.2 + *4.8*	79.3 + *1.2*	87.8 + *1.2*	82.6 + *1.2*	27.2 + *1.2*	85.2 + *1.6*	56.3 + *1.4*	57.8 + *6.7*	82.2 + *2.2*	72.5 + *4.0*
MobileNetV2 [32]	2.7	0.025	41.4	76.6	62.7	77.3	84.7	80.2	20.6	82.9	51.8	46.8	79.9	66.7
			50.9 + *9.5*	79.5 + *2.9*	68.2 + *5.5*	79.6 + *2.3*	85.1 + *0.4*	81.8 + *1.6*	23.7 + *3.1*	82.9 + *0*	53.4 + *1.6*	51.9 + *5.1*	81.8 + *1.9*	69.9 + *3.2*
ShuffleNetV2 [33]	1.6	0.018	48.7	77.3	66.0	77.8	86.1	81.0	26.8	84.4	55.7	48.6	79.6	67.3
			53.7 + *5.0*	79.2 + *1.9*	69.1 + *3.1*	78.7 + *0.9*	87.1 + *1.0*	82.0 + *1.0*	28.1 + *1.3*	84.4 + *0*	56.4 + *0.7*	52.3 + *3.7*	80.3 + *0.7*	69.1 + *1.8*
GhostNet [34]	4.5	0.028	48.7	77.3	66.0	76.5	85.1	79.9	19.2	83.1	51.2	44.0	79.9	65.6
			50.3 + *1.6*	79.2 + *1.9*	67.7 + *1.7*	77.8 + *1.3*	86.8 + *1.7*	81.3 + *1.4*	21.8 + *2.6*	84.5 + *1.4*	53.2 + *2.0*	47.3 + *3.3*	82.5 + *2.6*	68.4 + *2.8*

TABLE III: Parameter(M) Comparison. The first row represents the parameter count of the original convolution model, while the second row indicates the total parameter count of our designed backbone framework applied to the convolution model.

Params (M)	VGG-19	GoogleNet	ResNet-50	DenseNet-169	RegNet	EfficientNetV2	EdgeNeXt	MobileNetV2	ShuffleNetV2	GhostNet
Ori	39.3	6.4	23.7	12.6	38.0	13.1	18.1	2.4	1.4	4.2
Ours	39.8 + *0.5*	6.6 + *0.2*	24.2 + *0.5*	13.1 + *0.5*	38.4 + *0.4*	13.4 + *0.3*	18.3 + *0.2*	2.7 + *0.3*	1.6 + *0.2*	4.5 + *0.3*

Table II underscores the impressive performance of our designed framework with various convolutional backbones. As demonstrated in the parameter comparison presented in Table III, our designed framework adds minimal parameters, with an increase of less than 0.5 million parameters when compared to the original applications of convolutional models in image classification. In comparison to the ResNet-50 model employing fusion modules like MFB [2], SAN [3], and BAN [4] with a substantial parameter count, our ResNet-50 only introduces an additional half a million parameters. When compared to the pre-trained models like MEVF [21] and CPRD [24] that rely on image and text extractors along with fusion modules, our convolutional models significantly outperform them in terms of effectiveness.

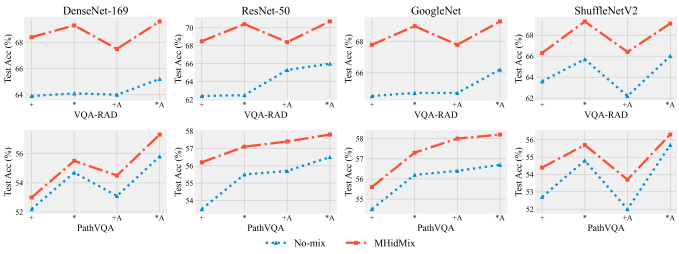

Fig. 3: Ablation of the backbone framework. The "+" operator represents element-wise addition, the "*" operator represents element-wise multiplication, and "A" indicates the usage of the GIA module to enhance the multimodal interaction process.

