Multimodal Understanding Amharic Video Question Answering using Bidirectional Cross Modal Attention

Helina Tefera Getahun

A Thesis Submitted to

The Department of Computer Science and Engineering

College of Electrical Engineering and Computing

Presented in Partial Fulfilment of the Requirement for the Degree of Master's in Computer Science and Engineering

Office of Graduate Studies

Adama Science and Technology University

June 2025

Adama, Ethiopia

Multimodal Understanding Amharic Video Question Answering using Bidirectional Cross Modal Attention

Helina Tefera Getahun
Adivisor: Worku Jifara Sori (Ph.D.)

A Thesis Submitted to the Department of Computer Science and Engineering

College of Electrical Engineering and Computing

Presented in Partial Fulfilment of the Requirement for the Degree of Master's in Computer Science and Engineering

Office of Graduate Studies

Adama Science and Technology University

June 2025
Adama, Ethiopia

DECLARATION

I, hereby declare that this Master Thesis entitled "**Multimodal Understanding Amharic Video Question Answering using Bidirectional Cross Modal Attention** " is my own work and has not been submitted for the award of any academic degree, diploma, or certificate in any other university. All sources of materials that are used for this thesis have been duly acknowledged through citation.

Helina Tefera Getahun	_____	_____
Name of Student	Signature	Date

RECOMMENDATION OF ADVISORS/SUPERVISORS

I, the advisor of this thesis, hereby certify that I have read the revised version of the thesis entitled "**Multimodal Understanding Amharic Video Question Answering using Bidirectional Cross Modal Attention** " prepared under my guidance by **Helina Tefera Getahun** submitted in partial fulfillment of the requirements for the degree of Masters of Science in Computer Science and Engineering (CSE). Therefore, I recommend the submission of the revised version of the thesis to the department following the applicable procedures.

Worku Jifara Sori (Ph.D.)	_____	_____
Major Advisor	Signature	Date

APPROVAL OF BOARD OF REVIEWERS

I, the advisor of the thesis entitled "**Multimodal Understanding Amharic Video Question Answering using Bidirectional Cross Modal Attention** " developed by **Helina Tefera Getahun**, hereby certify that the recommendation and suggestions made by the board of examiners are appropriately incorporated into the final version of the thesis.

Worku Jifara Sori (Ph.D.)	_____	_____
Major Advisor	Signature	Date

We, the undersigned, members of the Board of Examiners of the thesis by **Helina Tefera Getahun** have read and evaluated the thesis entitled "**Multimodal Understanding Amharic Video Question Answering using Bidirectional Cross Modal Attention** " and examined the candidate during the open defense. This is, therefore, to certify that the thesis is accepted for the partial fulfillment of the requirement of the degree of Master of Science in Computer Science and Engineering (CSE).

_____	_____	_____
Chairperson	Signature	Date
_____	_____	_____
Internal Examiner	Signature	Date
_____	_____	_____
External Examiner	Signature	Date

Final approval and acceptance of the thesis proposal is contingent upon submission of its final copy to the Office of Postgraduate Studies (OPGS) through the Department Graduate Council (DGC) and College Graduate Committee (CGC).

_____	_____	_____
Department Head	Signature	Date
_____	_____	_____
School Dean	Signature	Date
_____	_____	_____
Office of Postgraduate Studies Dean	Signature	Date

ACKNOWLEDGMENTS

First, I would like to express my deepest gratitude to Almighty God for granting me the opportunity to start and complete my endeavors and for providing me the strength to persevere in all my pursuits.

I also extended my heartfelt thanks to my advisor, **Worku Jifara Sori (Ph.D.)**, who guided me from the beginning to the end of this journey. His positive ideas and comments on my work encouraged me greatly, and I will never forget the support he gave to all of us.

Lastly, I would like to thank my family members and friends for their unwavering support and encouragement during the ups and downs of my time at the University. I expressed my gratitude to all ASTU computer science and engineering staff members for their advice, motivation, and guidance in leading me into the academic research world.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF CODE SNIPPETS

LIST OF ABBREVIATIONS AND ACRONYMS

AI Artificial Intelligence

AMD Advanced Micro Devices

Amharic VQA Amharic Video Question Answering

BCMA Bidirectional Cross modal Attention Mechanism

Bi-LSTM Bidirectional Long Short-Term Memory

CCVQA CLIP based Cross modal Video Question Answering

CLIP Contrastive Language–Image Pre-training

CMA Cross modal Attention

CMDA Cross modal Dynamic Graph Attention

CNN Convolutional Neural Network

DNN Deep Neural Network

EC-GNN Event-Correlated Graph Neural Network

Englilsh VQA English Video Question Answering

FN False Negative

FP False Positive

FRCNN Fast Region based Convolutional Network (Fast RCNN)

GCMA Graph based Cross modal Attention

GCN Graph Convolutional Network

GNN Graph Neural Network

GPU Graphics Processing Unit

K-Means K-Means Clustering

L-GCN Location-aware Graph Convolutional Network

LSTM Long Short-Term Memory

mBERT Multilingual BERT

MCLIP Multilingual CLIP

MSRVTT Microsoft Research Video-to-Text

MSRVTT-QA Microsoft Research Video-to-Text Question Answering

MSVD Microsoft Research Video Description

MSVD-QA Microsoft Research Video Description Question Answering

MSVD-QA MSVD Question Answering

NLP Natural Language Processing

PCMA Pairwise Cross modal Attention

RNN Recurrent Neural Network

RoI Region of Interest

RPN Region Proposal Network

TGIF-QA Text based GIF Question Answering

TImeSformer Time-Space Transformer

TN True Negative

TP True Positive

VideoBERT Video BERT

ViLBERT Vision-and-Language BERT

ViT Vision Transformer

ViQA Video Question Answering

VQA Visual Question Answering

XLM-RoBERTa Cross-lingual RoBERT

ABSTRACT

Amharic Video multi modal Understanding for Amharic Video Question Answering using Bidirectional Cross Modal Attention is a novel deep learning approach designed to enhance the comprehension of Amharic video content through a fusion of visual and textual modalities. One of the primary challenges in video question answering is the heterogeneous nature of visual and textual data, especially in low resource languages like Amharic. Conventional approaches often rely on randomly sample video frames,did not consider sematic relation between object,and all of them are for english. To overcome these limitations, this study introduces a Bidirectional Cross Modal Attention mechanism with CLIP based best frame selection, which models fine grained interactions between video representations CLIP features, temporal embeddings, object features and the question encoding using BERT. Previous models either aggregate all visual features at once or treat the question as a global embedding, which results in loss of word level alignment and spatial temporal correspondence. In contrast, the Bidirectional Cross Modal Attention model allows both visual and textual tokens to attend to each other iteratively, improving semantic alignment between questions and relevant visual content.To further enhance understanding, multiple visual cues such as CLS tokens, CLIP embeddings, object detections from FastRCNN, and temporal spatial features are integrated. An bidirectional corss modal attention based fusion layer selectively combines these features. The Bidirectional Cross Modal Attention Bidirectional Cross Modal Attention VQA model not only introduces the first ever benchmark for Amharic Video Question Answering (Amharic VQA) but also achieves significant improvements over current state of the art methods on the English MSVD QA dataset. The Amharic Video Question Answering model achieved 48.21% accuracy, the English model using English MSVD QA reached 58.71%, showing a notable improvement compered with Yu et al. 2024 with final accuracy with 48.2% on MSVD QA and Tang et al. 2024 with final result 39.1%. These results highlight the effectiveness of fine grained, bidirectional attention in enhancing semantic fusion between video content and questions, improved Video Question Answering performance, particularly in English.

Keywords: Amharic Video Question Answering, Multilingual Video Understanding, Bidirectional Cross Modal Attention, BERT, CLIP, FastRCNN, multi modal Fusion, MSVD QA, Low Resource Language Benchmarking.

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

As deep learning evolved, it brought significant advancements to both computer vision and natural language processing, establishing them as central domains in AI research. Video Question Answering (Video QA) arises at the intersection of these fields, aiming to understand dynamic visual content and provide accurate, context aware responses to natural language questions. Unlike conventional tasks such as object detection, or static image based question answering, Video QA necessitates reasoning across both spatial and temporal dimensions. On the other paper it introduce other complexity as models must not only extract visual and semantic features from individual frames but also track the dependencies over time with step. This research is centered on constructing Video QA framework by smartly integrating pretrained vision and language encoders that enable multi modal reasoning over video sequences and textual queries(Lerner et al. 2024; Maaz et al. 2023; Ayyubi et al. 2025; K. Li et al. 2023; Fan et al. 2024).

Visual Question Answering is question answering method for visual input, in which the users issue queries as natural language sentences. The system provides answers from the visual input by understanding the relation between question and visual input. Most of the visual question answering systems focus on image question answering. While Image QA has drawn significant attention but Video QA is largely unexplored (Khurana and Deshpande 2021).

Video Question Answering is a task to give predict result for correct answer based on a question and a video. It is a challenging task which requires deep visual and textual understanding. It consists of video analysis and answering questions regarding its content based on extraction of corresponding visual features and context interpretation (Yang et al. 2020). This entails merging techniques such as video feature extraction, attention mechanisms, and natural language processing(NLP) models to bridge the vision language gap. This area has received extensive attention and studies from the scientific community (Z. Wu et al. 2025a; Kim et al. 2024; Song et al. 2025).

The Video Question Answering (Video QA) task have two general sub tasks, one is to represent text and video feature in a common representation maintaining critical visual and textual information and generating a valid and contextual sufficient answer from such a representation. This is particularly difficult due to the nature of videos as temporal,

wherein there is a need to represent the sequence of the frames and temporal interaction among objects.Early research in Video Question Answering (Video QA) primarily focused on frame level visual features and sequence modeling using CNN-RNN architectures. The MovieQA dataset by (Tapaswi et al. 2016; Chalk et al. 2016) was a pioneering effort that used movie clips with subtitles and scripts, laying the groundwork for multi modal reasoning. The TGIF QA dataset by (Jang et al. 2017) introduced spatiotemporal questions requiring models to reason over frame sequences, using C3D networks and LSTMs. MarioQA (Tran et al. 2017) explored synthetic video datasets to isolate temporal understanding challenges, while(Gidaris and Komodakis 2018) Cross Modal Attention Co-Memory Networks, which incorporated dual attention mechanisms for better contextual alignment between question and visual memory. These models were limited by coarse video features and lacked fine grained object representation, highlighting the need for improved spatial and temporal modeling techniques in future work.

Video Question Answering (Video QA) has significantly improved using cross modal deep learning techniques, multi modal fusion and object level modeling. These techniques enable models to process and align visual and textual modality information, which enables models to learn a better comprehension of video content based on natural language questions (D. Gao et al. 2022). New architectures like Cross modal Collaborative Generation (MCG) model have been introduced to further integrate visual and textual information to improve the accuracy and relevance of generated responses. Other models like Multi modal Iterative Spatial Temporal Transformer (MIST) seek to efficiently extract spatial and temporal dynamics from videos and improve the question answering process (Yu et al. 2024). These advancements underscore the potential of cross modal strategies toward increasing the effectiveness of Video QA systems as a step stepping standard more natural and efficient human machine interactions in all contexts(Z. Wu et al. 2025b).

Most of the paper mentioned have limitation lies in their restricted global context comprehension models tend to overlook the holistic frame level understanding and instead focus narrowly on local features. This leads to an incomplete interaction modeling across frames and their ability to handle complex temporal dependencies. Moreover, these models often depend heavily on the initial quality of object labling and lack mechanisms for capturing detailed relationships, semanticaly correct labling or interactions between objects, both spatially and temporally. This results in weak object level reasoning and limited capabilities in understanding relative object positions or meaningful relations. Additionally, many models ignore the importance of keyword aware object representation, which could otherwise improve alignment between textual queries and visual content (Tang et al. 2024; Yin et al. 2023; S. Ye et al. 2023).

Existing VQA systems are designed primarily for common languages like English. The latest census results have shown that Amharic is spoken by approximately 57.5 million people in Ethiopia of which 25.1 million individuals within the country have adopted it as a second language. In addition, next to Arabic, Amharic is the second most widespread Semitic language (Hailemariam et al. 2025). For being the second most widespread Semitic language, Creating an Amharic VQA system is critical for improving accessibility and information retrieval for Amharic users. It would enable more interaction with video content across education, media, and assistive technologies. However, to achieve that, it entails preparing Amharic VQA dataset, training models capable of understanding Amharic text, and adapting vision language techniques to work through the linguistic and structural characteristics of the language.

The development of an Amharic Video Question Answering system is therefore crucial not only to enhance question answering but also to ensure inclusive access to educational content, public service messaging, and assistive tools in video based platforms. This paper allow native Amharic speakers to interact with visual content through natural language, unlocking new possibilities in AI driven media literacy, remote learning, and digital inclusion(Joshi et al. 2024; Q. Wu et al. 2016)

Video QA remains a challenging and evolving field due to its need to integrate both visual understanding and language comprehension over time. Traditional Video QA systems often fall short in object level grounding, temporal reasoning and and precise cross modal alignment. These limitations become more prominent when dealing with low resource languages like Amharic, which lack sufficient training data and model support(Joshi et al. 2024; Khurana and Deshpande 2021)

To address these gaps, our research use a multi level attention mechanism to jointly embed video, frame, and object level visual cues with Amharic language questions. TimeSformer (Bertasius et al. 2021) is used for capturing temporal dynamics, while Faster RCNN detects key visual entities box. MCLIP serves as the shared semantic space, allowing the model to relate Amharic questions with visual inputs effectively. Multilingual Contrastive Language Image PreTraining (**MCLIP2022**), a multilingual extension of CLIP, is employed to accommodate Amharic text embeddings, enabling broader accessibility and semantic alignment. By applying cross modal attention over all visual abstraction levels, the system gains the ability to reason over spatio temporal content and understand complex queries, offering a robust foundation for Amharic Video QA applications in education, media, and assistive contexts.

1.2 Motivation of the Study

I grew up speaking Amharic, and I've always felt that people who speak it deserve better access to modern technology. That's why I'm working on building a video question answering system in Amharic. I want to help close the gap between English and other widely supported languages, so Amharic speakers can benefit from tools like AI in everyday life whether it's in education, media, or getting helpful information from videos. This project is my way of making sure Amharic has a place in the future of technology. I'm also really interested in how we can better understand flow of information in videos, so the system can give good answers.

As I've worked on building question answering systems for videos, I've come to see how important it is to really understand the flow of time and how things change from frame to frame. Many current models fall short they often miss details by only looking at random or evenly spaced frames, and they don't fully capture how objects interact over time with their semantic meaning. That's why I'm motivated to create something better. My approach uses Bidirectional Cross modal Attention with CLIP to bring together information about the visual feature, and textual feature. I want to make these systems able to key objects to the question, and give more accurate answers.

1.3 Statement of the Problem

With progress in artificial intelligence across computer vision, natural language processing, and , visual question answering (VQA) has garnered significant interest as a challenging Cross modal task (Tan and Sun 2025). There are many researchers trying to address the visual question answering problem. Earlier approaches for constructing models of Video QA only considered the frame level information from the videos. However, Video QA requires finding clues accurately on both spatial and temporal dimensions (D. Patel et al. 2021), recent approaches leverage powerful architectures such as Transformers, Graph Neural Networks (GNNs), and attention mechanisms to enhance video understanding and generate accurate answers.

Video QA have limitations some methods rely on heavily sub sampled information from videos or like using sparely sampled frames, to manage computational complexity (Guda et al. 2024) they mainly focus on this weakly sampled frames question answering and others only focus on only frame level representation (Y. Ye et al. 2017; Yang et al. 2022; Y. Li et al. 2022). Video QA aim to learn object, and frame interactions within a video. New approaches utilize a temporal aware multi modality to represent video content, and it has yielded encouraging outcomes (Z. Wu et al. 2025b). However, these approaches often struggle to effectively capture the dynamic changes of objects over time and lack support

for Amharic in video question answering. Among the reviewed papers, none addressed or support Amharic or employed advanced strategies for frame sampling Most studies focus primarily on extracting visual features (from sampled frames and spatial temporal relations) and textual features (from questions), while neglecting video captions which provide rich semantic summaries of key events which can be generated from the question and answer key words. Additionally, many models rely solely on visual features without integrating external semantic knowledge, limiting their reasoning capabilities and overall performance.

The complex characteristics and the vast character set of Amharic latter contributes to the limitations of the model's ability to learn patterns due to the predominantly trained on latin script like English. As a result the model struggles to understand Amharic input and tends to produce in accurate results on different tasks.

This research addresses the challenge of semantic knowledge in video question answering by considering temporal and spatial information and the usability of Amharic in the field of Visual QA specifically for video. This work also tackles the unexplored world of Amharic language AI systems for video question answering.

1.4 Research Questions

For the mentioned issues earlier, here are the questions to be answered by the study

1. How a bidirectional cross modal attention mechanism can be developed to support Amharic video question answering systems.
2. How can Amharic semantic information from video be utilized to effectively represent video information?
3. How to develop model that can integrate Amharic textual information and visual feature for better performance?

1.5 Objective

1.5.1 General Objective

This study aims to develop Amharic Video Question Answering (ViQA) system by using Bidirectional Cross Modal Attention.

1.5.2 Specific Objective

- To translate and prepare an Amharic dataset for video captioning and question answering using existing public English MSVD QA.

- To extract and align visual features and Amharic text embeddings for each Question Answer instance.
- To implement and train a Bidirectional Cross Modal Attention model that fuses Amharic textual and video features for accurate answer prediction.

1.6 Scope and Limitation

1.6.1 Scope of the Study

The study focuses on only two modality visual(video) and text for developing an Amharic Video Question Answering (ViQA) system by creating a dedicated dataset tailored to the Amharic language, addressing the current lack of resources. It incorporates Cross modal attention techniques to model spatial and temporal dynamics in videos more effectively. Additionally, the system enhances object level feature extraction using context aware methods to improve understanding of visual content. The Microsoft Research Video Description(MSVD) and Microsoft Research Video Description question answering(MSVD-QA) datasets have diversity in its content which make it challenging, and MSVD-QA for accessing question answer pairs aligned with video content. The system is trained and evaluated using standard performance metrics to identify the most efficient model for Amharic VQA.

1.6.2 Limitations of the Study

This study is prone to various limitations, mainly due to the lack of available datasets and resources for Amharic video question answering. Since there is no dedicated Amharic VQA dataset, the study relies on translating and adapting the MSVD and MSVD-QA datasets, which introduces translation inconsistencies and linguistic problems.

1.7 Contribution and Beneficiaries of the Study

1.7.1 Contribution of the Study

We integrated two pre trained models that were originally developed for separate and unrelated tasks and adapted them to work together for the purpose of Video Question Answering (VQA). We used TimeSformer which a transformer based model trained for video classification. While it performs well in identifying high-level patterns in video sequences but it is not designed for question guided reasoning and it simply outputs a predicted class based on the entire video without taking any external input like a question into account. On the language side we used BERT large pre trained language model developed for natural language understanding. BERT can generate rich semantic embeddings for questions and answers but it does not process or incorporate any visual

information by default. We developed a new architecture that allows these two distinct modalities share information. We achieved this using multi head bi directional cross modal attention. Our multi head bi directional cross modal attention model let the two modality communicate with each other followed by classifier network for answer classification. For the object level representation we use keyword guided object level representation instead of relaying on the fasrRCNN final class for the objects.
Here is a list of contract contributions of the study:

- **Novel Frame Selection Method Based on MCLIP:** A new approach is Bidirectional Cross Modal Attention for selecting the most informative video frames by performing K-means clustering followed by MCLIP based semantic similarity scoring between each frame and the video caption,question, and answer. This ensures that selected frames carry maximal semantic alignment with the content.
- **Semantic Object Representation via MCLIP Word Matching:** Instead of traditional object classification, objects detected using Fast RCNN are semantically labeled using similarity with CLIP encoded keywords from the question caption context. This enriches object features with textual grounding and improves reasoning capabilities and it use two different models.
- **Establishment of a New Baseline for Amharic Video QA:** This research sets a new benchmark for future Amharic language Video QA systems and contributes methods that can be generalized to other low resource languages in multi modal learning.

1.7.2 Beneficiaries of the Study

This study offers benefits to multiple area across different domains:

- **Amharic Speaking Communities:** The Bidirectional Cross Modal Attention system enables native Amharic speakers to engage with multimedia content and assistive technologies.
- **Researchers in Low Resource Natural Language Processing and Computer Vision:** The work presents a novel contribution to the field of video question answering in low resource settings. It introduces methods for multilingual and multi modal processing, which are applicable to other underrepresented languages.
- **Educational and Accessibility Platforms:** Learning platforms and accessibility tools can integrate the system to support Amharic language users in understanding video content, fostering inclusive digital education.

- **Developers of multi modal AI Systems:** The innovative strategies for frame selection, object representation, and Cross modal attention fusion are valuable for developers working on robust vision language systems across languages and domains.

1.8 Organization of the Thesis

The remainder of this thesis is structured as follows:

1.8. Organization of the Thesis

This thesis is structured into the following chapters:

Chapter Two presents the foundational concepts and related literature. It covers essential topics such as text encoding techniques, attention mechanisms, and the principles of video question answering. Additionally, it discusses the theoretical background of machine learning, deep learning, and Cross Modal Attention(CMA).

Chapter Three outlines the research methodology, including data collection procedures, preprocessing steps, and the tools and platforms used for model development. It also elaborates on the evaluation metrics employed throughout the study.

Chapter Four introduces the Bidirectional Cross Modal Attention model architecture in detail. This includes the structure of the text encoder, and visual feature extraction networks. Mathematical formulations, pseudocode, and the rationale behind each design choice are also provided.

Chapter Five describes the implementation of the Bidirectional Cross Modal Attention model. It discusses the setup of the working environment, the integration of Bidirectional Cross Modal Attention, and the training procedures. Different experimental configurations are explored and presented using code snippets.

Chapter Six presents the experimental results and analysis. It compares the outcomes of the Bidirectional Cross Modal Attention model with baseline methods, interprets the findings, and discusses their significance in relation to the research objectives.

Chapter Seven concludes the thesis by summarizing the key contributions and offering recommendations for future research directions.

CHAPTER TWO

LITERATURE REVIEW AND RELATED WORKS

Traditionally, software systems were developed according to rigid rule based approaches with a focus on numerical logic and static formulas. These approaches proved inadequate to model the dynamic and complex nature of real world data, particularly in visual and language understanding tasks. Artificial intelligence came into existence as a reaction to these inadequacies to create systems that can mimic human like perception and reasoning. In the Video Question Answering context, this break is even more prominent, as it enables machines to interpret visual information and reply to natural language questions in ways that simulate human perception. Such a revolution has also been central in academic controversy on the cognitive features and morals of creating AI technologies capable of multi modal data reasoning (A. Patel et al. 2025; Xiao, Zhou, Chua, et al. 2022; Bengio et al. 2013; Lei et al. 2018).

Early AI systems were built to work in isolated domains of data, resulting in autonomous research avenues for vision and language. In computer vision, (Krizhevsky et al. 2017)introduced AlexNet, a deep convolutional neural network that dramatically improved image classification performance on the ImageNet benchmark. The model was a breakthrough in visual representation learning based only on image data, with no use of textual or audio input. At the same time, natural language processing also evolved with Word2Vec models of (Mikolov et al. 2013),which learned distributed word vector representations by predicting word co-occurrence in large corpora. This model was able to identify semantic relationships between words but operated only in the text space without any external modalities such as vision. Similarly, (Hinton et al. 2012) developed a deep belief network for speech recognition that significantly improved the performance of acoustic models. This system processed only audio signals, demonstrating the strength of deep learning in unimodal speech tasks but again lacking the capacity for integrating visual or textual context.

The transition to multi modal learning began with challenges related to joint reasoning over vision and language. (Vinyals et al. 2014) introduced the Show and Tell model, which employed CNNs and RNNs jointly to generate image captions, as an early instance of visual and textual modality fusion. (Karpathy and Fei-Fei 2014) followed this up with fine grained correspondences between image regions and words to enable more detailed image description. Soon afterwards, (Agrawal et al. 2015) released the ViQA dataset, which established the task of image question answering, propelling the area towards more

profound Cross modal understanding and setting the foundation for attention based multi modal frameworks.

The employment of transformers further solidified multi modal learning. (Lu et al. 2019)introduced ViLBERT, which generalized BERT to vision language tasks with the use of two stream transformers. (Su et al. 2019) designed VideoBERT for joint modeling of video and text for video language learning. Such architectures opened the way to recent models like CLIP (Radford et al. 2021), which aligns vision and language in a common space by using contrastive learning.

Traditional deep learning models equally weight all input data, assigning the same weight to every element. In real world applications like translation, captioning images or videos, not every part of the input is as significant. For instance, in Video QA, every frame or object is not as useful in answering a question. The issue is addressed by attention mechanisms by allowing models to conditionally focus on the most significant parts of the input, hence prediction becomes more efficient and sensible.

The concept of attention first gained prominence through neural machine translation models. (Bahdanau et al. 2014) introduced the idea of learning where to attend in a sequence by computing attention weights dynamically. Later, the Transformer architecture by (Vaswani et al. 2017) replaced recurrence entirely with self attention, allowing models to compute dependencies regardless of distance in the sequence. Understanding how self attention operates through the interaction of query, key, and value matrices is fundamental for appreciating the mechanism behind Cross modal attention.

2.1 Attention Mechanism

Attention, essentially, is an active weighting system that distributes different proportions of the input from various parts based on their salience to a particular task. Attention scores are computed relative to a set of keys and then applied to weigh the corresponding values. The attention module's output is weighted sum. The benefit of attention is it allows models to pay attention to salient features adaptively rather than viewing all input equally which makes it good building block for our thesis.

Attention other than NLP tasks it was extended to vision tasks.(Vinyals et al. 2014) introduced attention to image captioning which let the models to selectively focus on different regions of an image when generating every word in the final caption. This demonstrated that attention could help models reason spatially in images. The same as

the image attention mechanisms were adopted in video modeling particularly for tasks like action recognition and video captioning because temporal attention helped models focus on relevant frames in the sequential order. This laid the groundwork for applying attention to multi modal tasks like Video QA.

2.1.1 Understanding Attention Mechanism

Attention, at its fundamental level, is a mechanism that dynamically allocates weights to different portions of the input based on their relevance to a particular task. It enables the model to selectively concentrate on relevant information and suppress parts that are not relevant. The attention mechanism is managed by three parts: queries (Q), keys (K), and values (V). These are learned projections from the input embeddings.

