# NORTH SOUTH UNIVERSITY

# VQA: Asking questions about image content

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF
ELECTRICAL AND COMPUTER ENGINEERING
OF NORTH SOUTH UNIVERSITY
IN THE PARTIAL FULFILMENT OT THE REQUIREMENTS
FOR THE DEGREE OF
BACHELOR OF SCIENCE IN
COMPUTER SCIENCE AND ENGINEERING

**CSE 499B, SUMMER 2022**
**SENIOR DESIGN PROJECT**

# Declaration

It is hereby acknowledged that:

- No illegitimate procedure has been practiced during the preparation of this document.
- This document does not contain any previously published material without proper citation.
- This document represents our own accomplishment while being Undergraduate Students in the North South University.

This CSE499B report, entitled "***VQA: Asking Questions about Image Contents***," has not been approved for credit toward any degree and has not been submitted in association with any application for credit toward another degree, we thus certify. Wish to consider accepting this report as partial satisfaction of the requirements for the Bachelor of Science in Electrical and Computer Engineering at North South University.

Sincerely,

**S M Gazzali Arafat Nishan**
1831513042

Ashfaq Uddin Ahmed

**Ashfaq Uddin Ahmed**
1911848042

Tasfia Tabassum

**Tasfia Tabassum**
1821391042

# Approval

This is to certify that the CSE499B report entitled "***VQA: Asking Questions about Image Contents***", submitted by **S M Gazzali Arafat Nishan** (Student ID: **1831513042**), **Ashfaq Uddin Ahmed** (Student ID: **1911848042**) and **Tasfia Tabassum** (Student ID: **1821391042**) are undergraduate students of the Department of Electrical Computer Engineering, North South University. This report partially fulfils the requirements for the degree of Bachelor of Science in Computer Science and Engineering on **September 01, 2022**, and has been accepted as satisfactory.

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation.

**Dr. Mohammad Ashrafuzzaman Khan**
Assistant Professor
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation.

**Dr. Mohammad Rezaul Bari**
Associate Professor & Chair
Department of Electrical and Computer Engineering
North South University
Dhaka, Bangladesh

# Abstract

VQA (Visual Question Answering) is a relatively new activity that requires algorithms to reason about the visual content of a picture in order to respond to a natural language question. VQA requires an algorithm to respond to text-based inquiries about photos. Because many open-ended replies comprise a few words or a closed set of options that may be supplied in a multiple-choice format, VQA adapts itself to automated review. To put it simply, the Visual Question Answering (VQA) job combines data processing issues with visual and linguistic processing challenges in order to answer basic 'common sense' questions regarding given images. VQA has piqued the interest of deep learning, computer vision, and natural language processing researchers. In terms of issue formulation, existing datasets, assessment criteria, and algorithms, we have attempted to assess the current state of VQA. We used a tailored version of the "MSCOCO VQA v1 validation" and "MSCOCO VQA v2 validation" datasets. We reviewed existing algorithms for VQA. (CNN-LSTM). To figure out where VQA its image understanding stands. Our proposed model was BERT-Vision Transformer. We applied multiple BERT-ViT approaches and found good Accuracies. The comparative study will be performed using state-of-the-art models such as CNN-GRU, CNN-CNN, BERT-ViT, and CNN-LSTM. This study aims to provide a detailed comparison between these models. Through this study, we hope to help future researchers decide which models could be used for the purpose of Visual Question Answering as VQA can aid in the development of user trust in machines and many more applications like Med-VQA, VQA-For Visual Aid, etc.

# Contents

# List of Tables

# List of Illustrations

# Acknowledgements

First and foremost, we want to express our gratitude to Allah for giving us the strength to fulfil our obligations and finish the report. The senior design project program, a component of the Bachelor of Computer Science and Engineering (BSc in CSE) curriculum, is immensely helpful in bridging the gap between academic knowledge and practical experience. This report aims to give theoretical knowledge through practical experience. We also express our sincere gratitude to all of the professors who played a crucial role in giving us the technical expertise and spiritual support we needed to finish the project successfully. We must thank **Dr Mohammad Ashrafuzzaman Khan**, an honorary member of our department, for his attention and support in helping us attain this goal. We will always be obliged to the ECE department at North South University for providing us with a course like **CSE-499B** that allowed us to work on and complete this project. We appreciate our friends and family's continuing support and moral assistance in seeing this endeavor through.

# 1 Introduction

In this chapter, we will discuss VQA, from where it originated, and give a general overview of the report.

Artificial intelligence (AI) has a branch called computer vision that enables computers and systems to recognize patterns and meaningful information in digital images, videos, and other visual inputs and then act or recommend actions based on that knowledge. If artificial intelligence allows computers to think, computer vision will enable them to see, watch, and comprehend. Computer vision functions similarly to human vision, except, Human vision is a step ahead of computer vision in terms of how it works. Lifetimes of context let human eyesight learn how to distinguish between objects, how far away they are, if they are moving, and if a picture is blurry. Computer vision trains computers to execute these duties. Still, it must do so in a much shorter time, using cameras, data, and algorithms rather than retinas, optic nerves, and the visual cortex. Because a system trained to inspect items or monitor a manufacturing asset can examine thousands of products or processes per minute, detecting imperceptible faults or anomalies can swiftly outperform human capabilities.

Multi-modal scene understanding is one of the ultimate aims of computer vision, which calls for a system to record various data types, including objects, actions, events, scenes, atmospheres, and their relationships at many distinct semantic levels. Despite substantial advancements in various recognition tasks in recent years, these works only concentrate on resolving comparatively straightforward recognition issues in controlled environments. As the computer vision research community has switched its emphasis away from "bucketed" identification and toward handling multi-modal problems, language and vision tasks like picture caption-in and visual question answering (VQA) has become more prominent. For instance, this impact has been found in the visual question answering and picture captioning tasks.

Research on VQA spans numerous AI fields, including CV, NLP, and Knowledge Representation & Reasoning (KR), which can reason about and extract ideas from processed media. Multiple studies cover diverse approaches to VQA methods. A Convolutional Neural Network (CNN) is used to encrypt the image in the VQA, and an RNN is used to encrypt and produce text. There is a lot of research on this strategy. The techniques for image captioning include, for instance, word embeddings trained on massive text corpora and CNNs trained on object identification to create a mapping from photos to text.

Visual question answering is a much more difficult challenge than image captioning, not the least because of the need to obtain data that is not present in the image. This could be common sense or specific information regarding the image's subject. There are various difficulties in learning where to look from question-image pairs. The whole picture might be the most helpful in identifying a specific sport when asked that question. Other queries require focusing on particular areas, such as "What is on the sofa?" or "What color is the woman's shirt?". So, we pay special attention to knowing where to look. This is a complex topic since it calls for integrating language and vision and learning to identify objects, use relations, and assess relevance. Questions like "What color is the walking stick?" or "Is it raining?" puddles, grey sky, or umbrellas in the picture can be utilized to detect the presence of rain, whereas paying close attention to the walking stick alone is necessary to determine its colour.

In our experiment, we attempted training and testing from both datasets, MSCOCO VQA v1 validation and MSCOCO VQA v2 validation, on VQA models CNN-LSTM, CNN-GRU, CNN-CNN, and ViT-BERT in a comparison study. Despite the fact that numerous types of research have been conducted on the introduction, management, and classification of the MSCOCO VQA dataset, our work is the first to compare computer vision-based systems of ViT-BERT against the traditional CNN-LSTM model; also, our work is the first to implement ViT-BERT of the MSCOCO VQA dataset. The current paper evaluates VQA proposals and explores how ViT-BERT can be used.

The rest of this article is organized as Section 2, which outlines the numerous previous studies in this field. Section 3 of the current paper presents the proposed methodology, followed by the experimental results in Section 4 with a detailed discussion. The report concludes in section 5, followed by possible future work of current work.

# 2 Related Works

In this research, A. Agrawal et al. [1] suggested a task that entails open-ended, free-form questions and human answers. Their purpose was to broaden the information and reasoning skills required to deliver the correct responses. This is crucial for success in this more demanding and unrestricted work. They used a VQA dataset that was two orders of magnitude larger than what had been used in prior VQA experiments. Other related actions are tied to the VQA work they submitted. In similar work, they recommended pairing an LSTM for the question with a CNN for the image to get a response. At each time step in their model, the CNN image features are used to condition the LSTM question representation, and the final LSTM hidden state is used to decode the response phrase sequentially. In contrast, the model proposed in this paper looks into "late fusion," in which the LSTM question representation and CNN image features are computed separately, fused via element-wise multiplication, and then passed through fully connected layers to generate a softmax distribution over output answer classes.

Their best model (deeper LSTM Q + norm I) performed better. The questions are encoded using a two-layer LSTM, while the visuals are encoded using VGGNet's final hidden layer. After that, the picture characteristics are normalized. The question and picture characteristics are translated to a common space and fused using element-wise multiplication to get distribution across replies. This is then sent via a fully connected layer followed by a softmax layer. To generate a unique embedding, the picture and question embeddings are joined. They concatenated the BoW Q and I embeddings for the BoW Q + I approach. The picture embedding is first changed to 1024-dim by a fully-connected layer + tanh nonlinearity to match the LSTM embedding of the question in LSTM Q + I and deeper LSTM Q + norm I approaches. Their best model (deeper LSTM Q + norm I, chosen using VQA test-dev accuracies on VQA test standard) has a 58.16 percent open-ended accuracy and a 63.09 percent closed-ended accuracy (multiple-choice). We can observe that their model outperforms both the vision-alone and language-alone baselines substantially.

In this research, For the objective of Visual Question Answering (VQA), J. Lu et al. [1] suggested negating these linguistic priors by making vision (the V in VQA) significant! Specifically, they attempted to balance the popular VQA dataset by gathering complimentary photos so that each question in their balanced dataset was connected with a pair of similar images, resulting in two possible responses to the question. By design, their dataset is more balanced than the original VQA dataset, with roughly twice the number of image-question pairs. They also used their balanced dataset to test various state-of-the-art VQA models. All models perform much worse on their balanced dataset, indicating that they have learnt to exploit linguistic priors. This discovery provides the first empirical proof for what appears to be a qualitative feeling among practitioners.

Several recent papers have suggested ways of creating 'explanations' for deep learning models' predictions, which are often 'black-box' and uninterpretable. In this paper, Y.

