# Visual Question Answering for Medical Images with Explainable AI

Deepananth K 195001027
Jayakrishnan S V 195001040
BE CSE, Semester 7

Dr. S Kavitha
Supervisor

**Project Review: 2** (25 March 2023)
Department of Computer Science and Engineering
SSN College of Engineering

---

## Abstract

Visual Question Answering (VQA) combines the fields of Natural Language Processing and Computer Vision to generate answers for the questions about the given input image. VQA involves fusion of features extracted from both image and corresponding question, and the fused feature vector is then used for training a Neural Network based model to generate answers. The trained model is then used for generating answers for the given input image and question. In this project, images from ImageCLEF 2019 VQA-Med Dataset are used and these images are complex to analyze and are of low resolution. VGGNet and BERT are used for extracting features from images and questions respectively. The features are concatenated and the answer generation is achieved using BERT. The trained model is applied on the test data and it gives 46.8% accuracy, 48.61 BLEU Score and 50.97 WBSS Score. The outcome of the VQA model is analyzed using Explainable AI (XAI). Combining XAI with VQA for the medical images gives analysis and supports the generated answer with justifications.

## 1 Introduction

Artificial Intelligence has grown exponentially over the past 10-15 years. The intelligent models or agents have solved many real-world problems and were able to learn or identify patterns among different kinds of data and provide the desired output. Now moving on to the next phase of learning, where the model tries to answer

the questions asked by the user related to some data provided along with the question. Visual Question Answering (VQA) is one such emerging task in the field of Artificial Intelligence and Computer Vision that aims to generate answers for the given questions by looking into the given image which corresponds to the question. VQA can be applied to various types of images like Natural Images, Medical Images or Cartoon Images. In this project, we aim to use different types of medical images like radiology images, CT scans, MRI scans etc., along with relevant questions and try to generate answers. Features are extracted from both image and question. The features are fused and are used to train a Neural Network architecture. The trained Neural Network architecture is used to generated answer for the input image and question. In order to justify the answer, some explanations are needed. There should be some features that correspond to the answer that is generated. Such features can be analyzed and identified using XAI tools.

## 1.1 Motivation

The medical domain is one, where there are new viruses and new diseases and also the one that needs faster analysis of different patients' conditions. Applying Artificial Intelligence techniques in the Medical field is more effective, where deep analysis of problems van be performed with the help of different AI techniques or algorithms. Visual Questions Answering in the medical domain would help doctors to analyze and get in-depth knowledge of medical images. Also, the doctors can submit their queries and get the required information [1]. Not only doctors, but even patients could use these VQA tools to get answers to their questions. Instead of searching and reading unknown articles from various websites, they can use these tools to get required information. There are Visual Question Answering models [3, 4, 5, 6, 7] for the medical domain that could generate answers for questions, but they do not give any justification for the predicted outcome. To overcome this limitation, the proposed model uses a Explainable AI (XAI) technique to justify the outcome of the VQA model.

## 1.2 Problem statement

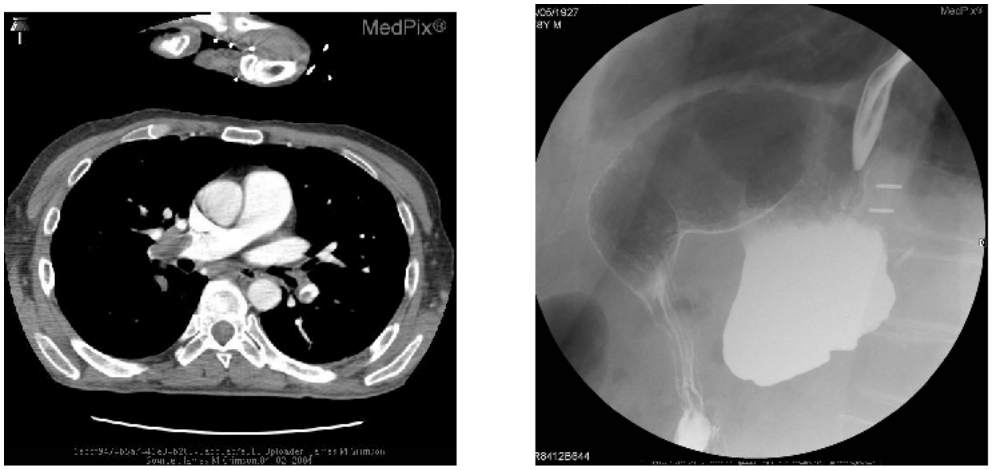
The aim of this project is to build an efficient VQA model that generates answers to questions related to Medical Images using deep learning techniques. In addition, the reason behind the generated answer has to be analyzed using Explainable AI tools like LIME (Local Interpretable Model-agnostic Explanations), SHAP (SHapley Additive exPlanations) to provide explanations on the outcome.

## 1.3 Input

The input is the medical images of different modalities, disease types, planes etc., and their relevant questions.

## 1.4 Output

For the given image and the query the proposed system predicts the answer which will be validated using Explainable AI technique. A sample image, query with the corresponding answer is shown in Figure 1.




(g) **Q**: which organ system is shown in the ct scan? **A**: lung, mediastinum, pleura

(h) **Q**: what is abnormal in the gastrointestinal image? **A**: gastric volvulus (organoaxial)

Figure 1: Sample Images and Questions with Corresponding Answers from ImageCLEF 2019 VQA-Med Dataset (Image IDs: synpic191614, synpic28495)

## 1.5 Objectives

- To collect and analyze the dataset of Visual Question Answering from the CLEF Forum.
- To build an efficient VQA model that generates the answers to the questions related to the given Medical Image using various transformer models like BERT, RoBERTa.
- To analyze the generated answer by using Explainable AI tools for providing explanations.

# 2 Literature survey

This section discusses various research papers on Visual Question Answering for medical images with its techniques and limitations and about XAI techniques used for Deep Learning algorithms.

## 2.1 Visual Question Answering

A task of Visual Question Answering was posted under the ImageCLEF forum which focuses on medical images [2]. The Visual Question Answering task involves generating answers for the given question provided the corresponding input image. The primary process in visual question answering involves extracting features from the image and question. They are then fused and the fused features are fed to a neural network to generate answers.

The dataset has four categories of questions such as Modality, Plane, Organ and Abnormality. Aisha Al-Sadi et al., [3] had used this characteristic of having different categories of questions in the dataset and had built ensembles of models for each category of the questions which had resulted with the Accuracy of 60.8% and a BLEU Score of 63.4.

An encoder-decoder type of model is proposed by Imane Allaouzi et al., [4] where the image features are extracted by using DenseNet which are then concatenated with the question features extracted using LSTM. These features are fed to a fully connected neural network to predict the answer. The answer generated in the previous iteration is then concatenated with the existing features and again fed to the same fully connected neural network to predict the next word in the answer. This is repeated until there is no answer generated by the neural network, marking the end of answer. This resulted with 55.6% accuracy and BLEU Score of 58.3.

Instead of just concatenating the image and question features, various feature fusing techniques are used for VQA tasks. Dhruv Sharma et al., [5] used Multimodal Factorized Bilinear Pooling (MFB) feature fusion technique to fuse the image feature extracted using ResNet-152 and question features extracted using pre-trained BERT. Some attention mechanisms like Image Attention and Image-Question Co-attention are used in their proposed model. An LSTM is used for answer generation similar to the answer generation technique proposed by Imane Allaouzi et al., [4]. The resulting accuracy is 63.8%.

Deepak Gupta et al., [6] used Batch Normalization to fuse the features from Inception-ResNet which extracts image features and Bi-LSTM which extracts question features. The fused features are then fed to a fully connected layer and the final answer is gen-

erated by recursive prediction procedure to find the individual words with the help of a Time-Distributed Layer.