To compute attention,let see the follows steps:

1. Compute the raw attention scores using the dot product between the query(Q) and all keys(K):

$$Q = XW^Q, \quad K = XW^K, \quad V = XW^V \tag{1}$$

Where:

- X is the input sequence matrix
- W^Q, W^K, W^V are learnable weight matrices for query, key, and value
- Q, K, V are the resulting query, key, and value matrices

2. Scale the scores by the square root of the key dimension(dk) to stabilize gradients:

$$\text{score}_{ij} = \frac{Q_i \cdot K_j^\top}{\sqrt{d_k}} \tag{2}$$

Where:

- Q_i is the query vector for the i^{th} input
- K_j is the key vector for the j^{th} input
- d_k is the dimension of the key vector (used for scaling)

3. Apply the SoftMax function to transform scores into attention weights:

$$\alpha_{ij} = \frac{\exp(\text{score}_{ij})}{\sum_{j'} \exp(\text{score}_{ij'})} \tag{3}$$

Where:

- α_{ij} is the attention weight for V_j with respect to Q_i

4. Multiply the weights with the corresponding values and sum them to produce the output:

$$\text{Attention}(Q_i, K, V) = \sum_j \alpha_{ij} V_j \tag{4}$$

Where:

- V_j is the value vector for the j^{th} input
- $\text{Attention}(Q_i, K, V)$ is the output for the i^{th} input

This is a process by which each output token gets to view over all the input tokens with varied weights. When used in sequences, this results in a dynamic mechanism that can manage contextual dependencies. In Multi head attention, several sets of Q, K, and V are used in parallel such that the model has an opportunity to capture different kinds of relationships and then mix them appropriately.

Based on the input sources, attention can also be categorized as self attention, where computed by using the same input vectors, and cross attention, where attention is computed by using vectors coming from 2 different sources. Further distinctions include global versus local attention, and single head versus Multi head attention, the latter being a key component of transformer architectures, allowing the model to capture information from multiple representation subspaces.

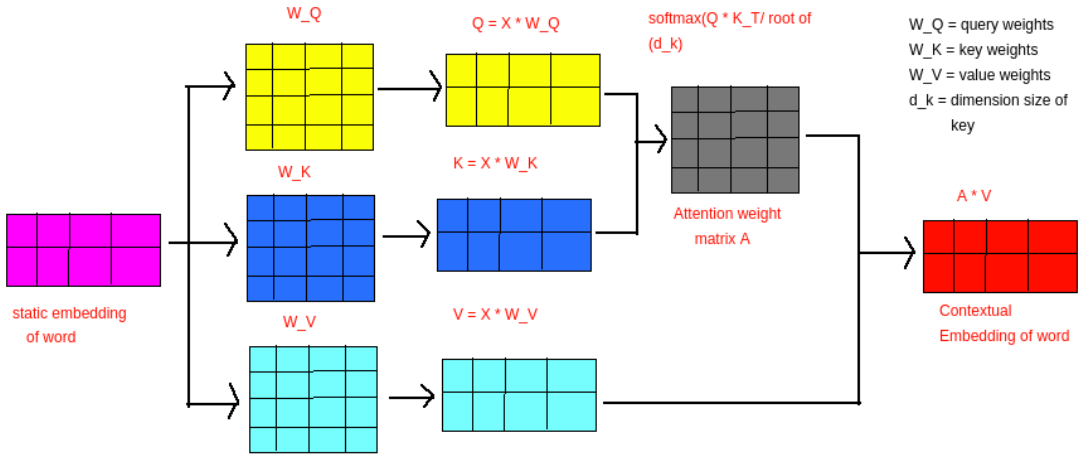


Figure 2.1: How self attention works Source: Taken from (A Review of Mechanism of Transformers — by Shreya Srivastava| Analytics Vidhya | Medium, n.d.)

2.1.2 Multi Head Attention

Multi head attention is a smart way to be one of our model building block for our model to look at different sub parts of a sentence and video at the same time. Instead of just

focusing on one thing, it breaks the input into smaller chunks called heads and lets each head pay attention to different parts. One head might focus on the subject of a sentence, another on the action, and another on the object and for the video part one head might focuses on object on different level. After all the heads do their job, the model puts everything back together to get a better overall understanding and the same parallel self attention that discussed on attention mechanism sub section. This helps our bidirectional cross modal attention learn more about the meaning and structure of the input, like how words or frames are related to each other. It works well because it can catch different patterns and meanings all at once, making the model smarter and more accurate in tasks like translating, answering questions, or understanding videos.

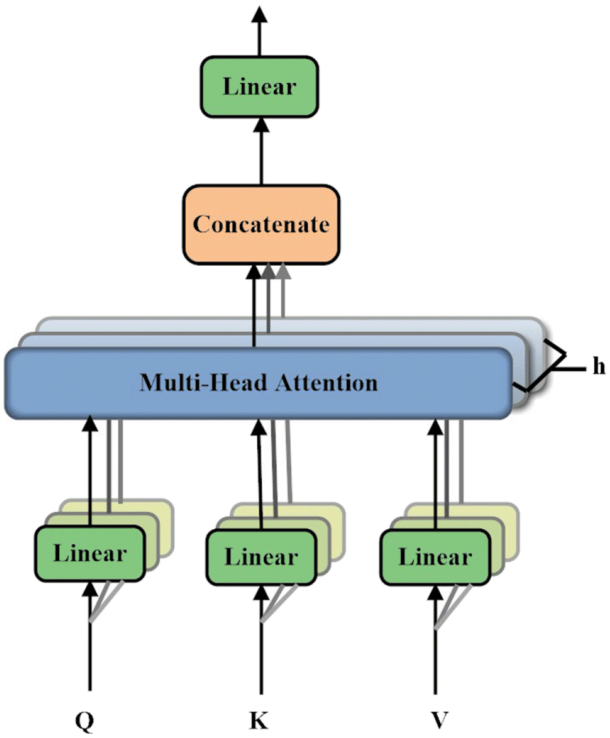


Figure 2.2: Architecture of Multi head Attention : Taken from (Vaswani et al. 2017)

2.2 Text Encoder

A text encoder is a model which maps input text into a dense vector representation to be used for downstream tasks. It plays a highly critical role in capturing the semantics of language and making it accessible to neural architectures. The encoder typically involves several stages like tokenization, embedding, and transformation through deep layers like transformers. It must be tokenized and embedded first before processing by the encoder. Tokenization step in text preprocessing is the process of converting raw text into word sized units of smaller units like words, subwords, or characters.

After tokenization the tokens are converted into dense numerical vectors through word

embeddings. These embeddings are either pretrained like GloVe or word2vec or learned during model training. Contextual embeddings introduced by models like Bidirectional Encoder Representations from Transformers(BERT) allow the meaning of a word to change depending on surrounding context. This is particularly important in attention based architectures. This method has evolved with the introduction of deep contextual embeddings. BERT introduced a method where word representations are derived from the context in which they appear through transformer layers. It differs from previous methods in that BERT provides different embeddings for the same word based on the sentence it appears in. The dynamic representation is suitable for syntax, semantics, and disambiguation. Contextual embeddings, introduced by models such as BERT, allow the meaning of a word to change depending on context. This is particularly important in attention based architectures.

The Transformer model accepts these embeddings as input and appends positional encodings to them, which represent token position. The Transformer architecture, introduced in the seminal paper "Attention Is All You Need" by Vaswani et al. 2017), consists of several layers, each consisting of Multi head self attention and feed forward sublayers. In self attention, one token attends to all the other tokens in the sequence so the model can create contextualized representations. These then get transformed through the feed forward networks before feeding into the subsequent layer.

2.2.1 Bidirectional Encoder Representations from Transformers

Bidirectional Encoder Representations from Transformers (BERT) is built entirely on Transformer encoder layers. It is trained using a masked language modelling objective, where random tokens are hidden and the model learns to predict them, enabling bidirectional context understanding. BERT's encoder captures rich semantic relationships and has been widely adapted for tasks involving text encoding, including those in multi modal settings like Video QA. One key factor that makes BERT suitable for multilingual tasks is its ability to share a common WordPiece vocabulary across languages and learn language agnostic representations during pretraining. Models such as Multilingual BERT (mBERT) and XLMRoBERTa have shown strong cross lingual transfer performance by training on large scale multilingual corpora (Devlin et al. 2018; Conneau et al. 2020). This enables effective generalization across languages, even for those with limited resources.

2.3 Multi modal Learning in Video Question Answering

multi modal representation learning addresses how to process and synchronize data across multiple modalities like video and text. Such learning is essential because data in the real

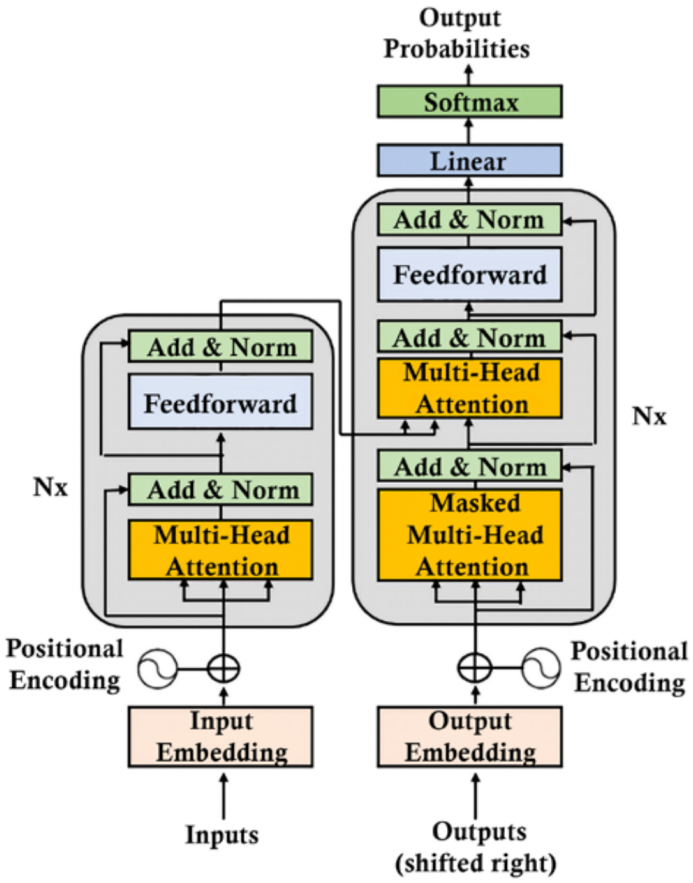


Figure 2.3: The Transformer – model architecture : Taken from (Vaswani et al. 2017)

world is seldom unimodal. For instance, a video's meaning depends on visual context, temporal information, and frequently text or speech descriptions. multi modal learning will bring together these separate and different but complementary sources of modal into one coherent model.

Earlier multi modal learning approaches were grounded in direct feature concatenation, where features preextracted from various modalities were concatenated before being input to a classifier. This couldn't learn intricate relationships between modalities. To address this, more advanced approaches were suggested like joint embedding spaces where data of each modality is mapped to a shared latent space that preserves semantic similarity.

The recent developments have revolved around deep learning based fusion mechanisms. They comprise bilinear pooling, gated fusion, and attention based fusion, which allow for more context sensitive and flexible modality integration. Transformer based architectures further improved multi modal learning by enabling dense interactions between modalities through cross attention layers (Tsai et al. 2019).

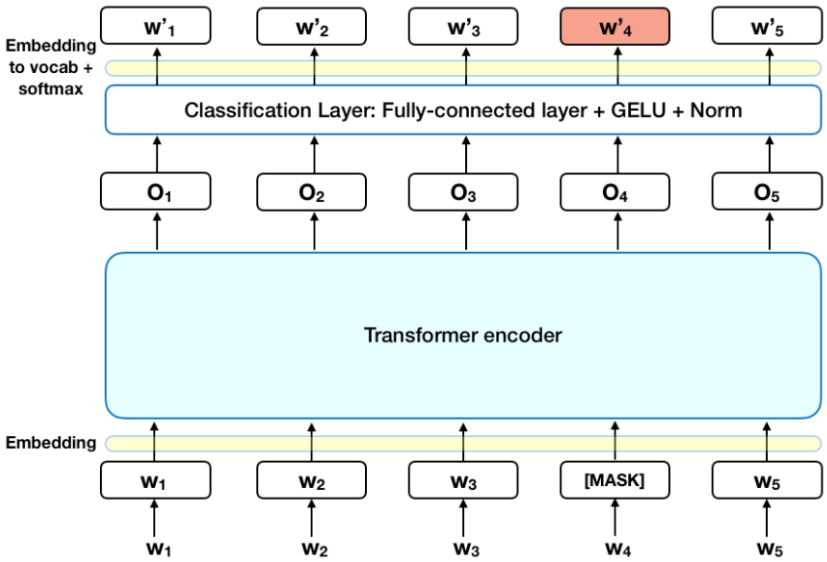


Figure 2.4: BERT Architecture Source: Taken from (BERT Explained — by Rani Horev| TDS Archive | Medium, n.d.)

2.3.1 Cross Modal attention

Effective multi modal representation learning in the case of Video QA allows a system to project video clips onto corresponding linguistic constituents by identifying what action is being asked about in a question and where it is located in the video. Convolutional Neural Network (CNN) or Vision Transformers (ViT) have ability to represent video frames through and text through BERT like models such fine grained multi modal reasoning is underpinned.

Cross modal attention architectures extend the principles of self attention to model interactions across modalities. Instead of attending within one modality, these architectures allow the model to use inputs from one modality (e.g., text) to query another (e.g., video). For example, ViLBERT (Lu et al. 2019) introduced a two stream architecture where vision and language are processed separately but interact through co-attention layers. This setup allows the model to learn fine grained associations between objects in frames and words in questions. Such Cross modal attention has become central to Video QA model VLAB (Xiao, Zhou, Yao, et al. 2023), which rely on transformer based modules to align and reason over video and text features.

One widely used design is the cross attention block, where the query originates from one modality like question tokens and the key and value are derived from another like video embeddings. This allows the model to generate modality specific attention maps that focus on corresponding regions or areas based on cross modal interactions. Cross modal attention unlike previous fusion methods it enables more flexible and richer reasoning

across modalities.

CMA – Cross Modal Attention Mechanism In this experiment we use a cross modal attention mechanism. Specifically, text representation as a query over the visual features is employed. Such attention allows text modality to adjust its representation conditioned on the visual feature and therefore encourages alignment of Amharic question with spatio temporal visual content.

BCMA – Bidirectional Cross Modal Attention Mechanism In this experiment we use a bidirectional attention mechanism where the textual and visual features pay attention to each other. The text representation is a query over visual features, and the visual features are a query over textual features. Through this cross modality attention both modalities can condition one's representation on the other and this will enhancing alignment between the Amharic question and spatio temporal visual content. This model is inspired by models like Modular Co-Attention Network (MCAN) which have achieved state of the art results on multi modal tasks.

2.3.2 Visual Question Answering

Visual Question Answering (VQA) is a core task in multi modal AI which is used to examine the degree to which vision language models can understand and reason across visual and text data. Existing VQA models are mainly trained on datasets cantered on English and some major languages and consisting of visual inputs frequently showing Western environments, so datasets often extend their linguistic range via translation or some other approaches.

Video Question Answering (ViQA) is a multi modal AI task that requires models to understand and analyze video content to provide accurate answers to natural language questions. Unlike traditional text based Q&A systems Videl QA integrates computer vision and natural language processing (NLP) to extract meaningful insights from videos. It involves processing spatial and temporal information by recognizing objects and actions then reasoning over sequential frames to generate context aware answers. Video Question Answering(Video QA) is a novel research activity for generating natural language responses to video content queries. The latest breakthrough in deep learning has enhanced natural language understanding and computer vision, pushing interdisciplinary challenges such as Video QA (Khurana and Deshpande 2021).

Video QA must handle more complicated visual content since videos consist of thousands of frames some of which may have background content that is irrelevant to the questions. Videos can have several actions but only a few are related to the questions. Video QA

frequently includes questions related to temporal cues for which one must consider the temporal position of objects and their interactions for reasoning. To be able to correctly answer questions, models need to understand their order in time. Unlike some image QA tasks, video QA requires an understanding of the interaction of objects. Location aware Graph Convolutional Networks (L-GCN) were Bidirectional Cross Modal Attention for encoding relations between detected objects. L-GCN helps to remove unnecessary background information and incorporate object location information into the graph for being location aware where a particular action is occurring (Huang et al. 2020).

The employment of transformers further solidified multi modal learning.(Lu et al. 2019) introduced ViLBERT, which generalized BERT to vision language tasks with the use of two stream transformers. (Su et al. 2019) designed VideoBERT for joint modeling of video and text for video language learning. Such architectures opened the way to recent models like CLIP (Radford et al. 2021), which aligns vision and language in a common space by using contrastive learning.

Some recent research has attempted to port neural network architectures designed for ImageQA to Video QA. For instance, MemexNet (**Jiang2017**) was built to answer questions regarding things that happen in images and videos, but it was limited by the scattered nature of information among video frames. Unlike images, where a solution can be inferred from one frame, Video QA must factor redundancy and temporal coherence to generate accurate answers. For example, a model that has been trained on Image QA may wind up counting the same person multiple times in different frames resulting in incorrect answers. Video QA thus requires models that can merge temporal knowledge and context based reasoning.

2.4 Related works

Video Question Answering (Video QA) has become increasingly popular in vision language navigation, multimedia recommendation, and communication systems. Video Question Answering (Video QA) is a challenging task that requires a model to analyze a video and reason about its visual content in relation to a given question to produce a meaningful answer. (S. Li et al. 2022) describes one framework for enabling zero shot multi modal task solution by combining the strength of pretrained models without additional training. PIC uses a generator model GPT or a diffusion model to produce candidate solutions and an ensemble of varied scorer models CLIP and classifiers to provide feedback, refining the output iteratively until agreement is achieved. Quantitative performance on two and three object relation tasks showed that including scorers from multiple camera views greatly improved the success rate. Video question

answering results on ActivityNet-QA data got success rate for 2 relations was improved from 35.0% for one view to 67.5% for five views.

(Bertasius et al. 2021) examines the use of space time attention networks specifically the TimeSformer in video understanding. The basic hypothesis is that an all transformer architecture can be well adapted to video without resorting to 3D convolutional networks (CNNs), which have been the conventional solution. 80.7 video level accuracy on Kinetics400.

(Guda et al. 2024) addresses limitations in single frame and end to end full video methods by proposing novel methods to improve causal, temporal, and descriptive Video QA on the NExT-QA benchmark dataset. It describes Pairwise Cross Modal Aggregation for efficient frame aggregation, Multi modal Action Grounding for grounding video content in actions and descriptions with CLIP for feature extraction, and Multi modal Robust Intervener for better causal reasoning using scene perturbations. These methods improve multi modal alignmen, and model robustness, achieving state of the art performance. The work also explores Video QA as a reinforcement learning problem to adaptively select frames. The Bidirectional Cross Modal Attention PCMA model achieves an overall accuracy of 46.27%.

(Tang et al. 2024) introduces a Spatio Temporal Graph Convolution (STGC) mechanism that constructs dynamic graphs, where nodes represent visual objects and edges model both spatial and temporal relationships. To further enhance reasoning capabilities, the authors integrate a Dynamic Graph Transformer which applies attention over these graphs to understand complex object dynamics. A key innovation of the model is its Cross modal Dynamic Graph Attention which aligns visual information with the given question by adaptively focusing on the most relevant objects.

(Zhu et al. 2023) introduces a novel CLIP guided Visual Text Fusion Transformer for video Pedestrian Attribute Recognition. Vision frames are encoded as video tokens with the help of a pretrained CLIP model, whereas attribute sets are converted into text descriptions and projected into CLIP's text encoding embedding spaces. These obtained video and text tokens are concatenated together and passed as inputs into the fusion Transformer with interaction between the multi modal information and then afterwards output into classification head to possess attribute recognition.

(L. Gao et al. 2024) module designed to enable image level Large Language Models (LLMs) to understand videos and output is a weighted average of input frame level visual tokens, ensuring the preservation of knowledge from the image-LLM. 67.7 accuracy was

archived on Next-QA dataset. (D. Gao et al. 2022) introduces s MIST (multi modal Iterative Spatial Temporal Transformer), a Video QA model that addresses challenges like multi event reasoning and multi grained visual understanding. MIST initially applies a frozen multi modal transformer like CLIP to obtain features, followed by iterative spatial temporal attention (ISTA) to sample question-associated video clips and areas through differentiable top-k and top-j choice and applying self attention between sampled features and the query. Predictions are finally produced through comparison of pooled representations to candidate answers. MIST – CLIP achive QA accuracies of 57.18% on NExT-QA val set.

(Tan and Sun 2025) is designed for Video Question Answering by modeling relationships between objects in videos. It builds two key graphs: a stereoscopic spatio-temporal graph (Gst) to capture spatial and temporal object relations using a Spatio-temporal GCN (ST-GCN), and an appearance graph to model object appearance relations with an Appearance GCN (A-GCN). Questions are processed using a BiLSTM, and a Bilinear Attention Network (BAN) fuses question features with visual features. The model predicts answers by reasoning over these fused representations. GBRR is evaluated on MSVD-QA and MSRVTT-QA, showing strong relational reasoning ability.

(Peng et al. 2023) proposes a novel framework designed to tackle the complex interactions between visual and textual modalities inherent in Video QA tasks. The model leverages a hierarchical synergy enhanced multi modal relational network that captures fine grained relationships and temporal dynamics within videos allowing for effective alignment and reasoning over both visual content and natural language questions. By incorporating hierarchical structures and relational reasoning, the approach models dependencies between objects, actions, and textual cues, while also addressing temporal alignment of events across video frames. Evaluated on benchmark datasets including MSVD QA, MSRVTT QA, and ActivityNet-QA, the framework achieves state of the art performance, demonstrating its efficacy in addressing the challenges of multi modal fusion and temporal reasoning in video question answering. This work significantly advances the capability of Video QA systems to understand and interpret complex video content in natural language contexts. (Yu et al. 2024) addresses the challenges of Long term Video Question Answering (Video QA) which involves understanding untrimmed videos and answering diverse free form questions. Traditional methods often rely on pre-extracted features, leading to representations that may not capture the intricate relationships between modalities. To overcome these limitations the authors propose an end to end framework named Multi granularity Contrastive Cross modal collaborative Generation (MCG).

(Cheng et al. 2023) model uses Cross modal interactions to perform Video Question Answering by integrating visual and textual information. It uses keyword aware question features to guide video graph construction, focusing on relevant objects and filtering out noise. Visual features undergo spatio-temporal reasoning and are then fused with question features using bilinear attention. This fusion enables accurate answer prediction by aligning and combining both modalities. The model is explicitly designed to predict answers based on both video and question inputs, following a Cross modal learning approach.

(Yin et al. 2023) is a Cross modal Video QA model that integrates video content, dense video captions, and question words. It constructs modality-aware graphs (video, caption, and question), performs intra-modal reasoning via GCNs, and applies a Cross modal Attention Mechanism (CAM) for inter-modal fusion. A question-guided multi modal fusion module refines these features for answer prediction. Evaluated on TGIF-QA and MSVD-QA, EC-GNN achieves state of the art or comparable performance, with ablation studies confirming the value of using dense captions and Cross modal reasoning. (Huang et al. 2020) model tackles Video QA by constructing a graph of detected objects enriched with spatial and temporal location features. Using GCNs models object interactions and temporal order helping to filter irrelevant background content. The model includes a Bi-LSTM question encoder, a location aware video encoder, and an attention based visual question interaction module for answer prediction. Evaluated on TGIF-QA, Youtube2Text QA, and MSVD QA, L-GCN reportedly achieves state of the art results, though exact metrics are not provided.

Table 2.1: Summary for Literature Reviews

Authors	**Datasets**	**Methods**	**Question Answering Type**	**Accuracy**	**Sampling**	**Gap and Limitation**
Guda et al. 2024	NExT-QA	Pairwise Cross Modal attention mechanism (PCMA)	Accuracy on causal, temporal, descriptive questions	46.27%	Random sampling	- Operate primarily on aggregated frame level features. - Does explicitly incorporate object level semantic information.
Tang et al. 2024	MSVD-QA, MSRVTT-QA	Cross modal Dynamic graph attention module (CMDA)	Multiple-choice and open-ended Video QA	MSVD-QA:39.1%, MSRVTT-QA: 43.3%	Random sampling	- Graph structure can become large and inefficient as number of objects - Performance can vary depending on the randomly sampled frames.
Yu et al. 2024	MSVD-QA, MSRVTT-QA	multi granularity Contrastive Cross modal collaborative Generation (MCG)	A accuracy (exact match)	MSVD-QA:48.2%, MSRVTT-QA: 44.0 %	Sparsely sampling	- Difficulty in object level Reasoning

Yin et al. 2023	MSVD-QA, TGIF-QA	Event Correlated Graph Neural Network (EC-GNN)	Open-ended words	MSVD-QA: 34.8%, TGIF-QA :81.2%	Not mentioned	- Struggled to capture fine grained relation information between objects - Neglected the interaction of information between objects in the temporal dimension
Cheng et al. 2023	MSVD-QA, MSRVTT-QA	Graph based relational reasoning network (GBRR)	Answer selection from a pre-defined set	MSVD-QA:41.5%, MSRVTT-QA:37.4%	Uniformly sampled	- Lacking complete frame level interactions - Graph structure can become large and inefficient as number of objects.

Most of the above Video QA models with Cross modal attention have achieved significant progress by aligning the video and language features. Nevertheless, there exist intrinsic limitations for these models. Firstly, these models are struggling to capture fine grained relational information between objects. Without explicit modeling of an object's interaction or transformation through frames, the system cannot reason about temporal dynamics or cause-effect relationships between entities. The inability to capture object dependencies sequentially leads to inferior object level understanding, weaken the response accuracy of the model on compositional questions requiring scene evolution tracking

Second, despite the use of attention mechanisms, most models lack a principled frame selection mechanism; random or uniform sampling strategies are employed, which increases the likelihood of missing frames with essential contextual or semantic information. To construct comprehensive and robust AI, future systems must engrain multilingualism at the level of representation, employ smart frame selection strategies, and address linguistic structure variability during training and testing protocols In addition, spatial relationships among objects across frames are typically disregarded, weakening the model's ability for event boundary comprehension or multi object reasoning.

Another major shortcoming is the absence of multilingual support in existing Video QA models. The dominant architectures are mostly trained on English MSVD QAs and overlook morphologically rich, low resource languages such as Amharic. This prevents effective deployment in multilingual settings from utilizing or benefiting from such intelligent systems.

This paper identifies absence of multilingual support and unguided frame selection as major shortcomings in existing Video QA models. To resolve the shortcomings, this paper presents a solution in the form of multilingual aware encoders and a smart question semantic guided frame selection scheme. These enhancements aim to preserve context support object reasoning across time and open up access to the previously underrepresented language AmhariC.