Goyal et al. [2] provided a natural language (sentence) description for picture categories. "Visual explanations," such as geographical maps superimposed on photographs, were provided to indicate the places the model focused on when making predictions. We offer a third explanation modality in this paper: counter-examples, which are cases that the model feels are similar to but not identical to the category predicted by the model. They hypothesize that properly training a model to answer questions on our balanced dataset would drive the machine to focus more on the visual signal as the linguistic signal has been degraded. They combined the dataset using VQA modelling. It incorporates a CNN embedding of the image, an LSTM embedding of the question, a pointwise multiplication to combine the two embeddings, and a multi-layer perceptron classifier to predict a probability distribution. This is a new attention-based VQA model that predicts a response by 'co-attending' to both the picture and the question. It uses the co-attention process to model the question and, as a result, the picture in a hierarchical way: at the word, phrase, and complete question levels. Y. Goyal et al. [2] also said, For the Visual Question Answering (VQA) task, the authors suggested overcoming various linguistic presumptions and making vision important. By gathering complimentary images, they precisely balanced the well-known VQA dataset so that every question in our balanced dataset was associated with one image and two similar photos that yielded two distinct responses to the question. Their dataset had around twice as many image-question pairs as the original VQA dataset and was, by design, more balanced. In addition to delivering an answer to the supplied (picture, question) pair, their data collection approach for detecting complementary photos allowed them to create a novel interpretable model that offered a counter-example-based explanation.

A. Khan et al. described [3] that to address Visual Question Answering, the authors introduced MMFT-BERT (MultiModal Fusion Transformer with BERT encodings) (VQA), maintaining separate and combined processing of a variety of input modalities. Their method benefited from multimodal data processing, adopting BERT encodings for (text and video) each separately and with a new transformer-based fusion technique to combine them.

L. Chen et al. [4] suggested a text and image processing Transformer-based multimodal item categorization (MIC) system. They evaluated a new picture classification model based on the Transformer. They looked into several methods of combining bi-modal data with a multimodal product data set obtained from a Japanese e-commerce giant.

A. Dosovitskiy et al. [5] demonstrated that there is no need for this dependency on CNNs and that pure transformers used directly on sequences of picture patches can get excellent results on image classification tasks. Vision Transformer (ViT) achieved good results when pre-trained on vast amounts of data and transferred to various mid-sized or tiny image recognition benchmarks (ImageNet, CIFAR-100, VTAB, etc.) while substantially requiring fewer CPU resources to train.

S. Antol et al. [6] advocated using an LSTM for the question and a CNN for the image in continuous processing to provide an answer. The final LSTM hidden state was utilised to successively decode the answer phrase in their model, which conditioned the LSTM question representation on the CNN image characteristics at each time step. However, the model created in this study experimented with "late fusion," where the CNN image features and the LSTM question representation were computed separately, combined via element-wise multiplication, and then passed through fully-connected layers to produce a softmax distribution over output answer classes. Concurrent with their work, they gathered Chinese questions and answers for COCO images that were then translated into English by humans. Employing COCO captions, four different question types (item, count, colour, and location) were automatically produced.

In this paper, J. Chenet al. [7] described that the authors developed several scalable model alternatives to factorise self-attention across the location, time, and modality dimensions. Additionally, they constructed and contrasted three different cross-modal attention mechanisms that could be effortlessly incorporated into the transformer building block further to examine the rich inter-modal interactions and their impacts.

# 3 Background & Design of the system

## 3.1 Transformer based Architecture Background Analysis & Design Principles

**3.1.1 Transformers:** The most prevalent sequence conveyance models are built on complicated recurrent or convolutional neural networks with an encoder and a decoder. The finest models additionally use an attention mechanism to connect the encoder and decoder. [8] Self-attention, also known as intra-attention, is an attention mechanism that connects several points in a single sequence to compute a depiction of the sequence. Reading comprehension, abstractive summarization, linguistic entailment, and learning task-independent sentence representations have all been effectively utilized with self-attention [9, 10].

**3.1.2 Vision Transformer:** The Vision Transformer (ViT) debuted in 2022 as a feasible alternative to convolutional neural networks (CNNs), which are now state-of-the-art in computer vision and hence widely deployed in many image classification applications. In terms of cognitive efficiency and accuracy, ViT models exceed the present state-of-the-art (CNN) by about x4. Transformer models have de facto become the norm in Natural Language Processing. Vision Transformers (ViTs) and Multilayer Perceptron (MLPs) have lately sparked interest in computer vision research. While the Transformer architecture has established the ultimate standard for Natural Language Processing (NLP) activities, its use cases in Computer Vision (CV) remain limited. Attention is utilized in computer vision either in tandem with convolutional networks (CNN) or to substitute some portions of convolutional networks while leaving their overall composition intact. ResNet-50 or ResNet-100, VGG-16 or VGG-19, YOLOv3, and YOLOv7 are examples of popular image recognition algorithms [11]. The Vision Transformer (ViT) model was introduced in a research article titled "An Image is Worth 16*16 Words: Transformers for Image Recognition at Scale" that was published as a conference paper at ICLR 2021. Neil Houlsby, Alexey Dosovitskiy, and ten other Google Research Brain Team researchers created and published it. The ImageNet and ImageNet-21k datasets were used to train the ViT models [5]. Transformers have been demonstrated to be quite effective in NLP and are now being applied to images. CNN leverages pixel arrays, whereas ViT splits the image into visual tokens. ViT split an image into fixed-size patches, linearly embeds each one, and adds positional embedding as an input to the Transformer Encoder. In terms of accuracy and computational efficiency, ViT outperforms state-of-the-art CNN. When trained on enough data, ViT beats state-of-the-art CNN with around four times less CPU resources. ViT's self-attention layer enables it to integrate information throughout the full picture. To recreate the visual structure from the training data, ViT learns to encode the relative placement of the patches. ViT is adjusted to a smaller dataset after being pre-trained on big datasets. The last pre-trained prediction head is eliminated during fine-tuning, and a zero-initialized feed-forward layer is then attached to predict the classes using the reduced dataset. A higher quality image than the one the model was pre-trained on can be used for fine-tuning, but the patch size should stay the same. Since Transformers lacks previous understanding of the visual structure, training the model takes longer and necessitates big datasets[12].

### 3.1.3 Vision Transformer Architecture

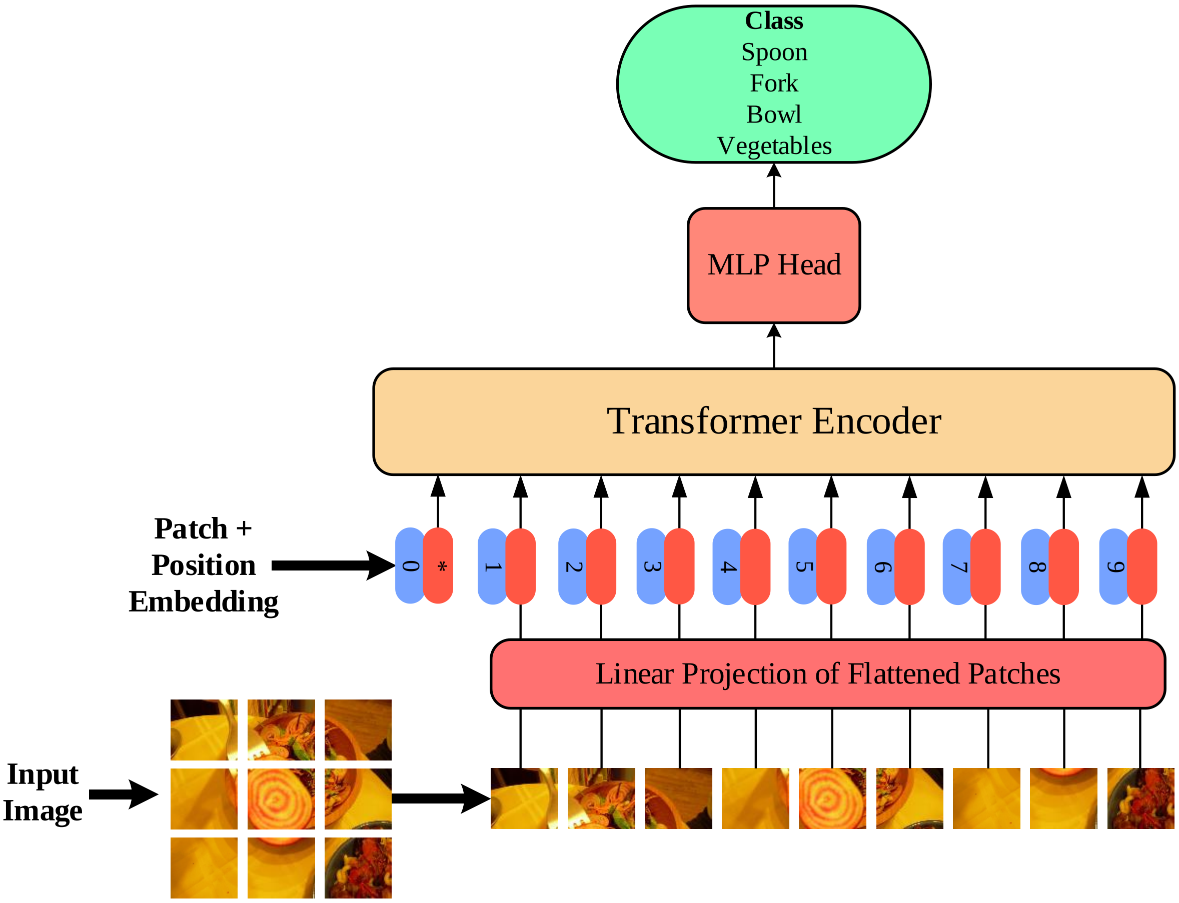


Fig.1 This Vision Transformer architecture is inspired from the study "AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE.[5]"

To begin, an image is transmitted through the vision transformer and then divided into fixed-size patches. The 2D picture of size H * W is divided into N patches, with N=H*W/P2. If the picture is 48 by 48 and the patch size is 16 by 16, the image will have 9 patches. Self-attention has a quadratic cost [12]. If we pass each pixel of the image as input, self-attention would need each pixel attending to every other pixel. The quadratic cost of self-attention will be prohibitively expensive and will not scale to practical input size; hence, the image is separated into patches. Linearly embed the 2D patches after flattening them to 1D patch embedding. By concatenating all pixel channels in a patch and then linearly projecting it to the required input dimension, each patch is flattened into a 1D patch embedding. To keep positional information, position embeddings are added to patch embeddings. Transformers are unconcerned about the structure of their input components. By adding learnable position embeddings to each patch, the model will be able to learn about the image's structure. At the beginning of the series, we add an extra learnable "classification token" to the patch embedding. This series of patch embedding vectors will be utilized as the Transformer Encoder's input sequence length[13].

