Advance Machine Learning (WOA7015)

Alternative Assessment

2025/2026 Sem 1

A Comparative Study of CNN Baseline Classifier and LLM-Based Model for Medical Visual Question Answersing

Group ZZ (OCC1)

NAME	MATIRX NO.
ZHONG JUN PEI	24214748
ZOU TING	24201617

ABSTRACT

Medical visual question answering (Med-VQA) aims to support clinical decision-making by jointly understanding medical images and natural language questions. This study investigates both the performance and safety of Med-VQA models by comparing a traditional classification-based baseline with a generative vision–language model under a low-resource setting. Experiments are conducted on the VQA-RAD dataset, covering both closed-ended diagnostic questions and open-ended descriptive questions.

The baseline model formulates Med-VQA as a classification task using a CNN–LSTM architecture, while the generative approach adopts a vision–language model with frozen pre-trained backbones and a lightweight mapping network. Model performance is evaluated using standard accuracy metrics for closed-ended questions and multiple automatic metrics for open-ended generation, complemented by qualitative analysis.

Experimental results show that the generative vision–language model outperforms the CNN–LSTM baseline on closed-ended questions, despite the limited size of the training data. For open-ended questions, quantitative results indicate low scores under strict lexical matching metrics; however, qualitative inspection reveals that many low-scoring cases are semantically correct and differ from the references only in wording or level of detail.

Overall, this study demonstrates that pre-trained vision–language representations provide stronger generalization than models trained from scratch in data-scarce medical settings. It also highlights the limitations of current automatic evaluation metrics in assessing the safety and clinical reliability of

generative Med-VQA systems. These findings suggest that future evaluations should combine quantitative metrics with qualitative analysis to better reflect real-world clinical applicability.

Word Count: 238 words

Keywords: Visual Question Answering · CNN · LSTM · Language Models · Prefix Tuning.

1. INTRODUCTION

1.1. Research Context and Problem Statement

Medical AI is advancing rapidly, and both patients and clinicians increasingly need systems that can interpret medical images to support diagnosis and decision-making(Hartsock & Rasool, 2024). While Visual Question Answering (VQA) has shown promise in general domains, applying it to radiology presents unique challenges, particularly regarding clinical reliability and safety. Our study focuses on quantitatively evaluating VQA models on both closed-ended and open-ended tasks. This evaluation is essential to ensure AI systems provide accurate answers while minimizing the risk of hallucination in high-stakes medical settings.

1.2. Research Questions and Objectives

We address two main research questions:

RQ1 (Closed-Ended Performance): Which model that we choose to compare performs better on closed-ended questions in radiology and how large is the performance gap?

RQ2 (Open-Ended Quality and Safety): How well do generative VLMs perform on open-ended medical questions, and what is the risk of hallucinated answers?

And establish two key objective for research questions:

Objective 1: To evaluate the traditional CNN-based baseline and a generative Vision-Language Model (VLM) using accuracy metric for closed-end questions and critically discuss the gap of these two model.

Objective 2: To evaluate answer quality using automated language similarity metrics such as BLEU-1, BLEU-4, BERTScore and qualitative analysis and compare the results with benchmarks.

1.3. Methodology Overview

We conducted experiments on the VQA-RAD dataset(Lau et al., 2018), a clinically verified benchmark of radiology images and question-answer pairs. We focus on their ability to handle the strict logic of closed-ended questions versus the generative demands of open-ended medical queries. Two model architectures were compared:

Baseline Method: A traditional model using RestNet-50 to extract visual features and an LSTM for text processing(Eldin & Kaboudan, 2023). It represents a standard classification approach.

Generative Model Approach: A generative VLM framework based on Van Sonsbeek et al.(Eldin & Kaboudan, 2023) with some adjustments. It uses a frozen CLIP encoder for vision, a frozen GPT-2 decoder for language, and a trainable lightweight mapping network connecting them.

2. METHODS

This section outlines the experimental setup used in this research to establish a baseline and a generative model for medical visual question answering (Med-VQA). The details provided include dataset preparation, model architecture, training procedures, and evaluation metrics.

2.1. Dataset

In this study, the VQA-RAD dataset were used for medical visual question answering, and the statistic feature of this dataset is shown in **Error! Reference source not found.**. VQA-RAD consists of 315 r adiology images such as X-rays and CT scans and 2,248 clinician-verified question–answer (QA) pairs. Compared with datasets that require external medical knowledge such as SLAKE(Liu et al., 2021), VQA-RAD focuses on pure visual questions, where answers can be inferred directly from the image content.

This design makes the dataset suitable for evaluating the visual representation learning and vision–language alignment capabilities of the model.

The questions include both closed-ended and open-ended types. As shown in Table 1, Closed-ended questions account for 57.7% of the dataset, while open-ended questions account for 42.3%. The questions cover multiple medical aspects, such as abnormality presence, anatomical location, and imaging modality. Notably, approximately 19.6% of the questions are duplicates, which requires careful evaluation to reduce potential textual bias.

Table 1 Statistics of VQA-RAD dataset

Item	**Description**
Number of images	315
Image modalities	X-ray, CT
Number of QA pairs	2,248
Closed-ended questions	57.7%
Open-ended questions	42.3%
Duplicate questions	19.6%

2.2. Baseline Model Architecture

This section will explain the architecture diagram of the baseline system, which is divided into four modules: Data Preprocessing, Data Transformation and Loading, Model Building, and Training Pipeline. Figure 1shows schematic diagram for specific functions. The following sections provide explanations for each individual module and the final training model.