CHAPTER THREE

RESEARCH METHODOLOGY

This sub section describes the methods adopted to develop an Amharic based Video Question Answering (Video QA) system through deep learning,translation, and multi modal comprehension techniques. Research methodology included data collection, preprocessing, analysis, and tools used to train and test the model.
Let see the general block diagram of the Bidirectional Cross Modal Attention model

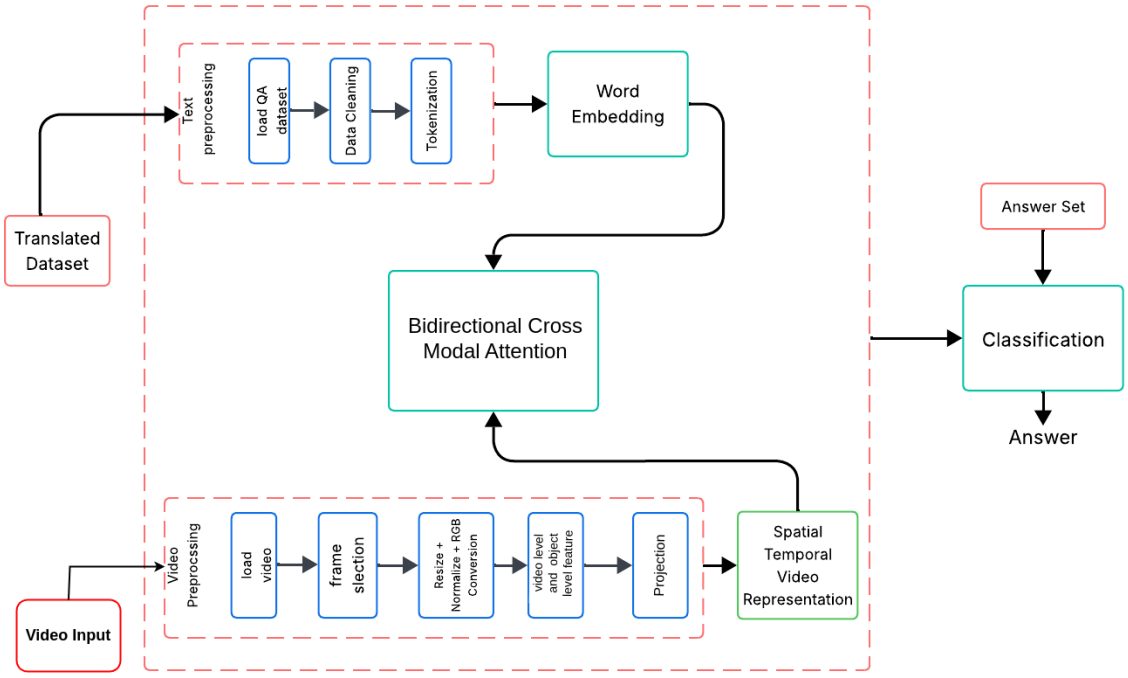


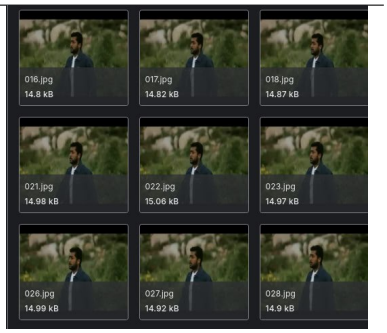
Figure 3.1: Block diagram of the Bidirectional Cross Modal Attention model

3.1 Benchmark Dataset

The benchmark datasets used in this work are the Microsoft Research Video Description Question Answering set and the Microsoft Research Video to Text Question Answering set. They were extracted from two popular video Question Answering datasets named Microsoft Research Video Description and Microsoft Research Video to Text (D. Patel et al. 2021). These datasets are publicly available and have been extensively used as benchmarks for testing the English language video question answering system's understanding and reasoning ability (Yin et al. 2023).
MSVD-QA dataset: We utilized the Microsoft Research Video Description (MSVD) dataset, which is widely used in training and evaluating video captioning and video

question answering (Video QA) models. The dataset consists of approximately 1,970 short video clips collected from YouTube, each typically lasting between 10 to 25 seconds. These videos depict a wide range of everyday human activities and interactions,

Table 3.1: MSVD datasets sample question and answer with a corresponding video frames samples.

Question	**Answer**	**Video Frames**
What is chewing on nut?	animal	
Who is walking down a pathway lined with greenery?	man	
Who is pouring marinade man from a bowl into a bag?	man	

Each video is annotated with multiple human written captions in English, offering varying sentence structures and vocabulary. This diversity makes the dataset valuable for training robust sequence and language models. Building on the caption annotations, the dataset is further extended into MSVD QA by automatically generating around 50,505 question answer pairs, enabling direct evaluation of Video QA systems. The questions are generally open ended and require semantic understanding of the video content. The answers are single words or short phrases, typically factual in nature like objects, actions, colors, or people. This answer format allows for straightforward classification and retrieval based QA evaluation. In Figure 3.2, we present examples of raw data from the dataset, including

sample video frames and corresponding questions and answers.

3.1.1 Data Preprocessing Techniques

Data preprocessing is a critical step in constructing deep learning models, particularly for complex multi modal tasks such as video question answering. In our study, visual and text data were preprocessed to ensure quality, consistency, and compatibility for training the model. Our primary source of data was the MSVD dataset that provides short video clips with multiple natural language captions and questions.

A. Visual Data Preprocessing:
Preprocessing visual data is a vital step in developing a robust video question answering (Video QA) system. Since videos typically contain hundreds of frames many of which are redundant or uninformative we adopted an efficient frame selection strategy to retain only the most meaningful visual content.

- **Frame Sampling:** For each video in the MSVD dataset, the sampling was already completed, and each frame was provided as a part of the dataset. But such extensive frame extraction resulted in many redundant or visually similar frames that did not provide any additional semantic information. Therefore, further processing was required to remove redundancy and retain only the most informative visual information.

$$F_i = \{f_{i1}, f_{i2}, \ldots, f_{in}\} \quad \text{where } F_i \text{ contains all frames extracted from video } V_i$$

- **Create 16 cluster:** To reduce redundancy from the large number of frames extracted, we used K-means clustering to group frames clip embedding according to visual similarity. We used 16 clusters per video, which we hoped would capture diverse visual contexts along the video time axis. In this manner, we are assured of coverage across scenes, objects, and actions without needing to process all the frames.

$$C_i = \{c_{i1}, c_{i2}, \ldots, c_{i16}\} \quad \text{where } c_{ij} \subset F_i, \text{ and } C_i \text{ is obtained by applying K-means on } F_i$$

- **CLIP based best frame selection:** We took the most representative frame from each of the 16 clusters. To do so, we used the CLIP model, which is capable of parsing images and text. We video's caption (brief description) for each frame within a cluster and used CLIP to measure how well the image and its caption matched semantically. The most representative frame in each cluster was selected as the representative frame. By doing this, we made sure that we were selecting the most important and diverse moments from each video without repeating

anything unnecessarily.

$$f_{ij}^{*} = \arg\max_{f \in c_{ij}} \text{CLIPSim}(f, \text{caption}(f)) \quad \text{for each cluster } c_{ij} \in C_i$$

- **Frame Preprocessing:** the 16 selected frames per video were then preprocessed to be model compatible with the models used in the system. The frames were resized to a fixed resolution 224×224 pixels so that they would be model compatible with models such as Vision Transformer (ViT), FastRCNN, and CLIP. The frames were converted to RGB format to have uniform color for all models and uniform color channels to process. Preprocessing thereby ensured that the data was in the correct form to train and make inferences with the models.

$$f_{ij}^{\text{pre}} = \text{RGB}\big((\text{Resize}(f_{ij}^{*},\ 224 \times 224))\big) \quad \text{for each } f_{ij}^{*} \in F_i^{*}$$

- **Object Cropping:** In addition to enriching the visual characteristics, we conducted object detection on every selected frame using FastRCNN. This was to identify key objects in the video that could be important to identify answers to questions regarding the video content. Using FastRCNN, we detected object bounding boxes within each frame and we cropped the object image only for the next step.

$B_{ij} = \text{FastRCNN}(f_{ij}^{\text{pre}})$ where B_{ij} is the set of detected bounding boxes in frame f_{ij}^{pre}

B. Textual Data Preprocessing:

- **Translation (English to Amharic):** generate a Amharic dataset by translating every question and answer per video. Our paper uses Google's translation quality and that it's sufficient to create a usable Amharic dataset, especially for research in low resource NLP even with some inconsistency.

- **Data cleaning and preparation:** this step remove punctuation, non Amharic data and symbols, keeping only letters and digits. This prevents malformed characters from interrupting downstream processing. To do this we use a regular expression to remove any characters that are not part of the Amharic unicode block, standard digits or spaces. This is important for text normalization more importantly when working with machine learning pipelines, as it avoids unexpected tokens, broken words, or invalid characters that could confuse the tokenizer or degrade model performance.

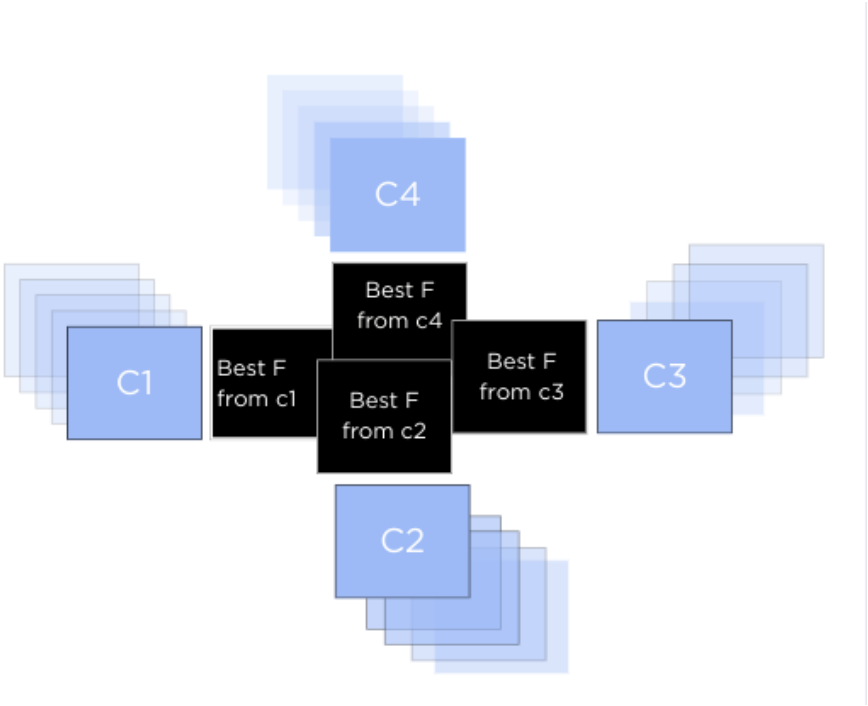
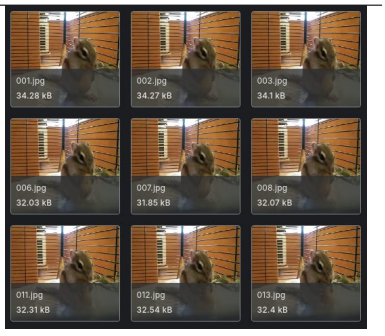


Figure 3.2: General workflow diagram of proposed frame selection.

- **Tokenization:** split each sentence into word level tokens with a regex tuned for mixed Amharic letters and numbers. These rules preserve meaningful morphemes while discarding extraneous whitespace.
- **Word embedding:** map every token to a dense vector with a multilingual encoder, contains the contextualized token embeddings for each word in the input, ready for attention and fusion with vision features.

Table 3.2: Sample translated MSVD datasets the question and answer with a corresponding video frames after translation.

Question	Answer	Video Frames
ለውዝ ሚያኝክ ምንድነው?	አንበሳ	

3.2 Development Tools

To design and implement the system Bidirectional Cross Modal Attention, a few tools and technologies were employed. These include UML modeling tools for system design,

and an assortment of support tools essential to research. The following sections briefly introduce each of the tools and its role in the development process.

3.2.1 Design Tools

Design tools are required to develop, visualize, and comprehend system concepts. During this project, Lucidchart was used as the primary tool for developing flowcharts and architectural diagrams. Lucidchart is a user friendly and collaborative tool for generating professional visualizations, including system workflows, network architectures, and data flow diagrams. Its simplicity and flexibility made it a handy tool for illustrating the overall design of the Bidirectional Cross Modal Attention system.

3.2.2 Hardware Development tools

To implement this research, the following hardware tools is used.

Table 3.3: Hardware tool

No.	Tools	Used for
1	GPU	To increase the computation and to fasten the training
2	Hard Disk	Used as storage for large datasets.
3	RAM	To work together with the GPU to speed up the training process.

3.2.3 Software Development Framework tools

To implement the Bidirectional Cross Modal Attention research, a variety of programming languages, frameworks, and development environments were utilized. Below is a brief description of the key tools used:

- **PyTorch:** We use popular open source machine learning library based on the Torch framework for this thesis. PyTorch is especially favored in academic and research settings for its flexibility, dynamic computation graph, and strong support for deep learning applications. We use 2.0.1+rocm5.4.2.
- **Jupyter Notebook:** We also use interactive development environment used extensively in data science and AI research. It allows researchers to write, execute, and visualize code in a step by step manner, making it ideal for experimentation and documentation. We use jupyter client 7.3.4.
- **Python:** The primary programming language used throughout the project. Python's versatility and rich ecosystem of libraries made it well suited for implementing machine learning models, handling data preprocessing, and managing system integration tasks. We use CPython 3.11.11.

3.3 Baseline Works

The performance of the Bidirectional Cross Modal Attention was experimented on various benchmark models. The Bidirectional Cross Modal Attention model was compared with video question answering models developed based on Cross Modal Attention, namely CCVQA (S. Ye et al. 2023), and EC-GNNs (Yin et al. 2023) , which were utilized as baseline models to conduct experiments.

- CCVQA used method leverages the Contrastive Language Image Pre training (CLIP) model to guide Cross modal learning for Video QA. It extracts video features using TimeSformer and text features using BERT, then utilizes CLIP to obtain visual text features from a general knowledge domain. A cross domain learning strategy is Bidirectional Cross Modal Attention to extract attention information between visual and linguistic features across the target and general domains, integrating these features for answer prediction.
- EC-GNNs introduces Event Correlated Graph Neural Networks (EC-GNNs) to perform Cross modal reasoning over three modalities: video, question, and dense captions. By incorporating dense video captions as an auxiliary modality, the model captures event correlations to enhance reasoning capabilities. It employs Cross modal attention mechanisms to model inter modal relationships and a question guided self adaptive multi modal fusion module to aggregate relevant information for answer prediction.

Why CCVQA and EC-GNNs as baseline models? Because these models represent two distinct yet relevant directions in recent Video QA research: the former employs CLIP based visual text alignment guided by pretrained general domain knowledge, while the latter emphasizes event-level reasoning through dense captions and graph attention. The purpose of selecting these models is to demonstrate how the Bidirectional Cross Modal Attention compares to both approaches that utilize semantic rich pretrained embeddings and those that model temporal and event correlations explicitly. Both models were evaluated on the benchmark dataset MSVD QA , making them strong and comparable baselines for assessing improvements in Cross modal understanding and answer accuracy.

3.4 Feature Extraction Network

This section explains how we converted both language and video inputs into structured formats suitable for learning. While Chapter 2 presented a range of encoding techniques, we here describe the real models implemented in our system.

For question encoding, we utilize BERT (Devlin et al. 2019), a transformer based language model that captures context from both directions using self attention. Unlike sequential models enabling deeper understanding of semantic relationships. In our setup, BERT generates contextual embeddings for each token that aligns effectively with visual information. It also supports multiple language.

To extract temporal and spatial information from videos, we employ TimeSformer (Bertasius et al. 2021), which applies self attention across both space and time. This model treats a video as a sequence of image patches, learning patterns over frames without using convolution. It is particularly effective in capturing the temporal dynamics of actions and interactions.

For object level feature extraction, we integrate Faster RCNN (Ren et al., 2015), a two stage object detector and MCLIP that proposes regions and classifies objects within them. It helps identify key entities in video frames, such as people, objects, or animals, which are often directly referenced in questions.

Why Faster RCNN? Because it remains a strong baseline for object detection, offering accurate localization and classification, especially important for tasks requiring fine grained reasoning (Ren et al. 2015). Faster RCNN because it remains a strong baseline for object detection, offering accurate localization and classification, especially important for tasks requiring fine grained reasoning.

To align text and vision semantically, we adopt CLIP (Radford et al. 2021), which jointly learns representations for images and text using contrastive learning. CLIP is trained on large scale image text pairs and embeds both modalities into a shared feature space. Why CLIP? Because it enables general purpose vision language alignment, providing robust similarity scores between text and images without task specific tuning. Together, BERT, TimeSformer, Faster RCNN, and CLIP provide a multi level encoding pipeline that captures semantic, spatial, temporal, and object level information, allowing for comprehensive reasoning in video question answering.

3.5 Evaluation Metrics

There are different evaluation approaches to measure the performance of a video question answering model on test data. These approaches include both qualitative observations and quantitative metrics. Among the widely used evaluation criteria in classification based tasks is **accuracy**, **precision**, **recall**, and **F1-score**. From these, **accuracy** was chosen to evaluate the performance of this system.

3.5.1 Accuracy

Accuracy is one of the most common metrics for evaluating classification tasks. It calculates the ratio between the number of correct predictions and the total number of predictions made by the model. In video question answering, accuracy reflects how often the predicted answer matches the correct answer from the ground truth.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \times 100\%$$

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.2}$$

Where TP and TN are true positives and true negatives, and FP and FN are false positives and false negatives, respectively. A higher accuracy indicates better model performance on the question answering task.

3.5.2 Precision

We also use precision as an evaluation metric that measures the proportion of correctly predicted positive samples out of all samples predicted as positive. In the context of Video Question Answering (Video QA), precision indicates how accurate the model is when it predicts a specific answer class. High precision means that when the model provides a certain answer, it is often correct.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3.3}$$

Where:

- TP (True Positives): The number of correct predictions where the model correctly identifies the correct answer.
- FP (False Positives): The number of incorrect predictions where the model incorrectly identifies an answer that is actually wrong.

3.5.3 Recall

Recall, also known as sensitivity or true positive rate, measures the proportion of actual positive samples that were correctly identified by the model. In Video QA, recall tells us how well the model can find all the correct answers from the ground truth.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{3.4}$$

Where:

- TP (True Positives): The number of correctly predicted correct answers.

- FN (False Negatives): The number of actual correct answers that the model failed to predict.

3.5.4 F1-Score

The F1-score is the harmonic mean of precision and recall. It provides a balanced evaluation metric when both precision and recall are important. It is particularly useful in Video QA tasks where class imbalance is a concern or where both false positives and false negatives carry significant impact.

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.5}$$

Where:

- Precision: As defined in Equation (3.3), it is the ratio of true positives to all predicted positives.
- Recall: As defined in Equation (3.4), it is the ratio of true positives to all actual positives.

CHAPTER FOUR

PROPOSED BIDIRECTIONAL CROSS MODAL ATTENTION MODEL

This chapter outlines the architecture of the Bidirectional Cross Modal Attention solution, including mathematical representations and graphical figures intended to serve the task of text to image generation. The chapter consists of eight. overall major sections. The first presents the general model structure that maps input text onto corresponding visual output. The second section addresses the text encoding mechanism. The third section addresses the structure of the bidirectional cross modal attention. The fourth section addresses the timesformer. The fifth section addresses clustering. The sixth section addresses CLIP. The seventh will address Fast RCNN. Finally, the eighth section provides a high-level the model steps.

4.1 Bidirectional Cross Modal Attention Model Architecture

multi modal Understanding for Amharic Video Question Answering using Cross modal Learning is the Bidirectional Cross Modal Attention model designed to address accurate answer classification and bridge the semantic gap between visual content and the given question. The model incorporates novel mechanisms such as temporal attention, object-guided alignment, and CLIP based similarity enhancement by extending standard TimeSformer based architectures.

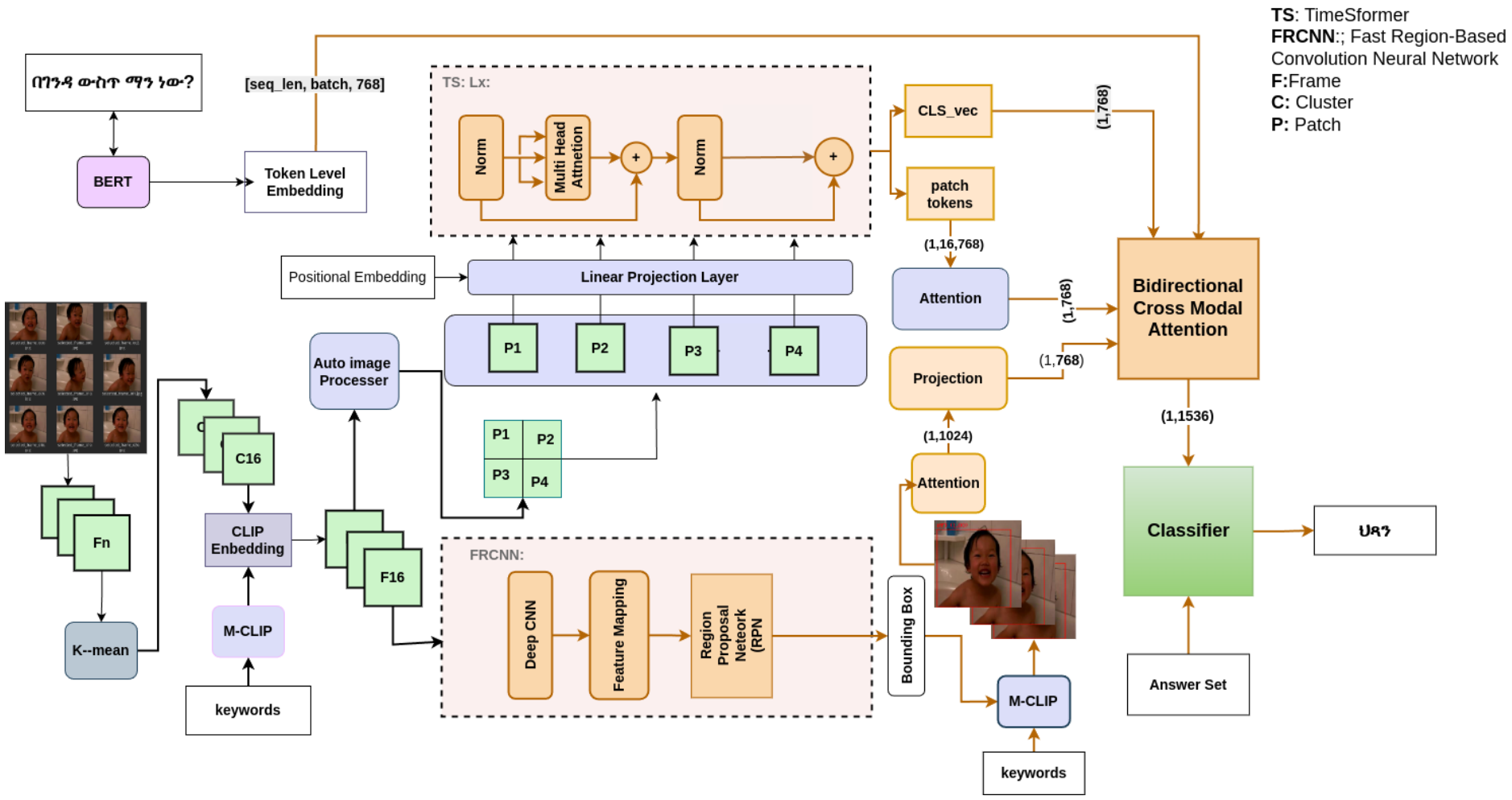


Figure 4.1: Bidirectional Cross Modal Attention Model architecture

4.2 Text Encoder

The text encoder is the initial component of this approach, responsible for converting the input question written in Amharic into a fixed length semantic representation. First, the input question is processed using a pretrained BERT tokenizer, which automatically handles punctuation, casing, and subword segmentation. The sentence is then transformed into a sequence of token indices and corresponding attention masks. These inputs are passed into a pretrained BERT model, which generates contextualized embeddings for each token in the sentence. Instead of using individual word vectors, the model extracts the pooled output corresponding to the `[CLS]` token. This vector have the overall meaning of the sentence and serves as the token level representation $s \in \mathbb{R}^{768}$, where 768 is the embedding dimension. This BERT based embedding captures both the syntactic and semantic structure of the question and is utilized in the subsequent stages of the model.

4.3 TimeSformer

To obtain useful spatiotemporal representations of the input video, we employs TimeSformer, which is a pure transformer architecture made specifically for video understanding tasks. In contrast to the conventional 3D convolutional networks, TimeSformer uses divided space-time attention to model spatial and temporal dependencies separately in a more efficient computation manner.

During preprocessing, To reduce visual redundancy and preserve different temporal information across video frames, a clustering based selection process was employed. Each of the videos was uniformly sampled into a sequence of frames, which were then divided into 16 clusters using the K-means clustering algorithm. Clustering was performed on visual similarity to ensure that each cluster contained distinct visual content of the video.

To identify the most indicative frame from every cluster, a CLIP based semantic matching approach was utilized. For every frame in a cluster and its corresponding video-level caption, the similarity score was computed based on the CLIP model, which embeds both text and image to a shared embedding space. The frame with the highest similarity score to the caption of every cluster was selected as the representative frame. This ensures that the top 16 frame set is not only visually rich but is also highly semantically connected to the overall description of the video, which makes it improve the quality of the temporal and contextual understanding in downstream tasks. The sampled video frames are processed in small chunks (4 frames at a time), with each chunk passed through a pretrained TimeSformer model. Each frame is divided into non-overlapping patches of

size 16×16, resulting in 14×14=196 patches per frame for standard 224×224 resolution inputs. These patches are linearly projected and combined with spatial and temporal positional encodings to form the input token sequence.

TimeSformer uses a divided space-time attention mechanism to process video data. In each transformer block, it separately applies temporal attention across frames (at the same patch location) and spatial attention within each frame (across different patches). This allows the model to learn how visual content changes over time while preserving spatial structure. When a chunk of frames is passed through the model, it produces two types of outputs:

- A CLS token, which summarizes the entire chunk.
- A set of patch token embeddings for each frame.

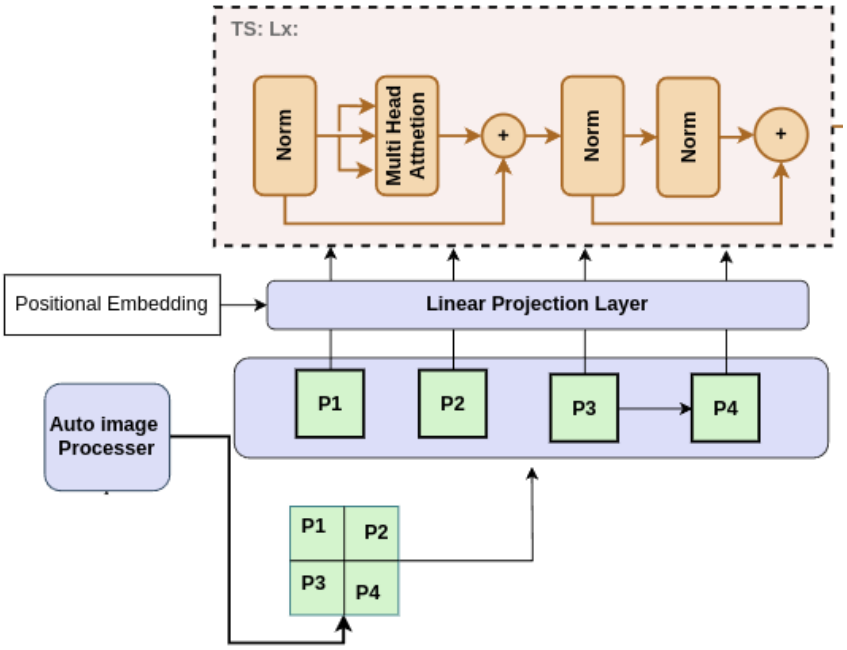


Figure 4.2: TimeSformer

In the paper, To extract the most informative moments, these concatenated frame embeddings are passed through a custom SimpleAttention module, which computes attention weights and outputs a weighted sum across frames. This yields a temporal summary vector that emphasizes the most relevant frames. In the end, each video is represented by two key features:

- A CLS based global feature, summarizing the overall video content with shape of (1,768).