### 3.1.4 Vision Transformer Encoder Design Principles

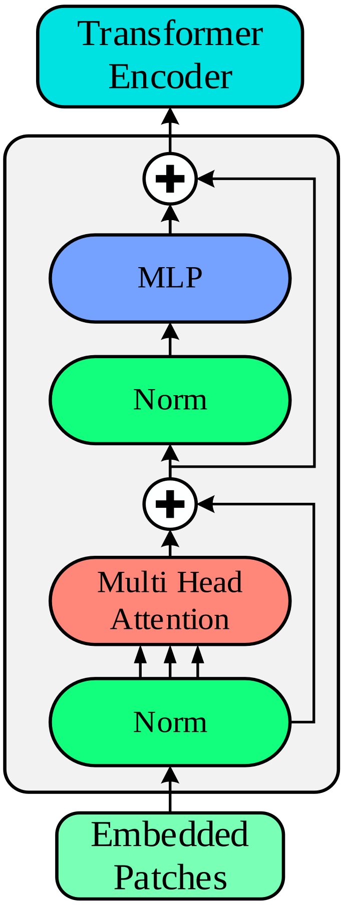


Fig.2 Vision Transformer Encoder

The Transformer Encoder uses a Multi-Head Self Attention Layer (MSP) to linearly concatenate various attention outputs to predicted dimensions. The image's various attention heads aid in learning local and global interdependence. MLPs are two-layer perceptron with Gaussian Error Linear Units (GELU)[14]. Layer Norm (LN) is used before each block since it introduces no additional dependencies between the training pictures. Assist in increasing training duration and generalization performance. Residual connections are used after each block because they allow gradients to travel straight through the network without going through non-linear activations. A classification head is constructed for image classification using MLP with one hidden layer during pre-training time and a single linear layer for fine-tuning. ViT's upper layers learn global characteristics, while the lower levels learn both global and local features. ViT can now learn more general patterns[15].

### 3.1.5 BERT Language Model

Bidirectional Encoder Representations from Transformers, or BERT[16], is a new work from Google AI Language researchers. By offering cutting-edge findings across a wide range of NLP tasks, such as Question Answering (SQuAD v1.1), Natural Language Inference (MNLI), and others, it has stirred up controversy in the machine learning field [17]. In the past, language models could only interpret text input sequentially, from right to left or from left to right, but not simultaneously. BERT is unique since it can simultaneously read in both directions. Bidirectionality is the name for this capacity, which the invention of Transformers made possible. BERT is pre-trained on two distinct but related NLP tasks—Masked Language Modeling and Next Sentence Prediction—using this bidirectional capacity. In Masked Language Model (MLM) training, a word is concealed within a phrase, and the software is instructed to guess the word that has been concealed (masked) based on the hidden word's context[18]. By masking (hiding) a word in a phrase and requiring BERT to utilize the words on either side of the covered word to anticipate the masked word, MLM facilitates or enforces bidirectional learning from text. No one had ever done this before. ML training can be efficiently parallelized thanks to the Transformer design. This makes it possible to train BERT on vast volumes of data in a short amount of time because to massive parallelization. Transformers employ an attention mechanism to notice how words relate to one another. Transformers are used in NLP models all around the world because to an idea that was first put out in the well-known "Attention Is All You Need" paper from 2017[19].

### 3.1.6 BERT Input Question Representation

The input question and passage are taken as a single packed sequence by BERT for the Question Answering System. Token and segment embeddings are added to form the input embeddings. At the start of the question, a [CLS] token is added to the input word tokens, and at the conclusion of the question and the paragraph, a [SEP] token is added. Each token receives a marking designating Sentence A or Sentence B. As a result, the model can discriminate between different texts. In the example below, every token with an A next to it belongs to the question, and every token with a B next to it belongs to the paragraph. BERT differentiates between the inquiry and the reference text using "Segment Embeddings." Simply put, BERT adds these two embeddings (for segments "A" and "B") to the token embeddings before sending them to the input layer. We give the start token classifier the final embedding of each token in the text. Every word is subject to the same set of weights when using the start token classifier. To create a probability distribution across all of the words, we first apply the SoftMax activation to the dot product of the output embeddings and the "start" weights. We choose the term that has the best likelihood of being the start token.

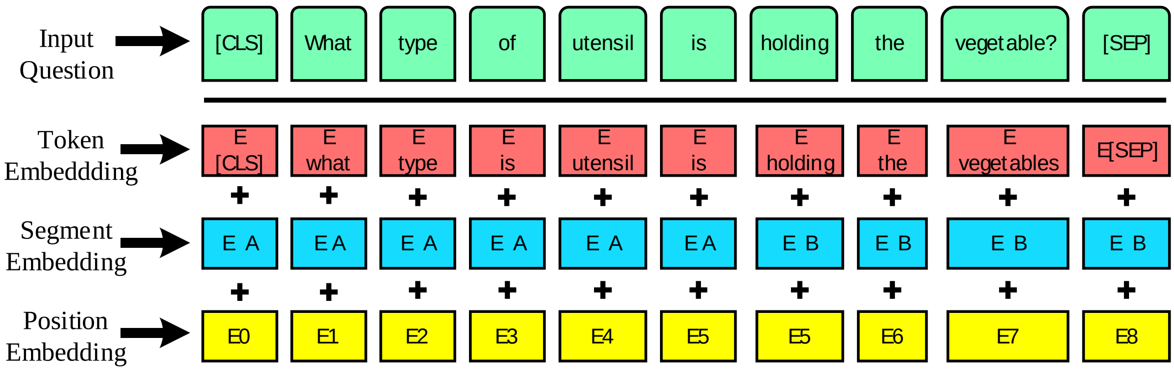


Fig.4 BERT Question Input representation

We utilized BERT base uncased pretrained model from hugging face for our project. Transformers serves as the foundation for BERT's model architecture. For language representations, it employs multilayer bidirectional transformer encoders. Two types of BERT models—BERT Base and BERT Large—are presented based on the complexity of the model architecture. The BERT Base model includes around 110M trainable parameters and 12 layers of transformer blocks with a hidden size of 768 and 12 self-attention heads. In contrast, BERT Large employs 24 layers of transformer blocks, has 16 self-attention heads, a hidden size of 1024, and around 340M trainable parameters. With only minor modifications, like as the addition of an output layer for classification, BERT employs the same model architecture for all tasks, including NLI, classification, and question-answering.

## 3.1.7 Building block of BERT Language Model

Transformers discard recurrent design in favor of relying exclusively on attention, resulting in enhanced parallelization and decreased training time. The transformer's encoder and decoder portions are made up of six identical layers of multi-head attention and feed-forward sublayers. Each sublayer takes a residual connection from previous inputs, adds it to the sublayer output, and normalizes it to generate the sublayer's final output. All sub-layers provide an output of size 512 to allow for residual addition. To overcome the vanishing gradient problem, all vectors Q, K, and V are packed into matrices for computationally optimum matrix multiplication after scaling by a factor K.

$$\text{Attentation}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{n}}\right)V$$