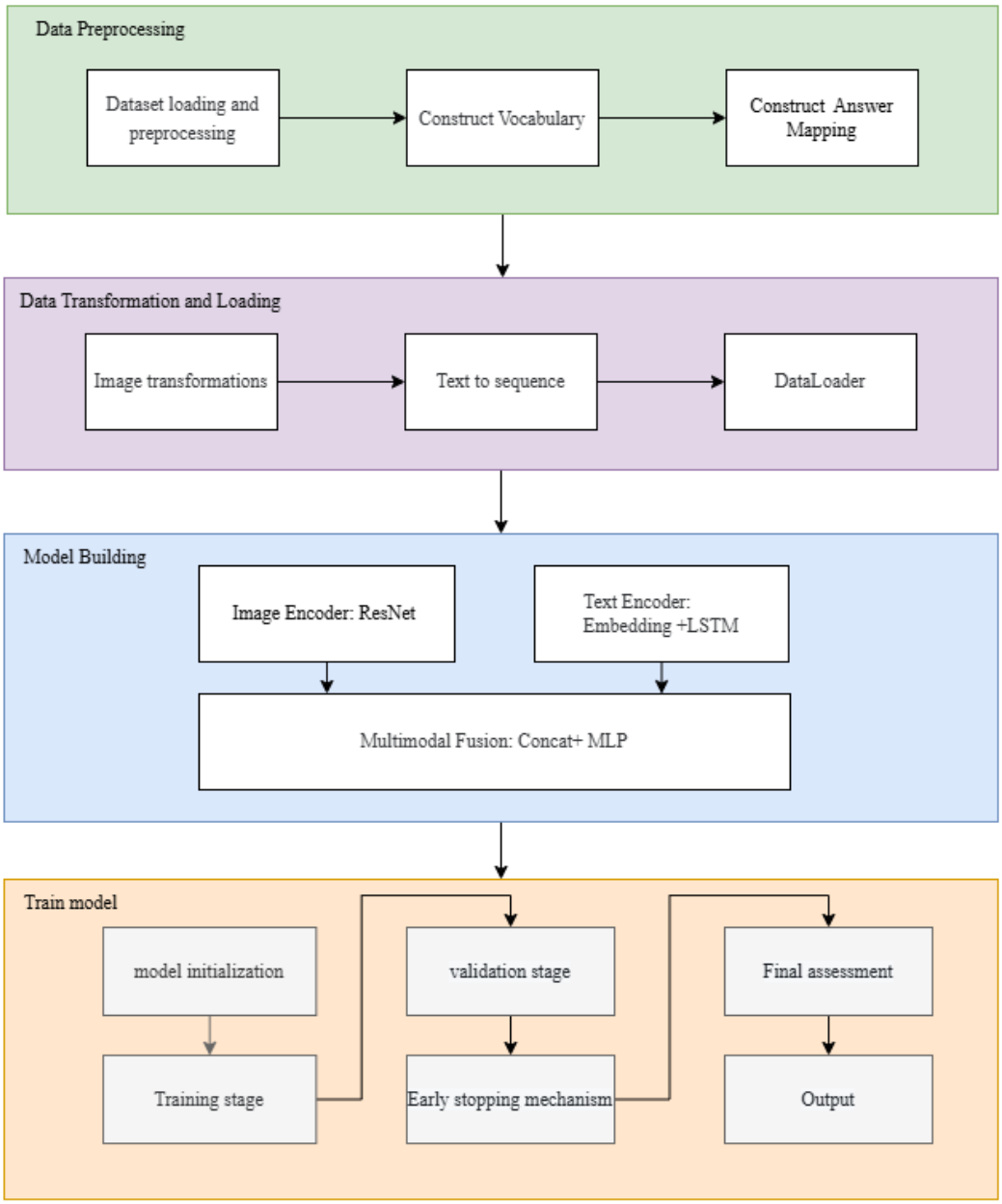

Figure 1 the architecture diagram of the baseline system

A. Data Preprocessing

Dataset preprocessing is divided into three steps: dataset loading and preprocessing, vocabulary construction, answer mapping construction.

First, load and preprocess the dataset. Read the data from the original file VQA_RAD Dataset Public.json, which is in JSON format. Each sample contains fields such as image_name (image file name), question

(questions related to medical imaging), answer (corresponding answer text), and answer_type (question type, such as OPEN/CLOSED).

During the data cleaning stage, check and remove the samples with missing key fields including image_id, question and answer. After statistics, there are no missing samples in the current dataset. Subsequently, the dataset was randomly divided into the training set (70%), validation set (15%) and test set (15%), with the random seed fixed at 42.

The distribution of question types in each dataset after division is shown in the Table 2.

Table 2 Question Type Distribution

Question Type	**Train**	**Validation**	**Test**
CLOSED	920 (58.4%)	195 (57.9%)	184 (54.6%)
OPEN	654 (41.6%)	142 (42.1%)	153 (45.4%)

The purpose of vocabulary construction is to convert text-based questions into numerical sequences that machines can process, providing standardized input of fixed dimensions for the model. The vocabulary list is constructed solely based on the question texts in the training set. During the construction process, first uniformly convert the problem text to lowercase and separate words by spaces. At the same time, remove punctuation marks and set the minimum word frequency threshold min_freq=2. Low-frequency words below this threshold will be regarded as unregistered words. In addition, add the following special tags: <PAD> (filler, ID: 0), <UNK> (unlogged word, ID: 1), <SOS> (sentence beginning tag, ID: 2), and <EOS> (sentence end tag, ID: 3). After statistics, the total number of words in the training set after removing duplicates is 2,158, and the size of the final constructed vocabulary list (including special tags) is 1,284. The statistical results of some high-frequency words are shown in the following

Table 3.

Table 3 Top 5 high-frequency words

Rank	Word	Occurrences
1	the	1,216
2	is	1,048
3	what	404
4	this	383
5	in	337

The purpose of constructing the answer mapping table is to convert text answers into category labels in classification tasks, achieving standardized mapping from open-domain answers to fixed category spaces. To prevent data leakage, the mapping table is constructed only based on the answers in the training set, treating each unique answer as an independent category. Since the test set may contain answers that do not appear in the training set, an additional <UNK> (unknown) category is added to the mapping table to uniformly map these unseen answers, thereby ensuring that all samples have valid labels and truly reflect the model's ability to handle new answers. The statistical results show that the number of unique answers in the training set is 1,412, and the distribution of answers presents a distinct long-tail characteristic with a high degree of category imbalance. The statistics of high-frequency answers are shown in Table 4.