- An attention based temporal feature, highlighting important frames across time with shape of (1,768).

4.4 Clustering

We used clustering on video frames which is a strategic approach to enhance diversity and mitigate redundancy in video based tasks like question answering. By grouping frames based on semantic or visual similarity using CLIP embeddings to capture high-level features, clustering ensures that selected frames represent distinct scenes, actions, or perspectives within the video. This method avoids overloading the model with repetitive or near-identical frames static backgrounds or prolonged shots, which can obscure critical information and waste computational resources. Our choice of 16 clusters strikes a balance it provides sufficient coverage to capture key temporal and spatial variations. Clustering help our model focuses on the most informative visual cues. For our Video QA task, where answers often depend on specific moments, this approach improves robustness by enforcing the model to analyze diverse visual contexts, reducing bias toward dominant or repeated patterns.

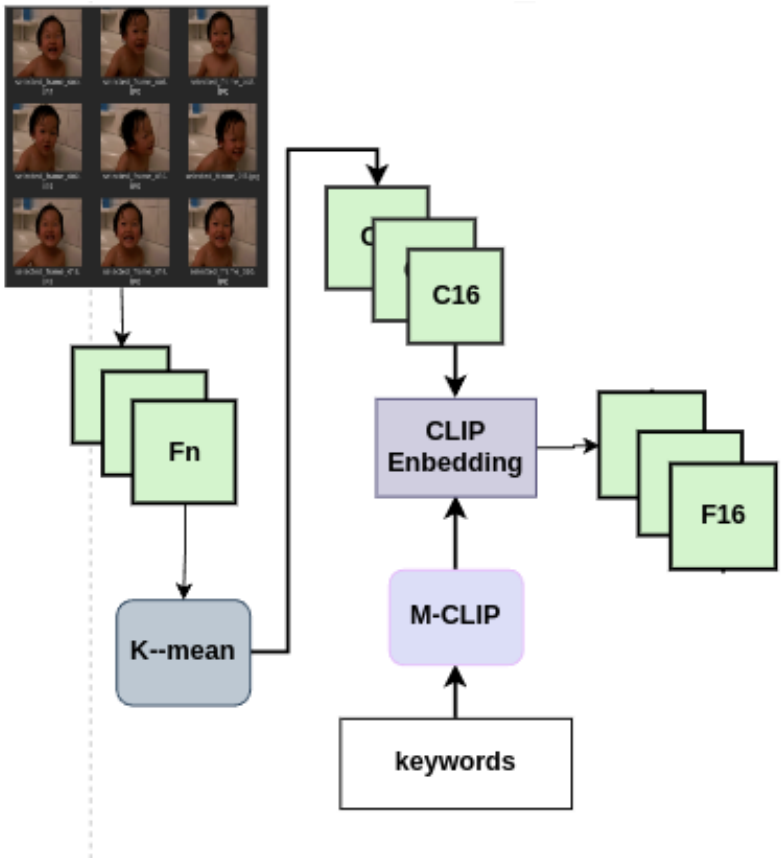


Figure 4.3: Frame selection part of the architecture

4.5 Contrastive Language-Image Pre-Training (CLIP)

Contrastive Language Image Pre Training's zero-shot visual text alignment capability makes it uniquely suited for object labeling in Video QA tasks, particularly when paired with question-answer (QA) keywords. By encoding both visual frames and candidate object labels (derived from QA terms) into a shared embedding space, CLIP computes similarity scores that identify the most semantically relevant objects for a given question (e.g., linking **"ቁጥር"** [number] in a question to numerals in a video). This approach bypasses fixed label sets, adapting dynamically to diverse Amharic vocabularies. Selecting the top 16 frames preserves temporal variety, capturing objects across different contexts (e.g., a **"መኪና"** [car] from multiple angles or lighting conditions), while the single best frame provides a distilled, high confidence visual anchor. This dual strategy balances comprehensive scene understanding (via 16-frame diversity) with noise reduction (via 1-frame precision), mimicking human attention patterns that alternate between broad observation and focused analysis. For Amharic video datasets, where object labels may lack standardized annotations, CLIP's language aware labeling combined with strategic frame sampling ensures robust multi modal grounding without overfitting to sparse training data.

4.6 Fast Region based Convolutional Network method (FAST RCNN

To extract object level semantic features from the frames of a video, We Fast RCNN as a typical two stage object detection network. For each frame selected in a video, Fast RCNN identifies several regions that may contain objects and returns precise bounding box locations as well as confidence values.

The detection pipeline begins by transforming the input image into a tensor and passing it through a backbone convolutional neural network ResNet-50 with Feature Pyramid Network (FPN) in this case. The backbone generates a dense spatial feature map. A Region Proposal Network (RPN) is then passed over the feature map to generate roughly 1,000 candidate regions for each image. These are then passed through objectness scores and non-maximum suppression to suppress redundancy and then only the top-ranked regions with a confidence score of 0.5 and above are retained—typically 5 to 20 high-confidence object regions per frame.

Each of frame's object is aligned using RoI Align and passed through a detection head that outputs a class prediction and refined bounding box coordinates. In this setup, only the bounding box coordinates are used. The detected regions are then cropped directly

from the original RGB frame.

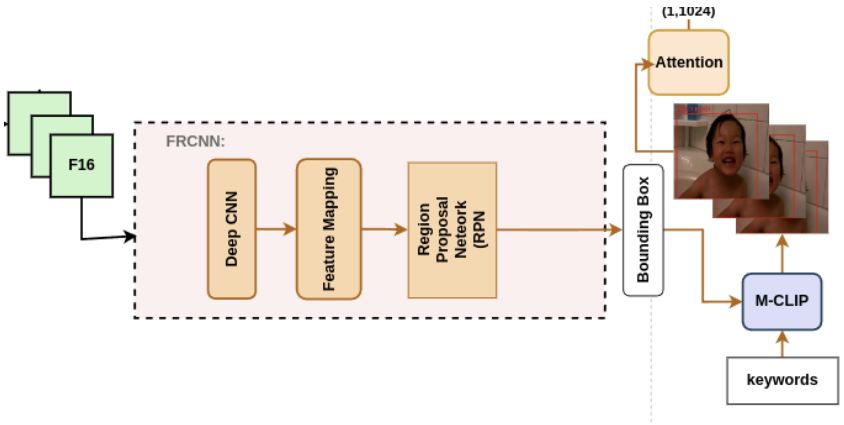


Figure 4.4: Frame selection part of the architecture

Each cropped object image is fed into a CLIP model, which encodes the visual region into a 512-dimensional feature vector. For this purpose, each Amharic video caption is tokenized into individual words, which are then encoded using a multilingual CLIP text encoder into a shared embedding space. By computing the cosine similarity between each object's visual embedding and the embeddings of all caption words, we identify the most semantically relevant word for each object. This allows the system to connect visual objects in the frame to the most pertinent keywords derived from the natural language description. In effect, the object features are now semantically grounded they are no longer simply spatial or appearance based , but are endowed with contextual meaning directly applicable to the question–answer space. This is especially important for Video QA tasks where reasoning is based not only on the detection of objects, but on the understanding of their salience to the linguistic query.

After extracting object features and aligning them with the most semantically relevant words from the video's caption, a temporal aggregation step is applied to enrich these representations with sequence-level context. This is necessary because the same keyword might appear across multiple frames, and treating each occurrence independently would lose important temporal continuity.

The TemporalAttention class is composed of three fully connected linear transforms q, k, and v that respectively generate the query, key, and value matrices from the sequence of frame wise CLIP features. Each of these transforms maps every feature vector to a shared hidden space, and attention weights are computed through scaled dot product similarity of query and key vectors. The output is a set of learned weights highlighting the most contextually relevant frames of a certain keyword across the entire video.

Figure 4.5: example frame with sematic information

In practice, for each keyword w associated with a object, used to track object across the frame. If a given frame does not contain an object aligned with w, a zero vector padding is inserted. This ensures that each keyword specific sequence maintains temporal structure while supporting variable object presence. The temporal attention module is then applied to this sequence, and its output a single vectoris computed as the mean of the attended value vectors. This final embedding represents a temporally aware, keyword specific object representation.

This custom layer is intentionally lightweight, with no multi head decomposition or residual connections, in order to reduce computational overhead while still benefiting from the inductive bias of attention. Its design all ows the model to focus on semantically important object appearances across time without relying on recurrent mechanisms, making it ideal for short video sequences where sparsity and variability in object presence are common. After processing, the feature vectors of all detected objects across the video are concatenated to form a comprehensive object level temporal representation.

4.7 Cross Modal Attention

Cross modal attention mechanism is a main component in process, particularly for video question answering tasks requiring understanding and integration of information from video, or question, these mechanisms allow models to dynamically locate and focus on the most noticeable segments of a video and the most useful words of a question to output an accurate answer. It facilitates more profound synergistic understanding of visual and

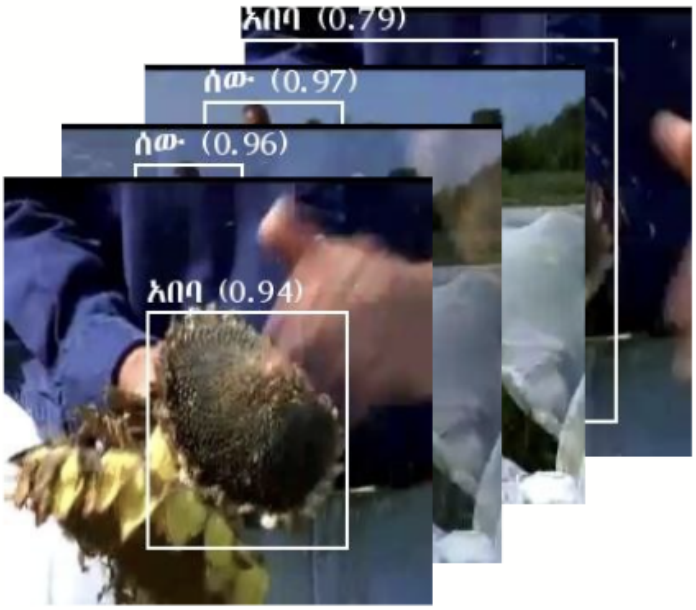


Figure 4.6: Objects with in a multiple frame

textual information, over and above simple concatenation or independent processing of modalities. Effectively, Cross modal attention permits one modality to ”query” another, selectively gating the importance of different items within the second modality in light of the information in the first. This interaction is necessary for Video QA, where the model needs to incorporate the semantic meaning of a question text into the spatio-temporal context of the video.

4.7.1 Query Projection

In Cross modal attention, the query Q is typically derived from the question representation. In our setup, we used a pretrained BERT model to encode the question, and then take the mean of the last hidden states across all tokens to obtain a token level embedding. This embedding captures the semantic intent of the question. To align it with the visual features, the vector is projected into the attention space using a learnable weight matrix:

$$Q = W_q \cdot \text{BERT}(x)$$

where $W_q \in \mathbb{R}^{d_q \times d}$ and d is the dimensionality of the BERT output.

4.7.2 Key and Value Projection

The keys K and values $V$are generated from the visual features. In a multi modal setup like Video QA, the visual modality can include several types of embeddings, such as frame level features, object level features, and temporal representations. These features are concatenated or stacked and then projected into the same space as the query using

separate linear layers:

$$K = W_k \cdot V_{\text{vis}}, \quad V = W_v \cdot V_{\text{vis}}$$

where $V_{\text{vis}} \in \mathbb{R}^{n \times d}$ is the stacked visual feature tensor, and $W_k, W_v \in \mathbb{R}^{d_k \times d}$ are learned parameters.

4.7.3 Scaled Dot-Product Attention

The core of the attention mechanism lies in computing how well each key matches the query. This is done using the dot product between the query and each key, scaled by the square root of the key dimensionality dk to maintain stability. These scores are then normalized using softmax to produce attention weights, which determine how much focus each visual value should receive:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$$

This operation allows the model to focus more on relevant visual features in relation to the question semantics.

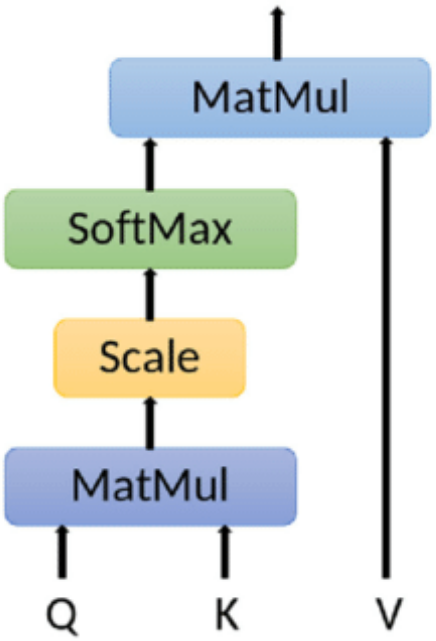


Figure 4.7: Scaled dot product Source: Taken from (Zhang et al. 2019)

4.7.4 Multi head Attention

To capture diverse types of interactions, multi head attention runs several attention operations in parallel. Each head operates on different subspaces of the input:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(head_1, \ldots, head_h)W^O$$

$$\text{where} \quad head_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Here, $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d_h \times d}$ are learned projection matrices for each head, and $W^O \in \mathbb{R}^{hd_h \times d}$ projects the concatenated outputs back to the original space.

4.7.5 Bidirectional cross attention Network

The Bidirectional Cross modal Attention architecture is a core of this Bidirectional Cross Modal Attention arctectrue which is designed to fuse textual and visual information through dual, interactive attention mechanisms. At its core, we employs a pretrained BERT encoder to extract rich linguistic features from text inputs, while simultaneously harmonizing diverse visual features such as object feature, temporal dynamics feature,and CLI embeddings attributes into a unified 768 dimensional space via projection layers. Our architecture's defining strength lies in its bidirectional cross attention design: the **text to visual attention** pathway refines visual features using language context , while the **visual to text attention** pathway enhances textual features with visual grounding by emphasizing words aligned with detected objects. This dual interaction ensures balanced, context aware fusion, preventing dominance by either modality. After mean pooling the attention outputs, our model concatenates the refined features and processes them through a classifier with dropout and ReLU activation, enabling robust multi modal decision-making. Key strengths our model include its ability to integrate heterogeneous visual features bridging the **modality gap** through projection layers, leverage pretrained BERT for strong textual priors, and maintain modularity for scalability across diverse visual inputs. By explicitly modeling bidirectional dependencies, the model offer high interpretability and accuracy in question answering on MSVD QA dataset, where aligning fine grained visual details with complex language queries is critical. Its design balances computational efficiency with expressive power, making it adaptable to both resource-constrained and high-performance multi modal applications.

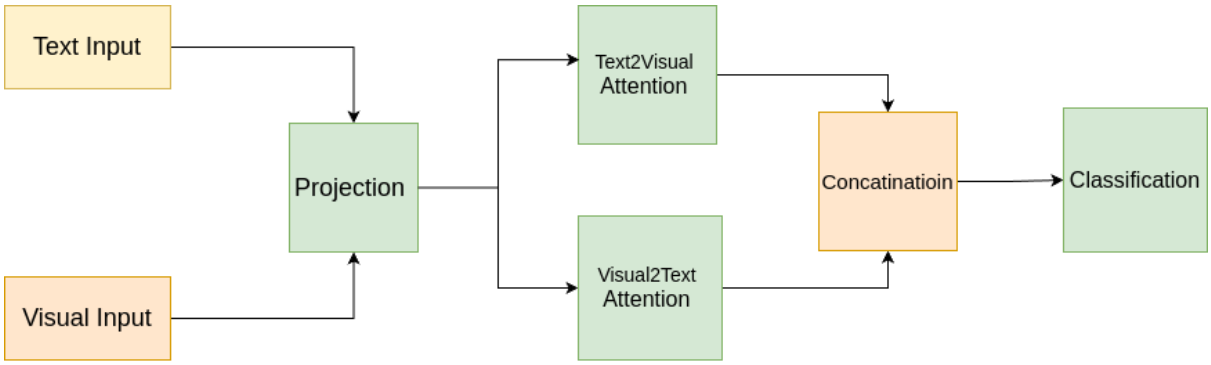


Figure 4.8: Inside Bidirectional Cross Modal Attention

4.7.6 Cross modal Fusion Output

The model produces fused representations through bidirectional attention and concatenation:

$$
\begin{aligned}
&\text{Text to Visual:} \quad \text{t2v\_out}, \_ = \text{MultiHead}(Q{=}\text{text\_feat}, K{=}V{=}\text{visual\_feat}) \\
&\text{visual to text:} \quad \text{v2t\_out}, \_ = \text{MultiHead}(Q{=}\text{visual\_feat}, K{=}V{=}\text{text\_feat}) \\
&\text{Mean Pooling:} \quad \begin{cases} \text{t2v\_vec} = \frac{1}{L}\sum_{i=1}^{L} \text{t2v\_out}^{(i)} \in \mathbb{R}^{B\times 768} \\ \text{v2t\_vec} = \frac{1}{M}\sum_{j=1}^{M} \text{v2t\_out}^{(j)} \in \mathbb{R}^{B\times 768} \end{cases} \\
&\text{Fusion:} \quad \text{fused} = [\text{t2v\_vec} \parallel \text{v2t\_vec}] \in \mathbb{R}^{B\times 1536}
\end{aligned}
$$

where:

- L: Text sequence length (from BERT's output)
- M=4: Number of visual features (object, temporal, scene)
- $\parallel$: Concatenation operation along feature dimension

The fused representation is processed by the classifier network:

$$\text{logits} = W_2\big(\text{Dropout}\big(\text{ReLU}(W_1(\text{fused}))\big) \in \mathbb{R}^{B\times N}$$

with:

- $W_1 \in \mathbb{R}^{1536\times 512}$: First linear transformation
- $W_2 \in \mathbb{R}^{512\times N}$: Final classification layer
- N: Number of answer classes (`num_answers`)

This dual-attention design explicitly preserves bidirectional interaction patterns between modalities, enabling the model to jointly reason about visual concepts grounded in language context and textual semantics grounded in visual evidence.

4.7.7 Classifier

The classifier serves as the final decision layer in the Cross modal Video QA architecture. It receives as input the fused representation produced by the Cross modal attention mechanism. This representation encodes the semantic relationship between the question and the visual content of the video. The classifier is implemented as a feed forward neural network consisting of two linear transformations with an intermediate non linear activation and dropout regularization. The first linear layer projects the 768 dimensional input into a hidden space, enabling the model to refine the fused features. A ReLU

activation introduces non linearity, which allows the model to learn complex mappings between inputs and output classes. A dropout layer is applied to reduce over fitting by randomly zeroing out elements of the hidden representation during training. The final linear layer maps the hidden features to a vector of logits, one for each answer class. These logits represent unnormalized confidence scores and are used during training in conjunction with a cross entropy loss function. During inference, the index of the maximum logit is selected as the predicted answer. This module plays a crucial role in transforming the multi modal, question aware visual representation into a concrete answer prediction.

CHAPTER FIVE

IMPLEMENTATION OF THE BIDIRECTIONAL CROSS MODAL ATTENTION

This chapter presents the implementation details of the Bidirectional Cross Modal Attention model architecture. It covers the working environment, the development of the multi modal UNDERSTANDING FOR AMHARIC VIDEO QUESTION ANSWERING USING Cross modal LEARNING, and the experiments conducted to evaluate the system.

5.1 Implementation Environment

The implementation and training of the Bidirectional Cross Modal Attention model were carried out across multiple computational platforms to accommodate different stages of development and experimentation.

- **Laptop Linux Workstation**: Used for high performance training, feature extraction, and heavy experimentation.
  - Device: ASUSTeK COMPUTER INC. ROG Zephyrus G14 GA402RJ
  - Operating System: Pop!_OS 22.04 LTS
  - Processor: AMD® Ryzen 9 6900HS with Radeon Graphics ×16
  - Graphics: AMD® Radeon RX 6700S + Integrated Radeon GPU
  - RAM: 16.0 GiB
  - Storage: 1.0 TB SSD
- **Kaggle Cloud Platform**: Employed for large scale training and experimentation using GPU resources.
  - GPU: Up to 30 hours per session using NVIDIA Tesla P100 / T4 GPUs
  - Environment: Hosted Jupyter Notebooks with access to accelerated hardware
  - Frameworks: PyTorch, TensorFlow, Hugging Face Transformers, OpenCV, etc.
  - Kernel Runtime: Shared RAM with up to 16 GB, and GPU acceleration

The implementation of the Bidirectional Cross Modal Attention model was carried out using a range of development tools, programming environments, and libraries suited for deep learning applications. Visual Studio Code was employed as the primary integrated development environment (IDE) for building and organizing the project structure. Python 3.6 was used as the main interpreter, while the model itself was implemented using the PyTorch framework, which provided the necessary tools for defining and training deep neural networks.

5.2 Environmental Setup

In addition to Visual Studio Code, several other software tools were utilized to facilitate model development and experimentation:

- **Anaconda**: Anaconda version 1.9.7 (64-bit) was used for managing the Python environment. It simplifies the installation and management of packages, dependencies, and environments required for machine learning workflows.
- **Jupyter Notebook**: Widely preferred in the machine learning community, Jupyter Notebook version 6.0.0 was used for interactive development, model testing, and visualization. It enabled a modular and transparent way to develop and debug code, making it especially useful for experimenting with different configurations and data flows.

5.3 Proposed Model Bidirectional Cross Modal Attention

5.3.1 Bidirectional Encoder Representations from Transformers

The text encoder is the first module in the model pipeline. Unlike traditional RNN based encoders such as Bi-LSTM, this implementation utilizes a pretrained BERT transformer model to encode the input question. The input text, written in Amharic, is first tokenized using the Bert Tokenizer, which handles token segmentation, truncation, padding, and special token addition. Each token is mapped to an index sequence and converted into embeddings using the BERT model. I employed BERT specifically for encoding Amharic questions and answers due to its pre trained language understanding capabilities because it was trained on more that 3 million Amharic data. Bi-LSTM lacks prior exposure to Amharic and is limited in capturing rich semantic representations which will li. My objectivemite the final model perfforamncegoes beyond modeling sequential dependencies—it requires deep linguistic understanding, which is crucial for question answering tasks, especially in a low-resource language like Amharic. Since Video QA involves not only temporal reasoning but also precise language comprehension, using a model that has been pre-trained and fine-tuned on

Amharic is essential. BERT provides contextual embeddings that effectively capture meaning and nuance, making it far more suitable for handling the complexities of Amharic in multimodal tasks like Video QA.

Instead of using per token embeddings or Bi-LSTM outputs, the model computes the mean of all token embeddings from BERT's final hidden layer to represent the entire question as a single fixed length vector. This 768 dimensional semantic embedding captures the syntactic and contextual information necessary for aligning the question with visual features. pretrained multilingual BERT for amharic already includes Amharic in its training, so it starts with some knowledge of Amharic structure and vocabulary. Bi-LSTM, on the other hand, has to learn everything from scratch and struggles with long range dependencies and sparse training data.

```
# Tokenize question
toks = self.tokenizer(q, padding="max_length", truncation=False,
                                max_length=self.max_q_len, return_tensors="pt")
```

Listing 5.1: Snippet code for Text tokenizer code

5.3.2 TimeSformer

The next step outlines the process of extracting temporal and global video features using the pretrained TimeSformer model. The process begins with the initialization of key components. A pretrained image processor is loaded, and the TimeSformer backbone is initialized. The model outputs a hidden state of size $D = 768$. To summarize frame level embeddings temporally, a lightweight attention module SimpleAttention is defined. This module applies a linear projection followed by softmax to compute attention weights, as shown in Snippet Code 5.2.1.

```
processor = AutoImageProcessor.from_pretrained(
"MCG-NJU/videomae-base"
)
model = TimesformerModel.from_pretrained(
    "facebook/timesformer-base-finetuned-k400"
).to(device).eval()

class SimpleAttention(nn.Module):
    def __init__(self, dim):
        super().__init__()
        self.proj = nn.Linear(dim, 1)

    def forward(self, x):
        w = torch.softmax(self.proj(x).squeeze(-1), dim=1)
        return (w.unsqueeze(-1) * x).sum(dim=1)
```

```
attn = SimpleAttention(model.config.hidden_size).to(device)
```

Listing 5.2: Snippet code for Model loading and attention setup

Selected frames are then divided into smaller chunks of 4 frames for efficient batch processing.

```
MAX_FRAMES, CHUNK = 16, 4

all_frames = sorted(
    p for p in os.listdir(fldr) if p.lower().endswith((
    ".jpg", ".png"
    ))
)
idx = np.linspace(
0,
len(all_frames) - 1,
min(MAX_FRAMES, len(all_frames)),
dtype=int
)
frames = [os.path.join(fldr, all_frames[i]) for i in idx]
```

Listing 5.3: Snippet code for Frame selection and sampling

The TimeSformer processes a chunk of 4 frames, each resized to 224×224, and tokenized into $14 \times 14 = 196$ patches per frame. The model outputs a tensor of shape $[1, 1 + 4 \times 196, 768]$, where the first token is a CLS token, followed by patch tokens. These patch embeddings are reshaped into per-frame structure and averaged spatially to obtain one embedding per frame.

```
for i in range(0, len(frames), CHUNK):
    imgs = [
    Image.open(p).convert("RGB") for p in frames[i:i + CHUNK]
    ]
    inp = processor(imgs, return_tensors="pt").to(device)

    with torch.no_grad():
        out = model(**inp).last_hidden_state  # [1, 1 + T*P, D]

    cls_list.append(out[:, 0, :])  # CLS token: [1, D]
    B, TP1, _ = out.shape
    T = len(imgs)
    tkn = out[:, 1:, :].reshape(B, T, -1, D).mean(2)
    frame_feats.append(tkn)
```

Listing 5.4: Snippet code for Snippet code for Chunk-level feature extraction

After all chunks are processed, their CLS tokens are averaged to produce a global video embedding. Meanwhile, all frame level vectors are concatenated and passed through the attention module to produce a temporal summary. Both outputs are serialized for later use.

```
cls_vec = torch.cat(cls_list, dim=0).mean(0, keepdim=True)
frames_cat = torch.cat(frame_feats, dim=1)
temporal = attn(frames_cat)  # [1, 768]

features[vid] = {
    "cls": cls_vec.cpu().numpy(),
    "temporal": temporal.detach().cpu().numpy()
}
with open(out_path, "wb") as f:
    pickle.dump(features, f)
```

Listing 5.5: Snippet code for Global and temporal feature storage

This two vector representation — one global and one temporal — is used in the next stage of the pipeline, where it is integrated with the corresponding question features using a Cross modal attention network.

