Enhancing Bangla Video Comprehension through Multimodal Feature Integration and Attention-Based Encoder-Decoder Captioning Models for Single-Action Videos

by

Saurav Das
20101100
Shammo Biswas
20101359
Taimoor Fahim
23241093
M.A.B.Siddique Sanjan
19201068
Tasnia Alam Tarannum
20301179

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
Bachelor of Science in Computer Science

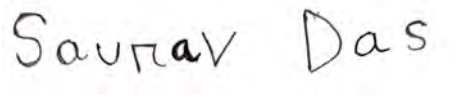
Department of Computer Science and Engineering
Brac University
May 2024

Declaration

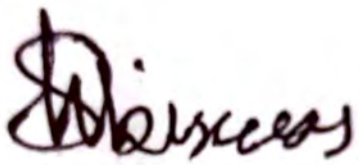
It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

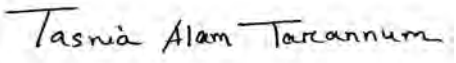
Student's Full Name & Signature:



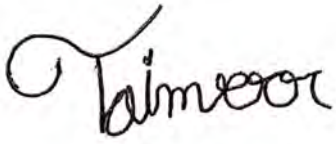
Saurav Das
20101100



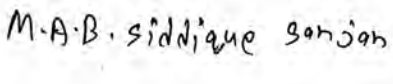
Shammo Biswas
20101359



Tasnia Alam Tarannum
20301179



Taimoor Fahim
23241093



M.A.B.Siddique Sanjan
19201068

Approval

The thesis/project titled "Enhancing Bangla Video Comprehension through Multimodal Feature Integration and Attention-Based Encoder-Decoder Captioning Models for Single-Action Videos" submitted by

1. Saurav Das(20101100)
2. Shammo Biswas(20101359)
3. Taimoor Fahim(23241093)
4. M.A.B.Siddique Sanjan(19201068)
5. Tasnia Alam Tarannum(20301179)

Of Spring, 2024 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May, 2024. **Examining**

Committee:
Supervisor:
(Member)

Dr. Md. Ashraful Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Md. Golam Rabiul Alam
Professor Thesis Coordinator
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Chair)

Md. Golam Rabiul Alam
Professor Thesis Coordinator
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, Ph.D.
Chairperson
Department of Computer Science and Engineering
Brac University

Ethics Statement

We hereby declare that this thesis is our own work and the discoveries are from our research. The sources have been appropriately acknowledged which we used in our study. Therefore, we are confirming that this thesis has not been submitted published, or presented in any other educational institution for receiving any degree.

Abstract

Video understanding and description have an important role to play in the field of computer vision and natural language processing. The capacity of automatically generating natural language descriptions for video content has many real-world applications, for example, quoting accessibility tools up to multimedia retrieval systems. Although understanding and describing video content in natural language is a challenging job, it is more so in resource-constrained languages like Bangla. This study investigates the integration of a feature fusion method and the attention-based encoder-decoder framework to improve comprehension of videos and to generate accurate captions for single-action video clips in Bangla. We propose a novel model based on multimodal fusion by combining visual features from video frames and motion information derived from optical flow. The adopted multimodal representations are then fed into an attention-based encoder-decoder architecture aiming to generate descriptive captions in the Bangla language. To facilitate our research, we collected and annotated a new dataset comprising single-action videos sourced from various online platforms. Extensive experiments are conducted on this newly created Bangla single-action videos dataset, with the models evaluated using standard metrics like BLEU, METEOR, and CIDEr. Among the models tested, including architectural variations, the GRU-Gaussian Attention model achieves the best performance, generating captions closest to the ground truth. As this is a new dataset with no previous benchmarks, the proposed approach establishes a strong baseline for Bangla video captioning, achieving a BLEU score of 0.53 and a CIDEr score of 0.492. Additionally, we analyze the attention mechanisms to interpret the learned representations, providing insights into the model's behavior and decision-making process. This work on developing solutions for under-resourced languages paves the way for enhanced video comprehension with potential applications in human-computer interaction, accessibility, and multimedia retrieval.