Table 4 Top 5 high-frequency answers

Rank	**Answer**	**Occurrences**
1	No	335
2	Yes	262
3	yes	168
4	no	81
5	Axial	19

B. Data Transformation and Loading

Data Transformation consists of two parts: Image transformations and text serialization.

Image transformations process first uses resize(224, 224) to unify the image size, and then normalizes with mean (0.485, 0.456, 0.406) and standard deviation (0.229, 0.224, 0.225) to convert the image pixel values to a nearly normal distribution. Finally, a normalized image tensor with a structure of (3, 224, 224) is obtained.

Text serialization is only applied to problem text. Set the maximum sequence length max_seq_len=25 to unify the lengths of all questions: sequences shorter than 25 words are filled with <PAD>, and sequences longer than 25 words are truncated with the first 25 words. Each word is mapped to the corresponding ID according to the vocabulary list. Words that do not appear in the vocabulary list are mapped to <UNK> (ID: 1). The retention of the first 25 words during truncation is based on the assumption that key information in medical problems is usually located at the beginning. The answers in the dataset have been converted into individual category ids through the answer mapping table and can be directly used as classification labels without the need for serialization processing.

After the data conversion is completed, an image existence check will be conducted. If the image is found to be missing, replace it with a zero tensor and log it. At the same time, normalize the answer type to CLOSED or OPEN.

DataLoader converts the original dataset into a batch iterable data stream, providing standardized and efficient data supply for model training. The configuration of the baseline model is as follows: the number of samples in each batch is 32, and the number of loading processes is 2 (when using a GPU). Among them, the shuffle parameter of the training dataset is set to true to ensure that the data order is randomly shuffled in each training epoch to prevent overfitting, set the shuffle parameter of the validation dataset

and the test dataset to false to facilitate tracking of model performance and ensure the reproducibility of results.

C. Model Building(MedVQA_ResNet_LSTM)

The overall architecture of MedVQA_ResNet_LSTM is a dual-stream fusion network: it uses ResNet50 as the image encoder for visual feature extraction, Embedding + LSTM as the text encoder for semantic feature extraction, and finally completes multimodal decision-making through the MLP fusion classifier. ResNet is a deep convolutional neural network specifically designed for image data processing, with its primary task being to extract features from images. After preprocessing the input image of VQA-RAD, ResNet extracts the high-level image features, and the output high-dimensional features will be transferred to the fusion module.

LSTM is a recurrent neural network (RNN) used to handle sequential data, such as text questions (for example, "What disease is this?"). And extract semantic features from it.

The text input into LSTM first converts the words into vector representations through the embedding layer and processes them into fixed-length sequences.

After the image and text features are extracted respectively, they enter the feature fusion module. The fusion process is divided into two steps: First, perform combination, directly combine the image features and text features into a joint feature vector, thereby increasing the feature dimension; Then, MLP (Multi-Layer Perceptron) is used for nonlinear transformation. Through several fully connected layers, high-order feature extraction is carried out on the concatenated joint features, and finally the classification result is output. The above is the architecture design of the baseline model.

D. Training Pipeline

The training and validation of the model are implemented based on the PyTorch framework and adopt the classic training-validation iterative strategy. After the initialization is completed, in each training epoch,

the model first receives the training data for forward propagation to calculate the prediction result, then uses the Cross Entropy Loss function to calculate the prediction error, and updates the model parameters through error backpropagation. Finally, evaluate the model on the validation set and record the losses and other performance metrics. To prevent the model from overfitting during the training process, we introduced the early stopping mechanism. During the validation phase, the performance of the model is judged by monitoring the validation loss: if the validation loss does not improve for five consecutive cycles (with the patience value set to 5), the training will automatically stop, thereby reducing the training time and improving the generalization ability of the model. The objective of training is to enhance the performance of the validation set while reducing its loss. After each round of training, the system will save the weights of the model with the best current performance to ensure that the model with the best effect can be selected in the end. Figure 2 shows the detailed process of the training-validation iteration strategy and the early stop mechanism.