5.3.3 Faster RCNN and CLIP based Object Feature Extraction

Each video frame is processed by Faster RCNN to generate bounding boxes for detected objects. The bounding boxes are filtered by a confidence threshold greater than 0.5, and the object region is cropped for each box. Faster RCNN operates using a two stage object detection pipeline. In the first stage, it generates region proposals likely to contain objects using a Region Proposal Network (RPN). In the second stage, these proposals are classified and refined. For each detected object, the model outputs bounding box coordinates and a confidence score. Only boxes with a confidence score above 0.5 are retained for semantic processing.

```
img = Image.open(img_path).convert("RGB")
inp = det_transform(img).to(device)
with torch.no_grad():
    pred = fastrcnn_model([inp])[0]
boxes = pred["boxes"].cpu().numpy()
scores = pred["scores"].cpu().numpy()
boxes = boxes[scores >= 0.5]
```

Listing 5.6: Snippet code for Object detection and region cropping

Extracts object level semantic information from each video frame by combining two key modules: a region proposal and classification model (Faster RCNN), and a multilingual

visual semantic embedding model (CLIP). Each detected object region is semantically aligned with the video Q and A to determine its relevance. The process is outlined below using Python code snippets and dimensional commentary. The multilingual caption is split into individual words, each encoded using the MCLIP model. These serve as reference text embeddings for identifying object word relevance.

```
words = [w.strip() for w in qa_dict[vid].split() if w.strip()]
with torch.no_grad():
    word_feats = model_mul.forward(words, tokenizer).to(device)
    word_feats /= word_feats.norm(dim=-1, keepdim=True)
```

Listing 5.7: Snippet code for Word encoding using MCLIP

For each cropped object, CLIP is used to extract visual embeddings. These embeddings are then compared to each word embedding using cosine similarity (via dot product since all vectors are normalized). The most semantically similar word is selected for annotation. To unify the number of object representations across frames and words, a temporal attention module is applied. This ensures that every word object pair has a single fixed length vector summarizing its temporal appearance. For each video, features are arranged per word across 16 frames, then attended using learned Q, K, V matrices to compute a weighted average.

```
for b in boxes:
    x1, y1, x2, y2 = map(int, b)
    crop = img.crop((x1, y1, x2, y2))
    clip_inp = preprocess(crop).unsqueeze(0).to(device)
    with torch.no_grad():
        feat = clip_model.encode_image(clip_inp)
        feat /= feat.norm(dim=-1, keepdim=True)
        sim = (100.0 * feat @ word_feats.T).softmax(dim=-1)
        best = sim.argmax().item()
```

Listing 5.8: Snippet code for object word semantic alignment using CLIP

For each video and each unique word, we builds a length 16 sequence of frame embeddings and after that we concatenate all word vectors for each video into one long feature vector, saves that structure, then we reloads it to compute the maximum vector length. The attention based approach aggregates variable numbers of object features (each 512 dimensional) from videos into a consistent 1024 dimensional representation, regardless of the number of objects because 50% the video have two object with dimension 1024 . This method uses a multi query attention mechanism to summarize object features dynamically, avoiding zero padding and ensuring fixed size outputs suitable for our task.

5.3.4 Bidirectional Cross modal Attention

The **Bidirectional Cross modal Attention VQA** model fuses visual and textual features through dual cross attention layers that learn fine grained semantic alignment between modalities. For the textual modality, the input question is tokenized and passed through a pre trained BERT encoder. The output consists of *contextualized token level embeddings* for each input word or sub word, capturing deep linguistic relationships across the entire sequence. These token embeddings, with shape $[L, B, 768]$ (where L is the sequence length with max length and B the batch size), are used as input to the Cross modal attention mechanism, enabling word level interaction with visual features. Unlike sentence level pooling (e.g., using `[CLS]` or mean pooling), this approach retains token-wise granularity, which is crucial for aligning specific words in the question with relevant regions or objects in the visual input.

```
text_out = self.bert(
input_ids=input_ids, attention_mask=attention_mask
)
text_feat = text_out.last_hidden_state.transpose(0, 1)
```

Listing 5.9: Snippet code for Text encoding using BERT

Four types of visual features—video-level global (CLS), temporal summary, CLIP object level features, and high dimensional semantic features from object tracking—are linearly projected into the same embedding space as the text. This ensures compatibility for attention computation. Each projection includes a BatchNorm layer to stabilize training. After projection, the features are stacked into a single tensor with dimension $[B, 4, 768]$, where B is the batch size and d is the embedding size.

```
visual_feats = torch.stack([
            self.visual_proj['cls'](cls_feat),
            self.visual_proj['temp'](temp_feat),
            self.visual_proj['obj'](obj_feat)
        ], dim=1)  # [bs, 4, 768]
        visual_feat = visual_feats.transpose(0, 1)# [4, bs, 768]
```

Listing 5.10: Snippet code for Visual feature projection

To fuse multi modal representations bidirectionally, multi head attention is applied in both directions. Specifically, the text embeddings act as queries while the four projected visual feature types serve as keys and values, allowing the model to learn which visual features are relevant to the textual context. Conversely, the visual features act as queries with the text embeddings as keys and values, enabling the model to understand which parts of the text are important given the visual information. Multi head attention is used

because it enables the model to jointly attend to information from different representation subspaces at multiple positions, enhancing the model's ability to capture diverse and complex Cross modal relationships. This bidirectional Multi head attention mechanism computes alignment scores that produce weighted, context aware feature representations in both modalities, fostering richer semantic fusion.

```
t2v_out, _ = self.text2vis_attn(
query=text_feat, key=visual_feat, value=visual_feat
)
            t2v_vec = t2v_out.mean(dim=0)  # [bs, 768]
```

Listing 5.11: Snippet code for Text to Visual Attention

```
    v2t_out, _ = self.vis2text_attn(
    query=visual_feat, key=text_feat, value=text_feat
    )
    v2t_vec = v2t_out.mean(dim=0)  # [bs, 768]
```

Listing 5.12: Snippet code for Visual to Test Attention

The outputs of both attention mechanisms are mean-pooled across sequence dimensions and then concatenated to form a unified feature vector. This concatenation fuses the complementary, context aware representations from both modalities into a single joint embedding, which is then passed to subsequent layers for classification or prediction.

```
fused = torch.cat([t2v_vec, v2t_vec], dim=1)  # [bs, 1536]
```

Listing 5.13: Snippet code for Fuse v2t and t2v

The attended multi modal representation is pooled using mean operation and passed to a classification module composed of a linear layer, ReLU activation, dropout for regularization, and another linear layer. This final step predicts the most likely answer class.

5.3.5 Classification Layer

The attended multi modal representation is aggregated by mean pooling and passed into a classification head. This head consists of a linear layer followed by a ReLU activation and dropout, then a final linear layer. The intermediate hidden layer allows the model to learn complex representations and enhances its capacity to capture subtle patterns across modalities. The dropout layer improves generalization by reducing overfitting. The final linear transformation produces a vector of size $\mathbb{R}^C$, where C is the number of answer classes. A softmax function is implicitly applied during loss computation to interpret these logits as class probabilities.

Internally, this can be expressed as:

$$\hat{y} = W_2(\text{Dropout}(\text{ReLU}(W_1 h))) \tag{5}$$

Where h is the attended representation, W_1 and W_2 are the learned weight matrices of the classifier.

```
self.classifier = nn.Sequential(
            nn.Linear(768 * 2, hidden_dim),
            nn.ReLU(),
            nn.Dropout(dropout_rate),
            nn.Linear(hidden_dim, num_answers)
        )
return self.classifier(fused)
```

Listing 5.14: Snippet code for Classification layer

Output: A vector $\in \mathbb{R}^C$ where C is the number of answer classes, representing the predicted category based on fused Cross modal information. This module effectively learns which visual channels are most relevant to a question and dynamically adapts attention weights for improved answer reasoning in Video QA.

5.4 Experiment Class

To evaluate variations in text and visual embedding strategies, 4 experiment classes were introduced using the Cross modal attention architecture:

- **Multilingual Cross modal Attention - BERT (CMA-MCLIP)**: The baseline experiment employs a multilingual CLIP model which mainly designed for non english language including Amharic in order to attain Cross modal attention. Question text is embedded, and cross attention is employed using the question as the query and visual features as keys and values. This enables the model to pay attention to semantic visual regions conditioned on the question so that text and video features can be aligned semantically.
- **Cross modal Attention - BERT(CMA-BERT)**: This experiment replaces the multilingual CLIP with BERT for text encoding. BERT demonstrates improved performance compared to the CLIP baseline, likely due to its stronger vision language pretraining. However, the attention remains unidirectional, using the question as the query and visual features as the key/value, which limits the model's ability to fully leverage bidirectional interactions. Unidirectional attention restricts the flow of information to a single direction—from the question to the visual features—meaning that the model cannot effectively incorporate

feedback from the visual modality back to the textual representation. This asymmetry can lead to suboptimal alignment between modalities, reducing the richness of multi modal context modeling. In contrast, bidirectional Cross modal attention allows for dynamic interaction where both textual and visual features iteratively inform and refine each other, enabling deeper understanding and better fusion of semantic cues across modalities. Hence, relying solely on unidirectional attention constrains the model's capacity to capture complex Cross modal relationships essential for accurate video question answering.

- **Bidirectional Cross modal attention - BERT(BCMA-BERT)**: This experiment extends the previous unidirectional Cross modal attention models by implementing bidirectional attention, meaning the model attends both ways: from text to visual features (text→visual) and from visual features back to text (visual→text). Unlike unidirectional attention where only the question queries the visual features bidirectional attention enables a richer and more interactive fusion between modalities. This allows the model to capture mutual dependencies and contextual interactions between the question and visual input, improving understanding by letting visual cues influence the interpretation of the question and vice versa. This holistic Cross modal interaction helps to better align semantics across modalities, often resulting in enhanced reasoning and prediction capabilities in video question answering tasks.

- **Bidirectional Cross modal Attention - MCLIP(BCMA-MCLIP)**: This experiment replaces BERT with multilingual CLIP for text encoding.

These variations assess the influence of embedding sources and attention fusion strategies on model performance. Each experiment provides a unique insight into how modality alignment, pooling strategies, and feature dimensionality impact the final classification accuracy and generalization across diverse Video QA scenarios.

5.5 Training Details

The training setup for all models uses a unified framework designed to ensure consistent evaluation and optimization across different Cross modal attention architectures. The dataset comprises Amharic video question-answer pairs alongside pre-extracted visual features from multiple modalities, such as Timesformer video features, fused multi modal embeddings, and object level CLIP features. The dataset is split into 61% training, 13% validation, and 26% testing to provide a balanced representation for model learning and also it is the standard partition for the dataset. Data loading employs a batch size of 24, balancing between GPU memory constraints and efficient training throughput, with

shuffling enabled in training to promote model generalization and pin memory for faster data transfer.

Our model architecture, typically instantiated as Bidirectiona CMA or its variants, The training utilizes AdamW optimizer with a relatively low learning rate of 3e-5 to carefully fine tune the models without destabilizing pretrained weights. Weight decay of 1e-5 is applied for regularization to prevent overfitting by penalizing large weights, while dropout at 0.3 further reduces co-adaptation of neurons. Label smoothing (0.1) in the cross-entropy loss encourages the model to be less confident on any single class, improving generalization. Gradient clipping with a threshold of 0.5 prevents exploding gradients during backpropagation, improving training stability.

Training is performed for up to 50 epochs with early stopping patience set at 5 epochs to avoid wasting resources when validation performance plateaus. Gradient accumulation over 2 steps is applied to effectively increase the batch size without increasing GPU memory usage. Additionally, a warm up period of 100 steps gradually increases the learning rate from zero, helping the optimizer start with stable updates. The learning rate scheduler reduces the learning rate by half when the validation loss stagnates, enabling finer optimization. The best performing model checkpoint based on validation loss is saved for final evaluation.

To comprehensively evaluate our model's performance and benchmark it against existing methods, the training validation and testing are primarily conducted on the Amharic dataset, but evaluation also includes an English MSVD QA. This allows direct comparison with current state of the art models and published benchmarks, providing insight into our model's cross-lingual generalization capabilities and overall effectiveness. By testing on both Amharic and English MSVD QAs, the approach ensures that improvements are not language specific and helps identify strengths and weaknesses across different linguistic contexts. This comparative evaluation is crucial for validating the model's utility in multilingual video question answering scenarios.

Table 5.1: Lists of experimental classes

Notation	**Experiment**	**Datasets**	Text Encoder	Attention Type	**Model used**
CMA-CLIP	Multilingual Cross modal Attention: employs a multilingual CLIP model which mainly designed for non english language including Amharic	MSVD-QA	MCLIP	Unidirectional (text→visual)	Cross Modal Attention
CMA-BERT	BERT based Cross modal Attention: BERT for text and encoding	MSVD-QA	CLIP	Unidirectional (text→visual)	Cross Modal Attention
BCMA-CLIP	Bidirectional Cross modal attention: employs multilangual CLIP with Bidirectional attention, the model attends both ways: from text to visual features (text→visual) and from visual features back to text (visual→text)	MSVD-QA	MCLIP	Bidirectional	Bidirectional Cross Modal Attention

BCMA-BERT	Bidirectional Cross modal attention: employs BERT with Bidirectional attention, the model attends both ways: from text to visual features (text→visual) and from visual features back to text (visual→text)	MSVD-QA	BERT	Bidirectional	Bidirectional Cross Modal Attention

CHAPTER SIX

RESULTS AND DISCUSSIONS

The experimental results of the Bidirectional Cross Modal Attention methods described in the previous chapters are presented. The performance of the Bidirectional Cross Modal Attention is evaluated and compared against existing state of the art approaches using relevant metrics such as accuracy, loss, and qualitative analysis. Specifically, this work investigates four different Cross modal attention architectures: multilingual CLIP based unidirectional attention, BERT based unidirectional attention, bidirectional Cross modal attention. The main objectives are to validate two key hypotheses: first, that incorporating bidirectional attention mechanisms enhances the interaction between visual and textual features, thereby improving overall model accuracy in visual question answering tasks. This chapter systematically discusses the experimental setup, training procedures, evaluation results, and comparative analysis to demonstrate the effectiveness of the Bidirectional Cross Modal Attention.

6.1 Video Question Answering

Video question answering requires the integration of textual and visual semantics to reason over a sequence of visual content. In this chapter, we present the experimental results obtained from multiple configurations of Cross modal attention models. Each experiment was designed to assess how different text encoders, feature integration strategies, and classifier input formats affect performance.

To develop the Bidirectional Cross Modal Attention model, four main experiments were conducted using different components and methods. The first experiment employed a multilingual CLIP based Cross modal attention mechanism in a unidirectional manner, where the question serves as the query and the visual features as key and value. While this established a baseline, the unidirectional attention limited the model's ability to fully capture complex interactions between modalities. MCLIP, with its fixed 77-token sequence length, often truncates longer questions, especially in Amharic, leading to loss of critical context and weaker question embeddings that impair Cross modal attention with visual features. While MCLIP is trained on multilingual image text pairs, its English centric pre training limits its effectiveness for low resource languages like Amharic. The second experiment replaced MCLIP with In contrast, BERT supports up to 512 tokens, capturing full question context without truncation, and it is fine tuned bert-base-multilingual-cased model on Amharic language texts, producing richer

embeddings. This enables better integration with the visual features, contributing to BERT's superior performance in our VQA task, which improved performance but still relied on unidirectional attention. To overcome this limitation, the third experiment introduced bidirectional Cross modal attention, enabling dynamic, two-way interactions between textual and visual features. This bidirectional mechanism allowed the model to better align and integrate information from both modalities, resulting in a significant boost in understanding and accuracy.

The initial experiments utilized object level feature representations by selecting key visual features based on keywords from the video captions, questions, and answers, employing a CLIP based variety of feature selection and representing the video using a single frame. While this improved alignment between visual and textual modalities, limitations remained in capturing the full interaction between the two. To overcome this, We propose a bidirectional Cross modal attention mechanism that enables dynamic, two-way interaction between visual and textual features. This approach enhances fine grained alignment and semantic consistency, allowing the model to better understand the relationships between video content and textual queries. Incorporating this mechanism significantly improves accuracy and reduces the semantic gap compared to previous baseline methods.

All experiments were conducted using the same dataset and environment for consistency. The Bidirectional Cross Modal Attention model was trained for up to 50 epochs with early stopping set to 5 epochs of no improvement to prevent overfitting. This early stopping helped reduce unnecessary training time while maintaining performance. Training hyperparameters are described in Chapter 5.Section 5.8. To stabilize training.

As discussed earlier, multiple experiments were conducted to evaluate the effectiveness of the Bidirectional Cross Modal Attention model. These experiments are assessed using qualitative evaluation metrics, and compared against existing baseline methods on the selected dataset. For objective evaluation, accuracy metrics were used. The training dynamics of the model are illustrated in the following figures To enhance the clarity of training dynamics.

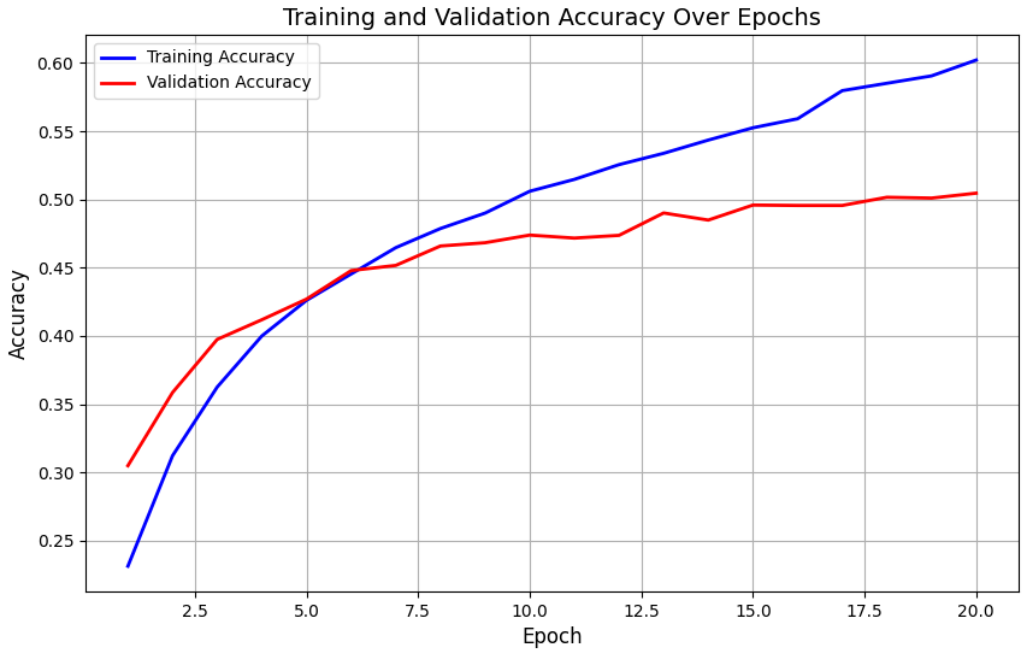


Figure 6.1: Training and validation accuracy curve of amharic VQA with BCMA model

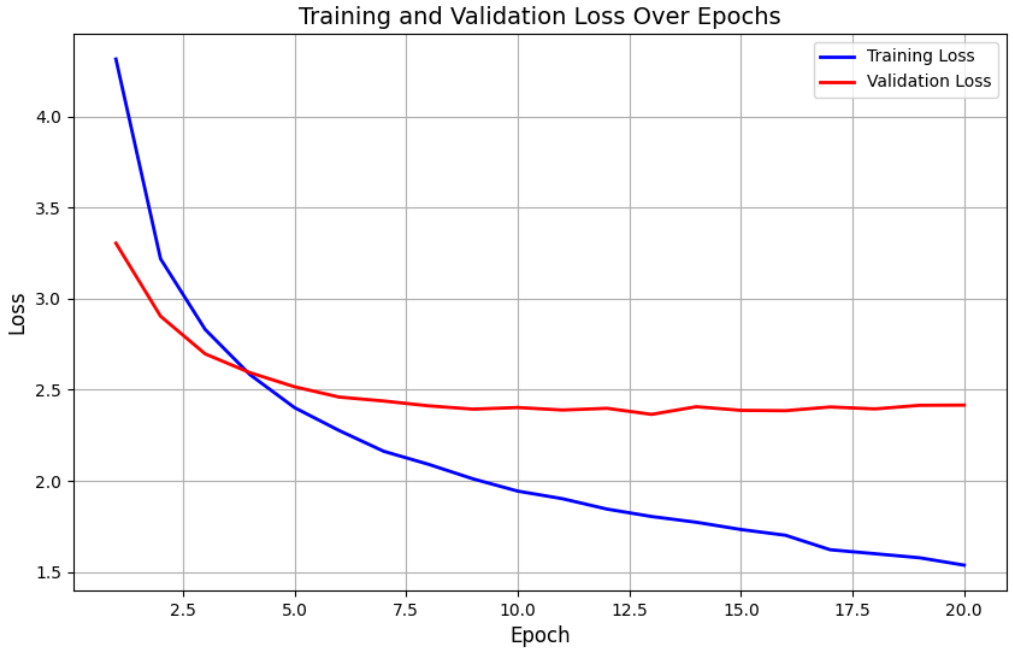


Figure 6.2: Training and validation loss curve of amharic VQA with BCMA model

Figure 6.3: Training and validation accuracy curve of amharic VQA with BERT-CMA model

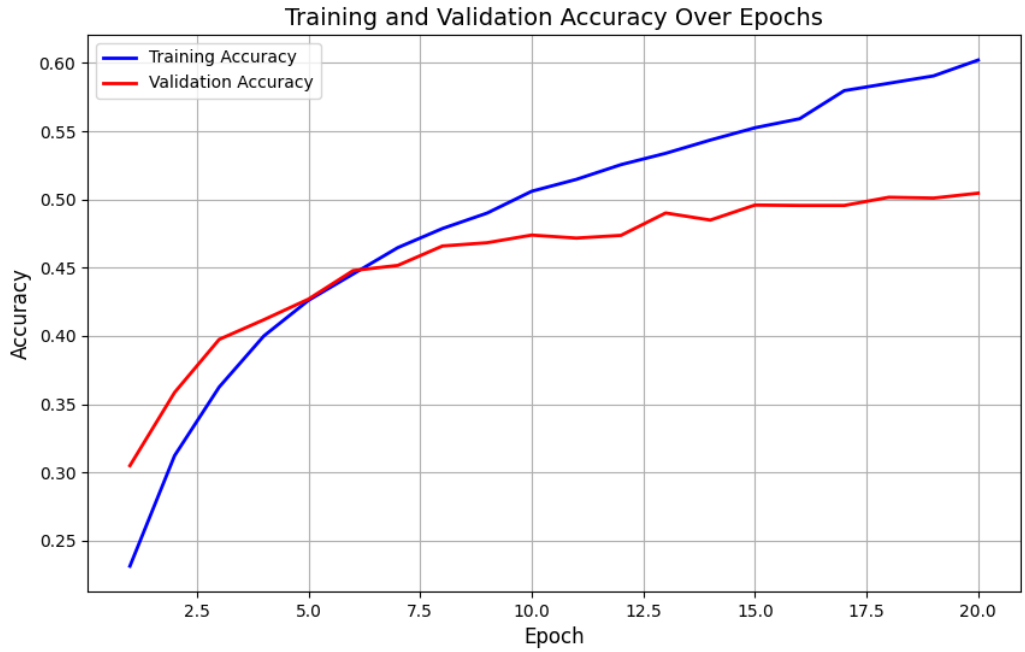


Figure 6.4: Training and validation loss curve of amharic VQA with BERT-CMA model

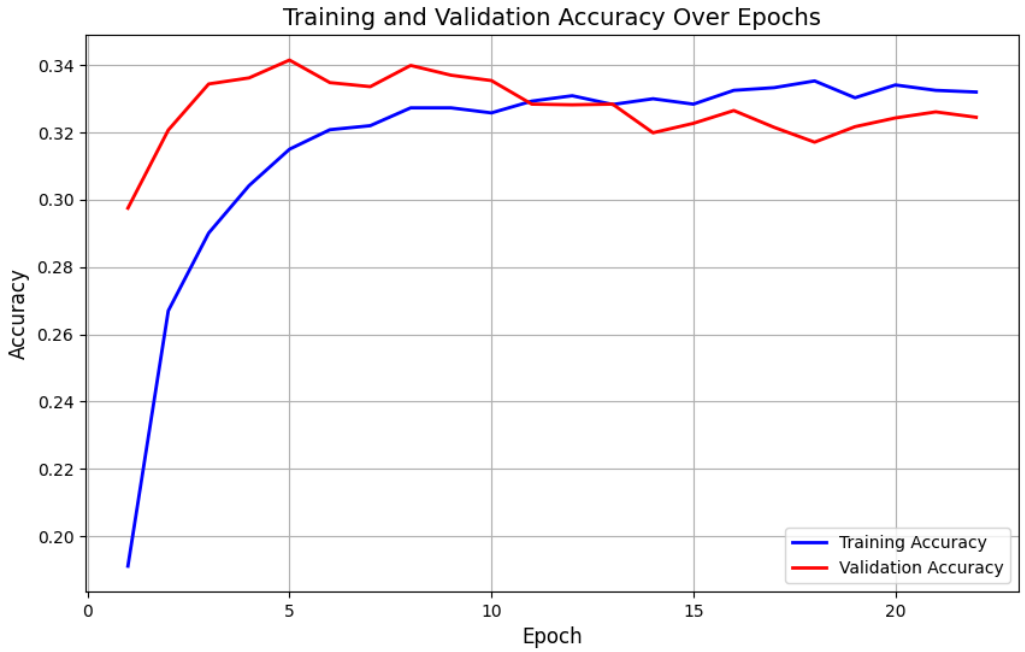


Figure 6.5: Training and validation accuracy curve of amharic VQA with CLIP-CMA model

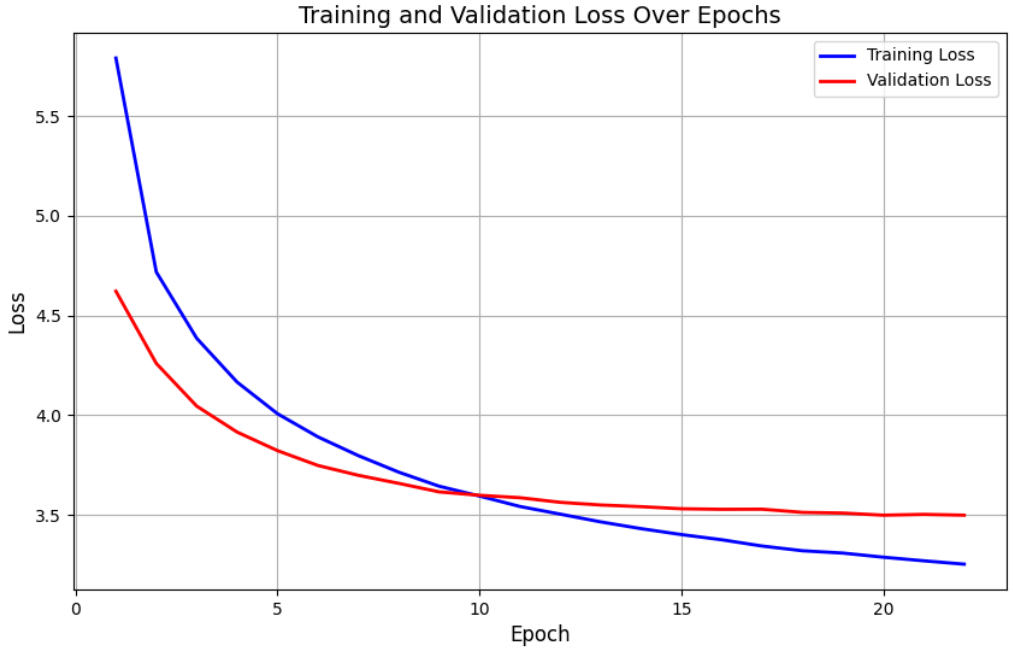


Figure 6.6: Training and validation loss curve of amharic VQA with CLIP-CMA model

For the english MSVD-QA traing accuray,training loss ,valaidation accuracy and validation loss is shown in the figures below.