Keywords: Video Captioning, Video Processing, Bangla Language, Computer Vision, Natural Language Processing (NLP), Feature Fusion, Encoder-Decoder Framework, Attention Mechanisms, Multimodal Fusion, Optical Flow, GRU-Gaussian Attention Model, Dataset Annotation, BLEU, CIDEr Score.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Video captions are the automatic production of natural language sentences to summarize video content [1]. What makes the task even more complicated is the problem of interpreting activities within a video, whereby all elements—be they humans, animals, or objects—are correctly identified and their interactions are understood. Machines aren't like the human eye and brain because they process information visually much differently. We just naturally see and understand what we see every day. On the other hand, computers do not see or experience information like humans. Teaching a computer to identify and understand the relationships between objects, humans, and animals within images or videos, and how the elements interact, is difficult. [2] Video captions are challenging because they require integrating visual and sentence context features and producing text that maintains the flow and meaning of the sentence [3].

The complexity of video captioning rises as not only this is a domain of computer vision as what the model can see but also what the model can read and give output which is the domain for Natural Language processing. Only correct and accurate sentences are needed for the perfect descriptions. Any change in the flow of the sentence can change the actual meaning of the sentence.[4] Video Captioning is challenging because it requires consideration of attention mechanisms, selection of salient features, and ensuring semantic consistency between sentence descriptions and video content. [5] Video content is complex and diverse, with varying vocabularies, expression styles, and multiple pieces of information making it difficult to caption.

This thesis explores the intricate domain of video captioning in the context of the Bangla language, a linguistic area that has received relatively less attention in the area of research in computational linguistics. Video captioning is an important technology meant for generating textual descriptions accompanying video content. Such captions enhance the accessibility of video content for the impaired and those who are non-native speakers of the language in which the video is spoken. Further, they contribute centrally in indexing content and enhancing search functionalities, hence is a critical tool in the evolution of digital media platforms.

With video consumption growing rapidly worldwide, the need for better video captioning technologies has increased, more so for languages like Bangla, which have not been studied extensively. The unique challenges posed by Bangla, noted for its complex syntactic structure and the richness in semantic depth, are such that only special kinds of computational models can be designed. These are models that are competent enough to translate the visual into coherent, contextually appropriate, and linguistically accurate captions.

The main focus of this thesis is on single-action videos. The single-action videos are good for analyzing one single or specific task that is happening inside of a video. This type of dataset can enhance the video captioning abilities of many models by getting familiarized with one action at a time in one video rather than a multi-action video where the models might find it hard to identify the depth of the video or what is happening since the actions are changing every few seconds. For our thesis, we have created a novel dataset focusing specifically on single-action videos in Bangla. To the best of our knowledge, this is the first ever dataset dedicated to video captioning in the Bangla language. This dataset can enable models to learn and accurately recognize objects, actions, and events depicted in low-resource language videos like Bangla, facilitating semantic and precise video descriptions.

This research proposes a hybrid approach toward feature extraction as a response to the multi-dimensional challenges posed by video captioning. The ResNet50 model extracts global features, while the VGG16 model was used for local feature extraction. In parallel, the optical flow technique calculates the dynamics of motion between frames which proves to be a very important feature for the understanding of the fluidity and progression of actions within the videos. This study is not based on one model but digs into several advanced modeling frameworks that generate syntactically consistent, linguistically accurate, and contextual text. We experimented with multiple neural network architectures to find the most effective model for video caption generation. One model used GRU with Bahdanau attention, while the second GRU layer enhanced the ability to understand deeper contexts. In another, a bidirectional GRU was adapted using a Gaussian attention mechanism that made the model reflect on its outputs for relevance to the generated captions. We then developed a bidirectional Long Short-Term Memory (BiLSTM) model that featured Bahdanau attention to maintain temporal coherence in the captions and ensure the captions have the ability to continuously describe video content. Last but not least, a two-layer bidirectional GRU with Bahdanau attention was experimented on to optimally focus on the most relevant parts of video content for the purpose of enhancing the precision and specificity of descriptions. Many more combinations were tried to explore the best possible architecture.