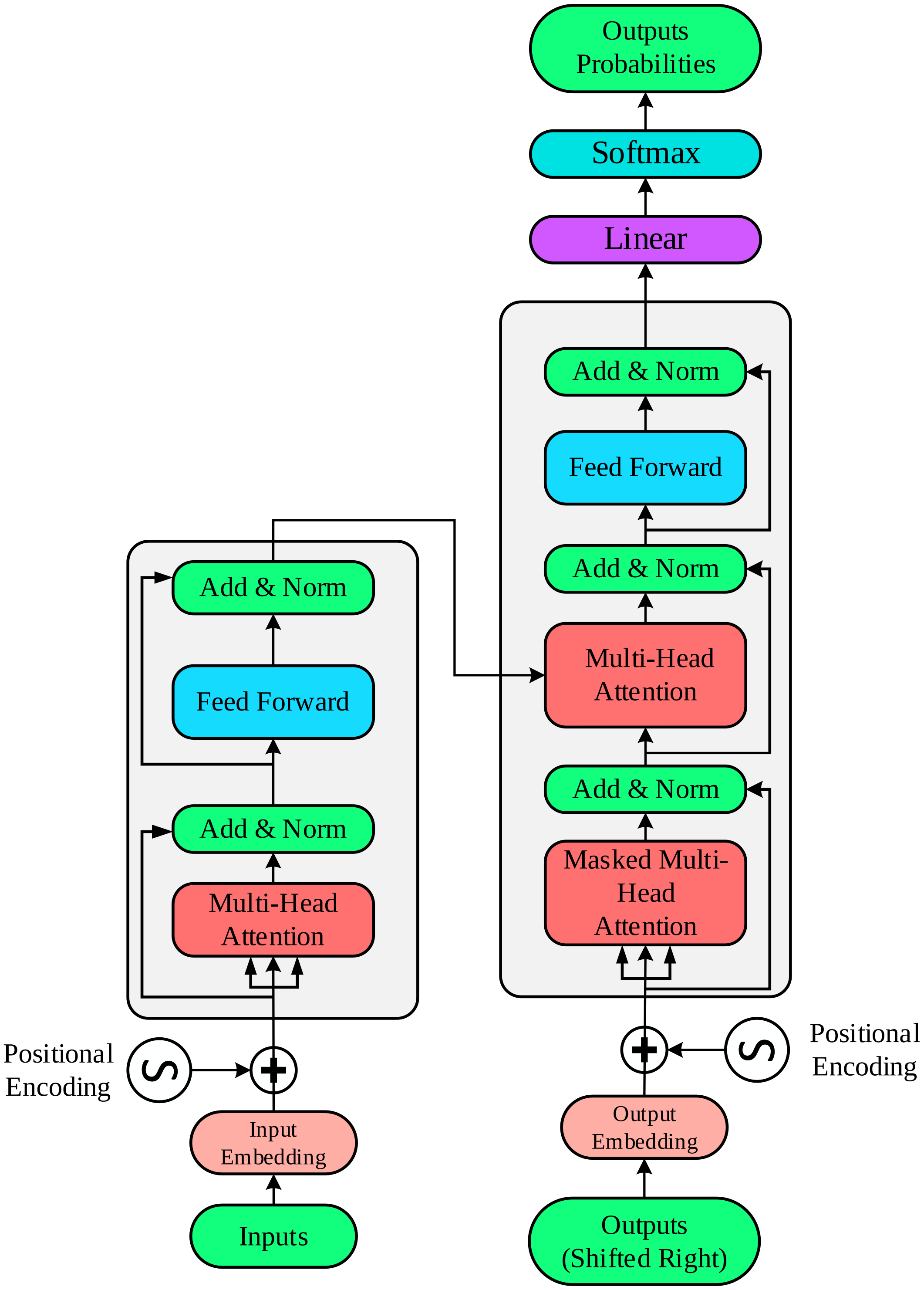


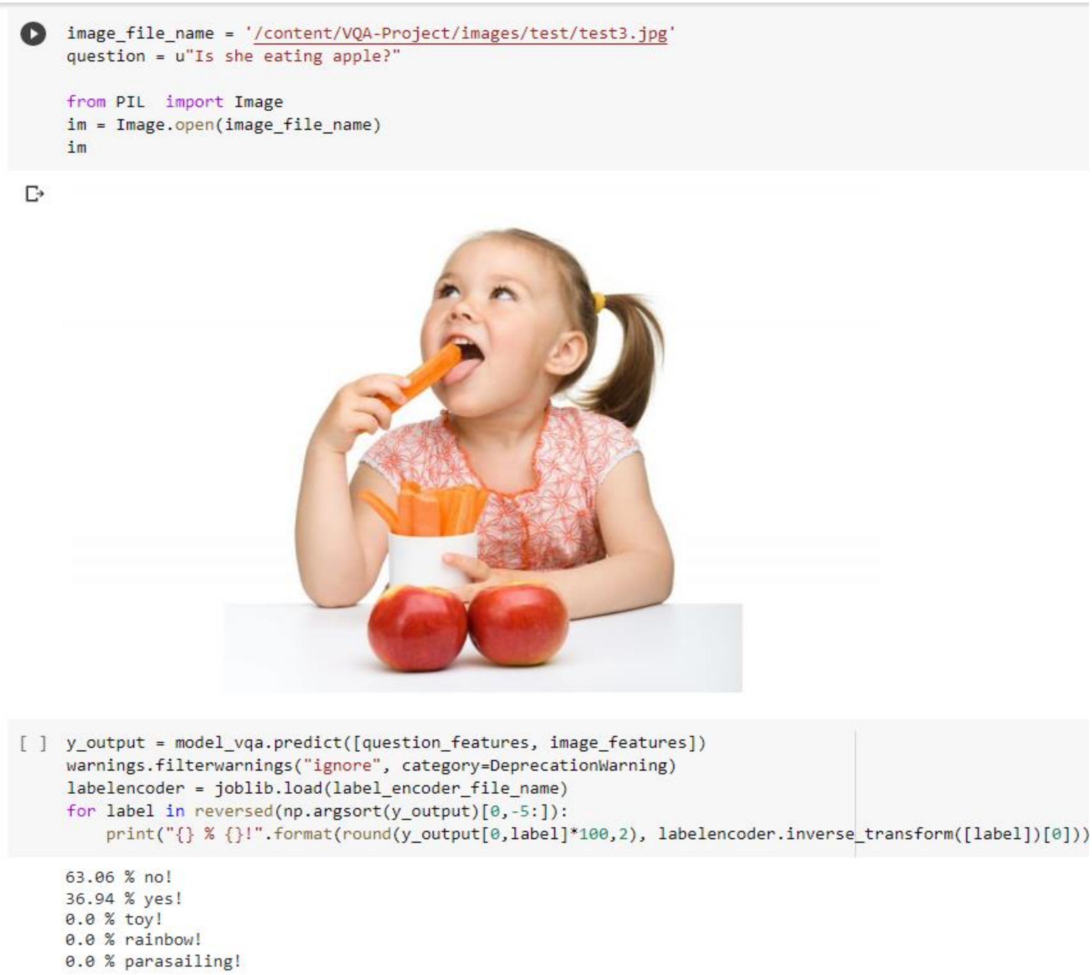
Fig.5 BERT Language Model's Encoder and Decoder

## 3.2 Usability, Manufacturability, Sustainability

Manufacturing is essential to the nation of origin because it creates high-paying jobs, spurs commercial innovation, aids in reducing the trade deficit, and contributes to environmental sustainability. Usability testing's objective is to understand how actual users interact with your product so that modifications may be made in reaction to the results. Sustainability enhances our quality of life, which also safeguards our ecosystem and protects natural resources for future generations. Together, these three determine the outcome of any project or work and are essential.

### 3.2.1 Usability

Our VQA project's objective is to make it as simple as possible for everyone to use. We will create an interface so everyone can access it efficiently, allowing our VQA system to work effectively as a robust system. Simply put, we much first have any random link from the internet place it in our model, write a question, and the VQA model will generate an answer. a GUI interface would have made it better, but our model is not deployed to a cloud service or in a hugging face-like platform; this is something we want to pursue in the future.



```
image_file_name = '/content/VQA-Project/images/test/test3.jpg'
question = u"Is she eating apple?"

from PIL  import Image
im = Image.open(image_file_name)
im
```

```
y_output = model_vqa.predict([question_features, image_features])
warnings.filterwarnings("ignore", category=DeprecationWarning)
labelencoder = joblib.load(label_encoder_file_name)
for label in reversed(np.argsort(y_output)[0,-5:]):
    print("{} % {}!".format(round(y_output[0,label]*100,2), labelencoder.inverse_transform([label])[0]))
```

```
63.06 % no!
36.94 % yes!
0.0 % toy!
0.0 % rainbow!
0.0 % parasailing!
```

Fig.6 VQA prototype built for capstone

### 3.2.2 Manufacturability

Specifically, VQA can take centre stage in computer vision, transforming numerous production areas. However, our work can be deployed to a cloud-based platform and be easily usable. Manufacturability is true for hardware; however, our work can only be deployed to a cloud server, and other developers use it to improve this topic further. On top of this, the work needs to be maintained. So, our products need no manufacturability, but the developers can utilize this, which can lead to the discovery of new work and resources. The ML algorithm will produce increasingly accurate insights and predictions as it consumes more data. Usually, training data is more extensive than testing data. This is because we want to provide the model with as much information as possible to identify and learn valuable patterns. Vit-BERT is overkill for the typical MSCOCO VQA dataset, so to take full advantage of this model, hopefully, more datasets will be made. The VQA community will undoubtedly appreciate this and pave the path for more work to be done.

### 3.2.3 Sustainability

It is possible that our efforts would not make much of a difference in the fight against climate change, but maybe it will be just one small step in the right direction. Using pre-trained models, we reduce climate change. Instead, this will drastically lower our electricity consumption. We might have needed four days or something, but thanks to this, we were able to complete our work in just a few hours. Using Google Colab Pro was another method we used to cut back on power usage as no GPU was used. In the long run, this will help cut down more power as more people work on VQA and our model.

# 4 Implementation of the System

## 4.1 Dataset

For this study, we have decided to use the official VQA dataset. The open-source image dataset MS COCO (Microsoft Common Objects in Context) contains 328,000 images of humans and familiar objects. Annotations in the dataset can be used to train machine learning models. There are 80 classes. VQA dataset is a recently collected dataset built with images in the MS-COCO dataset. However, each question has several linked answers that other people have annotated. We assess the suggested model using the VQA dataset's Real Image (Open-Ended) task. There are 81434 testing photos, 40504 validation images, and 82783 training images. The MS-COCO dataset contains images tagged with three questions, each with ten potential answers. There are 248349, 121512, and 244302 QA (QUESTION-ANSWER) pairings for training, testing, and validation, respectively. We solely use the single-word answer QA pairings in the VQA dataset, which make up 86.88% of all QA pairs in this dataset, to test the effectiveness of our strategy; 98% of answers do not exceed three words. For more details, you can refer to the following link: https://visualqa.org/vqa_v1_download.html.

The DAQUAR dataset, based on NYUDv2, is the first miniature QA benchmark constructed from RGB-D photographs of an indoor scenario. For training and testing, DAQUAR-all has 6,795 questions and 5,673 questions, respectively, with 894 categories of answers. In this dataset, some questions have a set of multiple answers rather than just one. BERT-ViT was previously only done using the DAQUAR dataset.

Multiple-choice questions are provided in the MS COCO Visual Question Answering (VQA) dataset. Three questions are associated with each image, and ten annotators recorded their free-response responses. Any response that comes from at least three annotators is considered correct.

The images in the dataset are all .jpg files. A unique image id is assigned to each image. JPG is a digital picture format that stores image data in compressed form. JPG pictures are quite small, with a compression ratio of 10:1. The JPG format saves vital image information. This is the most widely used picture format for exchanging photos and other images over the internet, as well as between mobile and desktop users.

The JSON file format is used to hold the questions. The data structure for the questions is as follows:

```
{
"info" : info,
"task_type" : str,
"data_type": str,
"data_subtype": str,
"questions" : [question],
"license" : license
}

info {
"year" : int,
"version" : str,
"description" : str,
"contributor" : str,
"url" : str,
"date_created" : datetime
}

license{
"name" : str,
"url" : str
}

question{
"question_id" : int,
"image_id" : int,
"question" : str
}
```

data_subtype: the type of data subtype (for example, mscoco's train2014/val2014/test2015, abstract v002's train2015/val2015).[3]

#### *4.1.1.1 3.4.3 Train/Test/Validation Input Annotations Format:*

The JSON file format is used to hold the annotations. The data structure of the annotations format is as follows:

```
{
"info" : info,
"data_type": str,
"data_subtype": str,
"annotations" : [annotation],
"license" : license
}

info {
"year" : int,
"version" : str,
"description" : str,
"contributor" : str,
"url" : str,
"date_created" : datetime
}

license{
"name" : str,
"url" : str
}
```

```
annotation{
"question_id" : int,
"image_id" : int,
"question_type" : str,
"answer_type" : str,
"answers" : [answer],
"multiple_choice_answer" : str
}
answer{
"answer_id" : int,
"answer" : str,
"answer_confidence": str
}
```

data_type: the image's source (ms coco or abstract v002).
data_subtype: the type of data subtype (for example, mscoco's train2014/val2014/test2015, abstract v002's train2015/val2015).
question_type: the question's type is specified by the question's first few words.
answer_type: the answer's type. "Yes/no," "number," and "other" are the only choices currently accessible.
multiple_choice_answer: most frequent ground-truth answer.
answer_confidence: the subject's belief in his or her ability to answer the question.

## 4.2 Pre-Processing

We iterated through the JSON data file and separated or extracted the question, question_id, images_id, answer, and question_len and then joined the relevant data to form a CSV file. This was used in the ViT-BERT model.

Find all the punctuations. Get all unique punctuation from the Top 200 questions. So, the punctuation we will focus on is " ' ", while we will remove it for the rest. For example: what's -> What is. We are going to use Wikipedia data. We need a "wikitable sortable" file from the HTML script. We have numbers here, and we will convert them to English names. Next, we will create a relationship between this coco data and training data. Each cocoid is related to the image_id of training and validation data. So, we will collect cocoid and create a mapping file for it, making the coco_vgg_id_map file. The Keras API's HDF5Matrix implementation can be used to directly load datasets in HDF5 format into memory, where they can then be used to train Keras deep learning models. We used the combination of algorithms CNN (for the Image part) and LSTM/GRU/CNN (for the linguistic part), and a second algorithm which uses Vision Transformer (Vit)(for the Image part) and BERT(for the linguistic part).

## 4.3 Traditional Approach or Early Fusion Architecture

For the traditional approach, we used CNN along with LSTM/GRU/CNN. Why are we

For CNN, as an image processing part, we used VGG16 trained images, which have been used in Visual Qa datasets. Since we are going to use the VGG16 architecture, we won't be training all the images. Instead, we will be using the image feature matrix from Stanford, which has already been trained on the VGG16 architecture. All these images are present in the training and validation data.

The vgg_feast.mat is the file where we have pre-trained model weights for the given dataset. So, it is trained on a very large dataset and we are using those features for our model.

The vgg_feature matrix is the features of images that have been fed into vgg16 architecture, with 16 layers of convolutional neural network and a Dense layer (with 4096 units), except the last dense layer (with 1000 units). So, each of the images has been fed into the model and which gave us the feature vector of each image, in the shape of 4096 X 1. Concatenating each feature vector of every image gives us the matrix of shape 4096 X 123287. Since, 82783 + 40504 = 1232287 (Training and validation data).

From the coco_vgg_id_map.txt file, we have images_id in the order of the columns of the vgg_feats matrix. Selecting top 10 image id from coco_vgg_id_map file, means selecting top 10 columns from the vgg_feats matrix. The vectors are dumped in the file full_img_features_train.h5.