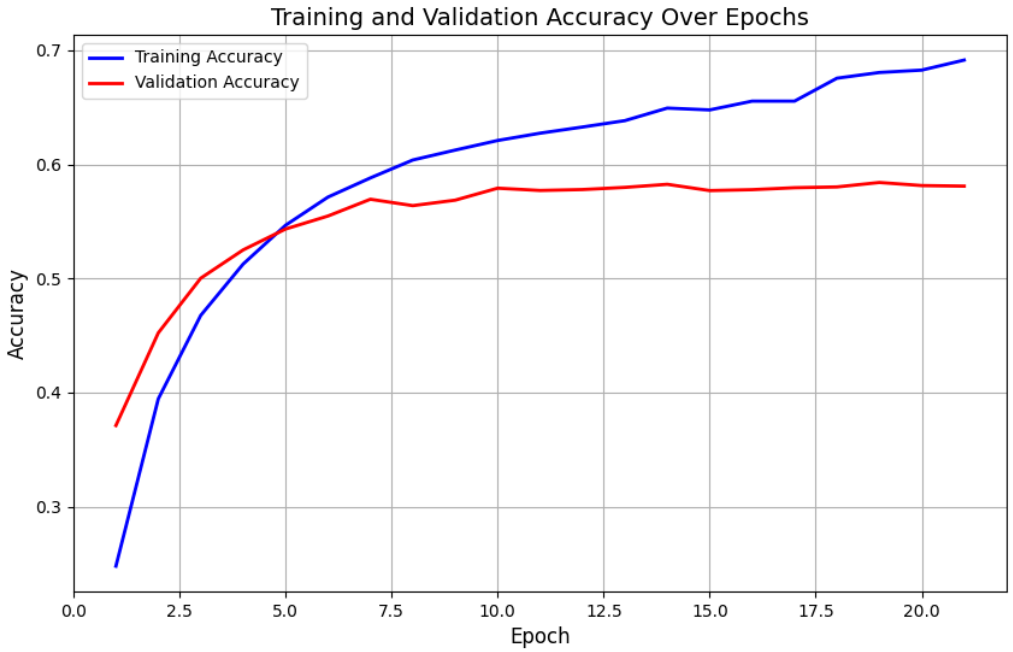


Figure 6.7: Training and validation accuracy curve of english VQA with BCMA model

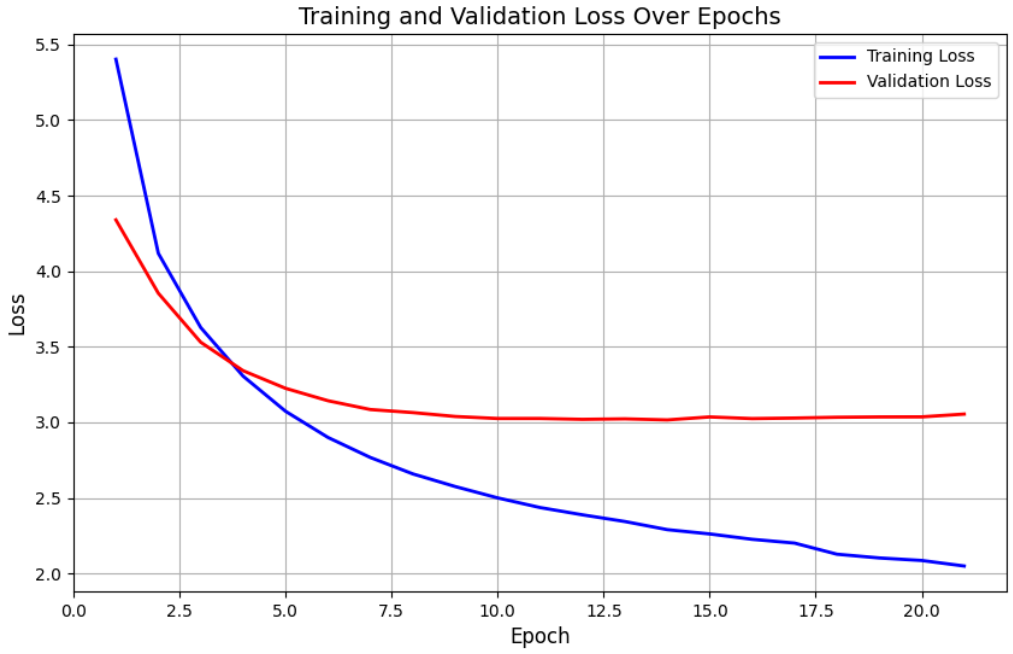


Figure 6.8: Training and validation loss curve of english VQA with BCMA model

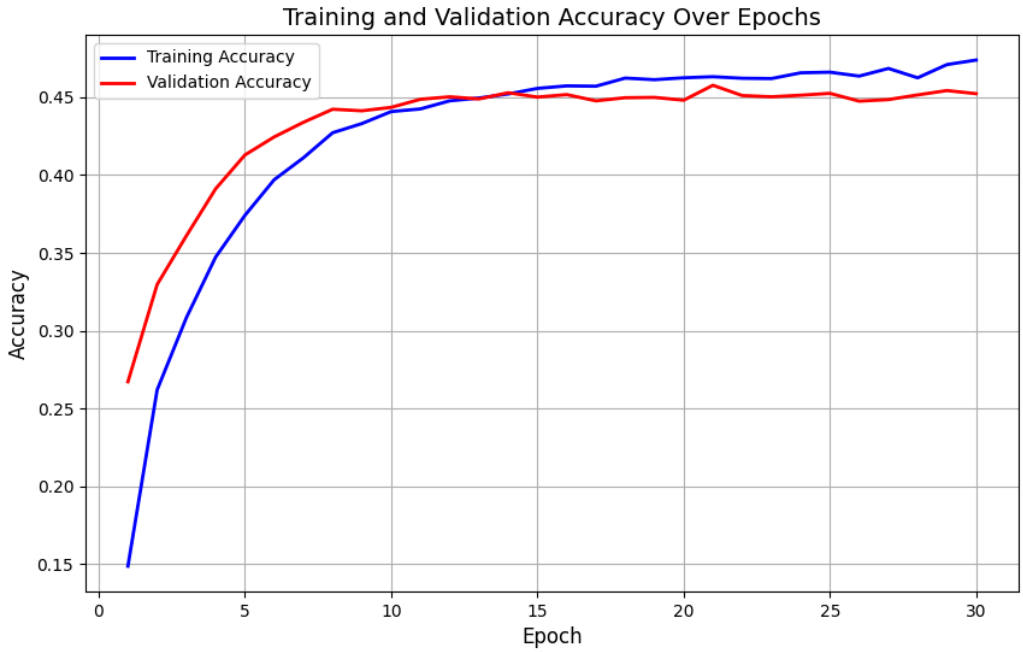


Figure 6.9: Training and validation accuracy curve of english VQA with BERT-CMA model

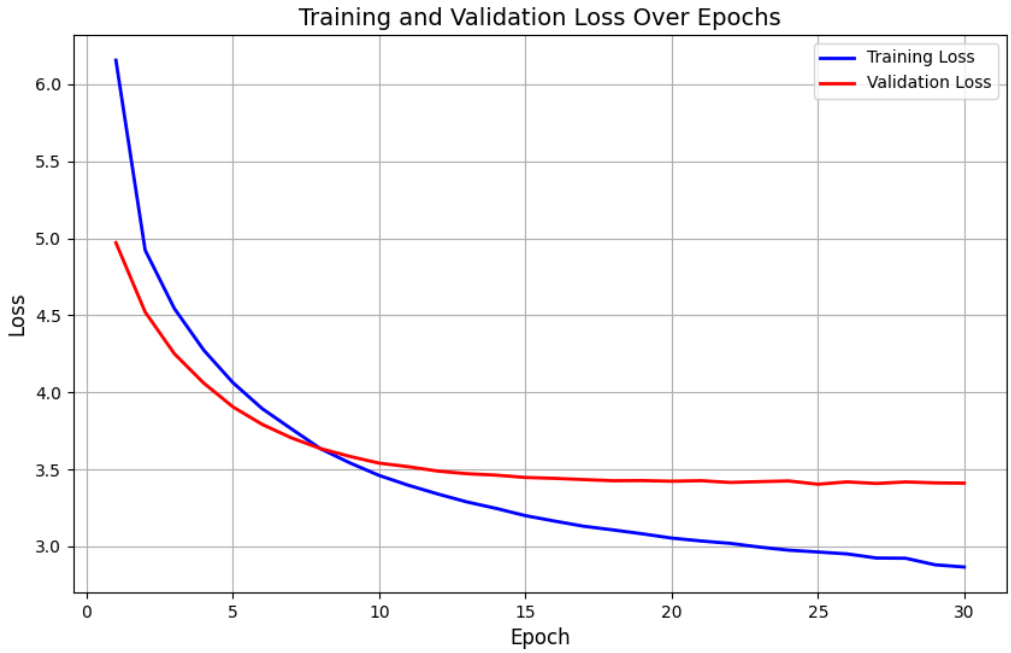


Figure 6.10: Training and validation loss curve of english VQA with BERT-CMA model

The model has converged on the validation set, reaching its generalization limit to determine that we use Early stopping around epoch so it would help prevent overfitting

and improve generalization. It stops training at the point where validation performance no longer improves, reducing the risk of the model memorizing training data rather than learning general patterns that work across videos and questions.

6.2 Evaluation Metrics

To assess the performance of the Bidirectional Cross Modal Attention video question answering models qualitative evaluation methods were employed. Among these, accuracy was used as the primary quantitative metric.

6.2.1 Accuracy

Accuracy was used as the core metric to evaluate how effectively the model answers questions based on video input. This metric reflects the proportion of correctly predicted answers over the total number of questions and is widely used in prior Video QA studies. The accuracy values reported in this work allow a direct comparison across different attention mechanisms and baseline models. Table 6.1 presents the accuracy results for the various experimental setups, clearly showing the performance improvements achieved through the Bidirectional Cross Modal Attention. Tag [en]represent english models and [am]tags shows result for amharic models.

Table 6.1: Accuracy comparison between all experiments.

No.	**Experiment**	**Model used**	**Test Accuracy**
1	[am]MCLIP-CMA	Cross Modal Attention(MCLIP)	31.608%
2	[am]BERT-CMA	Cross Modal Attention(BERT)	32.873%
3	[am]CLIP-BCMA	Bidirectional Cross Modal Attention(MCLIP)	32.873%
4	[am]BERT-BCMA	Bidirectional Cross Modal Attention(BERT)	**48.208 %**
5	[en]BERT-CMA	Cross Modal Attention(BERT)	45.041%
6	[en]BERT-BCMA	Bidirectional Cross Modal Attention(BERT)	**58.712%**

From above table 6.1, The bold values of accuracy show the best result in the experiments tag [am]represent experiment in Amharic MSVD-QA dataset and [en]tag is for the english experiment for MSVD-QA.

In the above Table 6.1, the performance of different Amharic and English multi modal models was compared based on their accuracy. Some models such as CLIP-CMA and BERT-CMA were implemented using different Cross modal attention strategies MCLIP, and BERT. The accuracy scores of each model show how well they perform on the multi modal video question answering task. Notably, the Bidirectional Cross Modal Attention [en]BCMA model achieves the highest accuracy at 58.712%, outperforming all other configurations. This indicates the effectiveness of using bidirectional Cross modal attention with BERT for aligning visual and textual representations. Similarly, [am]BERT-BCMA shows improved results among the Amharic models, suggesting the benefit of BCMA in handling low resource language features. These results demonstrate that the Bidirectional Cross Modal Attention approaches are capable of better capturing the semantic alignment between video content and the corresponding question, leading to more accurate answers.

The next figure illustrates the accuracy scores of different experiments conducted on English MSVD QAs. The graph presents each model alongside its corresponding accuracy score to visually compare the performance of various current benchmark papers.

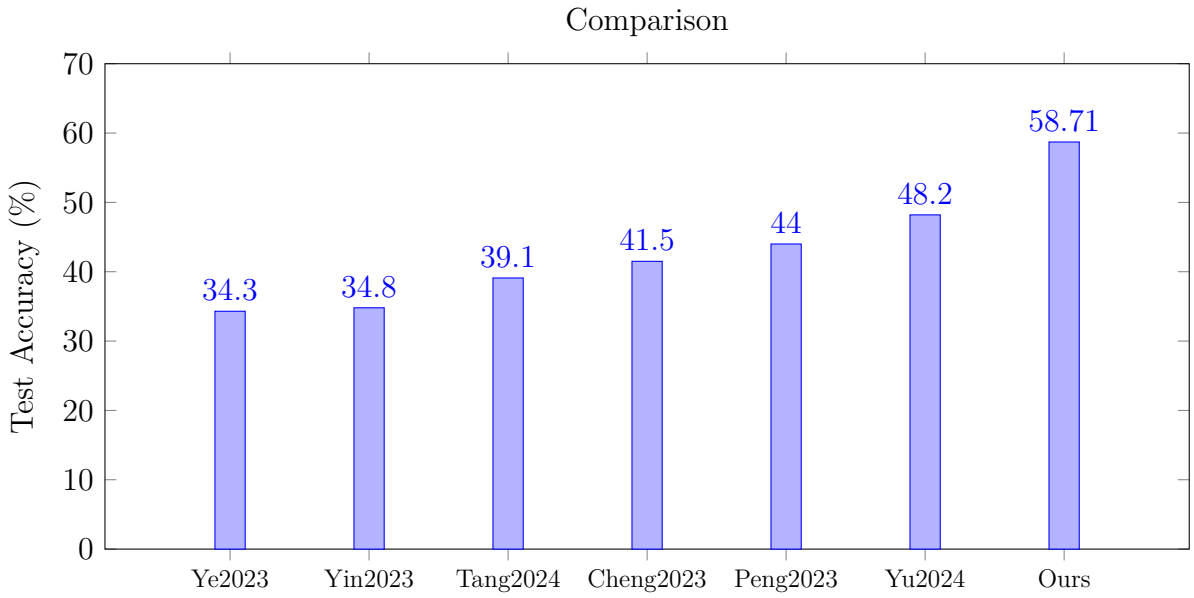


Figure 6.11: Comparison of Video QA model accuracies on the English MSVD QA.

6.2.2 Evaluation Matrix for Different Question Type

In this work, we evaluate precision not only on overall predictions but also with respect to different types of questions, categorized by their tags **What**, **When**, **Who**, **How** and **Where**. This tag based evaluation provides a more fine grained understanding of model performance across different linguistic and reasoning requirements.

Table 6.2: Evaluation metrics for the best performing model [am]BERT-BCMA by question type.

Metric	**ምን (7595)**	**መቼ (66)**	**ማን (4564)**	**እንዴት (66)**	**የት (101)**	**ሌሎች (500)**
Accuracy	34.9045%	72.7273%	71.0342%	39.3939%	25.7426%	44.4000%
Precision	99.3256%	97.9592%	81.7448%	100.0000%	100.0000%	96.1039%
Recall	99.8117%	90.5660%	96.9498%	96.2963%	96.2963%	91.3580%
F1 Score	99.5681%	94.1176%	88.7004%	98.1132%	98.1132%	93.6709%

The evaluation metrics reveal varied performance across different Amharic question types for the [am]BERT-BCMA model. The model shows outstanding precision and recall for the **"ምን" (What)** category, with an F1-score of 0.996, indicating highly reliable predictions for fact based questions. Similarly, **"እንዴት" (How)** and **"የት" (Where)** questions achieved very high precision and F1-scores (both 0.981), showing strong ability to handle procedural and locational queries. For **"ማን" (Who)** questions, although recall is high (0.97), precision is relatively lower (0.82), suggesting the model tends to overpredict person based answers. **"መቼ" (When)** questions also perform well with a balanced precision (0.98) and recall (0.91), yielding an F1-score of 0.94. The "other" category, which includes uncategorized questions, maintains solid scores across all metrics. Despite strong per-category performance, the overall accuracy and F1-score remain at 0.482, indicating room for improvement in generalization and class balance, especially as the dominant question types **(like "ምን")** influence the overall metric heavily due to their high support.

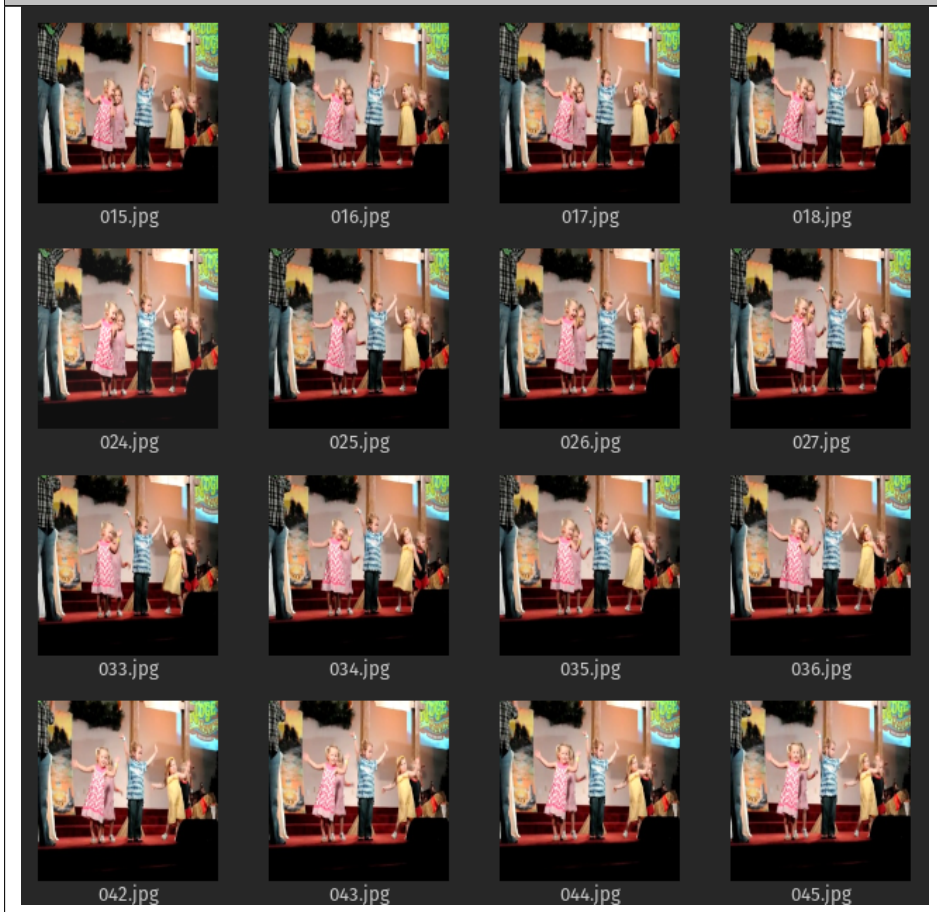
Table 6.3: Evaluation matrix for best performing model for [en]BERT-BCMA.

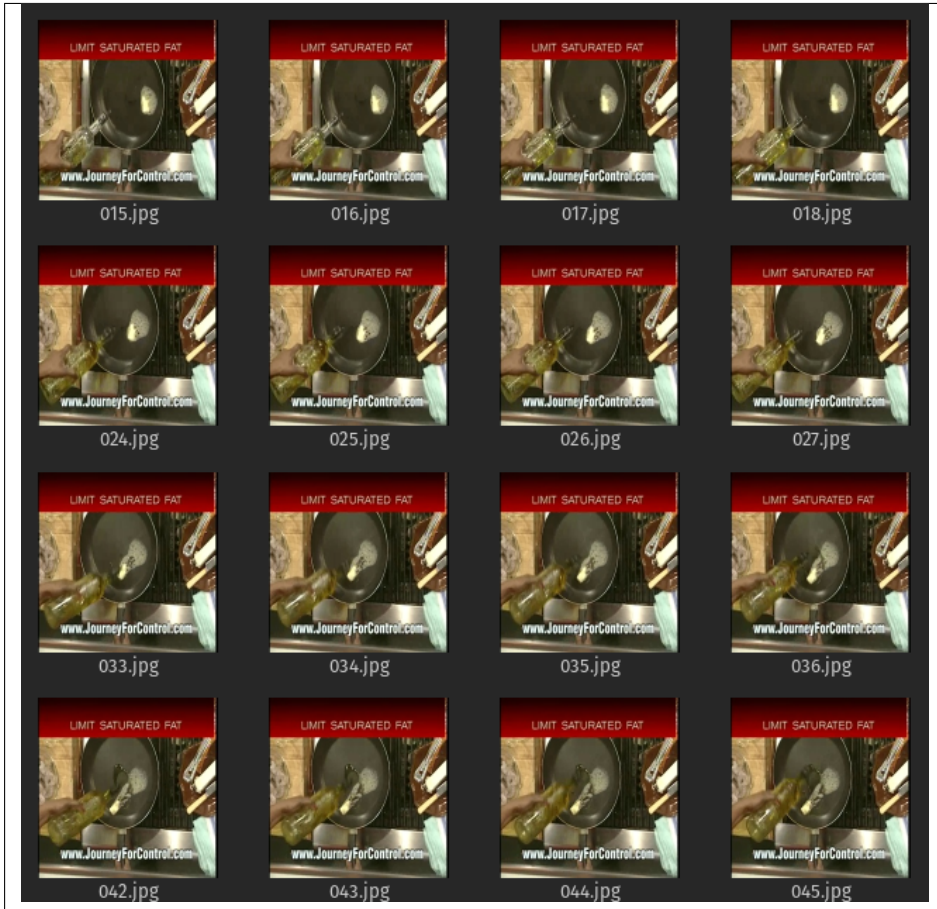
Metrix	**What(8229)**	**When(68)**	**Who(4359)**	**How(343)**	**Where(29)**
Accuracy	53.0198%	79.4118%	66.2537%	95.3353%	58.6207%
Precision	75%	91.0714%	30.6859%	97.2222%	81.2500%
Recall	0.281803	0.248397	0.139676	0.823815	0.238889
F1 Score	0.250127	0.245370	0.125046	0.872529	0.236559

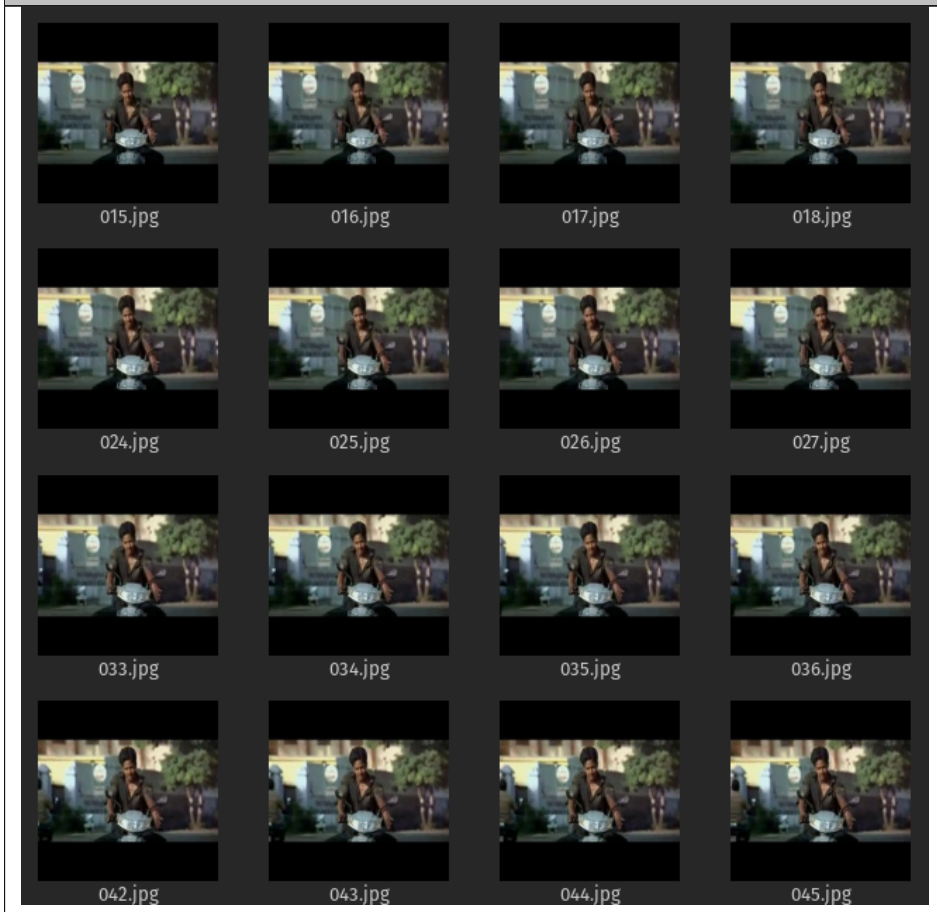
The model for English MSVD QA exhibits varied performance across different question types in the video question answering task. For **"What"** questions, which form the majority of the dataset (8,229 samples), the model achieves moderate accuracy (53.02%) and high precision (75%), but a low recall (28.18%), indicating that while it is often correct when it makes a prediction, it frequently misses relevant answers. **"When"** questions, though fewer in number (68 samples), show high accuracy (79.41%) and precision (91.07%) but also suffer from low recall (24.84%), suggesting the

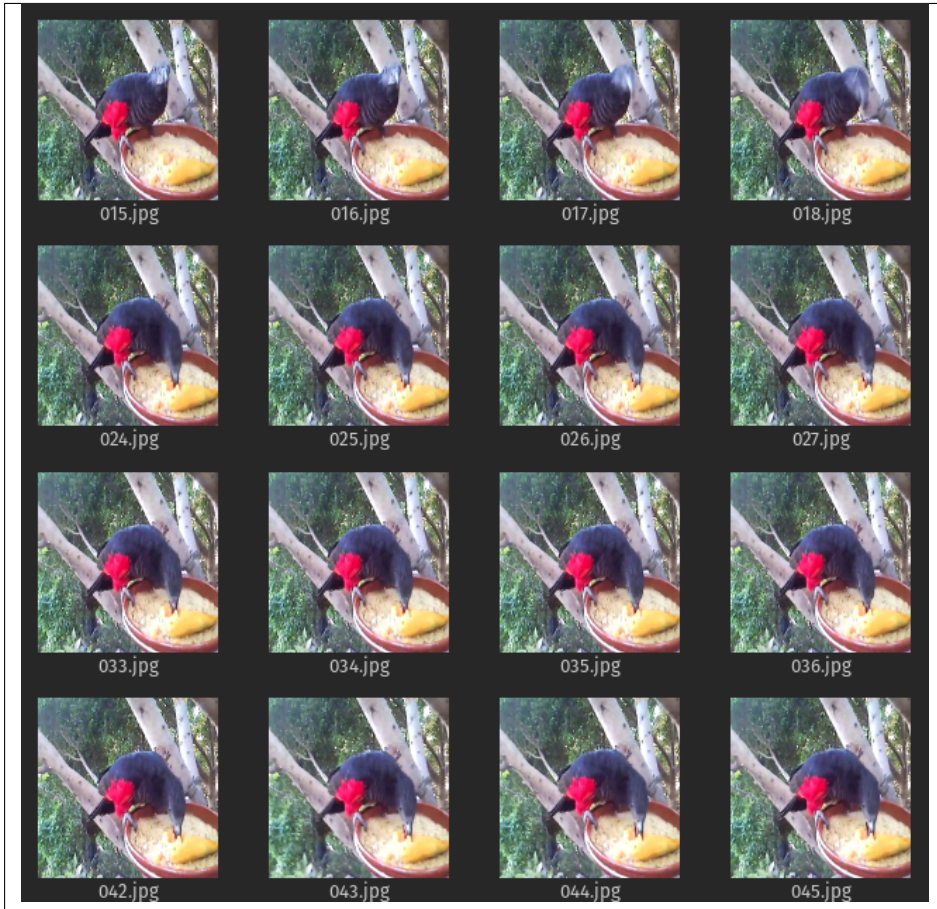
model is highly confident but selectively answers temporal queries. For **"Who"** questions (4,359 samples), the model performs with decent accuracy (66.25%), yet both precision (30.69%) and recall (13.97%) are low, pointing to many false positives and missed detections. In contrast, **"How"** questions (343 samples) are handled exceptionally well, with very high accuracy (95.34%), precision (97.22%), and recall (82.38%), showing the model's strong ability to address procedural or descriptive inquiries. Lastly, **"Where"** questions (29 samples) yield high precision (81.25%) and acceptable accuracy (58.62%), but again, the recall (23.89%) is low, indicating that while spatial answers are often correct, many are overlooked. Overall, the model favors precision over recall, especially in less frequent question types.