Transformer models such as BERT had been proposed for answer generation by Fuji Ren et al., [7] where different models for different types of questions including different models for Open-Ended questions were built. For closed-ended questions like finding the modality, plane and organ, a classification BERT model was used. For questions which deal with the abnormality in the image, a generative model is proposed. Here the BERT model is trained for generating the answer by masking random words and predicting them, resulting with the accuracy of 62.4% and BLEU Score of 64.4.

## 2.2 Explainable Artificial Intelligence

Explainable AI is a domain that deals with techniques and tools that helps to explain the predictions made by ML/DL Models [8]. These tools analyze the model and find the reasons behind the prediction made by the model.

There are several XAI tools and techniques that have been widely used for different types of Architecture. Knapič S et al., [9] analyzed the predictions made by a CNN model trained to classify the bleeding and non-bleeding images that correspond to the endoscopy image of the gastrointestinal tract. XAI tools like LIME and SHAP were used. CIU (Contextual Importance and Utility) is a XAI technique which is also used in the proposed system for generating explanations. The predictions of the model were analyzed and the analysis of the three XAI tools/techniques were compared. The comparison showed that CIU outperformed both LIME and SHAP.

Apart from the CIU technique and LIME and SHAP tools, there are several other XAI techniques like Grad-CAM, Anchors and Integrated Gradients are available.

Ramprasaath R. Selvaraju et al., [10] proposed a technique called Gradient-weighted Class Activation Mapping (Grad-CAM) for explaining and understanding CNN based models. This technique helps to visualize the important regions on the input image, that corresponds to the predicted outcome. This technique helps to understand many CNN-based models such as Image-Captioning and VQA models.

Integrated Gradients and Anchors techniques are combined with LIME and SHAP tools for analyzing and providing explanations for a model that detects COVID-19 from the X-ray images [11]. The results of each of the above tools are combined to give explanations.

The discussion about various VQA models and XAI techniques are summarized in the Table 1.

Table 1: Literature Survey

| **Paper Title** | **Methodology** | **Limitations** |
|---|---|---|
| Visual question answering in the medical domain based on deep learning approaches: A comprehensive study [3] | The Questions are classified into 4 categories and multiple models are trained for each type of question<br>**Dataset**: ImageCLEF 2019 VQA-Med<br>**Image Feature Extraction**: VGGNet16<br>**Answer Generation**: Ensemble of Classification models<br>**Analysis**: Accuracy-60.8, BLEU Score-63.4 | All models built for each question categories are classification models which is completely a black-box approach |
| An Encoder-Decoder model for visual question answering in the medical domain [4] | **Dataset**: ImageCLEF 2019 VQA-Med<br>**Image Feature Extraction**: DenseNet-121<br>**Question Feature Extraction**: LSTM<br>**Feature Fusion**: Feature Concatenation<br>**Answer Generation**: Fully Connected Neural Network<br>**Analysis**: Accuracy-55.6, BLEU Score-58.3 | For each query, entire image needs to be looked up, while a attention based mechanisms could be used to look up only the question centric regions |
| MedFuseNet: An attention-based multimodal deep learning model for visual question answering in the medical domain [5] | **Dataset**: ImageCLEF 2019 VQA-Med, PathVQA<br>**Image Feature Extraction**: ResNet152<br>**Question Feature Extraction**: BERT<br>**Feature Fusion**: Multimodal Compact Bilinear Pooling (MCB)<br>**Answer Generation**: LSTM<br>**Analysis**: Accuracy-63.6 | Answer Generation is based on LSTM and comparatively transformer based models like BERT work efficiently |

| | | |
|---|---|---|
| Hierarchical deep multi-modal network for medical visual question answering [6] | **Dataset**: ImageCLEF's VQA-Med dataset (2018), VQA-RAD<br>**Image Feature Extraction**: Inception-ResNet-V2<br>**Question Feature Extraction**: Bi-LSTM<br>**Feature Fusion**: BAtch Normalization<br>**Answer Generation**: Fully Connected Layer, Time-Distributed Layer<br>**Analysis**: BLEU Score-41.1 | Answer Generation is based on LSTM and comparatively transformer based models like BERT work efficiently |
| CGMVQA: A New Classification and Generative Model for Medical Visual Question Answering [7] | **Dataset**: ImageCLEF 2019 VQA-Med<br>**Image Feature Extraction**: ResNet-152<br>**Question Feature Extraction**: BERT Tokenizer<br>**Answer Generation**: BERT<br>**Analysis**: Accuracy-62.4, BLEU Score-64.4 | The proposed solution for VQA is building different models for different types of question such as Modality, Plane, Organ and Abnormality. But in reality, the exact type of a question may not be known. |
| Explainable Artificial Intelligence for Human Decision Support System in the Medical Domain [9] | **Dataset:** Red Lesion Endoscopy data<br>**XAI tools:** LIME, SHAP, CIU<br>A CNN is trained using the dataset. XAI tools are then used for visualization in terms of heatmap. The result of visualization is then compared | |

| | | |
|---|---|---|
| Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization [10] | Proposed Gradient-weighted Class Activation Mapping (Grad-CAM) for explaining and understanding CNN based models. This technique helps to visualize the important regions on the input image, that corresponds to the predicted outcome. This technique helps to understand many CNN-based models such as Image-Captioning and VQA models. | |
| LISA : Enhance the explainability of medical images unifying current XAI techniques [11] | **Dataset:** COVID-19 Dataset<br>**XAI tools:** LIME, SHAP, Anchors<br>**Other XAI techniques:** Integrated Gradients Transfer Learning is used for the detection of COVID-19. The XAI tools LIME, SHAP Anchor and Integrated Gradient techniques' results are combined to give explanations. | |

## Inference

Pre-trained models such as VGGNet, ResNet and DenseNet were majorly used for extracting features from images and to extract question features, LSTM and BERT were primarily used. For answer generation, neural network based models like LSTM or a simple Fully Connected Neural Network or BERT are used. The VGGNet + LSTM combination is widely used for VQA tasks. In this project, VGGNet and BERT combination is used, which were not used together for VQA tasks. A BERT model is a popular NLP model and is used for answer generation as it is efficient for Masked Language Modeling (MLM) tasks.

Though there are existing VQA models [3, 4, 5, 6, 7] that are developed, those models do not provide any explanations. Applying XAI techniques to the ML/DL models, not only provides an analysis on the prediction, but also helps us to improve the developed model.Further for XAI, tools like LIME, SHAP, CIU and Grad-CAM are to be explored and appropriate techniques are to be applied.

# 3 Proposed System

The proposed system aims to develop a VQA model for the ImageCLEF 2019 VQA-Med Dataset using VGGNet and BERT. The ImageCLEF 2019 VQA-Med Dataset is chosen for this project, since it has four categories of questions and also it has images with different modalities (like CT, MRI, Ultrasound and etc.,.), different planes (like axial, lateral, sagittal and etc.,.) and different organs (like lung, skull, spine, muskuloskeletal and etc.,.). These images are complex to analyze and are of low resolution. VGGNet and BERT are used in this project for extracting features from images and questions respectively. A BERT model is trained for Masked Language Modeling (MLM) which generates answers for the corresponding input by predicting the masked words. Further, the results of the VQA model are to be analyzed using XAI tool and techniques like LIME and SHAP. The detailed system architecture diagram is shown in Figure 2.