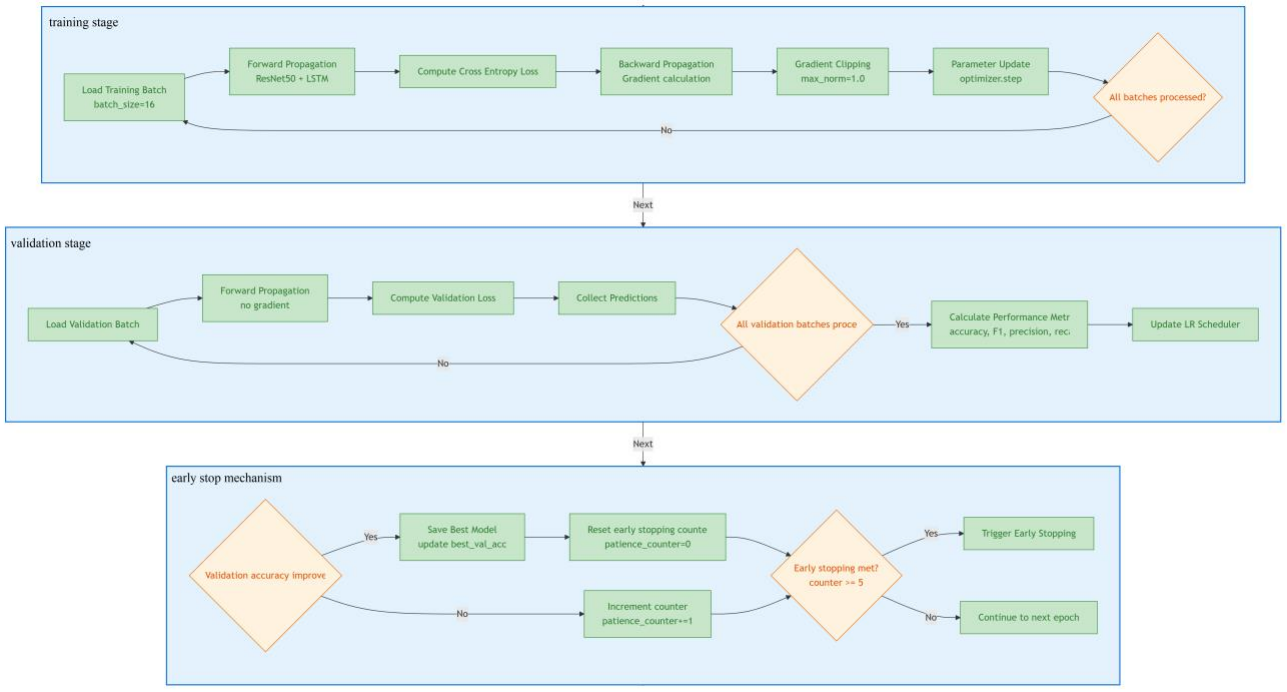

Figure 2 Process of the training-validation iteration strategy and the early stop mechanism

During training, we set the key hyperparameters as summarized in Table 5. A batch size of 16 balances training efficiency and GPU memory usage. The initial learning rate is 0.0001, adjusted dynamically using a ReduceLROnPlateau scheduler based on validation performance. The models are trained for 30 epochs. To evaluate model performance, we use cross-entropy loss as the primary optimization objective, measuring the difference between predicted and true labels. Classification performance is assessed with accuracy, while F1-Score and precision provide a more comprehensive evaluation, especially on imbalanced datasets.

Table 5 Training Hyperparameters and Evaluation Metrics

Category	Parameter / Metric	Details / Description
Hyperparameters	Batch Size	16 (balances training efficiency and GPU memory usage)
	Learning Rate	0.0001 (adjusted dynamically via ReduceLROnPlateau based on validation performance)
	Epochs	35
Evaluation Metrics	Cross-Entropy Loss	Measures the difference between model predictions and true labels
	Accuracy	Classification performance on the validation set
	F1-Score & Precision	Comprehensive metrics for classification, especially on imbalanced datasets

2.3. Generative Model(Sonsbeek et al.) Architecture

Our model architecture follows the prefix-based vision–language framework proposed by Sonsbeek et al. (Sonsbeek et al., 2023) for medical visual question answering. The model consists of three main components: a pre-trained CLIP image encoder, a pre-trained GPT-2 language model, and a lightweight trainable mapping network that aligns visual and textual representations.

In this framework, a medical image is first encoded by a frozen CLIP image encoder (ViT-B/32) to obtain a global visual feature. This visual embedding is then projected by a mapping network, composed of two fully connected layers with a ReLU activation function, into a sequence of visual prefix tokens. The generated visual prefixes are concatenated with the token embeddings of the question–answer prompt and jointly fed into the GPT-2 model for autoregressive answer generation.

Consistent with the design in Sonsbeek et al., both the CLIP encoder and the GPT-2 model are kept frozen during training to reduce computational complexity and mitigate overfitting on the small-scale VQA-RAD dataset. Only the parameters of the mapping network are optimized using a causal language modeling objective. The loss is computed exclusively on the textual tokens, while the visual prefix tokens are masked, encouraging the model to learn effective cross-modal alignment without disrupting the pre-trained language modeling capability.

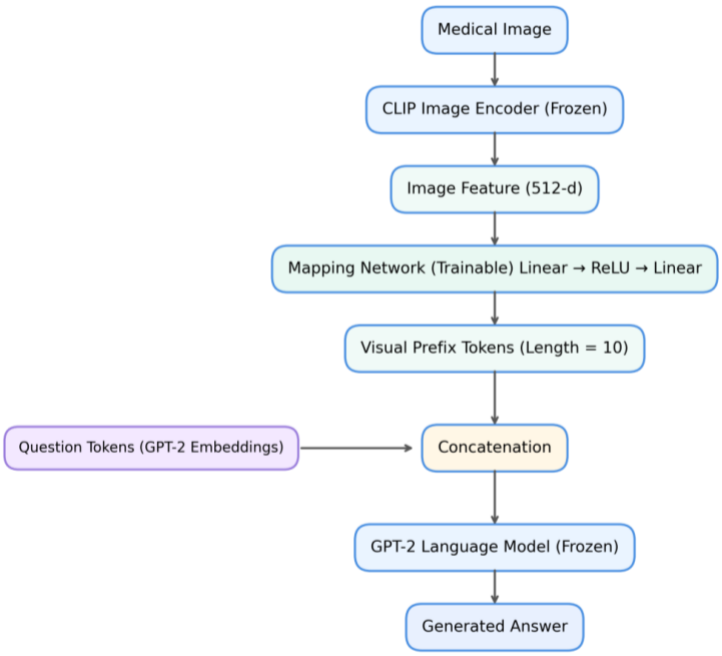

Figure 3 Overview of the prefix-based MedVQA architecture adopted in this study, following the framework proposed by Sonsbeek et al. (Sonsbeek et al., 2023).

As shown in Figure 3, the MedVQAModel in this study comprises three main components:

a. CLIP Image Encoder (frozen) : outputs 512-d image feature vectors.
b. Mapping Network (trainable) : two-layer linear network with ReLU activation projecting CLIP features to GPT-2 embedding space, generating 10 visual prefix tokens.
c. GPT-2 Language Model (frozen) : receives concatenated visual prefixes and question embeddings for answer generation.

Consistent with the design in Sonsbeek et al.**,** both the CLIP encoder and the GPT-2 model are kept frozen during training to reduce computational complexity and mitigate overfitting on the small-scale VQA-RAD dataset. Only the parameters of the mapping network are optimized using a causal language modeling objective. The loss is computed exclusively on the textual tokens, while the visual prefix tokens are masked, encouraging the model to learn effective cross-modal alignment without disrupting the pre-trained language modeling capability.