Next, let us examine how the best models perform in both amharic and english answering questions based on the visual content. As illustrated in Fig 6.14 and 6.15 will show 2 samples for each,the first row displays the input image, the second row shows the question provided, and the third row shows predicted answer and the fourth row presents the ground truth answer.

Video Frames	Model	Question	Predicte Answer	Actual Answer
	[am]BERT-BCMA	**ልጆች ምን እያደረጉ ነው?**	**ዳንስ**	**ዳንስ**

	[am]BERT-BCMA	**አንዲት ሴት ወደ ፓን ውስጥ ምግብ ማብሰል ምን አደረገች?**	**ቅቤ**	**ቅቤ**

Video Frames	Model	Question	Predicte Answer	Actual Answer
	[en]BERT-BCMA	who is riding a motorized bicycle in a town?	man	man

	[en]BERT-BCMA	what did the bird eat the food in the tree from?	bowl	bowl

The [am]BERT-BCMA model produces the most accurate and semantically consistent answers than the other model. It effectively fuses question embeddings with key video features across time. This leads to better attention over objects and actions, reducing answer ambiguity and demonstrating the strength of the Bidirectional Cross Modal Attention approach in Amharic Video Question Answering.

For English MSVD QA the [en]BERT-BCMA model improves by introducing bidirectional attention between the textual and visual modalities. This allows the model to better capture context across video frames and align it with the question, yielding significantly improved performance. The model shows better focus on relevant visual regions and actions that are directly tied to the question.

6.2.3 Results Discussion on video question answering

The Bidirectional Cross Modal Attention model is compared with several state of the art methods such as (Yu et al. 2024),(Tang et al. 2024),(S. Ye et al. 2023), (Yin et al. 2023),(Cheng et al. 2023), and (Peng et al. 2023) attention based mechanisms. These baseline approaches have been widely used in English-language VQA tasks and demonstrate strong performance in aligning video features with textual questions. However, the Bidirectional Cross Modal Attention model is specifically designed to handle Amharic language video question answering by incorporating multilingual understanding through BERT and CLIP, combined with enhanced Cross modal attention fusion. Unlike previous models that primarily focus on English MSVD QAs, our approach integrates both object level and global video context with Amharic question embeddings, leading to improved semantic alignment and answer accuracy. Let us now explore the qualitative results in detail.

6.2.4 Quantitative Results

Several experiments were conducted to arrive at the final Bidirectional Cross Modal Attention Video Question Answering (ViQA) model. Initial experiments on the Amharic dataset explored Cross modal attention mechanisms with different text embedding. The [am]MCLIP=CMA and [am]BERT-CMA models with CLIP and BERT respectively, achieving 31.608% and 32.873% accuracy. These models provided basic multi modal alignment but lacked the granularity needed for more precise answer classification.

The [am]BERT-BCMA model introduced token level Cross modal attention using BERT, enabling finer alignment between individual words in the question and relevant visual features in the video. While its accuracy remained at 48.208%

On the English MSVD QA, stronger performance was observed. The [en]BERT-BCMA model, which used token level bidirectional Cross modal attention with CLIP features, achieved 58.712% accuracy. The highest performing model, [en]BERT-BCMA, applied bidirectional token level attention using BERT and it is indicating the benefit of rich, two way interactions between language and vision features. The results show that leveraging token level bidirectional Cross modal attention enhances the semantic fusion of visual and textual modalities, leading to improved VQA performance — especially evident in both Amharic and English settings.

Moreover, the [en]BERT-BCMA model, which incorporates bidirectional Cross modal attention, further improves the accuracy to 58.712%. This represents a substantial gain over previous state of the art methods, demonstrating that the bidirectional attention mechanism enables more effective fusion of visual and textual features. These improvements highlight the effectiveness of advanced attention mechanisms and transformer based text encoding in enhancing the semantic alignment between video content and questions, ultimately leading to better question answering performance on the English MSVD QA.

6.2.5 Research Question Discussion

Answer for Q1: A Video Question Answering (ViQA) system supporting the Amharic language was developed by utilizing bidirectional Cross modal attention mechanisms. These mechanisms effectively integrate visual features extracted from video frames with Amharic textual inputs, addressing the language specific challenges using multilangual text encoders using fine tuned bert-base-multilingual-cased model on Amharic language texts.

Answer for Q2: Semantic information from videos was harnessed by combining multi level visual features—such as frame level, object level, and temporal features with textual question embeddings using attention based fusion techniques. This allowed the system to focus on the most relevant parts of the video and improve contextual understanding, leading to better alignment between video content and the questions posed. The model is the fist benchmark for Amharic and improver current benchmark paper in english.

Ansewer for Q3: To improve performance in video question answering, we develop a model that integrates textual and visual features using bidirectional Cross modal attention. The question is encoded using BERT, while visual features—such as global, temporal, and object level representations—are extracted from the video. The model applies attention in both directions: the question attends to relevant visual regions, and

visual features attend to key question tokens. This bidirectional attention allows the model to capture fine grained interactions between language and vision, leading to more accurate and context aware answers.

CHAPTER SEVEN

CONCLUSION AND FUTURE WORK

7.1 Conclusion

For both high level language like english and for low level language like amharic developing Video Question Answering is challenging task. Existing approaches often rely on global video level representations or sentence level embeddings, which fail to capture fine grained information details necessary for accurate question answering. These models also typically lack robust alignment between visual content and linguistic context, especially when applied to low resource languages that suffer from a lack of pretrained resources and annotated datasets.

To address these issues, this study proposes a unified, multi modal Videl QA framework that leverages both visual and textual modalities in a novel and efficient manner. One of the key contributions is a frame selection method based on MCLIP caption similarity. Instead of treating all frames equally or uniform sampling using fixed Frame per second sampling method, the system first applies K-means clustering for all the video to reduce redundancy, and then selects the most semantically relevant frames by computing cosine similarity between MCLIP encoded frames and the video caption. This ensures that only the most informative frames are passed on to subsequent processing steps, improving efficiency and semantic focus.

In addition to better frame selection, the system introduces text guided object representation. Rather than relying solely on object classification labels from Fast RCNN, each detected object identified using Faster RCNN is semantically aligned with the question and caption context using CLIP embeddings. Objects are matched to relevant keywords from the linguistic input, creating a context aware and richly grounded visual representation. This object level grounding enhances the system's reasoning ability, particularly for complex questions that depend on understanding object roles, relationships, and attributes. One of the core innovations is the use of TimeSformer, a Transformer based architecture specifically designed for video understanding, which captures temporal dependencies across frames. This model enables the system to learn rich video level features by processing spatiotemporal information jointly, providing a more better representation of the video context necessary for accurate question answering.

To bridge the gap between textual and visual modalities, multiple class were

implemented and evaluated, including Cross Modal Attention with multilangual Contrastive Language Image Pre training and BERT base multilingual cased fine tuned Amharic then Bidirectional Cross Modal Attention with multilangual Contrastive Language Image Pre training and bert-base-multilingual-cased-finetuned-amharic. These models facilitate deeper interaction between the encoded textual question and the visual features by allowing the model to dynamically attend to relevant parts of both inputs.

Through a series of experiments, the Bidirectional Cross Modal Attention demonstrated significant improvements over baseline and existing works. On the English MSVD QA, the best model (BCMA with CLIP features) achieved an accuracy of 58.712%, outperforming earlier models that lacked fine grained object level reasoning and frame selection. In the case of Amharic, which lacks extensive resources and pretrained models, our GCMA model still achieved a strong performance of 33.427%, setting a new benchmark for Amharic VQA tasks. These results underscore the importance of combining semantic aware frame selection, object level reasoning, and bidirectional Cross modal attention in building a robust and language inclusive VQA system.

In summary, by combining temporal modeling (TimeSformer), semantic aware best frame selection (CLIP + keyword similarity), contextual object representation, and the Bidirectional Cross Modal Attention system significantly improves VQA performance across languages and modalities. This study contributes a robust and scalable solution that not only advances VQA for English but also pioneers research in low resource settings Amharic.

7.2 Future Work

While this study presents a comprehensive framework for Video Question Answering in both English and Amharic, there remains substantial room for improvement, especially for low resource languages. One major direction for future work is the fine tuning of a dedicated language model for Amharic, optimized specifically for embedding video captions, questions, and Answer. The current use of multilingual BERT and CLIP embeddings, although effective, does not fully capture the linguistic semantic richness of Amharic due to the scarcity of domain specific training data. By fine tuning a transformer based language model such as BERT on Amharic video and question datasets, the system's ability to understand and respond to Amharic queries could be significantly improved.

Additionally, Relying on translation for Amharic Video Question Answering miss some

semantically information. This limitation should be addressed in future work. While the current model performs well on the MSVD dataset, achieving broader generalization will require more datasets and task specific adaptations to improve performance beyond the current results.

REFERENCES

Ayyubi, Hammad et al. (2025). *ENTER: Event Based Interpretable Reasoning for VideoQA*. DOI: 10 . 48550 / ARXIV . 2501 . 14194. URL: https://arxiv.org/abs/2501.14194.

Hailemariam, Nebiyou Daniel, Blessed Guda, and Tsegazeab Tefferi (2025). *AmaSQuAD: A Benchmark for Amharic Extractive Question Answering*. DOI: 10.48550/ARXIV.2502.02047. URL: https://arxiv.org/abs/2502.02047.

Patel, Alkesh, Vibhav Chitalia, and Yinfei Yang (2025). *Advancing Egocentric Video Question Answering with Multimodal Large Language Models*. DOI: 10.48550/ARXIV.2504.04550. URL: https://arxiv.org/abs/2504.04550.

Song, Zijie, Zhenzhen Hu, Yixiao Ma, Jia Li, and Richang Hong (Apr. 2025). "Video Flow as Time Series: Discovering Temporal Consistency and Variability for VideoQA". In: URL: http://arxiv.org/abs/2504.05783.

Tan, Tao and Guanglu Sun (Jan. 2025). "Graph-based relational reasoning network for video question answering". In: *Machine Vision and Applications* 36 (1). ISSN: 14321769. DOI: 10.1007/s00138-024-01645-w.

Wu, Zhixuan et al. (Jan. 2025a). "VideoQA-TA: Temporal-Aware Multi-Modal Video Question Answering". In: *Proceedings of the 31st International Conference on Computational Linguistics*. Ed. by Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert. Abu Dhabi, UAE: Association for Computational Linguistics, pp. 7239–7252. URL: https://aclanthology.org/2025.coling-main.483/.

– (Jan. 2025b). "VideoQA-TA: Temporal-Aware Multi-Modal Video Question Answering". In: *Proceedings of the 31st International Conference on Computational Linguistics*. Ed. by Owen Rambow, Leo Wanner, Marianna Apidianaki, Hend Al-Khalifa, Barbara Di Eugenio, and Steven Schockaert. Abu Dhabi, UAE: Association for Computational Linguistics, pp. 7239–7252. URL: https://aclanthology.org/2025.coling-main.483/.

Fan, Yue et al. (2024). *VideoAgent: A Memory-augmented Multimodal Agent for Video Understanding*. DOI: 10.48550/ARXIV.2403.11481. URL: https://arxiv.org/abs/2403.11481.

Gao, Lishuai, Yujie Zhong, Yingsen Zeng, Haoxian Tan, Dengjie Li, and Zheng Zhao (Dec. 2024). "LinVT: Empower Your Image-level Large Language Model to Understand Videos". In: URL: http://arxiv.org/abs/2412.05185.

Guda, Bhanu Prakash Reddy, Tanmay Kulkarni, Adithya Sampath, and Swarnashree Mysore Sathyendra (July 2024). "Causal Understanding For Video Question Answering20". In: URL: http://arxiv.org/abs/2407.20257.

Joshi, Raviraj et al. (2024). *Adapting Multilingual LLMs to Low-Resource Languages using Continued Pre-training and Synthetic Corpus*. DOI: 10.48550/ARXIV.2410.14815. URL: https://arxiv.org/abs/2410.14815.

Kim, Wonkyun, Changin Choi, Wonseok Lee, and Wonjong Rhee (Mar. 2024). "An Image Grid Can Be Worth a Video: Zero-shot Video Question Answering Using a VLM". In: *Findings of the Association for Computational Linguistics: EMNLP 2024*. Association for Computational Linguistics, pp. 15155–15165. DOI: 10.1109/ACCESS.2024.3517625. URL: http://arxiv.org/abs/2403.18406.

Lerner, Paul, Olivier Ferret, and Camille Guinaudeau (2024). *Cross-modal Retrieval for Knowledge-based Visual Question Answering*. DOI: 10.48550/ARXIV.2401.05736. URL: https://arxiv.org/abs/2401.05736.

Tang, Jiahao et al. (2024). "Spatio-temporal graph Convolution Transformer for Video Question Answering". In: *IEEE Access* 12, pp. 131664–131680. ISSN: 21693536. DOI: 10.1109/ACCESS.2024.3445636. URL: http://dx.doi.org/10.1109/ACCESS.2024.3445636.

Yu, Ting, Kunhao Fu, Jian Zhang, Qingming Huang, and Jun Yu (Oct. 2024). "Multi-granularity Contrastive Cross-modal Collaborative Generation for End-to-End Long-term Video Question Answeringn". In: DOI: 10.1109/TIP.2024.3390984. URL: http://arxiv.org/abs/2410.09379%20http://dx.doi.org/10.1109/TIP.2024.3390984.

Cheng, Yi, Hehe Fan, Dongyun Lin, Ying Sun, Mohan Kankanhalli, and Joo-Hwee Lim (July 2023). "Keyword-Aware Relative Spatio-Temporal Graph Networks for Video Question Answering". In: URL: http://arxiv.org/abs/2307.13250.

Li, KunChang et al. (2023). *VideoChat: Chat-Centric Video Understanding*. DOI: 10.48550/ARXIV.2305.06355. URL: https://arxiv.org/abs/2305.06355.

Maaz, Muhammad, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan (2023). *Video-ChatGPT: Towards Detailed Video Understanding via Large Vision and Language Models*. DOI: 10.48550/ARXIV.2306.05424. URL: https://arxiv.org/abs/2306.05424.

Peng, Min, Xiaohu Shao, Yu Shi, and Xiangdong Zhou (Dec. 2023). "Hierarchical Synergy-Enhanced Multimodal Relational Network for Video Question Answering". In: *ACM Transactions on Multimedia Computing, Communications, and Applications* 20.4, pp. 1–22. ISSN: 1551-6865. DOI: 10.1145/3630101. URL: http://dx.doi.org/10.1145/3630101.

Xiao, Junbin, Pan Zhou, Angela Yao, et al. (Nov. 2023). "Contrastive Video Question Answering via Video Graph Transformer". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.11, pp. 13265–13280. ISSN: 1939-3539. DOI: 10.1109/tpami.2023.3292266. URL: http://dx.doi.org/10.1109/TPAMI.2023.3292266.

Ye, Shuhong, Weikai Kong, Chenglin Yao, Jianfeng Ren, and Xudong Jiang (Mar. 2023). "Video Question Answering Using CLIP-Guided Visual-Text Attention". In: pp. 81–85. DOI: 10.1109/icip49359.2023.10222286. URL: http://arxiv.org/abs/2303.03131.

Yin, Chengxiang, Zhengping Che, Kun Wu, Zhiyuan Xu, Qinru Qiu, and Jian Tang (2023). *Cross-Modal Reasoning with Event Correlation for Video Question Answering*. Tech. rep. URL: www.aaai.org.

Zhu, Jun, Jiandong Jin, Zihan Yang, Xiaohao Wu, and Xiao Wang (Apr. 2023). "Learning CLIP Guided Visual-Text Fusion Transformer for Video-based Pedestrian Attribute Recognition". In: URL: http://arxiv.org/abs/2304.10091.

Gao, Difei, Luowei Zhou, Lei Ji, Linchao Zhu, Yi Yang, and Mike Zheng Shou (Dec. 2022). "MIST: Multi-modal Iterative Spatial-Temporal Transformer for Long-form Video Question Answering". In: URL: http://arxiv.org/abs/2212.09522.

Li, Shuang, Yilun Du, Joshua B. Tenenbaum, Antonio Torralba, and Igor Mordatch (Oct. 2022). "Composing Ensembles of Pre-trained Models via Iterative Consensus". In: URL: http://arxiv.org/abs/2210.11522.

Li, Yicong, Xiang Wang, Junbin Xiao, and Tat-Seng Chua (2022). *Equivariant and Invariant Grounding for Video Question Answering*. DOI: 10.48550/ARXIV.2207.12783. URL: https://arxiv.org/abs/2207.12783.

Xiao, Junbin, Pan Zhou, Tat-Seng Chua, and Shuicheng Yan (July 2022). "Video Graph Transformer for Video Question Answering". In: URL: http://arxiv.org/abs/2207.05342.

Yang, Antoine, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid (2022). *Zero-Shot Video Question Answering via Frozen Bidirectional Language Models*. DOI: 10.48550/ARXIV.2206.08155. URL: https://arxiv.org/abs/2206.08155.

Bertasius, Gedas, Heng Wang, and Lorenzo Torresani (Feb. 2021). "Is Space-Time Attention All You Need for Video Understanding?" In: URL: http://arxiv.org/abs/2102.05095.

Khurana, Khushboo and Umesh Deshpande (2021). "Video Question-Answering Techniques, Benchmark Datasets and Evaluation Metrics Leveraging Video Captioning: A Comprehensive Survey". In: *IEEE Access* 9, pp. 43799–43823. ISSN: 21693536. DOI: 10.1109/ACCESS.2021.3058248.

Patel, Devshree, Ratnam Parikh, and Yesha Shastri (Jan. 2021). "Recent Advances in Video Question Answering: A Review of Datasets and Methods". In: DOI: 10.1007/978-3-030-68790-8_27. URL: http://arxiv.org/abs/2101.05954%20http://dx.doi.org/10.1007/978-3-030-68790-8_27.

Radford, Alec et al. (Feb. 2021). "Learning Transferable Visual Models From Natural Language Supervision". In: URL: http://arxiv.org/abs/2103.00020.

Conneau, Alexis et al. (July 2020). "Unsupervised Cross-lingual Representation Learning at Scale". In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Ed. by Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault. Online: Association for Computational Linguistics, pp. 8440–8451. DOI: 10.18653/v1/2020.acl-main.747. URL: https://aclanthology.org/2020.acl-main.747/.

Huang, Deng, Peihao Chen, Runhao Zeng, Qing Du, Mingkui Tan, and Chuang Gan (Aug. 2020). "Location-aware Graph Convolutional Networks for Video Question Answering". In: URL: http://arxiv.org/abs/2008.09105.

Yang, Antoine, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid (2020). *Just Ask: Learning to Answer Questions from Millions of Narrated Videos*. DOI: 10.48550/ARXIV.2012.00451. URL: https://arxiv.org/abs/2012.00451.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (June 2019). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Ed. by Jill Burstein, Christy Doran, and Thamar Solorio. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. DOI: 10.18653/v1/N19-1423. URL: https://aclanthology.org/N19-1423/.

Lu, Jiasen, Dhruv Batra, Devi Parikh, and Stefan Lee (Aug. 2019). "ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks". In: URL: http://arxiv.org/abs/1908.02265.

Su, Weijie et al. (2019). "VL-BERT: Pre-training of Generic Visual-Linguistic Representations". In: DOI: 10.48550/arXiv.1908.08530. URL: https://github.com/jackroos/VL-BERT..

Tsai, Yao-Hung Hubert, Shaojie Bai, Makoto Yamada, Louis-Philippe Morency, and Ruslan Salakhutdinov (Aug. 2019). "Transformer Dissection: A Unified Understanding of Transformer's Attention via the Lens of Kernel". In: URL: http://arxiv.org/abs/1908.11775.

Zhang, Canlin, Daniel Biś, Xiuwen Liu, and Zhe He (Dec. 2019). "Biomedical word sense disambiguation with bidirectional long short-term memory and attention-based neural networks". In: *BMC Bioinformatics* 20.S16. ISSN: 1471-2105. DOI: 10.1186/s12859-019-3079-8. URL: http://dx.doi.org/10.1186/s12859-019-3079-8.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova (Oct. 2018). "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding". In: URL: http://arxiv.org/abs/1810.04805.

Gidaris, Spyros and Nikos Komodakis (Apr. 2018). "Dynamic Few-Shot Visual Learning without Forgetting". In: URL: http://arxiv.org/abs/1804.09458.

Lei, Jie, Licheng Yu, Mohit Bansal, and Tamara L. Berg (2018). *TVQA: Localized, Compositional Video Question Answering*. DOI: 10.48550/ARXIV.1809.01696. URL: https://arxiv.org/abs/1809.01696.

Jang, Yunseok, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim (Apr. 2017). "TGIF-QA: Toward Spatio-Temporal Reasoning in Visual Question Answering". In: URL: http://arxiv.org/abs/1704.04497.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton (May 2017). "ImageNet classification with deep convolutional neural networks". In: *Communications of the ACM* 60.6, pp. 84–90. ISSN: 1557-7317. DOI: 10.1145/3065386. URL: http://dx.doi.org/10.1145/3065386.

Tran, Du, Jamie Ray, Zheng Shou, Shih-Fu Chang, and Manohar Paluri (Aug. 2017). "ConvNet Architecture Search for Spatiotemporal Feature Learning". In: URL: http://arxiv.org/abs/1708.05038.

Vaswani, Ashish et al. (June 2017). "Attention Is All You Need". In: URL: http://arxiv.org/abs/1706.03762.

Ye, Yunan, Zhou Zhao, Yimeng Li, Long Chen, Jun Xiao, and Yueting Zhuang (2017). "Video Question Answering via Attribute-Augmented Attention Network Learning". In: DOI: 10.48550/ARXIV.1707.06355. URL: https://arxiv.org/abs/1707.06355.

Chalk, Matthew, Olivier Marre, and Gasper Tkacik (May 2016). "Relevant sparse codes with variational information bottleneck". In: URL: http://arxiv.org/abs/1605.07332.

Tapaswi, Makarand, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun, and Sanja Fidler (Dec. 2016). "MovieQA: Understanding stories in movies through question-answering". In: 2016-December, pp. 4631–4640. ISSN: 10636919. DOI: 10.1109/CVPR.2016.501. URL: http://arxiv.org/abs/1512.02902.

Wu, Qi, Damien Teney, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel (2016). *Visual Question Answering: A Survey of Methods and Datasets*. DOI: 10.48550/ARXIV.1607.05910. URL: https://arxiv.org/abs/1607.05910.

Agrawal, Aishwarya et al. (May 2015). "VQA: Visual Question Answering". In: URL: http://arxiv.org/abs/1505.00468.

Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun (2015). *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*. DOI: 10.48550/ARXIV.1506.01497. URL: https://arxiv.org/abs/1506.01497.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio (Sept. 2014). "Neural Machine Translation by Jointly Learning to Align and Translate". In: URL: http://arxiv.org/abs/1409.0473.

Karpathy, Andrej and Li Fei-Fei (Dec. 2014). "Deep Visual-Semantic Alignments for Generating Image Descriptions". In: URL: http://arxiv.org/abs/1412.2306.

Vinyals, Oriol, Alexander Toshev, Samy Bengio, and Dumitru Erhan (Nov. 2014). "Show and Tell: A Neural Image Caption Generator". In: URL: http://arxiv.org/abs/1411.4555.

Bengio, Y., A. Courville, and P. Vincent (Aug. 2013). "Representation Learning: A Review and New Perspectives". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.8, pp. 1798–1828. ISSN: 2160-9292. DOI: 10.1109/tpami.2013.50. URL: http://dx.doi.org/10.1109/TPAMI.2013.50.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean (Jan. 2013). "Efficient Estimation of Word Representations in Vector Space". In: URL: http://arxiv.org/abs/1301.3781.