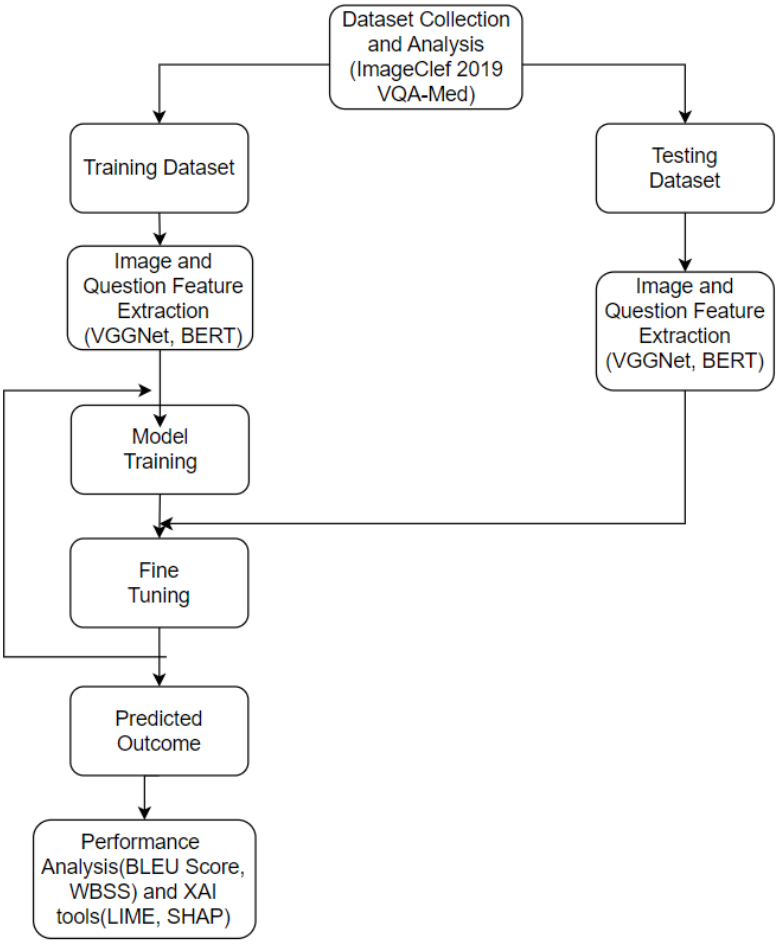



Figure 2: System Design

The system design is split into four modules. They are:

1) Dataset Collection and Analysis

2) Feature Extraction

3) VQA Model Building

4) Performance Analysis using Quantitative metrics and XAI techniques

## 3.1 Dataset Collection and Analysis

ImageClef 2019 VQA-Med Dataset is used in this project. In this module, the dataset is collected and analyzed in terms of the categories of question such as Modality, Plane, Organ and Abnormality. Within these categories of questions, a pivot table is used to analyze the classes available under each categories. The analysis of data is further discussed under the Section 5.1.

## 3.2 Feature Extraction

### 3.2.1 Image Pre-Processing

The images are pre-processed by resizing the image to a constant size of (224,224).The resized images are then used for Image Feature Extraction. The function for image pre-processing is summarized in Algorithm 1.

**Algorithm 1** Image Pre-Processing

**Input** :$Image\ of\ different\ sized$ ▷ Input Image

**Output** :$Resized\_Image\ of\ size\ 224 \times 224$ ▷ Resized Image

**function** IMAGEPREPROCESS(Image)

$Resized\_Image \leftarrow cv2\_resize(Image, (224, 224))$

**return** $Resized\_Image$

**end function**

### 3.2.2 Image Feature Extraction

A pre-trained VGGNet is used to extract features from the images. The last layer of the VGGNet model is replaced with a dense layer of 960 units. When an image is given to this VGGNet Model, the values at the newly added dense layer are the required image features.

The architecture of VGGNet which is modified by replacing the last layer is shown in Figure 3.

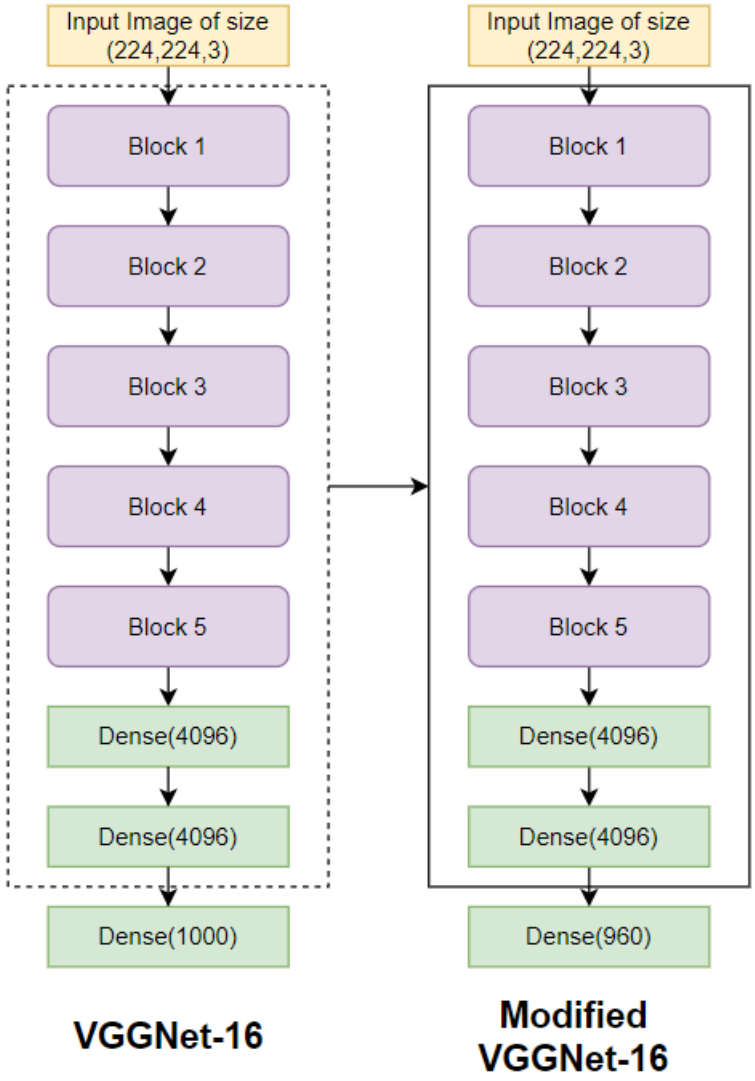



Figure 3: Modified VGGNet Architecture

The summary of the modified VGGNet model is shown in Figure 4.

| Layer (type) | Output Shape | Param # |
|---|---|---|
| input_1 (InputLayer) | [(None, 224, 224, 3)] | 0 |
| block1_conv1 (Conv2D) | (None, 224, 224, 64) | 1792 |
| block1_conv2 (Conv2D) | (None, 224, 224, 64) | 36928 |
| block1_pool (MaxPooling2D) | (None, 112, 112, 64) | 0 |
| block2_conv1 (Conv2D) | (None, 112, 112, 128) | 73856 |
| block2_conv2 (Conv2D) | (None, 112, 112, 128) | 147584 |
| block2_pool (MaxPooling2D) | (None, 56, 56, 128) | 0 |
| block3_conv1 (Conv2D) | (None, 56, 56, 256) | 295168 |
| block3_conv2 (Conv2D) | (None, 56, 56, 256) | 590080 |
| block3_conv3 (Conv2D) | (None, 56, 56, 256) | 590080 |
| block3_pool (MaxPooling2D) | (None, 28, 28, 256) | 0 |
| block4_conv1 (Conv2D) | (None, 28, 28, 512) | 1180160 |
| block4_conv2 (Conv2D) | (None, 28, 28, 512) | 2359808 |
| block4_conv3 (Conv2D) | (None, 28, 28, 512) | 2359808 |
| block4_pool (MaxPooling2D) | (None, 14, 14, 512) | 0 |
| block5_conv1 (Conv2D) | (None, 14, 14, 512) | 2359808 |
| block5_conv2 (Conv2D) | (None, 14, 14, 512) | 2359808 |
| block5_conv3 (Conv2D) | (None, 14, 14, 512) | 2359808 |
| block5_pool (MaxPooling2D) | (None, 7, 7, 512) | 0 |
| flatten (Flatten) | (None, 25088) | 0 |
| fc1 (Dense) | (None, 4096) | 102764544 |
| fc2 (Dense) | (None, 4096) | 16781312 |
| new_fc (Dense) | (None, 960) | 3933120 |

Figure 4: Model Summary of VGGNet

The values from the newly added layer are added with 100 and are rounded to its highest integer value as BERT does not accept float values. The rounded values are the encoded image features. The values are added with 100 so that the encoded feature values do not match the token Ids of the special tokens. This image feature encoding is depicted in the Algorithm 2.