### 4.3.1 LSTM

The Long Short-Term Memory Network, often known as LSTM, is a type of recurrent neural network (RNN) that is very good at predicting long sequences of data over time, such as words and market prices. Due to the presence of a feedback loop in its construction, it is different from a typical feedforward network. An LSTM cell has three separate gates: an input gate, an output gate, and a forget gate.

### 4.3.2 TextCNN

Convolutional neural networks for text, or TextCNN, are effective deep learning algorithms for tasks like sentiment analysis and question categorization that require classification of sentences. We need a word embedding layer and a one-dimensional convolutional network to use TextCNN. In order to process neural information in kernel space, or high-dimensional space, CNN uses an activation function [A-1].

### 4.3.3 GRU

Another recurrent neural network is gated recurrent units, or GRUs. The sole difference between it and an LSTM is that it only has two gates: a reset gate and an update gate. It also noticeably lacks an output gate. GRUs are typically simpler and quicker to train than LSTMs since they have fewer parameters.

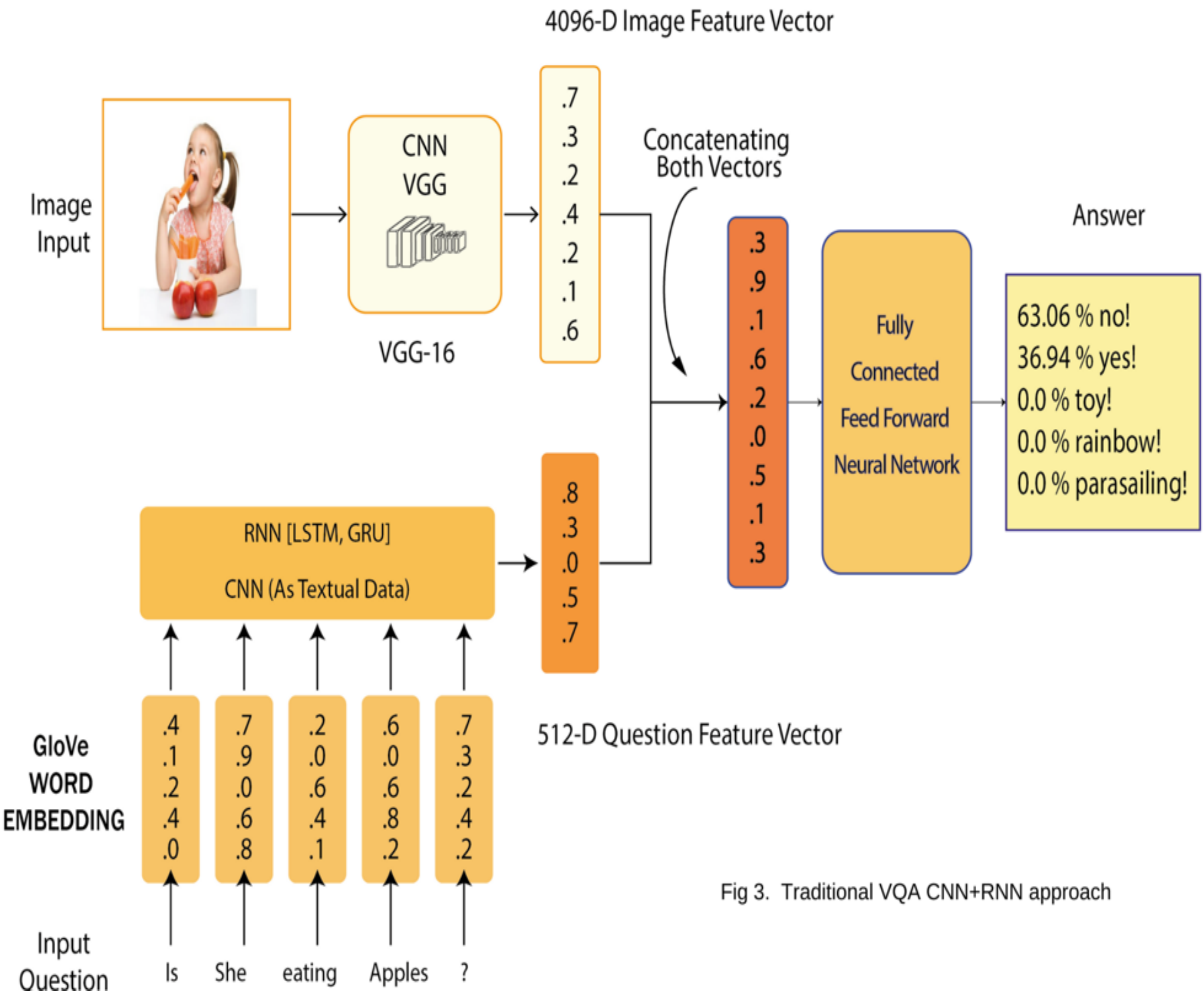


Fig 3. Traditional VQA CNN+RNN approach

### 4.3.4 **Implementation of the traditional approach**

The image vector and the word vector are concatenated and then passed through a feed forward neural network then being trained on the Training and Validation datasets. The top 10000 answers are then placed in a Sklearn Encoder and when the application is used, the answer is given via classification. The most common approach to VQA has been ABC-CNN or VAnilla VQA, or to put it simply, an implementation of CNN and LSTM. This model uses convolution kernels, also known as configurable convolutions, and the kernels are set using the LSTM-extracted question characteristics. The convolution kernels are then convolved with the image to produce a feature map that lends more weight to the relevant sections of the image.

## 4.4 Late Fusion Architecture Approach for the BERT-ViT Transformers

To collect data from the text and visual modalities as well as some cross-modal interaction, multimodal models can take on a variety of shapes. To complete the downstream job, fusion models merge the data from the text and image encoders into a single representation. A common VQA system fusion model includes the following steps:

### 4.4.1 **Featurization of image and question:**

After tokenizing the question, we obtained the embeddings by extracting features from the image. Simple embeddings (like GLoVe), Seq2Seq models (like LSTMs), or transformers can be used to feature the query. Similar to how primitive CNNs (convolutional neural networks), early layers of object recognition or image classification models, or image transformers may be used to extract picture information (Vision Transformer).

### 4.4.2 **Feature Fusion:**

In our case, it was necessary to jointly represent the characteristics from both modalities since VQA requires comparing the semantic information included in the picture and the query. Typically, a fusion layer was employed to do this, allowing cross-modal interaction between text and picture information to produce a fused multimodal representation.

### 4.4.3 **Answer generation:**

The right answers might be produced using only natural language generation (for lengthy or descriptive replies) or a straightforward classifier model (for one-word or phrase answers existing in a defined answer space), depending on how the VQA challenge was modelled. The individual feature extraction and feature fusion phases are carried out using the late fusion model described below:

### 4.4.4 Implementation of the Late fusion approach

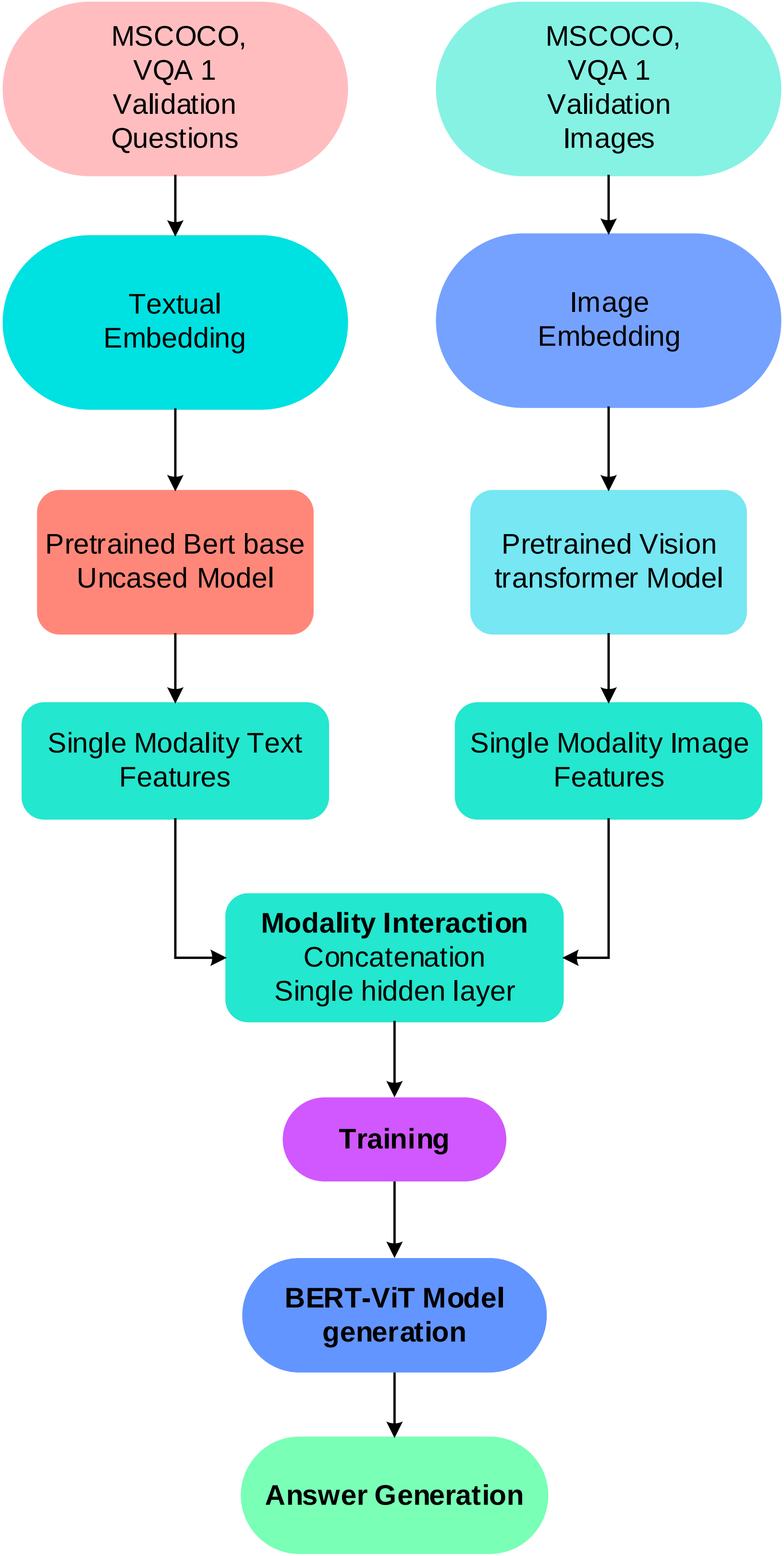

MSCOCO, VQA 1 Validation Questions
MSCOCO, VQA 1 Validation Images
Textual Embedding
Image Embedding
Pretrained Bert base Uncased Model
Pretrained Vision transformer Model
Single Modality Text Features
Single Modality Image Features
Modality Interaction
Concatenation
Single hidden layer
Training
BERT-ViT Model generation
Answer Generation

Figure.7 Implementation of Late Fusion Model Architecture.