The model that combined GRU with Gaussian attention turned out to be the best performer. This model balanced well between computational efficiency and accuracy and was particularly suitable for our tasks of video captioning. In the next sections, we will describe the architecture of these models, detail its architectural components and operational dynamics, and point out specific advantages that promote superior performance in generating coherent and contextually relevant captions.

The subject of discussion in this thesis is the Bangla language, which certainly is not among the most popular languages. This is necessary for AI in multimedia and opens up development for other areas regarding captions of videos in languages where few references are available. [6] Video Captioning improves video comprehension and benefits improvement in memory, concentration, and understanding of video content, especially to learners, people who speak other languages besides the video content language, and persons with disabilities. We used quite many techniques in our experiments and the one that gave us the best performance was the GRU with Gaussian attention. The aim would be to develop models that can understand and annotate videos at a high level in order to ensure that the usage of videos by users of different languages and cultures would be easier. From what has been established, the research aims at filling the gap between the spoken and the visual languages to assist in the understanding of information that is available through the enhancement of digital global narratives in terms of access to media consumption.

1.2 Motivation

Since technology is rising, the field of video captioning is rising as well with that. Video captioning is now becoming a popular field among fellow researchers who are trying to make better models, datasets, and new methods or approaches to automatically describe video with accurate descriptions. Since video content is rising every day at a fast rate, it is very much needed to automatically generate meaningful and contextually correct descriptions. This can not only help the visually impaired but also can be used in content retrieval systems using video descriptions. [7]Video captioning improves accessibility and information retrieval by leveraging multiple visual features and semantic attributes. The research in this area is growing due to the demand for how it can be used in smarter assistive technologies and content management changing how we interact with video content daily. This can help many researchers, developers, person who is working in any industry, or any person who consumes media. The video captioning sector holds much potential due to many factors

1.2.1 Language integration with AI

With immense growth in the field of Artificial Intelligence and Natural Language Processing, many languages are yet to undergo technological advancement. One of the languages—Bangla—spoken by millions, has yet to develop sufficient research in video documentation. The thesis would, therefore, try to fill this gap with the development of models captioning the peculiarities of the Bangla language to make it more inclusive and usable with AI technology. There is a huge need to develop computational resources and models that can understand and process Bangla to a level at par with global languages like English. This research brings in the linguistic tools and technologies essential for processing Bangla and sets the foundation for further advancement in the field.

1.2.2 Enhancing Relevance

Today's digital environment makes video a key medium for communication and information distribution. Effective video annotations not only enhance the visually impaired but also provide additional support for non-native speakers and help improve comprehension in noisy environments This study will significantly increase the accessibility and usability of video content, thereby improving the quality of content in Bangla. This is a key inclusion requirement, where the wealth of information and interactive opportunities offered by video content is accessible to a wider audience. Enhanced themes will contribute to better understanding and greater engagement, and make the digital content accessible and usable by greater engagement.

1.2.3 Improved content discovery and indexing

Enhanced video captioning has many implications for content discovery and indexing, which are particularly important in languages like Bangla, where digital resources are scarce. Making videos searchable and indexable with accurate, detailed captions will support their use, for example, in education and professional development by enabling quick access to relevant video content. It not only raises the efficiency of specialized field video databases but also supports cultural preservation and education through the creation of a rich, accessible video content repository, which furthers the visibility and utility of Bangla digital resources globally.

1.2.4 Single-Action Captioning Challenge

Enhanced video captions prove to be a valuable means for improving information discovery and indexing, particularly for a language like Bangla with scarce digital resources. Precise and detailed text embedded in videos makes them far more searchable and indexable, a factor increasingly important in places like educational institutions and professional development programs. That will enable the users to search for specific features quickly and with accuracy, improving the performance of video databases across disciplines. Besides, such development helps preserve and teach culture using comprehensive, accessible video archives. Measures like this are bound to increase education significantly by helping users gain a deeper understanding of the cultural heritage and linguistic nuances of Bangla. Such improved visibility and usefulness of digital resources not only support local educational and professional efforts but also make Bangla better known in the world, encouraging better intercultural understanding and exchange. Subsequently, one event dataset can be turned into several event datasets by merging some videos.