**Algorithm 2** Image Feature Encoding

**Input** :*Image* ▷ Input Image
**Output** :*Image_Encoding of length* 960 ▷ Encoded Image Features
$VGGModel \leftarrow VGG16()$
$VGGModel.layers[-1] \leftarrow Dense(units = 960, activation = "relu")$
**function** GETIMAGEENCODING(Image)
$Preprocessed\_Image \leftarrow imagePreProcess(Image)$
$Image\_Features \leftarrow VGGModel(Preprocessed\_Image).layers[-1].values$
$Image\_Encoding \leftarrow list((ceil(x) + 100)\ for\ x\ in\ Image\_Features)$
**return** *Image_Encoding*
**end function**

### 3.2.3 Question Pre-Processing

The text data usually is associated with punctuation and special characters and hence the question needs to be pre-processed by removing these special characters and punctuation. Also, the text is converted to lower case. The question pre-processing is summarized as Algorithm 3.

**Algorithm 3** Question Pre-Processing

**Input** :*Question* ▷ Input Question for pre-procesing
**Output** :*Preprocessed_Question* ▷ Preprocessed Question
**function** TEXTPREPROCESS(Question)
$Lower\_Case\_Qn \leftarrow Question.lower()$
$Preprocessed\_Question \leftarrow Lower\_Case\_Qn.replace("[^a-z0-9]"," ")$
**return** *Preprocessed_Question*
**end function**

### 3.2.4 Question Feature Extraction

The question features are the set of tokens generated by the BERT-Tokenizer. For tokenizing the question, a vocabulary is initially built with the text data available in the dataset. This vocabulary file is used with BERT to tokenize the question which is the required question feature. The Question Feature Extraction Process is explained in Algorithm 4.

**Algorithm 4** Question Feature Encoding

**Input** :*Question* ▷ Input Question for Feature Extraction
**Output** :*Question_Encoding*
*Tokenizer* ← *BERT_Tokenizer*(*VocabFilePath*) ▷ Question Tokens ie., Features
**function** GETTOKENIZEDQUESTION(Question)
*Preprocessed_Question* ← *textPreProcess*(*Question*)
*Question_Encoding* ← *Tokenizer*(*Preprocessed_Question*)
**return** *Question_Encoding*
**end function**

## 3.3 VQA Model Building

The image and question features are extracted using VGGNet and BERT. These features are then fused by concatenating the feature vectors. VQA model building involves developing an answer generating model that takes the encoded image features and the tokenized question features as input and generates the corresponding answers. In this project, a BERT Model is used for generating answers by taking the fused features as input. A BERT model is very efficient for the tasks of Next Sentence Prediction (NSP) and Masked Language Modeling (MLM). MLM involves predicting the masked token in the given sentence or a paragraph. The idea of MLM is used in this project, to generate the answers for the given image and question using the fused feature vectors.

Training the BERT using MLM involves constructing the input for BERT with the image and text encoding along with the respective tokenized answers. The parts of the answers are masked and the model is trained to predict the masked words. The input for training is first constructed in the following format as explained by Algorithm 5:

**[CLS] ENCODED-IMAGE-FEATURES [SEP] QUESTION-FEATURE [SEP] MASKED-ANSWER [SEP]**

**Algorithm 5** Constructing Input for BERT Training

**Input:** Image, Question, Answer

**Output:** Tokens ▷ A list of token vector

**function** CONSTRUCTINPUT(Image,Question,Answer)

$MaxLen \leftarrow 1000$

$ImageEncoding \leftarrow getImageEncoding(Image)$

$QuestionEncoding \leftarrow getTokenizedQuestion(Question)$

$AnswerEncoding \leftarrow getTokenizedQuestion(Answer)$

$Input \leftarrow [CLS] + ImageEncoding + [SEP] + QuestionEncoding + [SEP] + AnswerEncoding + [SEP]$

$Padding \leftarrow MaxLen - len(Input)$

$i \leftarrow 0$

$Tokens \leftarrow []$ ▷ Empty List

**while** $i < Padding$ **do**

$temp\_tokens \leftarrow Input$

$mask\_positions \leftarrow len(QuestionEncoding + ImageEncoding) + 3 + i$

$temp\_tokens[mask\_positions] \leftarrow MASK$ ▷ Masking

$Tokens.append(temp\_tokens)$

$i \leftarrow i + 1$

**end**

**return***Tokens*

**end function**

Figure 5 shows the working of the BERT model to predict the masked word.

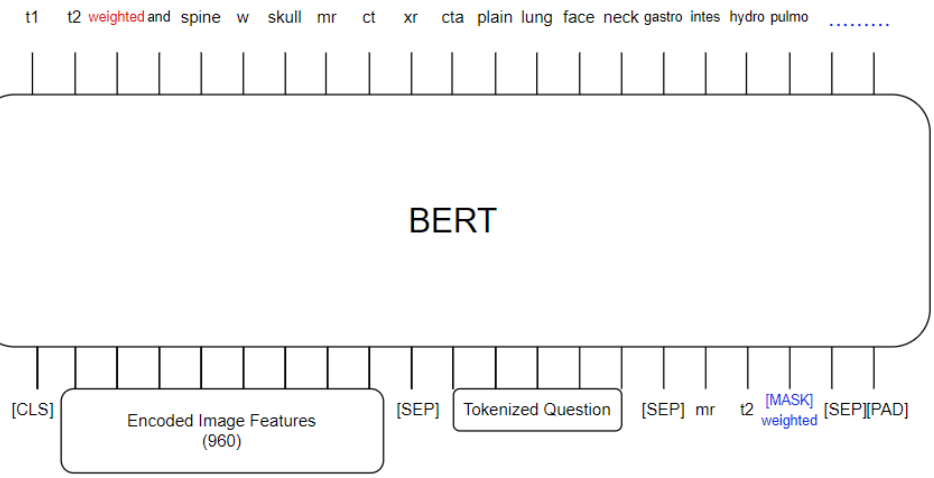



Figure 5: BERT model predicting the masked word

The trained model is then used for generating answers. For generating answers from the trained model, the input data is of the form:

**[CLS] ENCODED-IMAGE-FEATURES [SEP] QUESTION-FEATURE [SEP] [MASK]**

The model now attempts to predict the word at the masked position. When the model predicts the word, then the word is concatenated to the input data and the [MASK] is appended to the end of it. Now the model tries to predict the word at the current position of [MASK]. The above process repeats until a [SEP] token is predicted, marking the end of the answer. This is summarized in the Algorithm 6.