# 5 Process of the development

The software development lifecycle, or SDLC, is a crucial component of the computer science industry. The SDLC has several steps. It mainly consists of the general steps that are used to create software. These procedures help in the activity planning of the software development team. The core steps, such as gathering requirements, defining the problem, designing the system, putting it into use, conducting tests, documenting it, providing support, and maintaining it, are typically the focus of the process. Although some of these processes may overlap, they serve to outline the project's phases generally. If some steps don't work out, the project might return to the previous one. Some businesses add or change a few extra steps since some procedures might need to be repeated to account for errors or changes. Some steps help the team understand the problem, which is the most crucial first step in discovering a solution. A system analyst is responsible for gathering requirements, outlining the issues, and creating the system.

For this project, we used an agile methodology. Agile refers to the capacity to react swiftly to requirements, technology, and people changes. Agility is dynamic, growth-specific, change-averse, and content-specific. It is used when you either don't know all of your needs or when they change. As part of the agile process, tasks are divided into smaller iterations or components that have no direct impact on long-term planning. Agile may produce quality that is considerably better than the waterfall model since it allows for constant testing and change. Quality is further improved by adding continuous integration to the system.

The characteristic of the agile process is modularity, which means dividing large projects into modules or activities. Another aspect is iterative; small modules are taken, which are mandatory. Essential tasks are taken in the first iteration, and the final product is made slowly by the end. Thus, if any changes are needed, they are applied in the next iteration. These tasks are usually time-bound. For example, we set a time limit of two weeks per iteration. Agile Project Development is also adaptive. During this process, there is a high chance of risk, so solution decisions are taken based on that. We complete the modules in each iteration as we go ahead, not in a single increment.

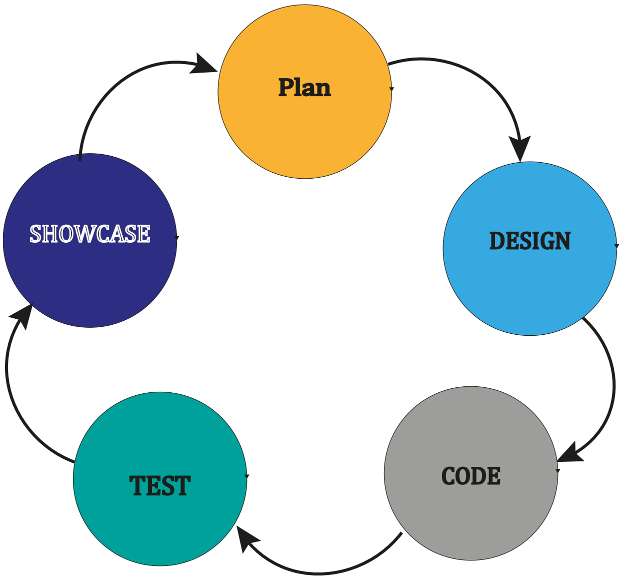


Figure.8 The Agile Project Development Methodology

We had initially planned to have a comparison between the traditional models; however, our supervisor insisted we take on the BERT-ViT model. Agile methodology helped us a lot during this transition. Also, agile methods allowed us to look back at our results and take a second training run with them, hoping that specific tweaks would help improve outcomes. We could stop correcting our errors and solving them as well. One particular problem lasted for two weeks; with agile, we had the freedom to work thoroughly and resolve it.

# 6 Economic, Social, Political, Health Impact

Computer vision is an area of artificial intelligence, as we all know. The goal of computer vision is to teach a computer how to "understand" a scene or image's qualities. Computer vision is a technology that is gaining traction, and it is critical that everyone involved in technology grasp the potential it offers as well as its current limits. Traditional computer vision is concerned with image and video processing, with the goal of reliably extracting information from pictures and videos. The human vision system, which is our most complex sense, is the driving force behind the development of computer vision. As a result, we examine real-world applications ranging from factory inspection systems to autonomous vehicles, from license plate recognition to robot interaction with the environment, and from face recognition to augmented reality. Visual Question Answering (VQA) is a relatively new problem in computer vision and natural language processing that has piqued the interest of deep learning, computer vision, and natural language processing communities. VQA requires an algorithm to respond to text-based inquiries regarding photos. Additional datasets have been provided since the initial VQA dataset was released in 2014, and several techniques have been developed.

In terms of issue formulation, existing datasets, assessment criteria, and algorithms, we intend to critically analyze the current status of VQA. We're particularly interested in learning about the limits of existing datasets in terms of their capacity to properly train and evaluate VQA algorithms. Then, instead of using typical LSTM-CNN, we aim to thoroughly analyze current VQA algorithms such as Vision Transformer (Vit) and other transformers. Finally, we'd want to see what future avenues VQA and image comprehension research may go. In this study, we will attempt to acquire access to ViT's self-attention layer, which allows us to embed information globally over the whole image. The model will also be able to learn from training data in order to encode the upward revision of picture patches in order to rebuild the image's structure.

Our ambition is to create a visual Question Answering (VQA) system that will allow you to obtain an answer from a picture when you ask a question about it. It is simple, efficient, and trustworthy to create the VQA model. Our VQA project will assist you in better understanding computer vision and implementing it more rigorously. We will first study our dataset before attempting to preprocess it. Finally, we use Vision Transformer (Vit) to create a model for the VQA that will be able to provide as accurate an answer as possible to the query regarding the provided image. Taking into consideration all of the factors, our objective is to create a VQA system that is as efficient as feasible, GPU-light, user-friendly, and dependable in order to provide you with an appropriate answer.

Compared to humans, a VQA system can concentrate on different aspects of an image, which can result in many options like, medical VQA, AI-based medical image understanding, associated medical question-answering, or even a VQA for persons who are blind. A VQA that automatically responds to common inquiries could enable blind individuals to live without visual limitations.

In the hot topics of Deep Learning, Computer Vision, and Transformers-based architecture, this research intends to shed light on 4th generation alternatives. This study aims to clarify some of these problems and provides a starting point for describing the environment for the next generation. The VQA community should pay close attention to the Vit & BERT paradigm to achieve performance capable of enhancing practical domain-specific applications. In the long term, we want to create an API that users can use in their software. To generate revenue, we intend to implement VQA for corporations and government organizations based on their needs, as this is far from over.

Humans and machines must work together to extract information from images; VQA immediately applies to various high-societal-impact applications requiring humans to elicit situationally-relevant information from visual data. This project has the potential to significantly advance daily life for users who are blind or visually impaired and to transform how society as a whole engages with visual information.

Advertising needs to be relatively straightforward to be easily understood by the largest possible audience while being entertaining and attention-grabbing. It's hardly surprising that VQA can keep working in the competitive advertising world. Understanding the advertisement and the underlying meaning is the first goal to be accomplished with VQA.

Our work is far from any of the things we want VQA to accomplish in the future. However, as said, it is the first ViT-BERT model with the official MSCOCO dataset; hence, it will be appreciated by the VQA community and pave the path for more work to be done on this.

# 7 Environment Considerations & Sustainability.

Researchers have discovered that the beginning of the Industrial Revolution, more than 180 years ago, was when climate change first started. This has resulted in an increase in sea level over time, which has threatened biodiversity, resulted in the loss of coastal land, altered precipitation patterns, and raised the risks of droughts and floods. Yes, there is scientific evidence for global warming and climate change, and there has been for many years.

The term "carbon emissions" refers to the release of carbon dioxide, a form of greenhouse gas that occurs naturally as well as a result of human activities like deforestation, electricity use, and industrial production. Emissions of greenhouse gases trap heat in the atmosphere, causing a variety of changes such as ozone layer deterioration, global warming, and ecosystem damage. Other greenhouse gases exist, but carbon dioxide is the most prominent.

While it is true that both plants and animals produce carbon dioxide, human activities such as the burning of fossil fuels, the production of electricity, and transportation have been a major contributor to the sharp rise in carbon dioxide emissions, which have reached levels that are beyond the capacity of nature to maintain a healthy balance.

Environmental considerations include taking into account social impacts such as respecting the human rights of indigenous peoples and other issues, as well as environmental effects on the air, water, soil, and ecosystems. It is possible that our efforts would not make much of a difference in the fight against climate change, but maybe it will be just one small step in the right direction. Using pre-trained models, we reduce climate change. Instead, this will drastically lower our electricity consumption. We might have needed four days or something, but thanks to this, we were able to complete our work in just a few hours. Using Google Colab Pro was another method we used to cut back on power usage. Google asserts that it has been carbon-neutral since 2007, which means that it has purchased an equal amount of renewable energy and carbon offsets to completely offset its net operational carbon emissions. For its data centers, Google Cloud strives for carbon-free energy.

Microsoft, a carbon-neutral enterprise, is the source of the MSCOCO VQA dataset. A Creative Commons Attribution 4.0 License governs the use of the MS COCO picture dataset. So long as you provide credit to the original author, this license permits you to share, remix, change, and expand upon your work, even for commercial purposes. This helps us save time and money. The photographs are also scraped from Flickr, which is an addition. Additionally, the annotations were somewhat human-assisted and partially computer-generated.

# 8 Ethical & Professional Responsibility

Even Though computer vision allows artificial intelligence systems to recognize faces, objects, places, and motions, it raises many ethical and privacy concerns. These are examples of fraud, bias, inaccuracy, and a lack of informed consent. There are no privacy or criminal problems because we are working with text and selected images. However, there is a problem of inaccuracy, which means that the model may mis-predict a word in cases where the test image presented is excessively blurry or deformed.

A big credit for the codes goes to Aditya Prakash[20] for work on VQA seven years ago. We implemented new features to his pre-existing work. But we still ran into huge problems, so we took codes from Jiasen Lu[21]. Merging their work with significant changes helped us run our traditional VQA approaches on our limited computational-powered devices. The ViT-BERT was always done on the Daquaor dataset, the result from Tazan Sahu[22] was from 2020, and significant changes had to be overtaken for the MSCOCO VQA dataset. We ran into over two weeks of problems, but we resolved them. Our work would not be complete without them; hence we want to give them credit.

The authors provide a cutting-edge, productive response in the paper VQA: Visual Question Answering. Only the traditional CNN and LSTM were used to implement this strategy. In contrast to their model, ViT and BERT produce very different kinds of prediction output. We suggest two additional algorithms in our study and train and test the BERT and ViT model using the MSCOCO VQA dataset.

The VQA's official website provides us with a dataset. The dataset was then downloaded to our local PC workstation. Then we uploaded the dataset to GoogleDrive and worked on Google Collab Pro. These are the zipped datasets, and the dataset is enormous.

The authors provide a cutting-edge, productive response in the paper VQA: Visual Question Answering. Only the traditional CNN and LSTM were used to implement this strategy. In contrast to their model, ViT and BERT produce very different kinds of prediction output. We suggest two additional algorithms in our study and train and test the BERT and ViT model using the MSCOCO VQA dataset.