1.2.5 Uses in robot vision

The video captioning improves the robotic vision systems for object and scene recognition, navigation, and spatial awareness in a better way, all helping in effective human-robot interaction. It helps robots understand difficult environments for enhanced task performances and monitoring, like navigating obstacles or even helping in an assembly line. Video captioning further provides a continuous basis for learning and adaptation through rich annotations that can be learned by the robots, and these contextually make them aware. It has particular application in the domain

of assistive robotics, where video captioning assists by providing much-required visual information in enhancing the safety and independence of users with disabilities. Video captioning will help robots operate more independently and more naturally within their environment.

1.2.6 Leveraging Advanced Neural Architectures

The intrinsically dynamic nature of video data places high demands on the robustness of the modeling techniques used, able to capture video data intricacies in the temporal and spatial dimensions. This thesis proposes to push the boundary of what is computationally possible for video captioning with a special emphasis on its capabilities in the Bangla language. Specifically, this research will use state-of-the-art neural network architectures and sophisticated attention mechanisms. These technologies are crucial for the processing of fast sequences of visual events in videos and intricate details within them in generating accurate and contextually relevant captions. This probing of state-of-the-art computational methods aims not only to increase the technical efficacy of video captioning but also to make serious contributions to accessibility and utility of digital content in Bangla, thus promoting broader linguistic inclusion and technological advance in the field.

1.2.7 Social Impact

While enhancing the technology, this research has huge implications for the Bangla-speaking society by making digital content more accessible and comprehensible. Improved video captioning can make education more inclusive, learning environments better, and enhance digital resource engagement. Other than ensuring a more informed public to support informed decision-making and civic engagement, it also aids in proper documentation of cultural narratives through better captions and hence preserves and celebrates the Bangla cultural heritage and continuity in relevance in a digitized world. Therefore, this research brings forward wide-ranging social implications by creating an educated, informed, and culturally rich society.

Altogether, these reasons drive home how important this research is for moving the field forward in the discipline of AI and machine learning, more so with a focus on language processing and the understanding of multimedia content. The expected results will increase technological capabilities but also bring important social benefits by making digital content more accessible and understandable to Bangla speakers.

1.3 Problem Statement

The thesis addresses the long-esteemed gap in the area of computational linguistics, an efficient and sophisticated video captioning system for the Bangla language. The amazing increase in global digital video content has resulted in disproportionately less representation and technological progress in captioning for the Bangla language. This is an alarming situation because millions of people speak and depend on this language, but it has been hugely underrepresented in linguistic research and technology development. The medium for information dissemination and entertainment is getting transformed to video content, and disparity in language technologies is

increasing accordingly. This thsis will bridge that gap by addressing some critical issues in video captioning for the Bangla language. If these critical challenges are not met, access to digital content for a huge portion of the global population may be very tough. This research, through a series of targeted objectives, deals not only with advancing the technological framework for Bangla video captioning but also with advancing the wider goals of inclusiveness and accessibility in multimedia content.

1.3.1 Lack of Advanced Modeling Techniques

The most advanced video captioning methodologies are developed for well-studied languages like English. The methods used are based on vast research and a big repository of linguistic data, something not so easily available for a less studied language like Bangla. Therefore, when these methodologies are applied to Bangla, they often fail because of the presence of unique linguistic structures, syntactic rules, and specific usage contexts that make it distinctly different from English. Our research addresses this important gap by proposing innovative modeling techniques that will be tailored to address the intricacies in the Bangla language. Models attuned with the semantic and syntactic peculiarities of Bangla, which can precisely model contextual relevance and accuracy, can significantly improve video captions. This would involve exploring novel neural network architectures that can deal better with the temporal sequences typical for video content and use context-aware algorithms to understand the difficult interplay of visual and textual elements in Bangla. The result will be a robust video captioning system that, beyond working on technical benchmarks, resounds with native speakers through naturally flowing and culturally coherent captions.