**Algorithm 6** Answer Generation

**Input:** Image, Question
**Output**: Answer
**function** GETANSWER(Image,Question)
$MaxLen \leftarrow 1000$
$ImageEncoding \leftarrow getImageEncoding(Image)$
$QuestionEncoding \leftarrow getTokenizedQuestion(Question)$
$Input \leftarrow [CLS] + ImageEncoding + [SEP] + QuestionEncoding + [SEP] + [MASK]$
$Padding \leftarrow MaxLen - len(Input)$
$i \leftarrow 0$
$Vocab \leftarrow loadVocabulary(VocabPath)$
$Answer \leftarrow$ "" ▷ Empty String
$MaskPosition \leftarrow len(Input)$
**while** $i < Padding$ **do**
$Prediction \leftarrow VQAModel(Input)$
$GeneratedWord \leftarrow Vocab[Prediction]$
**if** $GeneratedWord =$ "$[SEP]$" **then**
**break**
**end**
$Answer \leftarrow Answer + GeneratedWord$
$Input[MaskPosition] \leftarrow Prediction$
$MaskPosition \leftarrow MaskPosition + 1$
$Input[MaskPosition] \leftarrow [MASK]$
$i \leftarrow i + 1$
**end**
**return** *Answer*
**end function**

For instance, to generate the answer **'bucket handle tear of meniscus'** (Image ID: synpic58267), the model generates answer as shown in the Figure 6

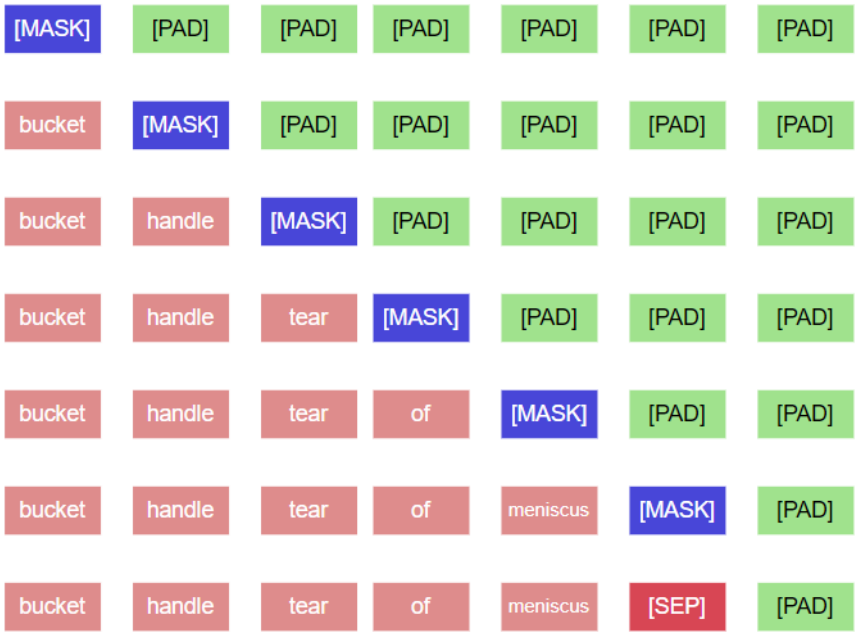



Figure 6: Answer Generation for a Sample with ID: synpic58267

## 3.4 Performance Analysis using Quantitative metrics and XAI

### 3.4.1 Performance Analysis of VQA using Quantitative metrics

The performance of Visual Question Answering Models can be analyzed using various metrics such as Accuracy, BLEU Score and WBSS. Accuracy is a measure of how accurate the generated answer matches with the actual answer. The perfect match gives score 1.0, otherwise 0. Accuracy is calculated as shown in Eq. 1.

$$Accuracy = \frac{No.\ of\ correctly\ generated\ answers}{Total\ no.\ of\ samples} \tag{1}$$

The BLUE Score or the Bilingual Evaluation Understudy Score is a score for comparison of the generated answer and the actual answers. The comparison here does not check if both the answers exactly match. It calculates the score based on how much of the answer words match in each of the generated and actual answers. The perfect match gives a BLEU score of 1.0 while a perfect mismatch gives 0. The BLEU score value can be between 0 and 1, depending on the percentage of match between the two answers. The steps involved in calculating BLEU Score are as follows.

The first step is to compute Precision scores for 1-grams. The Precision scores for 1-grams is calculated as given by Eq. 2

$$PrecisionOneGram = \frac{No.\ of\ CorrectlyPredictedOneGrams}{No.\ of\ TotalPredictedOneGrams} \tag{2}$$

The next step is to calculate the Brevity Penalty using the values of c, which is the number of words in the generated sentence and r, which is the number of words in the target sentence. The Brevity Penalty is calculated as shown in Eq. 3.

$$BrevityPenalty = \begin{cases} 1, & \text{if } c > r \\ e^{(1-r/c)}, & \text{if } c <= r \end{cases} \tag{3}$$

Finally to calculate BLEU Score, the Brevity Penalty is multiplied with Precision 1-gram value as given by Eq. 4

$$BLEU\ =\ BrevityPenalty\ .\ PrecisionOneGram \tag{4}$$

Word-based Semantic Similarity (WBSS) is used to compare the Wu-Palmer similarity (WUPS) between the words in each of the actual and generated answers. For a generated answer A and the actual answer or the ground truth T, the WUPS[12] is calculated as depicted in Eq. 5.

$$WUPS(A,T) = \frac{1}{N} \times \sum_{i=1}^{N} \times min\left\{ \prod_{a \in A^i} max_{t \in T^i} WUP(a,t), \prod_{t \in T^i} max_{a \in A^i} WUP(a,t) \right\} \times 100 \tag{5}$$

### 3.4.2 Explainable AI (XAI)

Explainable AI or XAI deals with explaining the predictions made by a ML/DL model. They try to analyze the output with respect to the input features that contributed to the arrival of that prediction. There are many XAI tools to interpret the various ML/DL models and give explanations for the output. In this project XAI tools LIME (Local Interpretable Model-agnostic Explanations) and SHAP (SHapley Additive exPlanations) are to be used to analyze the predictions made by the VQA model.

LIME takes a prediction function of a model as input along with a input data for the model to predict. Lime now produces an analysis of how the model arrives at the result for the input data by analyzing the prediction function's working.

On the other hand, SHAP takes a prediction function and a set of data as input and

try to find the SHapley values that expresses model predictions as linear combinations of binary variables.

# 4 Feasibility study

## 4.1 Availability of Dataset

ImageClef 2019 VQA-Med Dataset
**Dataset Link:** https://www.aicrowd.com/clef_tasks/29/task_dataset_files?challenge_id=220

## 4.2 Timeline

| Review | Module | Jan | Feb | Mar | Apr |
|---|---|---|---|---|---|
| Review 1 | Data Collection and Analysis | | | | |
| | Image Feature Extraction | | | | |
| Review 2 | Text Feature Extraction | | | | |
| | VQA Model Building | | | | |
| Review 3 | Performance Analysis (Accuracy, BLEU Score, WBSS) and XAI (LIME, SHAP) | | | | |

## 4.3 Hardware and Software Requirements

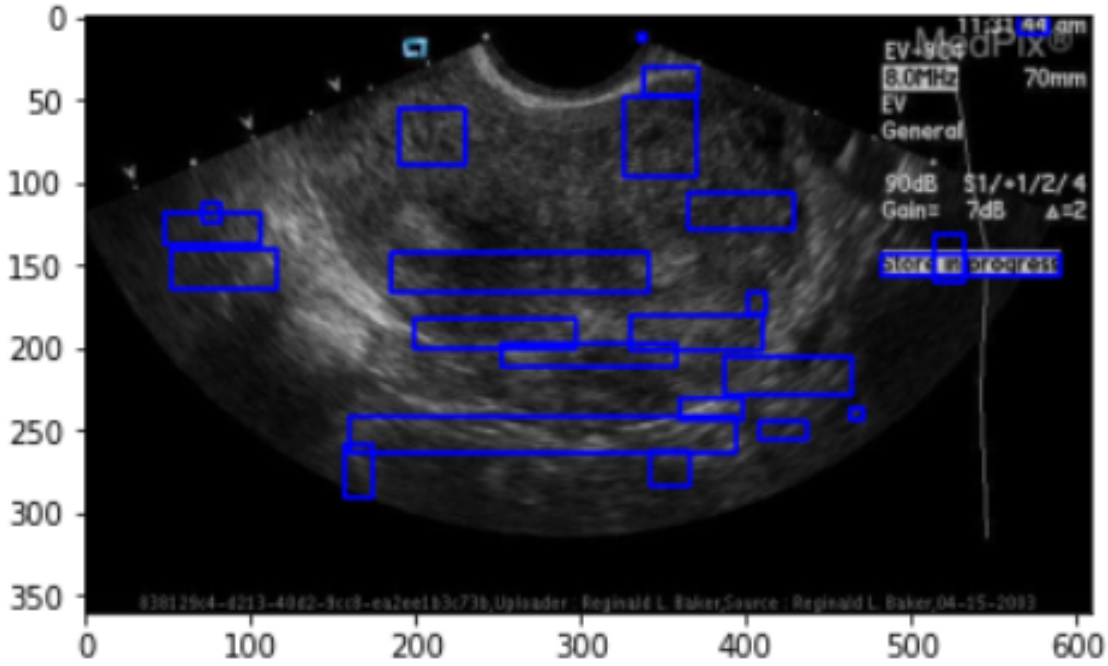
High processing computers with GPUs are used for training the model faster and efficiently. Machine Learning and Deep Learning libraries like Tensorflow, Pandas, Numpy, Pytorch, CV2 are used.