# 9 Tools and Technology used

Our test framework was entirely done on Google Colab pro. Colab is particularly well suited to machine learning, data analysis, and education and allows anyone to create and execute arbitrary python code through the browser. It had Tesla T4 GPU and high-memory VMs (25 GB RAM) and 16GB GPU.

| Type Of Tool Or Technology | Description |
|---|---|
| String Tokenizer | A string can be divided into tokens by an application using the string tokenizer class. Tokenizers divide strings into lists of substrings. **NLTK** provides a function called **word_tokenize()** for splitting strings into tokens (nominally words). It splits tokens based on white space and punctuation. For example, commas and periods are taken as separate tokens. Contractions are split apart (e.g., “What's” becomes “What” “'s “). |
| Embedding layer | Using the embedding layer, we may turn each word into a fixed-length, pre-defined vector. Instead of only having 0s and 1s, the resulting vector is dense and contains actual values. A list of every word and its accompanying embeddings is referred to as an embedding matrix.<br>By tokenizing the data, you can restrict the areas where sensitive information is permitted and give tokens to users and programs that need to perform data analysis. This preserves the confidentiality of the original sensitive data while enabling a wide range of apps and processes to access the token data. |
| Glove and Word2Vec. | GloVe is a technique for producing vector representations of words using unsupervised learning. It is a Stanford-developed unsupervised learning technique that creates word embeddings by combining the global word-word co-occurrence matrix from a corpus. We built a GloVe word embedding.<br>We use a pre-defined word embedding that is accessible through the library. Most embeddings, including Glove and Word2Vec, are open-source and generally available if the data is not embedded. |

<table>
<tr><td>HDF5</td><td>Large, complicated, heterogeneous data can be stored in the open-source file format known as HDF5, or Hierarchical Data Format version 5. HDF5 is a cross-platform file format that contains data native to a computing platform binary format. The binary format is more effective for computers than text formats since it is a native format of computers. Datasets in HDF5 format can be loaded into memory and then used to train Keras deep learning models using the HDF5Matrix implementation of the Keras API.</td></tr>
<tr><td>Categorical_crossentropy</td><td>It is used as a loss function for a multi-class classification model with two or more output labels since we modelled the VQA task as multi-class classification.</td></tr>
<tr><td colspan="2">Firstly, we transformed the dataset from JSON formatted questions and annotations to CSV format for better comprehension. Then, we have used BERT (Bidirectional Encoder Representations from Transformer) text transformer for encoding questions and ViT (Vision - Transformer): 'google/vit-base-patch16-224-in21k' image transformer for encoding images. To process the question, we have used text_model='bert-base-uncased'. We have used two pre-trained models from the Hugging Face AI community.</td></tr>
<tr><td>Keras</td><td>Use the Keras module from TensorFlow Keras, a Python interface for artificial neural networks provided by an open-source software package. For the TensorFlow library, Keras serves as an interface. Many Keras libraries of which the stand will be the Keras's implementation of LSTM, GRU. To implement TextCNN and CNN, we imported from TensorFlow.Keras.utils Sequential, Model, SGD, Adadelta, Convolution2D, MaxPooling2D, ZeroPadding2D, Dense, Activation, Dropout Flatten, Embedding, concatenate, Conv1D, Input, Embedding, MaxPooling1D and many more.</td></tr>
<tr><td>Pytorch</td><td>For the pre-trained BERT and ViT models, the Pytorch framework is employed. An open-source machine learning (ML) framework called Pytorch is built using the Torch library and the Python programming language. One of the most popular platforms for deep learning research is this one.</td></tr>
</table>

| | |
|---|---|
| AutoTokenizer, and AutoFeatureExtractor | Imported from transformer packages from hugging face, the library will instantiate this generic tokenizer class as one of its tokenizer classes. |
| vgg_feast.mat | vgg_feast.mat is the file where we have pretrained model weights for the given dataset. So it is trained on very large dataset and we are using those features for our model. From coco_vgg_id_map.txt file, we have image_id in order of the columns of vgg_feats matrix. Selecting top 10 image id from coco_vgg_id_map file, means selecting top 10 columns from vgg_feats matrix. This axis dimension is required because VGG was trained on a dimension of 1, 3, 224, 224 (first axis is for the batch size even though we are using only one image, we have to keep the dimensions consistent |
| Activation function | A mathematical "gate" between the input driving the current neuron and its output is driving the next layer. They decide on whether or not to activate the neuron. They aid in concentrating our learning in a stream of practical changes. I'll make an effort to make the description less complicated. We used the likes of RELU, GELU, tanh, Softmax. |
| The Wu & Palmer similarity | We tested The Wu & Palmer similarity, a performance metric determining the semantic similarity between two words or sentences. It does so by taking into account the depth of the LCS (Least Common Subsumer) and the depth of the two synsets in the WordNet taxonomy. |

Table. 2: Tools and Tech.

# 10 Results Analysis

| Dataset | Model | Accuracy | Epoch |
|---|---|---|---|
| V1- VQA Validation | BERT-ViT | 0.46 | 20 |
| V1- VQA Validation | LSTM-VGG | 0.43 | 100 |
| V1- VQA Validation | GRU-VGG | 0.43 | 100 |
| V1- VQA Validation | CNN-VGG | 0.44 | 100 |

Table. 1: Results of the models used.

- The long short-term memory networks (**LSTM**) output shapes and gate recurrent unit networks (**GRU**) are similar. This is because they have similar structures. LSTM has three gates, one more than GRU: input, output, and forget. GRU is less complex than LSTM as it has a lower number of gates, only input and output. This is represented by the parameters being different. GRU performs better with long text and smaller datasets, while LSTM is claimed to perform better in other situations. GRU is around 25% faster than LSTM for model training speed while processing the same dataset. Although the dataset in this instance is not small, the text is lengthy, so the results are comparable.

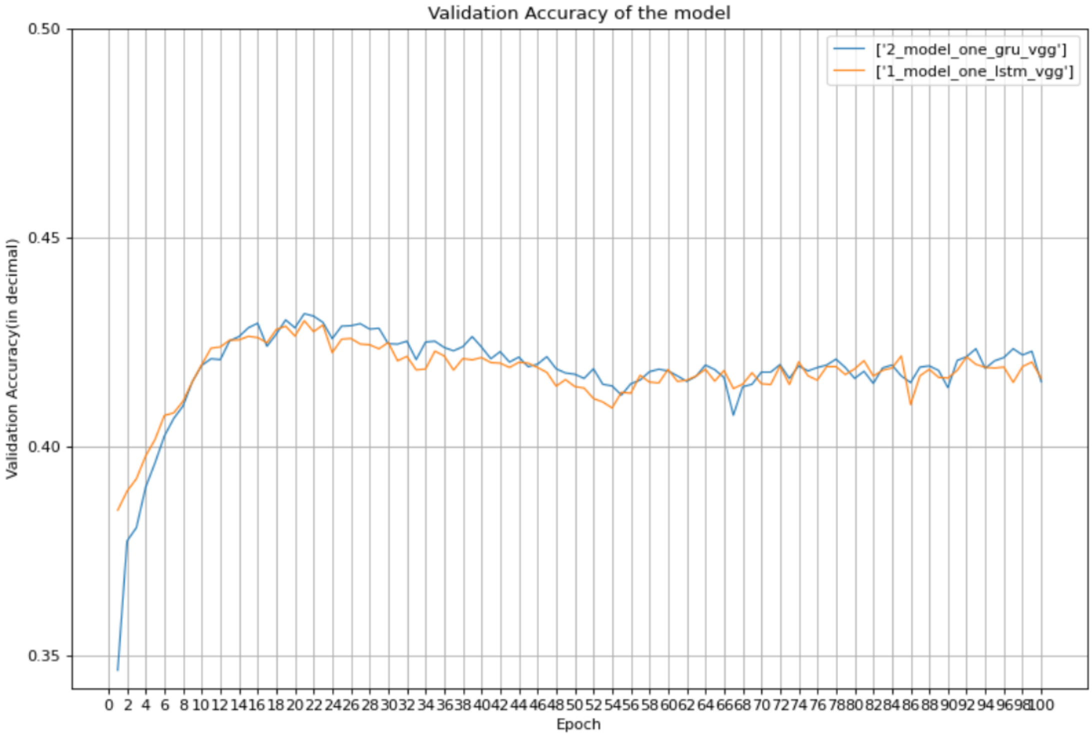
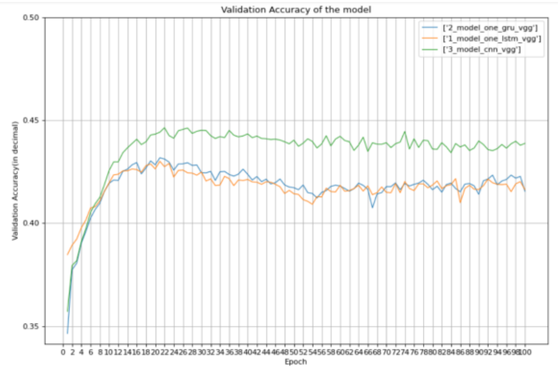


Figure.9 LSTM vs GRU

- TextCNN, or convolutional neural network (**CNN**) for processing textual data, seems much faster than a recurrent neural network (**RNN**). While a CNN learns to detect patterns over space, an RNN is trained to recognize patterns across time. Some RNN models include limited gates, or even a forget gate since RNNs are better suited to evaluating temporal, sequential inputs such as text or movies. An RNN's architecture differs from that of a CNN. CNNs are "feed-forward neural networks," combining pooling layers and filters. Compared to RNN, CNN appears to be much faster in computing time. Despite appearing to overfit throughout the training of our dataset, CNN's test accuracy was acceptable.