C. Backbone Ablation Analysis

To explore the potential of the backbone framework and MHidMix data augmentation, we conducted ablation experiments on popular convolutional models. The experiments were conducted on the VQA-RAD and PathVQA datasets, which are the most commonly used datasets in the MedVQA task. In the convolutional framework, we integrate three components: GIA, EML, and element-wise multiplication after the average pooling of final image and text features. EML is used only

TABLE IV: Model performance was assessed in two scenarios: (1) using only the original convolutional network (OI) for image feature extraction without text interaction, and (2) employing only the GIA module (OA) for text-image interaction.

Model	Method		VQA-RAD			PathVQA		
	OI	OA	OP(%)	CL(%)	ALL(%)	OP(%)	CL(%)	ALL(%)
DenseNet-169	✓		5.58	56.2	36.2	0.35	53.3	26.9
		✓	46.3	75.8	64.1	22.1	84.1	53.2
ResNet-50	✓		7.26	57.0	37.4	0.0	54.4	27.3
		✓	45.8	73.2	62.3	22.9	55.1	54.1
GoogleNet	✓		3.35	58.4	36.7	0.0	54.4	27.3
		✓	**47.0**	**77.3**	**65.3**	**24.1**	**84.3**	**54.3**
ShuffleNetV2	✓		5.03	58.8	37.6	0.0	54.3	27.3
		✓	**47.0**	71.5	61.8	23.4	82.9	53.2

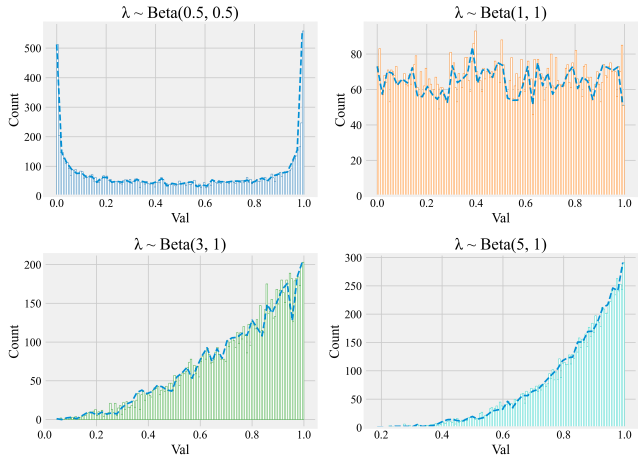

Fig. 4: Sample histograms show the distribution of randomly generated 10,000 points under different Beta distributions. The X-axis represents the 0-1 interval, and the Y-axis indicates the number of points corresponding to each subinterval.

when the final multimodal features require point multiplication or addition; otherwise, it is excluded.

Figure 3 shows the impact of element addition, element multiplication, and GIA. Element multiplication performs better than element addition, and GIA further improves the performance of the model by promoting text-image interaction. MHidMix consistently enhances the model performance in all cases. Table IV presents an analysis of the performance of the model without and with GIA to text-image interaction. The table reveals that the framework without text interaction performs poorly on the test set, relying solely on probability-based predictions. Incorporating GIA improves the training outcomes for the convolutional backbone. However, the impact of GIA alone is somewhat limited. The integration of the EML module and the final step of multi-modal feature point multiplication proves to be essential for further improvement.

D. MHidMix Ablation Analysis

To determine the optimal performance of MHidMix, we conducted ablation experiments using different Beta distributions. Notably, M-Mixup [18] has been shown to achieve the best data augmentation effect with a $Beta(5,1)$ distribution. In our experiments, we explored different Beta distributions,