# 5 Implementation & Results

## Review 1 - Follow up

| S. No. | Question/Suggestions | Follow up |
|---|---|---|
| 1 | Use different CNN models for image feature extraction for different modality images | There isn't any increase in accuracy of the VQA model |
| 2 | Remove text from the image | Some of the organ features are also marked as text and removing the text would remove the features of the organ area in the image as shown in Figure 5. |

Table 2: Review 1 - Follow up




For review 2, the dataset collection & analysis, feature extraction and VQA model building are completed. The flow of the work carried out for review 2 is depicted in Figure 7.

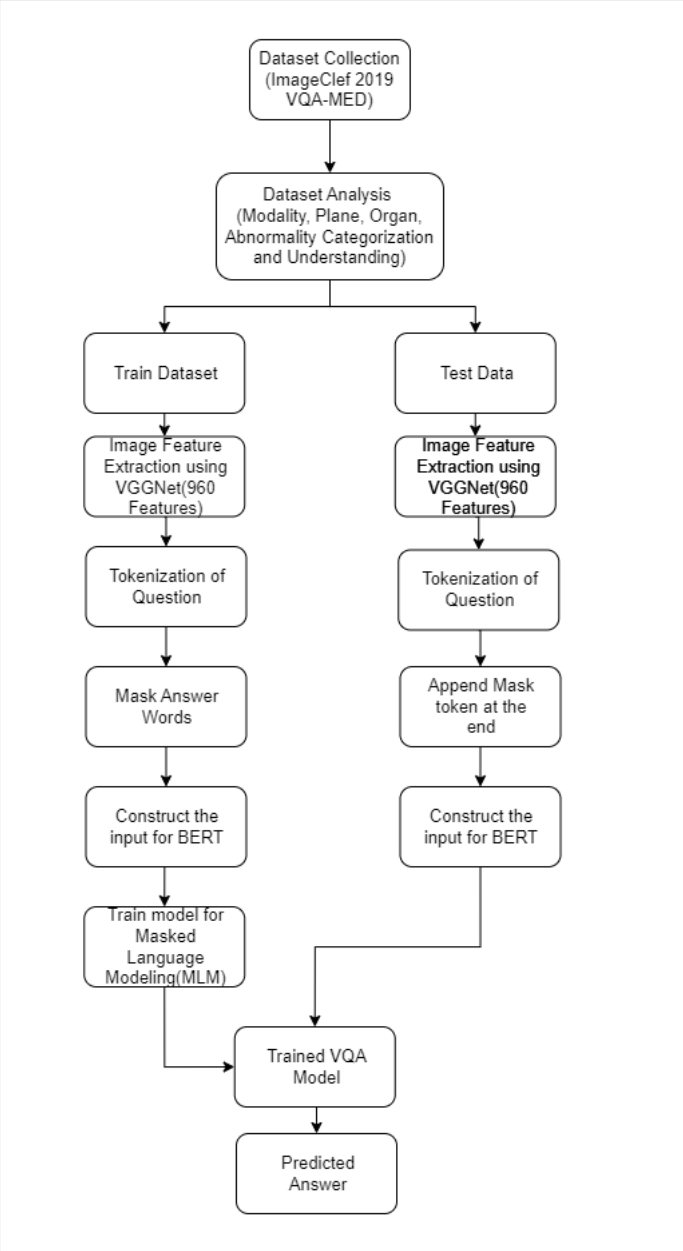



Figure 7: Review 2 Flow Diagram

## 5.1 Dataset Collection & Analysis

Many Datasets are available for the desired tasks and they contain many Medical Images along with many relevant questions for each image. A ImageCLEF task has been posted for VQA for Medical Images and the dataset (VQA-Med 2019) is under AI Crowd.The dataset contains different medical images and their corresponding Question-Answer pairs. There are 3200 training medical images. The questions are categorized into four major types - Modality, Plane, Organ System and Abnormality.

There are totally 12,792 Question-Answer pairs.

A text file for each category of questions is given in the dataset which includes the Question-Answer pair along with the image ID which corresponds to the name of the image file.

The validation set contains 500 images and 2000 Question-Answer pairs. The test set consists of 500 images and 500 Question-Answer pairs.The result of the analysis is given in the Table 3.

| Dataset | Question Category | No. of Questions | No. of Classes |
|---|---|---|---|
| Training Dataset | Modality | 3200 | 44 |
| | Plane | 3200 | 15 |
| | Organ System | 3200 | 10 |
| | Abnormality | 3192 | 1484 |
| Testing Dataset | All | 500 | - |

Table 3: Dataset Analysis

A sample image along with four types of question and corresponding answers is shown in Figure 8.

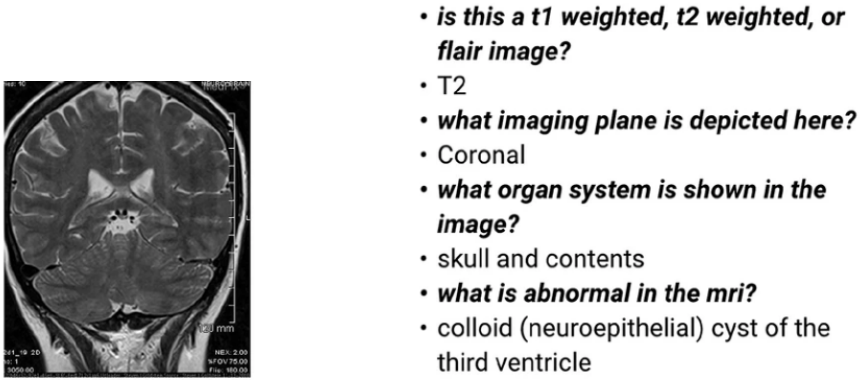



Figure 8: A Sample Image and QA pair from Dataset (Image ID: synpic16994)

## 5.2 Feature Extraction

### 5.2.1 Image Feature Extraction

The image features are extracted and are encoded as explained in Algorithm 2 from the VGGNet model. The efficiency of VGGNet in extracting features from the images is analyzed using a heatmap which depicts the activations of the last Convolution Layer. Figure 9 shows the generated heatmap superimposed onto the original image.

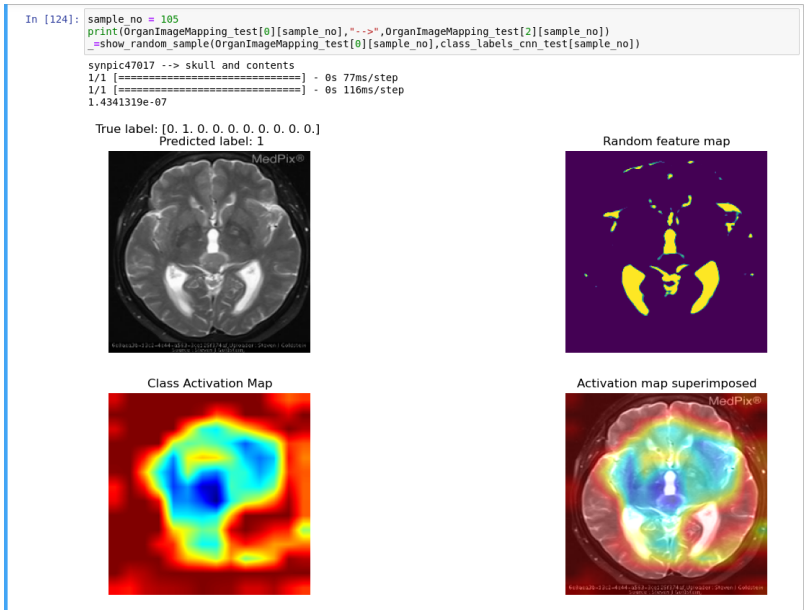



Figure 9: Pre-trained VGGNet

### 5.2.2 Question Feature Extraction

The questions are tokenized using BERT-Tokenizer. To tokenize the question, a vocabulary is built using the text data available in the dataset. This vocabulary is used to tokenize the question using BERT. The tokenized questions is the required question feature.