For model training and evaluation, the dataset was randomly split into training and test sets with a ratio of 80:20. All images were resized to 224×224pixels and normalized to match the input requirements of the pre-trained CLIP image encoder. Text data were tokenized and formatted into a causal language modeling prompt such as *Question: [Q] Answer: [A]* using function *MedVQADataset()* to support generative learning. The performance of this model will be assessed model performance using two metric:

a. BLEU-1 (Papineni et al., 2001)for open-ended questions to quantify n-gram overlap with the reference.
b. BERTScore-F1(Zhang et al., 2020) using PubMedBERT to capture semantic similarity of generated vs reference answers.

The meanings of this two metrics will be explained in Section 2.4 and the performance will be discussed in the Section 3.1 using these evaluation metric, similarly Section 3.1 will compare our results with Sonsbeek et al. and explain the possible reason for the differences in our experiments.

2.4. Evaluation Metrics

To ensure a rigorous and fair assessment of model performance on the VQA-RAD dataset, this study adopts a set of evaluation metrics that account for both closed-ended and open-ended question types. Due to the diverse nature of Med-VQA tasks, ranging from simple diagnostic decisions to detailed clinical descriptions, no single metric suffices. Instead, a combination of classification-based and generation-oriented metrics is employed, following established practices in prior Med-VQA literature.

Closed-ended questions, where answers come from a fixed set like abnormality presence, are evaluated primarily using accuracy. Since the baseline model treats Med-VQA as a classification task over a fixed set of answers, accuracy directly reflects how often the model gives the correct diagnosis. It is also the most commonly reported metric in VQA-RAD benchmarks, enabling meaningful comparison with existing studies.

To complement accuracy, threshold-independent diagnostic discrimination metrics are additionally considered. AUC-ROC is used to assess the model's ability to distinguish between positive and negative clinical conditions across varying decision thresholds, while AUC-PRC offers further insight under class-imbalanced settings, where positive cases are relatively rare. Although these metrics are particularly informative for binary disease detection tasks, they are treated as supporting indicators in this study. The primary quantitative comparison for closed-ended evaluation remains accuracy, in order to maintain consistency with the evaluation of generative models.

In contrast, evaluating open-ended questions presents a fundamentally different challenge. Clinically correct responses may be expressed using diverse yet equally valid linguistic forms, making strict label

matching insufficient. To address this, a multi-dimensional evaluation strategy is adopted to capture both diagnostic correctness and linguistic quality.

For short, determinate answers such as anatomical locations or categorical descriptors, **exact** match accuracy is employed. In addition to strict matching, a *soft exact match* variant is applied by normalizing generated outputs through lowercasing and punctuation removal. This approach avoids penalizing superficial formatting variations while preserving strict clinical validity.

For longer, free-form responses, Bilingual Evaluation Understudy (BLEU) serves as the primary metric for quantitative comparison. By measuring n-gram overlap between generated answers and reference annotations, BLEU evaluates whether the model produces appropriate medical terminology and coherent phrasing. As such, it provides an effective measure of the expressive capability of generative architectures, particularly in comparison to rigid classification-based baselines.

To further assess semantic alignment and informational completeness, Recall and BERTScore are incorporated as complementary metrics. Recall measures the extent to which key diagnostic tokens from the ground truth are covered in the generated output, while BERTScore leverages contextual embeddings to capture semantic similarity and account for synonymous medical expressions such as lesion and tumor. These metrics provide deeper insight into semantic adequacy beyond surface-level lexical overlap.

While advanced semantic metrics offer valuable interpretative support, the core quantitative evaluation in this study focuses on accuracy for closed-ended questions and BLEU for open-ended responses. This choice ensures alignment with standard Med-VQA benchmarks and facilitates consistent comparison across different model paradigms.

Table 6 The metrics for evaluating the performance of generative Med-VQA model in benchmarks

Metric	Question Type	Evaluation Focus	Uesd By
Accuracy	Closed-ended and open-Ended	Diagnostic correctness	All work
AUC-ROC,AUC-PRC	Closed-ended	Diagnostic discrimination and robustness under class imbalance	(Sharma et al., 2021)
Recall	Open-Ended	Information integrity	(Li et al., 2023)
BLEU	Open-Ended	Linguistic quality	(Sonsbeek et al., 2023)
BERTScore	Open-Ended	Semantic similarity	(Sonsbeek et al., 2023)

Table 6 summarises the evaluation metrics adopted in this study and their corresponding application scenarios. Metrics are selected to reflect the heterogeneous nature of Med-VQA tasks, covering both classification-based diagnostic decisions and free-form clinical descriptions. While a diverse set of metrics is reported for completeness, the primary quantitative comparison focuses on accuracy for closed-ended questions and BLEU for open-ended responses, in alignment with standard evaluation practices in existing Med-VQA benchmarks.

3. RESULTS

3.1. Baseline Model Result

A. Core Parameters

In this experiment, we adopted a standard data partitioning strategy, dividing the VQA-RAD dataset into a training set (1,574 samples), a validation set (337 samples), and a test set (337 samples) in a ratio of 70%-15%-15% to ensure that the model could achieve reliable generalization performance evaluations based on sufficient training data.

The experiment was conducted in the T4 GPU environment on the Google Colab platform, which provided 15.0GB of GPU memory and 12.7GB of system memory, fully meeting the resource requirements for model training. The core parameters of baseline model are shown in Table 7.

Table 7 Core training parameters of baseline model

Parameter	**Value**	**Description**
Epochs	35	Total number of training cycles
Batch Size	16	Number of samples processed in each batch
Optimizer	Adam	Optimization algorithm for weight updates
Learning Rate	10^(-4)	Learning rate controlling the step size of updates
Dropout	0.3	Dropout rate used to prevent overfitting

B. Overall performance

Figure 4 shows the trend that the loss value of the model continuously drops during the training process, while the training accuracy and validation F1 score gradually increase to a stable high level with each training round. This indicates that as the training progresses, both the fitting ability and generalization performance of the model have been effectively improved and optimized.