1.3.2 Inadequate Datasets

A major barrier to progress in the domain of Bangla video captioning is the severe dearth of comprehensive, high-quality datasets specific to this language. Existing datasets lack detailed annotations and do not offer the depth in context that complex captioning models require. The shortage of well-annotated datasets poses very serious constraints on our ability to develop models that are capable of understanding and reproducing the complex, multi-layered linguistic nuances and cultural subtleties that characterize Bangla. To this end, one major emphasis in this research will be to build and curate a richly annotated video dataset that reflects the different and dynamic aspects of Bangla speech and text. This dataset will encompass a wide array of video content from everyday activities to culturally-specific events each supported by several contextually appropriate captions. These captions shall be constructed to capture different expressions, idiomatic uses, and syntactic structures that characterize Bangla, thus providing a robust grounding for training a much more advanced and accurate video captioning model. The production of such a dataset will not only facilitate the fine-tuning of the performance of our models but also set a new benchmark for dataset quality and comprehensiveness in the computational linguistics domain for under-represented languages.

1.3.3 Complexity of Accurate Feature Extraction

Effective video captioning in Bangla requires the development of state-of-the-art feature extraction techniques capable of precisely capturing the intricacy of details at the micro level and the overall context at the macro level in the video content. This involves capturing the subtle visual and audio cues, such as action by individual elements and broader narrative elements, and understanding the temporal progression in the video stream. Our aim in this research is to extend these capabilities by using state-of-the-art deep learning models: Convolutional Neural Networks (CNNs) for extracting static visual features and Recurrent Neural Networks (RNNs), especially Long Short-Term Memory (LSTM) networks or Gated Recurrent Units (GRUs), in handling temporal dynamics. We also try to connect different attention mechanisms that let the model selectively focus on the most important elements at any point in time; this way, it emulates human cognitive focus and increases the contextual appropriateness of the captions. These enhanced feature extraction methods are essential in generating quality, contextually appropriate captions, which will help evoke the cultural and linguistic subtleties of the Bangla language and lift the performance and usability of video captioning systems.

1.3.4 Evaluation Metrics and Model Benchmarking

The video captioning systems in the Bangla language need evaluation metrics that are reliable and strong enough to assess both linguistic accuracy and relevance and contextual appropriateness of the captions they come up with. The evaluation metrics are important in assessing the quality of generated captions and, hence, that they do not stray very far from the intended message or video content context. Assessing model performance and progress is important by comparing it with clear benchmarks that would be set. Therefore, setting benchmarks will help provide standard criteria for comparison and evaluation of different models with an aim of achieving uniformity in reviewing progress within the field. Further, setting such thresholds can be used to clearly point out areas in need of further improvement and stimulate innovation in Bangla language processing. It seeks to ensure that the developed captioning systems actually help end-users by providing them with accurate and meaningful video descriptions.

1.3.5 Integration of Cultural and Contextual Sensitivities

The problem for video captioning concerning this language is not only linguistic accuracy but deep insight into cultural and contextual nuances that resonate with local audiences. This statement implies that the captioning system should be capable of understanding and incorporating cultural references, idioms, and contextual cues of the regions in which the language is spoken. Captions have to be context-sensitive and socially apt to the several cultures that use the language, capturing subtleties in regional variations, humor, and cultural allusions meaningful to the audience. It would assist in evoking a deep, personal level of engagement from the viewers beyond the simple transfer of factual content. In this regard, the development process would be coupled with linguistic expertise and cultural consulting in the form of the detection and interpretation of these cultural subtleties with natural language processing tools. The goal is to create a video captioning system that is culturally

aware and capable of producing captions that are both informative and emotionally appealing in the viewer's experience with the speakers of Bangla.