Hinton, Geoffrey et al. (2012). "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups". In: *IEEE Signal Processing Magazine* 29 (6), pp. 82–97. ISSN: 10535888. DOI: 10.1109/MSP.2012.2205597.

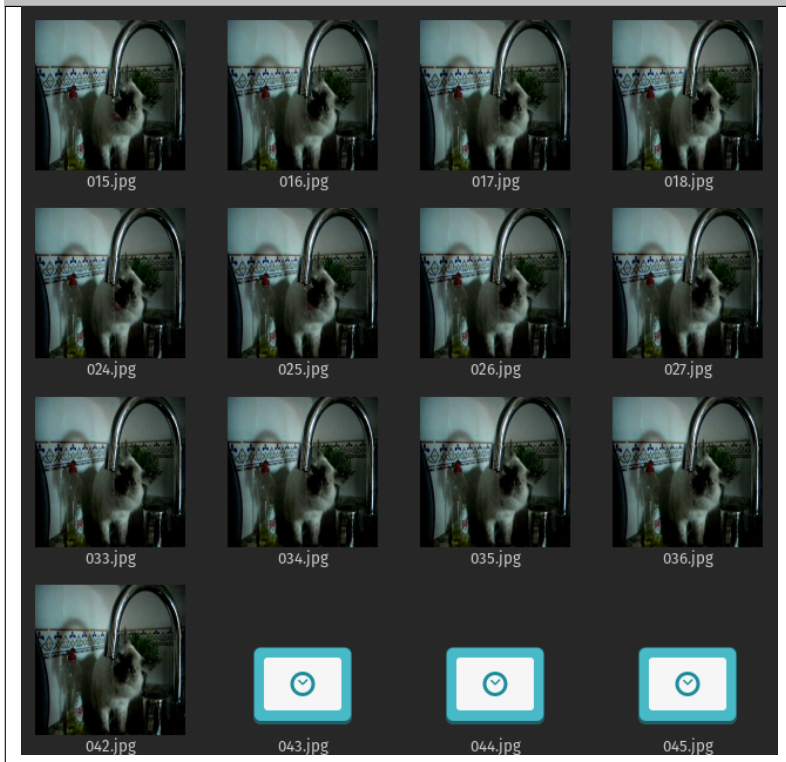
APPENDIXES

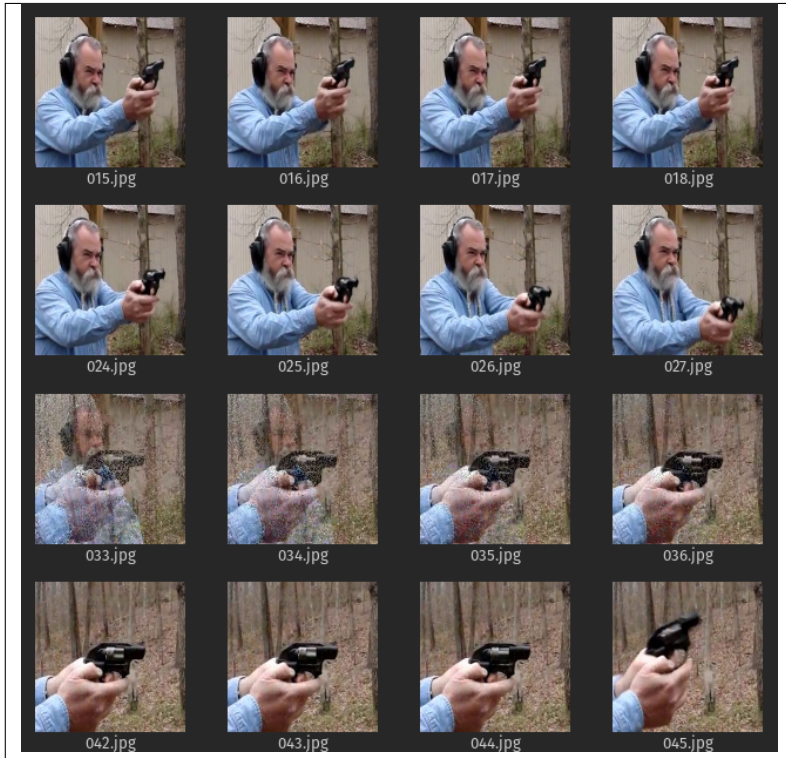
APPENDIX A: ABLATION STUDY

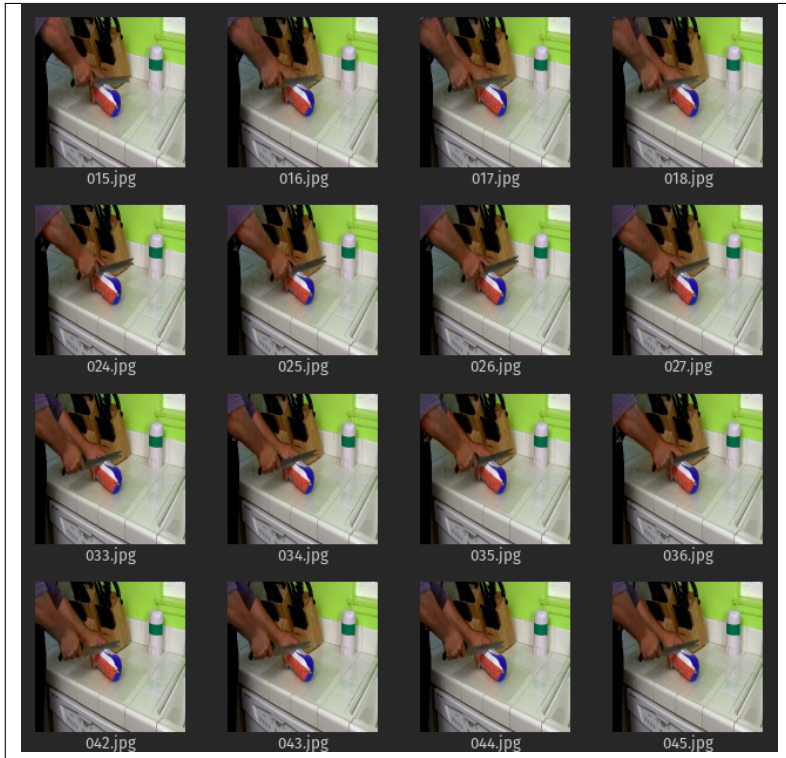
This ablation study was conducted to evaluate the effectiveness of various design choices within the Bidirectional Cross Modal Attention multi modal Video Question Answering (ViQA) system, particularly focusing on embedding strategies, Cross modal attention mechanisms, and semantic frame selection. The study aimed to address two primary questions: which combination of visual and textual embedding techniques along with attention strategies yields the highest question answering accuracy, and what feature combinations most effectively align the visual and textual modalities to enhance reasoning performance. Several experiments were conducted, beginning with baseline models whihc lacks sematic object alignment and sematic frames selection. While this setup provided a solid foundation, it lacked object level semantics and frame wise alignment, resulting in moderate accuracy. Building on this, an enhanced model incorporated CLIP based object level features and Fast RCNN detected object representations, semantically grounded using CLIP similarity to keywords derived from the question and video captions. This significantly improved performance, demonstrating the advantage of leveraging pretrained vision language models for object grounding.

Further experiments compared BERT vs. CLIP as the textual encoder: BERT performed well on syntactic question understanding, while CLIP, trained on large scale image text pairs, offered superior multi modal alignment when used in cross attention modules. The introduction of a bidirectional Cross modal attention mechanism (text to vision and vision to text) further improved the system by allowing fine grained interactions between question representations and visual features. Additional experiments explored the impact of the frame selection strategy. Using a CLIP caption similarity scoring method after k-means clustering allowed the model to select semantically informative frames, reducing noise and improving reasoning quality. Overall, the ablation study confirms that the combination of CLIP based embeddings, object level semantic grounding, and bidirectional attention mechanisms provides a strong advantage in multi modal reasoning. It also shows that Cross modal alignment and frame level relevance are key to VQA accuracy, especially for low resource languages like Amharic. These findings establish a clear guideline for optimizing VQA models through careful architectural and embedding choices.

APPENDIX B: AMHARIC VIDEO QUESTION ANSWERING RESULT ON DIFFERENT MODELS

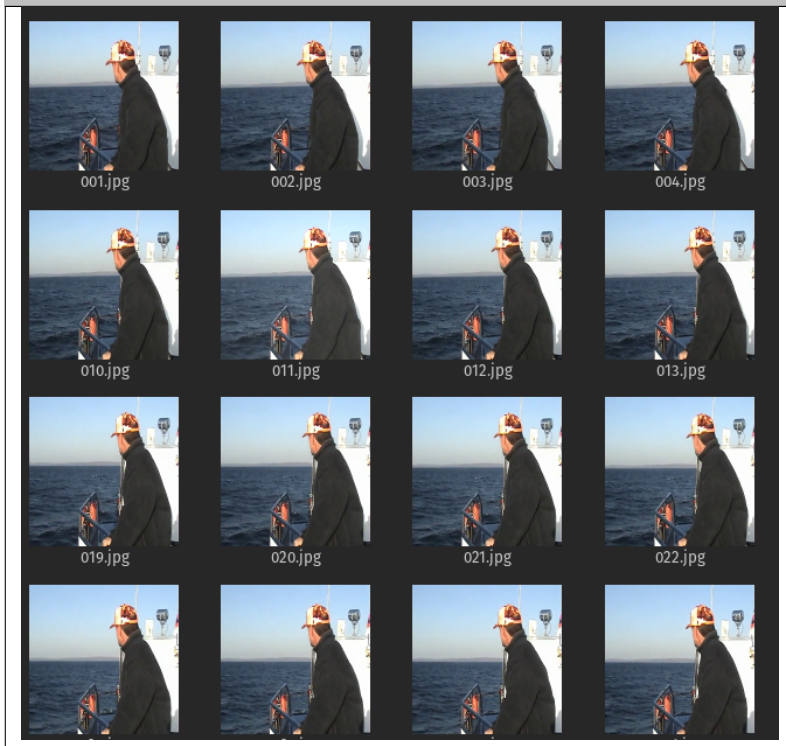
Video Frames	Model	Question	Predicte Answer	Actual Answer
	CLIP-CMA	ከውኃ ላይ ውሃ የሚጠጣው ነገር ምንድን ነው?	ድመት	ውሻ

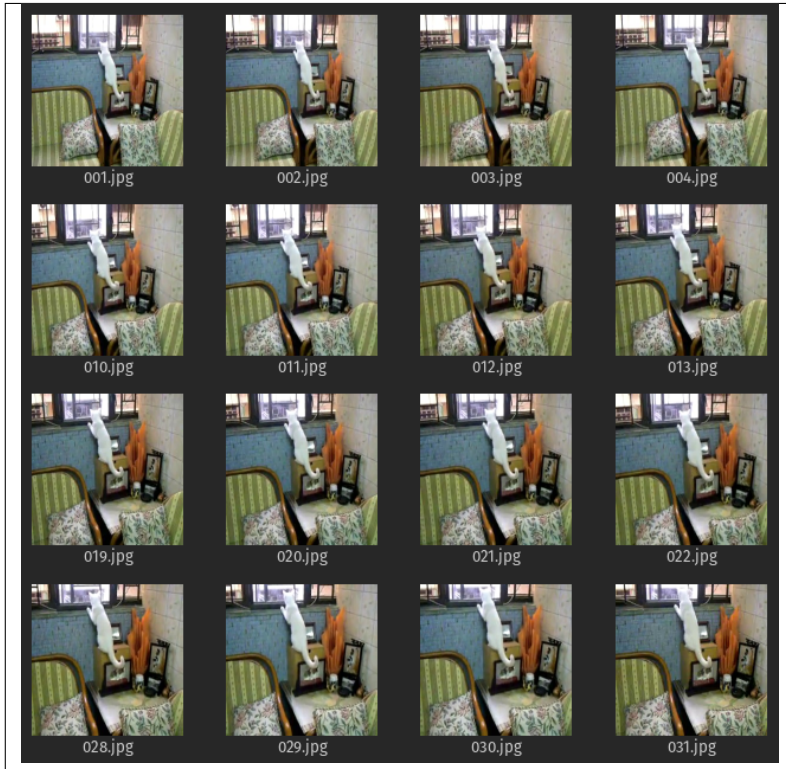
	BERT-CMA	**አንድ አዛውንት ሲነካው ምንድነው?**	**ተኩስ**	**ጠመንጃ**

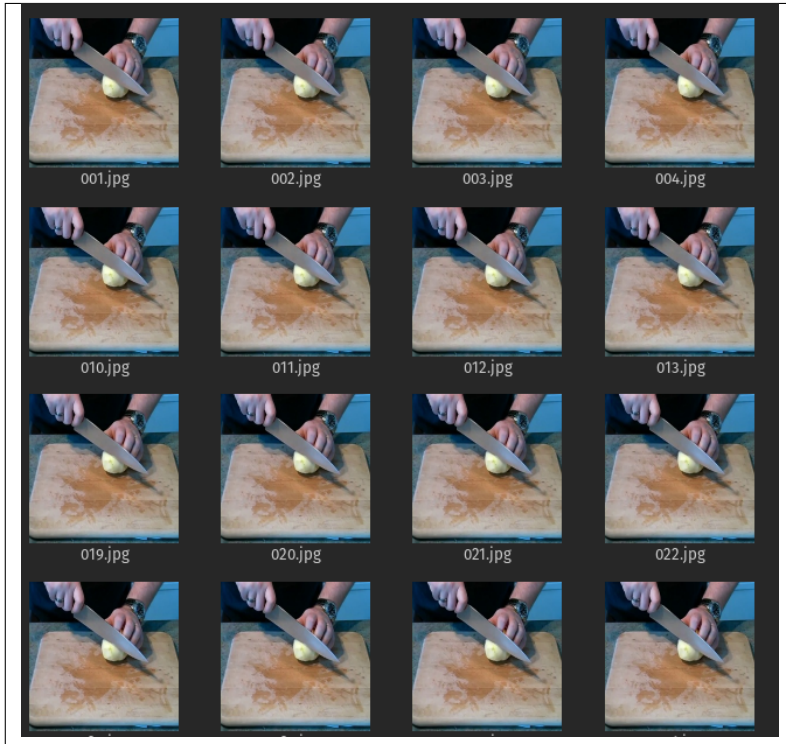
	MCLIP-BCMA	**በክብት የበሬ ሥጋ ጥቅል ውስጥ የሚቆረጥ ማነው?**	**ሰው**	**ሰው**

	BERT-BCMA	**በበረዶ ውስጥ የምትጋልብላት ሴት ምንድነው ?**	**ፈረስ**	**ፈረስ**

APPENDIX C: ENGLISH VIDEO QUESTION ANSWERING RESULT ON DIFFERENT MODELS

Video Frames	Model	Question	Predicte Answer	Actual Answer
	BERT-BCMA	what did the man look out at from his boat?	ocean	ocean

	MCLIP-CMA	what is a white cat perched on a small wooden cabinet doing?	window	look

	BERT-CMA	who is cutting potato?	man	someone

APPENDIX D: SAMPLE CODE

Code for best 16 frame selection

```
def load_annotations(annotation_file):
    annotations = {}
    with open(annotation_file, "r", encoding="utf-8") as f:
        for line in f:
            line = line.strip()
            if not line:
                continue
            parts = line.split(maxsplit=1)
            if len(parts) < 2:
                continue
            video_id, caption = parts
            if video_id in annotations:
                annotations[video_id].append(caption)
            else:
                annotations[video_id] = [caption]
    return annotations

def select_caption(captions, max_words=50):
    for caption in captions:
        try:
            _ = clip.tokenize([caption])
            return caption
        except RuntimeError:
            continue
    shortest = min(captions, key=lambda x: len(x.split()))
    truncated = " ".join(shortest.split()[:max_words])
    return truncated

def load_frames_from_folder(folder_path):
    frame_files = sorted([
        os.path.join(folder_path, f)
        for f in os.listdir(folder_path)
        if f.lower().endswith(('.jpg', '.jpeg', '.png'))
    ])
    frames = []
    for file in frame_files:
        frame = cv2.imread(file)
        if frame is not None:
            frames.append(frame)
    return frames

def compute_frame_features(frames):
    features = []
    for frame in frames:
```

```
        frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
        pil_img = Image.fromarray(frame_rgb)
        image_input = preprocess(pil_img).unsqueeze(0).to(device)
        with torch.no_grad():
            feature = model.encode_image(image_input)
        features.append(feature.cpu().numpy().squeeze())
    return np.array(features)

def compute_caption_embedding(caption):
    text_input = clip.tokenize([caption]).to(device)
    with torch.no_grad():
        text_features = model.encode_text(text_input)
    return text_features.cpu().numpy().squeeze()

def cosine_similarity(a, b):
    return np.dot(a, b) / (np.linalg.norm(a) * np.linalg.norm(b))

def select_representative_frames(frames, caption, num_selected=16):
    if len(frames) == 0:
        return []
    frame_features = compute_frame_features(frames)
    caption_embedding = compute_caption_embedding(caption)

    if len(frames) < num_selected:
        return frames

    kmeans = KMeans(
    n_clusters=num_selected, random_state=42, n_init=10
    )
    labels = kmeans.fit_predict(frame_features)

    selected_frames = []
    for i in range(num_selected):
        idxs = np.where(labels == i)[0]
        if len(idxs) == 0:
            continue
        sims = [cosine_similarity(
        frame_features[j], caption_embedding) for j in idxs]
        best_idx = idxs[np.argmax(sims)]
        selected_frames.append(frames[best_idx])
    return selected_frames

if __name__ == "__main__":
    video_base_folder = "/kaggle/input/msvd-video-caption/train"
    annotation_f="/kaggle/input/msvd-video-caption/annotations.txt"
    output_base_folder = "/kaggle/working/train_16"

```

```
    if not os.path.exists(output_base_folder):
        os.makedirs(output_base_folder)

    annotations = load_annotations(annotation_file)

    video_ids = os.listdir(video_base_folder)
    for video_id in tqdm(video_ids, desc="Processing videos"):
        video_folder_path = os.path.join(
        video_base_folder, video_id
        )
        if not os.path.isdir(video_folder_path):
            continue
        captions = annotations.get(video_id, None)
        if captions is None:
            continue

        # Select one caption that is not too long
        caption = select_caption(captions, max_words=50)

        frames = load_frames_from_folder(video_folder_path)
        if len(frames) == 0:
            continue

        selected_frames = select_representative_frames(
        frames, caption, num_selected=64
        )
        if len(selected_frames) == 0:
            continue

        video_output_folder = os.path.join(
        output_base_folder, video_id
        )
        if not os.path.exists(video_output_folder):
            os.makedirs(video_output_folder)
        for idx, frame in enumerate(selected_frames):
            cv2.imwrite(
            os.path.join(
            video_output_folder, f"selected_frame_{idx:03d}.jpg"
            ), frame
            )
```

Listing 7.1: Snippet code for Frame selection

Code for Object labeling

```
# Load or create results
if os.path.exists(output_path):
    with open(output_path, "rb") as f:
        results = pickle.load(f)
```

```
else:
    results = {}


video_folders = os.listdir(video_frames_root)

for video_id in tqdm(video_folders, desc="Processing videos"):
    vid = video_id.strip()
    if vid not in qa_dict or vid in results:
        continue

    words = [w.strip() for w in qa_dict[vid].split() if w.strip()]
    with torch.no_grad():
        word_feats = model_mul.forward(words, tokenizer).to(device)
        word_feats /= word_feats.norm(dim=-1, keepdim=True)

    folder_path = os.path.join(video_frames_root, vid)
    if not os.path.exists(folder_path):
        continue

    vdata = []
    for idx, frame in enumerate(sorted(os.listdir(folder_path))):
        img_path = os.path.join(folder_path, frame)
        if not os.path.exists(img_path):
            continue

        img = Image.open(img_path).convert("RGB")
        inp = det_transform(img).to(device)
        with torch.no_grad():
            pred = fastrcnn_model([inp])[0]

        boxes = pred["boxes"].cpu().numpy()
        scores = pred["scores"].cpu().numpy()
        boxes = boxes[scores >= 0.5]

        for b in boxes:
            x1, y1, x2, y2 = map(int, b)
            crop = img.crop((x1, y1, x2, y2))
            clip_inp = preprocess(crop).unsqueeze(0).to(device)
            with torch.no_grad():
                feat = clip_model.encode_image(clip_inp)
                feat /= feat.norm(dim=-1, keepdim=True)
                sim = (100.0 * feat @ word_feats.T).softmax(dim=-1)
                best = sim.argmax().item()

            vdata.append({
                "frame_id": idx,
```

```
                "box": [x1, y1, x2, y2],
                "word": words[best],
                "clip_feature": feat.cpu().numpy(),
                "confidence": sim[0, best].item()
            })

    if vdata:
        results[vid] = vdata
        with open(output_path, "wb") as f:
            pickle.dump(results, f)
```

Listing 7.2: Snippet code for Object labeling

Code for Bidirectional Cross Modal Attention

```
# Model Class with Bidirectional Cross modal Attention
class BidirectionalCrossModalAttention(nn.Module):
    def __init__(self, bert_model_name, num_answers, hidden_dim=512,
                 dropout_rate=0.1, num_heads=8):
        super().__init__()
        self.bert = BertModel.from_pretrained(bert_model_name)
        self.visual_proj = nn.ModuleDict({
            'cls': nn.Sequential(nn.Linear(768, 768), nn.BatchNorm1d(768)),
            'temp': nn.Sequential(nn.Linear(768, 768), nn.BatchNorm1d(768)),
            'obj': nn.Sequential(nn.Linear(1024, 768), nn.BatchNorm1d(768))
        })
        self.text2vis_attn = nn.MultiheadAttention(
        embed_dim=768, num_heads=num_heads, dropout=dropout_rate
        )
        self.vis2text_attn = nn.MultiheadAttention(
        embed_dim=768, num_heads=num_heads, dropout=dropout_rate
        )
        self.classifier = nn.Sequential(
            nn.Linear(768 * 2, hidden_dim),
            nn.ReLU(),
            nn.Dropout(dropout_rate),
            nn.Linear(hidden_dim, num_answers)
        )
        self.apply(self._init_weights)

    def _init_weights(self, module):
        if isinstance(module, nn.Linear):
            nn.init.kaiming_normal_(
            module.weight, mode='fan_out', nonlinearity='relu'
            )
            if module.bias is not None:
                nn.init.constant_(module.bias, 0)
        elif isinstance(module, nn.BatchNorm1d):
            nn.init.constant_(module.weight, 1)
```

```
            nn.init.constant_(module.bias, 0)

    def forward(
    self, input_ids, attention_mask, cls_feat, temp_feat, obj_feat
    ):
        # Text Encoding
        text_out = self.bert(
        input_ids=input_ids, attention_mask=attention_mask
        )
        text_feat = text_out.last_hidden_state.transpose(0, 1)

        # Visual Encoding
        visual_feats = torch.stack([
            self.visual_proj['cls'](cls_feat),
            self.visual_proj['temp'](temp_feat),
            self.visual_proj['obj'](obj_feat)
        ], dim=1)  # [bs, 3, 768]
        visual_feat = visual_feats.transpose(0, 1)  # [3, bs, 768]

        # Text to Visual Attention
        t2v_out, _ = self.text2vis_attn(
        query=text_feat, key=visual_feat, value=visual_feat
        )
        t2v_vec = t2v_out.mean(dim=0)  # [bs, 768]

        # visual to text Attention
        v2t_out, _ = self.vis2text_attn(
        query=visual_feat, key=text_feat, value=text_feat
        )
        v2t_vec = v2t_out.mean(dim=0)  # [bs, 768]

        fused = torch.cat([t2v_vec, v2t_vec], dim=1)  # [bs, 1536]
        logits = self.classifier(fused)
        return logits
```

Listing 7.3: Snippet code for Bidireactional Cross Modal Attention

```
class VideoQADataset(Dataset):
    def __init__(
    self, qa_csv, features_pkl, obj_feat, tokenizer_name, max_q_len=32
    ):
        qa = pd.read_csv(qa_csv)
        with open(features_pkl, "rb") as f:
            self.feat_data = pickle.load(f)
        with open(obj_feat, "rb") as f:
            self.obj_data = pickle.load(f)

        # Thorough filtering of valid rows
```

```
        valid_rows = []
        for _, row in qa.iterrows():
            vid = row["video_id"]
            if (vid in self.feat_data and self.feat_data[vid] is not None and
                "cls" in self.feat_data[vid]
                and self.feat_data[vid]["cls"] is not None and
                "temporal" in self.feat_data[vid]
                and self.feat_data[vid]["temporal"] is not None and
                vid in self.obj_data and self.obj_data[vid] is not None):
                valid_rows.append(row)

        self.qa = pd.DataFrame(valid_rows).reset_index(drop=True)
        self.tokenizer = BertTokenizer.from_pretrained(tokenizer_name)
        self.max_q_len = max_q_len
        self.ans2idx = {
        ans: idx for idx, ans in enumerate(
        self.qa["answer"].unique())
        }
        self.idx2ans = {idx: ans for ans, idx in self.ans2idx.items()}

    def __len__(self):
        return len(self.qa)

    def __getitem__(self, idx):
        row = self.qa.iloc[idx]
        vid, q, a = row["video_id"], row["question"], row["answer"]

        # Tokenize question
        toks = self.tokenizer(q, padding="max_length", truncation=True,
                              max_length=self.max_q_len, return_tensors="pt")

        # Ensure input_ids is torch.long
        input_ids = toks["input_ids"].to(dtype=torch.long).squeeze(0)
        attention_mask = toks["attention_mask"].to(dtype=torch.long).squeeze(0)

        # Get features
        feat = self.feat_data[vid]
        cls_feat = torch.tensor(
        feat["cls"], dtype=torch.float).squeeze(0)
        temp_feat = torch.tensor(
        feat["temporal"], dtype=torch.float).squeeze(0)
        obj_feat = torch.tensor(
        self.obj_data[vid], dtype=torch.float).squeeze(0)

        return (
            idx,
            input_ids,
```

```
            attention_mask,
            cls_feat,
            temp_feat,
            obj_feat,
            self.ans2idx[a]
        )
```

Listing 7.4: Snippet code for Feature set up

Dr. Mesfin Abebe

Multimodal Understanding Amharic Video Question Answering using Bidirectional Cross Modal Attention

Thesis proposal

Thesis proposal (MSc students)

Adama Science and Technology University

Document Details

Submission ID
trn:oid:::1:3418573369

Submission Date
Nov 20, 2025, 1:03 PM GMT

Download Date
Nov 20, 2025, 1:11 PM GMT

File Name
odal_Understanding_for_Amharic_Video_QuestionAnswering_using.PDF

File Size
4.5 MB

90 Pages

23,056 Words

130,975 Characters

16% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography

Exclusions

- 1 Excluded Source

Match Groups

287 Not Cited or Quoted 14%
Matches with neither in-text citation nor quotation marks

47 Missing Quotations 2%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

11% Internet sources
12% Publications
5% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.