The vocabulary built using the text from the dataset has 4914 words including the special tokens for BERT such as [PAD], [UNK], [CLS], [SEP] and [MASK].

A sample set of tokens and their corresponding token ID is shown in the Table 4.

| **Token ID** | **Token** |
|---|---|
| 0 | [PAD] |
| 1 | [UNK] |
| 2 | [CLS] |
| 3 | [SEP] |
| 4 | [MASK] |
| 92 | th |
| 94 | ##at |
| 97 | what |

Table 4: A sample set of tokens and their IDs from the vocabulary

## 5.3 VQA Model Building

The feature from images and questions are extracted and are fused. For answer generation, a BERT model is built and trained for generating answer tokens. The model is trained for 20 epochs with batch size of 40. Figure 10 shows the training and validation of the model.

```
Epoch: 1
Iter (loss=0.065): : 1042it [21:09,  1.22s/it]
Iter (loss=0.802): : 176it [02:29,  1.18it/s]
Epoch: 2
Iter (loss=0.023): : 1042it [21:18,  1.23s/it]
Iter (loss=0.902): : 176it [02:28,  1.19it/s]
Epoch: 3
Iter (loss=0.014): : 1042it [21:20,  1.23s/it]
Iter (loss=0.752): : 176it [02:27,  1.19it/s]
Epoch: 4
Iter (loss=0.006): : 1042it [21:09,  1.22s/it]
Iter (loss=0.870): : 176it [02:28,  1.19it/s]
Epoch: 5
Iter (loss=0.007): : 1042it [21:21,  1.23s/it]
Iter (loss=0.631): : 176it [02:27,  1.19it/s]
Epoch: 6
Iter (loss=0.004): : 1042it [21:07,  1.22s/it]
Iter (loss=0.455): : 176it [02:28,  1.18it/s]
Epoch: 7
Iter (loss=0.002): : 1042it [21:10,  1.22s/it]
Iter (loss=2.232): : 176it [02:27,  1.20it/s]
```

Figure 10: Training and Validation of Model

The trained model is now used for generating answers for the given question along with the corresponding input image. Figures 11 - 15 shows samples of answers generated by the model.

Figure 11 shows a sample image queried about the Organ.

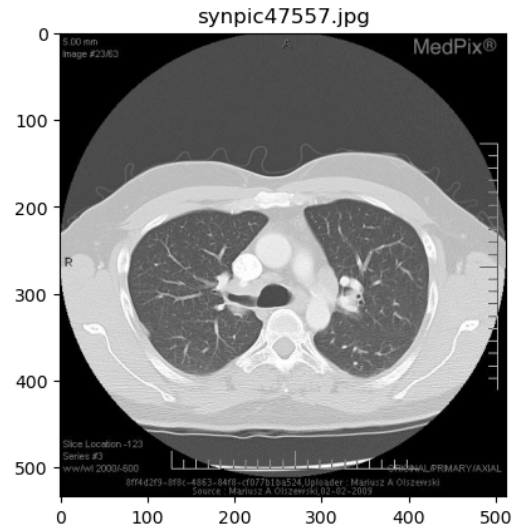
Figure 12 shows a sample image queried about the Modality.

Figure 13 shows a sample image queried about the Organ.

Figure 14 shows a sample image queried about the Plane.

Figure 15 shows a sample image queried about the Abnormality.

```
generateAnswer('synpic47557','what organ system is imaged?')
```

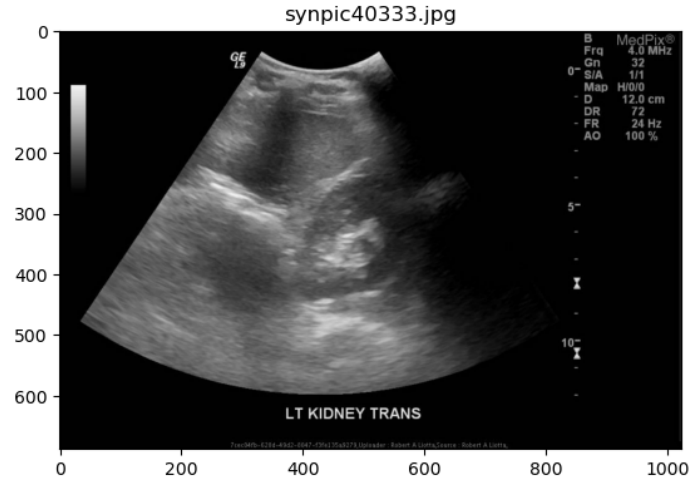



```
Generating Answers...
lung
mediastinum
pleura
[SEP]

'lung mediastinum pleura '
```

Figure 11: Sample Answer Generation for the image with ID: synpic47557 and queried about the Organ captured (Correct Answer)

```
generateAnswer('synpic40333','what imaging modality was used to take this image?')
```

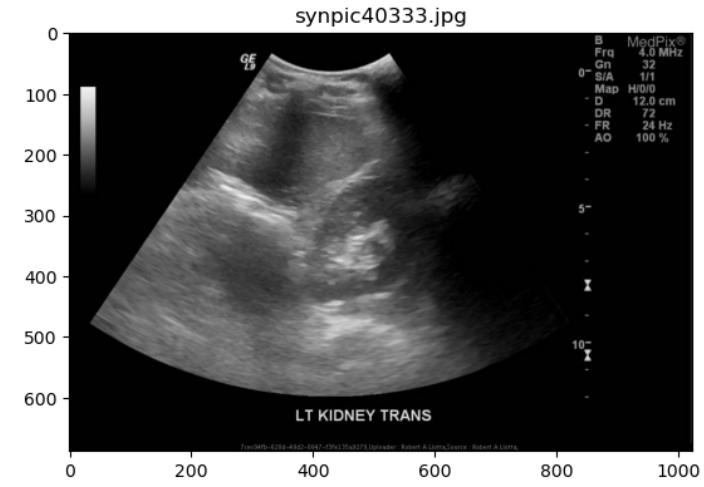



```
Generating Answers...
us
ultrasound
[SEP]

'us ultrasound '
```

Figure 12: Sample Answer Generation for the image with ID: synpic40333 and queried about the Modality (Correct Answer)

```
generateAnswer('synpic40333','what organ system is imaged?')
```

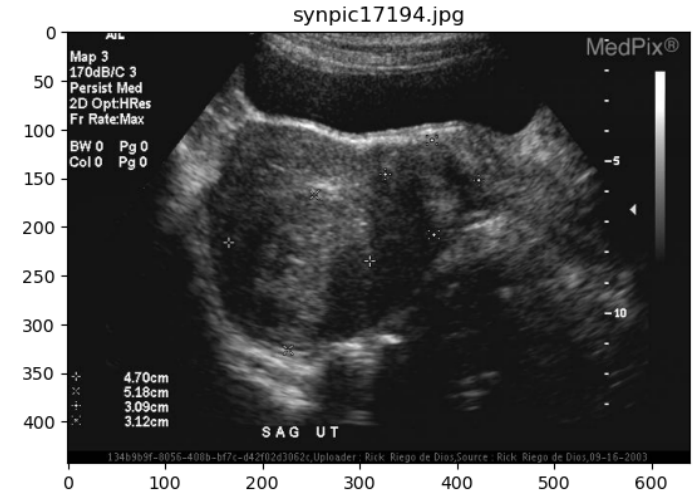



```
Generating Answers...
gastrointestinal
[SEP]

'gastrointestinal '
```