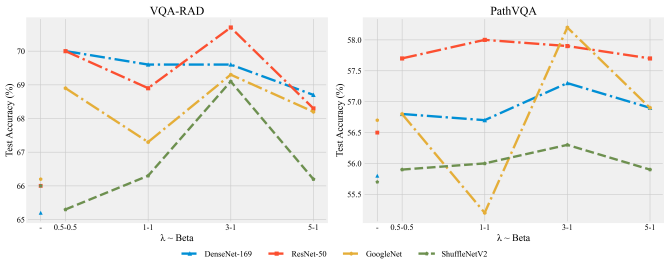

Fig. 5: The performance of different backbones is evaluated under various Beta distributions of MHidMix, where the "-" on the X-axis indicates the absence of MHidMix.

including $\lambda \sim Beta(0.5,0.5)$, $Beta(1,1)$, $Beta(3,1)$, and $Beta(5,1)$, to identify the best results of MHidMix. Figure 4 presents sample histograms depicting 10,000 randomly generated points in the 0-1 interval for each Beta distribution. Notably, when $\lambda \sim Beta(3,1)$, the generated points predominantly cluster between 0.6 and 1.

Figure 5 illustrates the impact of different Beta distributions on the performance of various convolutional backbone models when utilizing MHidMix. In most cases, the models exhibit positive performance when using different Beta distributions. However, there are instances where the performance is worse compared to not using MHidMix at all. The best performance is consistently achieved when the mixing coefficient λ follows the $Beta(3,1)$ distribution. This finding aligns with the observation made by M-Mixup, which suggests that when following the $Beta(0.5,0.5)$ or $Beta(1,1)$ distributions, the generated samples contain excessive content from both original samples, hindering the effective extraction of useful information. When following the $Beta(5,1)$ distribution, the generated samples exhibit relative simplicity, which limits the ability of the model to extract complex information and consequently leads to limited performance gains.

IV. Conclusion

In this paper, we present a novel framework that leverages convolutional models as the backbone to address challenges in MedVQA. Within this convolutional framework, we design the GIA module to effectively integrate text embeddings and image features before they enter the convolutional backbone. The processed image features are then extracted through the convolutional backbone, and the final output is generated by performing dot multiplication between the text information output by the EML module and the deeper visual features of the convolutional backbone. By applying MHidMix data augmentation within the convolutional framework, we mix the hidden states of paired image and text samples during training, significantly enhancing the model performance. These methods empower various convolutional models to excel across different datasets, demonstrating the effectiveness and generalization of this framework in the field of MedVQA.

Acknowledgments. This work was funded by National Natural Science Foundation of China (Grant No. 62366036), National Education Science Planning Project (Grant No.

BIX230343), Key R&D and Achievement Transformation Program of Inner Mongolia Autonomous Region (Grant No. 2022YFHH0077), The Central Government Fund for Promoting Local Scientific and Technological Development (Grant No. 2022ZY0198), Program for Young Talents of Science and Technology in Universities of Inner Mongolia Autonomous Region (Grant No. NJYT24033), Inner Mongolia Autonomous Region Science and Technology Planning Project (Grant No. 2023YFSH0017), Science and Technology Program of the Joint Fund of Scientific Research for the Public Hospitals of Inner Mongolia Academy of Medical Sciences (Grant No.2023GLLH0035), Supporting the Reform and Development of Local Universities (Disciplinary Construction) and the Special Research Project of First-Class Discipline of Inner Mongolia (Grant No. YLXKZX-ND-036).

REFERENCES

[1] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[2] Z. Yu, J. Yu, J. Fan, and D. Tao, "Multi-modal factorized bilinear pooling with co-attention learning for visual question answering," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1821–1830.

[3] Z. Yang, X. He, J. Gao, L. Deng, and A. Smola, "Stacked attention networks for image question answering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 21–29.

[4] J.-H. Kim, J. Jun, and B.-T. Zhang, "Bilinear attention networks," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[6] L. Liu, X. Su, H. Guo, and D. Zhu, "A transformer-based medical visual question answering model," in *2022 26th International Conference on Pattern Recognition (ICPR)*. IEEE, 2022, pp. 1712–1718.