Figure 4 Loss Curve / Accuracy /F1 score Comprarison of Baseline Classifier

The model's training accuracy rate is 78.65% in Table 8, proving that the baseline architecture has the basic ability to handle medical VQA tasks. The training process converges stably, and the training loss is controlled at a low level of 0.6623, indicating that the optimization algorithm configuration is reasonable and the model can effectively extract features from the training data. However, the model exhibited a severe overfitting phenomenon, with a training-test accuracy gap as high as 44.82%, verifying that its generalization ability on unseen data was significantly insufficient. All kinds of evaluation indicators are generally low (F1 score 13.64%, accuracy 33.83%, recall 13.27%), reflecting a serious category imbalance problem that affects the prediction quality.

Table 8 Core training parameters of baseline model

Metric	Training Set	Validation Set	Test Set
Accuracy	78.65%	44.21%	33.83%
F1 Score	-	16.21%	13.64%
Precision	-	16.04%	14.54%
Recall	-	-	13.27%
Loss	0.6623	3.8041	-

Figure 5 shows that the model has the highest accuracy rate (48%) on CLOSED problems, but the overall performance indicators (F1, precision, recall) are all at a relatively low level (about 13%-14.5%). This indicates that the model performs slightly better in judging simple problems, but its overall performance in comprehensive evaluations is poor, and its generalization ability is relatively low.

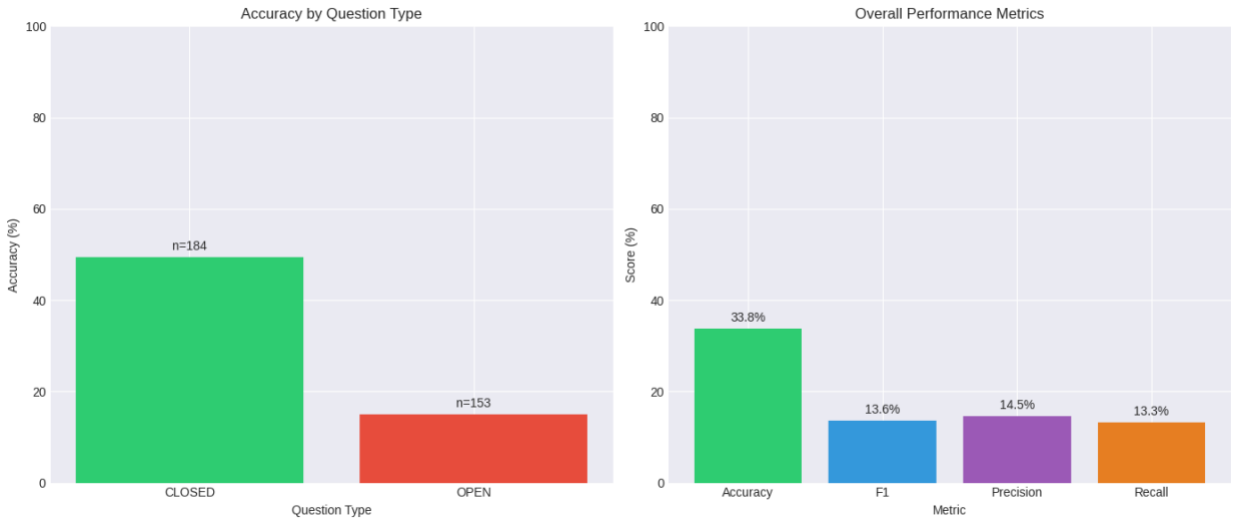

Figure 5 Accuracy by question type and overall performance on test dataset

3.2. Generative Model (Sonsbeek et al.) Result

The training loss shows a rapid decline in the initial stage in Figure 6, with a particularly steep drop from epoch 0 to epoch 1. This is expected given that the vision encoder (CLIP) and the language model (GPT-2) are pre-trained and frozen, providing strong feature extraction and language generation capabilities. The mapping network only needs to learn a simple linear transformation to align visual and textual features, allowing the model to quickly converge toward a reasonable solution.

After approximately Epoch 5, the loss curve enters a plateau, stabilizing around 0.3 with minimal fluctuations. This smooth progression indicates that the learning rate (1e-4) and the AdamW optimizer are appropriately set, avoiding gradient instability. The early stabilization also reflects the relatively small

size of the VQA-RAD dataset, enabling the model to rapidly learn the mapping rules. The mapping network's limited capacity, together with the frozen backbone, constrains the model from capturing more complex patterns beyond these simple alignments.

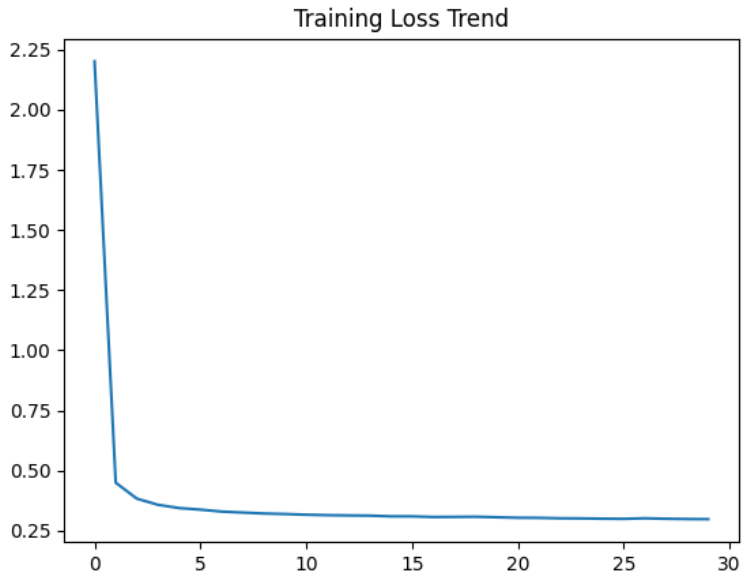

Figure 6 Training loss curve of generatic model

Next, we will analyze the results of the experiment. The following table shows the statistics of the results. For closed-end problems, the accuracy metric is used for evaluation, and for open-end problems, the BULE and BERTScore are used for analysis.