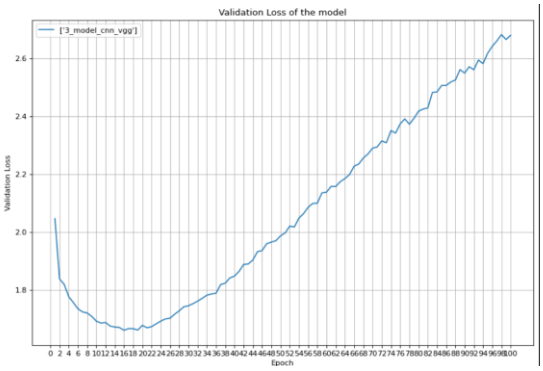
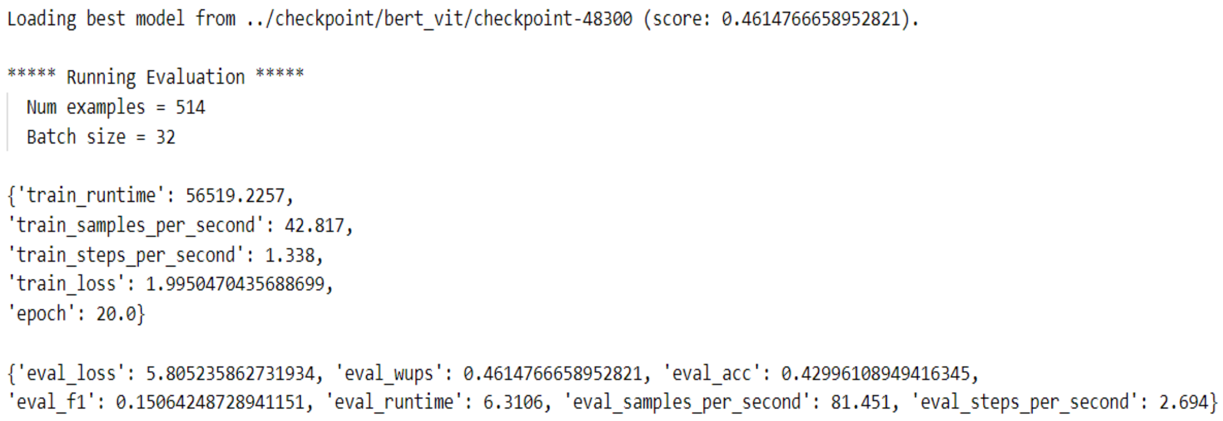


Figure.11 All Traditional models

Figure.10 Overfitting of TextCNN

- The standard model for visual data up until now has been CNNs. Recent research has demonstrated that (Vision) Transformer models (ViT) can perform as well as, if not better, on image classification tasks. Regardless of differing training settings, transformers are fundamentally more resilient than CNNs. By dividing a picture into fixed-size patches, accurately embedding each one, and including positional embedding as an input to the transformer encoder, the visual transformer can transform an image. As a result, the model may concentrate on a particular portion of the image rather than the entire thing at once, yielding a more accurate output. While training, Vision Transformer (ViT) produces excellent results with noticeably less computing power. That is overkill. Self-attention-based architectures, in particular Transformers, have emerged as the preferred model for natural language processing (NLP). The most popular approach involves pre-training on a significant text corpus and then fine-tuning it on a smaller dataset that is task-specific. Vision Transformers (ViT) are similar to Transformers in terms of their architecture, analyzing natural language, and CNNs in vision tasks. The pre-trained BERT performed marginally better than RNN, given the same resources and time, but there was no discernible difference. Although starting from scratch when training the BERT model could result in substantially better results, the necessary resources and costs are outside the scope of this study. Google results claim that **ViT beats state-of-the-art CNN** with four times fewer CPU resources when trained on enough data.

- Only the most crucial areas are given attention by Visual Transformer, which also uses self-attention to relate spatially distant concepts into a small number of visual tokens. These visual tokens can be projected back onto the feature map for semantic segmentation or utilized for picture classification. ViT employs multi-head self-attention to eliminate image-specific inductive biases during computer vision. To comprehend the local and global characteristics present in the picture, ViT separates the images into a series of positional embedding patches that are then processed by a Transformer Encoder. On a dataset that can be sustained as being large, ViT offers greater accuracy with less training time.



```
Loading best model from ../checkpoint/bert_vit/checkpoint-48300 (score: 0.4614766658952821).

***** Running Evaluation *****
  Num examples = 514
  Batch size = 32

{'train_runtime': 56519.2257,
'train_samples_per_second': 42.817,
'train_steps_per_second': 1.338,
'train_loss': 1.9950470435688699,
'epoch': 20.0}

{'eval_loss': 5.805235862731934, 'eval_wups': 0.4614766658952821, 'eval_acc': 0.42996108949416345,
'eval_f1': 0.15064248728941151, 'eval_runtime': 6.3106, 'eval_samples_per_second': 81.451, 'eval_steps_per_second': 2.694}
```

# 11 Conclusion

With our experiment, we aim to be able to tackle the problem in the VQA with the official VQA dataset from https://visualqa.org/, a large dataset and provide a proper comparative study. We have tested our recommended technique, and the literature we researched and the material we studied indicate a success. Even though numerous ways have been demonstrated to tackle the attention-grabbing part of the image problem in a dataset, we believe that fixing the problem via Vision Transformer and BERT will be a much more efficient procedure. The final results we obtained are compact and efficient.

We first introduced the datasets that we will use to benchmark the performance of VQA algorithms. The dataset introduced included real-world images, synthetic images, and additional annotations such as supporting world facts necessary to answer some visual questions. We then introduced different algorithms—a Vision Transformer algorithm and three traditional algorithms, which we grouped into specific categories based on the main contribution of the algorithm.

Instead of having models that perform well on some datasets but badly on others, future work would include more holistic models capable of answering issues that require external world knowledge, complicated reasoning, and dealing with synthetic data. Our focus will be on more common VQA datasets.

We utilized two sorts of transformers in our VQA project. BERT is one of them, while Vision Transformer is another. They are both pre-trained transformers. In this case, we acquired the transformers from the hugging-face AI community. Because the transformer architecture is pre-trained, it saves us time while training our BERT-Vit model. Transformers were first intended to aid with language translation. Unlike recurrent networks, transformers allow us to model lengthy relationships between input sequence pieces and facilitate parallel processing. Everyone wants a universal model that can accomplish many tasks accurately and efficiently.

Transformer models, like MLPs, are universal function approximators that approximate sequence-to-sequence functions. The concept of attention mechanism is used by both the Vision Transformer and the BERT. Google's Vision Transformers (ViT) are similar to Transformers in their architecture, analyzing natural language and CNNs regarding vision tasks. Images in ViTs are represented as sequences, with class labels anticipated, allowing models to learn picture structure independently. The adoption of the transformer architecture enabled substantial parallelization and translation quality optimization. CNNs, on the other hand, worked with a fixed-sized window and had difficulty capturing relationships at the pixel level in both the spatial and temporal domains. Transformers originally intended to tackle machine translation now show promising results in computer vision. The fact that ViTs outperformed CNNs in picture categorization was a significant achievement. However, they need time-consuming pre-training on large external datasets. ConViT beats the ViTs on ImageNet while providing much higher sampling efficiency. These findings suggest that Transformers can outperform CNNs in many computer vision applications.

There are basic LSTMs that may be trained from scratch for text categorization and variations of LSTMs that have been pre-trained to resemble BERT's Language Model. The distinction is that LSTMs are used to train language models and fine-tune them for classification, whereas BERTs employ transformers. The pre-trained BERT performed substantially better than LSTM but not significantly better because we provided the same resources to both models, but they required different amounts of time. Although the requisite resources and costs are outside the scope of this study, it is possible that building the BERT model from scratch using the same VQA dataset may lead to significantly excellent results.

On the official MSCOCO VQA dataset, we have successfully developed four VQA models, possibly the first to implement ViT and BERT, which the VQA community will very much appreciate. We have not yet created a demo for that model, though. We have not yet developed an API as suggested in this report, but we anticipate doing so after we make demos of all of our algorithms.

# 12 References

[1] A. Agrawal, J. Lu, S. Antol, M. Mitchell, C. Zitnick, D. Batra, & D. Parikh (2016, October 27). VQA: Visual question answering. Retrieved August 23, 2022, from https://arxiv.org/abs/1505.00468

[2] Y. Goyal, T. Khot, D. Summers-Stay, D. Batra, & D. Parikh (2017, May 15). Making the V in VQA matter: Elevating the role of image understanding in visual question answering. Retrieved August 25, 2022, from https://arxiv.org/abs/1612.00837

[3] A. U. Khan, A. Mazaheri, N. da V. Lobo, and M. Shah, "MMFT-BERT: Multimodal Fusion Transformer with BERT Encodings for Visual Question Answering," *arXiv:2010.14095 [cs]*, Oct. 2020, [Online]. Available: https://arxiv.org/abs/2010.14095

[4] L. Chen, H. Chou, Y. Xia, and H. Miyake, "Multimodal Item Categorization Fully Based on Transformer," *ACLWeb*, Aug. 01, 2021. https://aclanthology.org/2021.ecnlp-1.13/ (accessed Aug. 31, 2022).

[5] A. Dosovitskiy *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *arXiv:2010.11929 [cs]*, Oct. 2020, [Online]. Available: https://arxiv.org/abs/2010.11929

[6] S. Antol *et al.*, "VQA: Visual Question Answering," *openaccess.thecvf.com*, 2015.https://openaccess.thecvf.com/content_iccv_2015/html/Antol_VQA_Visual_Question_ICCV_2015_paper.html

[7] J. Chen and C. M. Ho, "MM-ViT: Multi-Modal Video Transformer for Compressed Video Action Recognition," *arXiv:2108.09322 [cs]*, Nov. 2021, [Online]. Available: https://arxiv.org/abs/2108.09322

[8] A. Vaswani et al., "Attention Is All You Need," arXiv.org, 2017. https://arxiv.org/abs/1706.03762

[9] A. P. Parikh, O. Täckström, D. Das, and J. Uszkoreit, "A Decomposable Attention Model for Natural Language Inference," arXiv:1606.01933 [cs], Sep. 2016, [Online]. Available: https://arxiv.org/abs/1606.01933

[10] R. Paulus, C. Xiong, and R. Socher, "A Deep Reinforced Model for Abstractive Summarization," *Openreview.net*,2018.https://openreview.net/forum?id=HkAClQgA-

[11] "Vision Transformers (ViT) in Image Recognition - 2022 Guide," *viso.ai*, Sep. 06, 2021. https://viso.ai/deep-learning/vision-transformer-vit/

[12] R. Khandelwal, "Visual Transformers: A New Computer Vision Paradigm," *The Startup*, Jul. 30, 2021. https://medium.com/swlh/visual-transformers-a-new-computer-vision-paradigm-aa78c2a2ccf2 (accessed Aug. 31, 2022).

[13] B. Wu *et al.*, "Visual Transformers: Token-based Image Representation and Processing for Computer Vision," *arXiv:2006.03677 [cs, eess]*, Nov. 2020, [Online]. Available: https://arxiv.org/abs/2006.03677

[14] “Transformers for Image Recognition at Scale,” *Google AI Blog*. https://ai.googleblog.com/2020/12/transformers-for-image-recognition-at.html

[15] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, “Transformers in Vision: A Survey,” *arXiv:2101.01169 [cs]*, Feb. 2021, [Online]. Available: https://arxiv.org/abs/2101.01169

[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” *arXiv.org*, Oct. 11, 2018. https://arxiv.org/abs/1810.04805

[17] Rani Horev, “BERT Explained: State of the art language model for NLP,” *Medium*, Nov. 10, 2018. https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270

[18] “What is BERT (Language Model) and How Does It Work?” *SearchEnterpriseAI*. https://www.techtarget.com/searchenterpriseai/definition/BERT-language-model

[19] “BERT 101 - State of The Art NLP Model Explained,” *huggingface.co*. https://huggingface.co/blog/bert-101

[20] "iamaaditya/VQA_Demo", link, 'https://github.com/iamaaditya/VQA_Demo'

[21] "jiasenlu/HieCoAttenVQA", link, 'https://github.com/jiasenlu/HieCoAttenVQA'

[22] "tezansahu/VQA-With-Multimodal-Transformers", link, "
https://github.com/tezansahu/VQA-With-Multimodal-Transformers