[7] Y. Khare, V. Bagal, M. Mathew, A. Devi, U. D. Priyakumar, and C. Jawahar, "Mmbert: Multimodal bert pretraining for improved medical vqa," in *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2021, pp. 1033–1036.

[8] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 6848–6856.

[9] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[10] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio, "Manifold mixup: Better representations by interpolating hidden states," in *International conference on machine learning*. PMLR, 2019, pp. 6438–6447.

[11] J. J. Lau, S. Gayen, A. Ben Abacha, and D. Demner-Fushman, "A dataset of clinically generated visual questions and answers about radiology images," *Scientific data*, vol. 5, no. 1, pp. 1–10, 2018.

[12] B. Liu, L.-M. Zhan, L. Xu, L. Ma, Y. Yang, and X.-M. Wu, "Slake: A semantically-labeled knowledge-enhanced dataset for medical visual question answering," in *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2021, pp. 1650–1654.

[13] X. He, Y. Zhang, L. Mou, E. Xing, and P. Xie, "Pathvqa: 30000+ questions for medical visual question answering," *arXiv preprint arXiv:2003.10286*, 2020.

[14] Y. Huang, X. Wang, F. Liu, and G. Huang, "Ovqa: A clinically generated visual question answering dataset," in *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2022, pp. 2924–2938.

[15] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[16] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. PMLR, 2015, pp. 448–456.

[17] H. Gong, G. Chen, M. Mao, Z. Li, and G. Li, "Vqamix: Conditional triplet mixup for medical visual question answering," *IEEE Transactions on Medical Imaging*, vol. 41, no. 11, pp. 3332–3343, 2022.

[18] L. Liu and X. Su, "How well apply multimodal mixup and simple mlps backbone to medical visual question answering?" in *2022 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*. IEEE, 2022, pp. 2648–2655.

[19] X. Hao, Y. Zhu, S. Appalaraju, A. Zhang, W. Zhang, B. Li, and M. Li, "Mixgen: A new multi-modal data augmentation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2023, pp. 379–389.

[20] Z. Chen, Y. Du, J. Hu, Y. Liu, G. Li, X. Wan, and T.-H. Chang, "Multi-modal masked autoencoders for medical vision-and-language pre-training," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part V*. Springer, 2022, pp. 679–689.

[21] B. D. Nguyen, T.-T. Do, B. X. Nguyen, T. Do, E. Tjiputra, and Q. D. Tran, "Overcoming data limitation in medical visual question answering," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2019, pp. 522–530.

[22] T. Do, B. X. Nguyen, E. Tjiputra, M. Tran, Q. D. Tran, and A. Nguyen, "Multiple meta-model quantifying for medical visual question answering," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part V 24*. Springer, 2021, pp. 64–74.

[23] L.-M. Zhan, B. Liu, L. Fan, J. Chen, and X.-M. Wu, "Medical visual question answering via conditional reasoning," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 2345–2354.

[24] B. Liu, L.-M. Zhan, and X.-M. Wu, "Contrastive pre-training and representation distillation for medical visual question answering based on radiology images," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2021, pp. 210–220.

[25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[26] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[28] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[29] I. Radosavovic, R. P. Kosaraju, R. Girshick, K. He, and P. Dollár, "Designing network design spaces," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 10 428–10 436.

[30] M. Tan and Q. Le, "Efficientnetv2: Smaller models and faster training," in *International conference on machine learning*. PMLR, 2021, pp. 10 096–10 106.

[31] M. Maaz, A. Shaker, H. Cholakkal, S. Khan, S. W. Zamir, R. M. Anwer, and F. Shahbaz Khan, "Edgenext: efficiently amalgamated cnn-transformer architecture for mobile vision applications," in *European Conference on Computer Vision*. Springer, 2022, pp. 3–20.

[32] A. Howard, A. Zhmoginov, L.-C. Chen, M. Sandler, and M. Zhu, "Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation," 2018.

[33] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 116–131.

[34] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "Ghostnet: More features from cheap operations," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1580–1589.