Table 9 The value of evaluation metrics for closed-end questions and open-end questions

Question Type	**Accuracy**	**BLEU-1**	**BERTScore (F1)**
Closed	63.03%	–	–
Open	–	0.084	0.911
Overall	36.00%	–	–

The evaluation of our model on the VQA-RAD dataset reveals a distinctive pattern across closed-ended and open-ended questions. Closed-ended (Yes/No) questions achieve a moderate accuracy of 63%, indicating that the model can reliably capture the presence or absence of specific findings in radiology images. In contrast, open-ended questions exhibit extremely low Exact Match (5.7%) and BLEU-1 (0.084), while maintaining a high BERTScore F1 of 0.91. This divergence highlights a critical distinction between surface-form metrics and semantic alignment in evaluating medical VQA.

Several implementation differences relative to prior approaches, such as those reported by Sonsbeek et al. (2023), can explain this outcome. First, the prefix tuning configuration in our study uses longer and fixed-length sequences (lx = 10, lq/la = 128) compared with shorter or dynamically calculated prefixes in prior work. Longer prefixes can dilute the injection of visual information into the language model, particularly affecting the generation of detailed open-ended answers, while having limited impact on binary closed-ended tasks.

Second, the linguistic model used here is GPT2-base rather than the larger GPT2-XL or BioMedLM models. The reduced parameter count and limited medical vocabulary restrict the model's ability to produce precise lexical forms, contributing to low Exact Match and BLEU-1 scores. At the same time, semantic content is largely preserved, as evidenced by the high BERTScore.

Third, the language model is frozen rather than fine-tuning, which prevents adaptation to dataset-specific answer distributions. Combined with a smaller dataset (VQA-RAD only) and a lower learning rate optimizer without warmup, this further constrains surface-form accuracy while retaining semantic correctness.

These observations lead to several interesting findings:

a. High BERTScore-F1 indicates that the model captures the clinically relevant meaning of answers even when lexical forms differ from the reference.
b. BLEU is highly sensitive to token-level differences and short answer lengths, which can misrepresent model capability in medical open-ended tasks.
c. Prefix length, model capacity, frozen LM, and small dataset size collectively contribute to low surface-form scores, while closed-ended accuracy remains robust.

While a direct comparison to the baseline model is not yet available, these findings suggest that some metrics may underestimate the clinical reasoning capability of the model. The combination of closed-ended accuracy and semantic metrics such as BERTScore provides a more nuanced assessment of performance in medical VQA tasks.

4. DISCUSSION

4.1. RQ1: Performance Gap (Baseline vs VLM)

To address RQ1, we compared a traditional CNN–LSTM baseline with a generative Vision-Language Model (VLM) on closed-ended questions. The results demonstrate a clear performance advantage for the generative approach.

Quantitatively, the baseline achieved a test accuracy of approximately 48%, which is close to random guessing for binary classification tasks. In contrast, the generative VLM reached an accuracy of 63.03%, representing a notable improvement of around 15 percentage points in diagnostic correctness.

This gap can largely be attributed to differences in representation learning. The baseline model exhibited severe overfitting, with high training accuracy but substantially lower test performance, indicating limited generalization when trained from scratch on a small dataset. By contrast, the generative VLM benefits from frozen, large-scale pre-trained backbones (CLIP and GPT-2). Even without updating these

backbones, the learned visual–semantic alignment provides a strong inductive bias, enabling better generalization under low-resource conditions.

4.2. RQ2: Expression Quality and Hallucination Risk

RQ2 examines the generation quality and safety risks of VLMs on open-ended medical questions. The results reveal an apparent paradox: while the model performs poorly on strict lexical metrics in Exact Match and BLEU, it achieves a very high semantic similarity score measured by BERTScore.

The high BERTScore suggests that the model correctly captures core medical concepts and aligns them with visual features. However, lexical metrics penalize clinically reasonable variations in terminology. For example, A closer inspection of failure cases with Exact Match = 0 shows that many errors arise from surface-level linguistic differences rather than factual hallucinations.

As illustrated in Table 10, the model frequently produces semantically correct answers that differ only slightly from the reference annotations. For instance, in Row 1, the prediction *brain* was penalized against the ground truth *the brain*, resulting in a very low BLEU-4 score despite identical clinical meaning. Similarly, in Row 2, the model correctly identified the anatomical location as *right* but was penalized for omitting the modifier *side*. In Row 3, the model successfully classified the imaging modality as *MRI*, although it failed to specify the *diffusion-weighted*.

From a safety perspective, this distinction is critical. Qualitative analysis indicates that most errors correspond to soft hallucinations, where the model paraphrases or substitutes clinically related terms, rather than hard hallucinations, which involve fabricating non-existent diseases. The frozen GPT-2 backbone supports fluent and coherent language generation, thereby reducing the likelihood of nonsensical outputs.

Table 10 Representative Failure Cases Illustrating Metric Strictness

Truth Answer	Prediction	Metric Outcome	Interpretation
the brain	brain	BLEU-4 = 0.065	Identical clinical meaning
right side	right	EM = 0	Missing modifier, correct spatial concept
diffusion weighted MRI	MRI	EM = 0	Correct modality, missing specific subtype

4.3. Limitations

Although our implementation follows the methodology of Sonsbeek et al., our results fall below those reported in the original study. This discrepancy can be explained by three key factors: dataset scale, domain knowledge, and fine-tuning strategy.