Figure 13: Sample Answer Generation for the image with ID: synpic40333 and queried about the Organ captured (Correct Answer)

```
generateAnswer('synpic17194','what is the plane shown in this image?')
```




```
Generating Answers...
sagittal
[SEP]

'sagittal '
```

Figure 14: Sample Answer Generation for the image with ID: synpic17194 and queried about the Plane (Correct Answer)

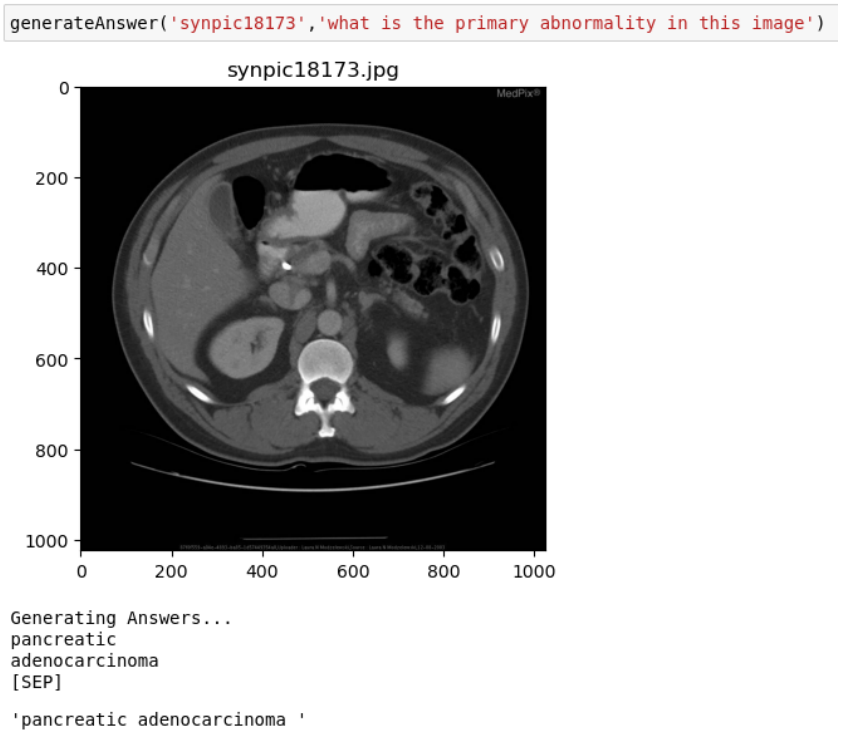



Figure 15: Sample Answer Generation for the image with ID: synpic18173 and queried about the Abnormality (Wrong answer - Actual answer: pancreatic duct adenocarcinoma)

## 5.4 Performance Analysis using Quantitative metrics

The performance of the VQA Model is analyzed using metrics such as accuracy, BLEU Score and WBSS. Table 5 shows the performance of the VQA model for each categories and overall test data.

| **Category** | **No. of Samples** | **Accuracy** | **BLEU Score** | **WBSS** |
|---|---|---|---|---|
| Modality | 125 | 65.6 | 68.79 | 71.66 |
| Plane | 125 | 64.8 | 64.8 | 65.35 |
| Organ | 125 | 50.4 | 53.19 | 54.82 |
| Abnormality | 125 | 6.4 | 7.65 | 12.03 |
| **Overall** | **500** | **46.8** | **48.61** | **50.97** |

Table 5: Performance analysis using Accuracy, BLEU Score and WBSS

### Inference

The performance metrics such as Accuracy, BLUE Score and WBSS of Modality based questions are high compared to other categories. In case of abnormality, the accuracy is very low due to unavailability of enough data for 1484 classes of abnormality in the training set (Refer Table 3 for analysis of the dataset). For other categories of questions, the classes are few and there is enough data for training.

## 6 Conclusion & Future Work

The dataset for the task of Visual Question Answering is collected and it is analyzed. The image features are extracted using VGG-16 from the newly added dense layer of 960 units and are encoded. The question is tokenized using BERT with the help of the vocabulary built from the text data in the dataset. The BERT model is trained for predicting the Masked token. From the extracted features of trained dataset, the BERT model is trained using the Masked Language Modeling (MLM).The trained model is used to predict answers for the test data by recursively feeding the model with the encoded features and the answer from the last iteration. The model gives a 46.8% Accuracy, 48.61 BLEU Score and 50.97 WBSS for test dataset. The accuracy of abnormality based questions are too low due to data scarcity and it reduces the overall accuracy. For the next review, XAI techniques are to be applied to generate explanations for the predicted outcome and the VQA model can be fine tuned for increasing the accuracy of abnormality based question.

## References


[1] Lin Z, Zhang D, Tac Q, Shi D, Haffari G, Wu Q, He M, Ge Z. *Medical visual question answering: A survey*. arXiv preprint arXiv:2111.10056. 2021 Nov 19.

[2] Asma Ben Abacha, Sadid A. Hasan, Vivek V. Datla, Joey Liu, Dina Demner-Fushman, Henning Müller. *Overview of the Medical Visual Question Answering Task at ImageCLEF 2019*. CEUR-WS. 2019 Sep 9.

[3] Al-Sadi, Aisha & Al-Ayyoub, Mahmoud & Jararweh, Yaser & Costen, F.. (2021). *Visual Question Answering in the Medical Domain Based on Deep Learning Approaches: A Comprehensive Study. Pattern Recognition Letters.* 150. 10.1016/j.patrec.2021.07.002.

[4] Allaouzi, Imane et al. *An Encoder-Decoder Model for Visual Question Answering in the Medical Domain*, Conference and Labs of the Evaluation Forum (2019).

[5] Sharma, D., Purushotham, S. & Reddy, C.K. *MedFuseNet: An attention-based multimodal deep learning model for visual question answering in the medical domain.* Sci Rep 11, 19826 (2021).

[6] Deepak Gupta, Swati Suman, Asif Ekbal. *Hierarchical deep multi-modal network for medical visual question answering*,Expert Systems with Applications, Volume 164, 2021, 113993, ISSN 0957-4174, https://doi.org/10.1016/j.eswa.2020.113993.

[7] F. Ren and Y. Zhou, *CGMVQA: A New Classification and Generative Model for Medical Visual Question Answering*, in IEEE Access, vol. 8, pp. 50626-50636, 2020, doi: 10.1109/ACCESS.2020.2980024.

[8] U. Pawar, D. O'Shea, S. Rea and R. O'Reilly, *Explainable AI in Healthcare*, 2020 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA), 2020, pp. 1-2, doi: 10.1109/CyberSA49311.2020.9139655.

[9] Knapič S, Malhi A, Saluja R, Främling K. *Explainable Artificial Intelligence for Human Decision Support System in the Medical Domain*. Machine Learning and Knowledge Extraction. 2021; 3(3):740-770.

[10] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, Dhruv Batra. *Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization*. International Journal of Computer Vision, vol. 128(2):336–359, Springer Science and Business Media, October 2019.

[11] S. H. P. Abeyagunasekera, Y. Perera, K. Chamara, U. Kaushalya, P. Sumathipala and O. Senaweera, *LISA : Enhance the explainability of medical images unifying current XAI techniques*. 2022 IEEE 7th International conference for Convergence in Technology (I2CT), Mumbai, India, 2022, pp. 1-9, doi: 10.1109/I2CT54291.2022.9824840.

[12] Mateusz Malinowski, Mario Fritz, *A Multi-World Approach to Question Answering about Real-World Scenes based on Uncertain Input*. Adv Neural Inf Process Syst. 2014;27:1682–90.