First, the VQA-RAD dataset used in this study is substantially smaller than the Slake dataset adopted in prior work, limiting the capacity of the mapping network to learn complex associations. Second, while state-of-the-art results relied on biomedical language models such as BioGPT or BioMedLM, our use of a general-purpose GPT-2 introduces a domain mismatch. Besides, advanced adaptation methods such as LoRA provide stronger fine-tuning capacity than the lightweight prefix or mapping networks used here, albeit at higher computational cost. Finally, current evaluation metrics for open-ended medical VQA remain imperfect, as strict lexical matching underestimates the clinical value of semantically correct answers, highlighting the need for more advanced evaluation protocols.

5. CONCLUSION

This study analyzed the performance and safety of models for medical visual question answering tasks, with a focus on comparing classification-based approaches and generative vision–language models. Based on experiments conducted on the VQA-RAD dataset, the evaluation was carried out from two aspects: closed-ended questions and open-ended questions.

For closed-ended questions, the generative vision–language model achieved better overall performance than the CNN–LSTM baseline, even under limited training data. This result indicates that pre-trained visual and language features help the model learn the relationship between medical images and textual questions more effectively, while models trained from scratch show limitations in generalization.

For open-ended questions, the results show that evaluation metrics based on lexical matching cannot fully reflect the model's understanding of the questions. Some low-scoring cases are not caused by incorrect medical facts, but by differences in wording between the predicted answers and the reference answers, despite having the same semantic meaning. Further qualitative analysis suggests that most errors are related to incomplete expressions or alternative wording, rather than the generation of non-existent diseases or imaging findings.

Overall, this study suggests that relying only on automatic evaluation metrics may underestimate the actual performance of generative models in medical visual question answering tasks. Therefore, qualitative analysis is necessary when assessing model performance and safety, in order to better understand the reliability of model outputs. Future work may consider using larger medical datasets and evaluation methods that are more aligned with clinical use cases to further improve the practical applicability of such models.

AUTHORS CONTRIBUTION

Zou ting [24201617] was primarily responsible for the implementation and evaluation of the baseline CNN–LSTM model.

Zhon Jun Pei [24214748] was primarily responsible for the implementation of the generative vision–language model and the comparative analysis between the baseline and generative approaches.

Both authors contributed to the experimental design, result interpretation, and writing of the final report.

REFERENCES

Eldin, W. S., & Kaboudan, A. (2023). *AI-driven medical imaging platform: Advancements in image analysis and healthcare diagnosis*. *14*.

Hartsock, I., & Rasool, G. (2024). Vision-language models for medical report generation and visual question answering: A review. *Frontiers in Artificial Intelligence*, *7*, 1430984. https://doi.org/10.3389/frai.2024.1430984

Lau, J. J., Gayen, S., Ben Abacha, A., & Demner-Fushman, D. (2018). A dataset of clinically generated visual questions and answers about radiology images. *Scientific Data*, *5*(1), 180251. https://doi.org/10.1038/sdata.2018.251

Li, C., Wong, C., Zhang, S., Usuyama, N., Liu, H., Yang, J., Naumann, T., Poon, H., & Gao, J. (2023). LLaVA-med: Training a large language-and-vision assistant for biomedicine in one day. *Advances in Neural Information Processing Systems*, *36*, 28541–28564.

Liu, B., Zhan, L.-M., Xu, L., Ma, L., Yang, Y., & Wu, X.-M. (2021). *SLAKE: A semantically-labeled knowledge-enhanced dataset for medical visual question answering* (No. arXiv:2102.09542). arXiv. https://doi.org/10.48550/arXiv.2102.09542

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2001). BLEU: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02*, 311. https://doi.org/10.3115/1073083.1073135

Sharma, D., Purushotham, S., & Reddy, C. K. (2021). MedFuseNet: An attention-based multimodal deep learning model for visual question answering in the medical domain. *Scientific Reports*, *11*(1), 19826. https://doi.org/10.1038/s41598-021-98390-1

Sonsbeek, T. van, Derakhshani, M. M., Najdenkoska, I., Snoek, C. G. M., & Worring, M. (2023). *Open-Ended Medical Visual Question Answering Through Prefix Tuning of Language Models* (No. arXiv:2303.05977). arXiv. https://doi.org/10.48550/arXiv.2303.05977

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). *BERTScore: Evaluating text generation with BERT* (No. arXiv:1904.09675). arXiv. https://doi.org/10.48550/arXiv.1904.09675

APPENDIX

A. GitHub Repository

The complete code, data preprocessing scripts, and instructions for reproducing all experiments are available at:

GitHub Repository: https://github.com/zhongjp21-coder/WOA7015_AlternativeAsscement.git

Commit / Tag: main

B. Generative Model Training Hyperparameters

The following table summarizes the main hyperparameters used in our experiments:

Parameter	Value	Description
Vision Encoder	CLIP ViT-B/32 (pre-trained, frozen)	Extracts 512-d visual features
Language Model	GPT-2 (pre-trained, frozen)	Token embeddings for question-answer modeling
Mapping Network	MLP: 512 → 256 → 10 × 768	Linear-ReLU-Linear mapping to generate visual prefixes
Prefix Length (lx)	10	Number of prefix tokens concatenated with question tokens

Parameter	Value	Description
Token Sequence Length	Dataset-dependent	Mean number of tokens + 3×SD, zero-padded to max length
Optimizer	AdamW	Learning rate = 1e-4, weight decay = 0.01
Batch Size	16	Limited by GPU memory
Epochs	30	Early stopping applied if no improvement
Warm-up Steps	0	No explicit warm-up for frozen backbone
Device	NVIDIA RTX 3060 / CUDA	Single-GPU training

C. Notes on Generative Model Dataset

a. Dataset: VQA-RAD

b. Train/Test Split: 80% / 20%

c. Images: 315 radiology images (X-rays and CT scans)

d. QA Pairs: 2,248 pairs (57.7% closed-ended, 42.3% open-ended)

e. Tokenization: GPT-2 tokenizer, max length 128, zero-padded