1.3.6 Scalability and Real-Time Processing

Video captioning is one of the fastest-growing areas, with live broadcasts and other real-time requirements growing at such a pace that new scalable solutions are needed to provide high-quality captions instantly. This makes video captioning at scale very challenging in terms of accuracy and contextual relevance under live performance constraints. We research the scalable architectures and real-time processing frameworks that could handle such demands. We will be working on innovations in computational process streamlining, optimization of algorithmic efficiency, and utilizing advanced machine learning techniques to quickly process video content without compromising the quality of output. Technological advancements will enable real-time systems to operate with seamless live events, providing Bangla speakers with instant, accurate, and contextually appropriate captions. This will not only help to make live content more accessible but also guarantee full ability to enable every viewer to engage with media in real-time irrespective of the person's hearing abilities or language skills.

1.3.7 Accessibility and Inclusivity

Improving video captioning of the Bangla language is a milestone toward digital content universal access, especially benefiting millions of Bangla speakers worldwide, including deaf and hard-of-hearing populations. The quality and accuracy of captions are critical to achieving the goals of technology inclusivity and accessibility. An effective captioning model is such that users with equal access to video content, be it for information, entertainment, or educational and public service broadcasts, understand it easily without any difference in auditory capabilities or linguistic background. This is in enhancing digital inclusiveness in technology to serve all people.

This study addresses these challenges by developing dedicated encoder-decoder architectures and an original dataset to set a new benchmark for video captioning in Bangla. This serves to contribute more linguistically diverse and effective accessibility in the AI domain, marking strong steps toward more inclusive and effective communication in multimedia.

1.4 Research Objectives

The main purpose of this thesis is to develop and evaluate customized encoder-decoder architectures for advancing the field of video captioning in the Bangla language. The following are the specific objectives:

- **To Design Specialized Encoder-Decoder Architectures:** Construct various advanced neural network structures that can generate Bangla captions for videos specifically. These models will contain counterparts with GRU, BiGRU, and BiLSTM cells, which use both Bahdanau and self-attention mechanisms to deal with temporal and contextual aspects related to video content.

- **To Create a Comprehensive Bangla Video Dataset:** Create a collection of single-action videos with appropriate Bangla captions addressing a wide range of activities and conditions. This dataset is a vital tool for training, validating, and testing the developed models so that they can fit well within the linguistic and cultural contexts of Bengali.

- **To Enhance Feature Extraction Techniques:** Investigate how global, local, and motion-based features may be extracted from video frames using up-to-date deep learning models such as InceptionV3 or VGG16 together with the optical flow-like techniques to efficiently capture full visual context.

- **To Evaluate Model Performance with Rigorous Metrics:** Make use of standard linguistic metrics like BLEU, METEOR, ROUGE-L, and CIDEr to rigorously evaluate the performance of each captioning model. These measures will help in gauging the accuracy, relevance, and level of language correctness of the generated captions.

- **To analyze the effectiveness of various architectural components:** Systematically determine how different components of encoder-decoder models that include bidirectional layers and attention mechanisms, among others influence video caption quality. Such an analysis would enable us to establish what works best for Bangla video captioning.

- **To Contribute to AI and Machine Learning Research in Bangla:** AI and machine learning approaches to process Bangla by providing insights, methodologies, open-source resources, etc., for other researchers and practitioners who may wish to adopt them as well as their potential application for other low-resource languages through adaptation.

  Through these objectives, this thesis desires to largely improve on the potentiality of AI systems to produce meaningful and contextually suitable video captions in Bengali thereby enhancing accessibility, content discovery overall utility of videos.

Chapter 2

Literature Review

Video captioning lies at the intersection of computer vision, natural language processing, and multimedia processing. The demand for systems capable of interpreting and narrating video content in human-like language is rising at an exponential rate with digital video content spread across platforms like YouTube, social media, and broadcast media. Today, research in this arena is pushing to develop the most sophisticated algorithms and models yet for generating accurate and contextually relevant video captions. Video captioning is a method that employs advanced computer vision techniques to analyze visual elements over time and NLP methods to express these observations in coherent, natural language captions.

Video captioning begins with the very difficult task of understanding the temporal dynamics and spatial details present in a video sequence. Unlike static images, videos represent moving elements and a narrative structure over time, making the process of captioning even more challenging. In the process of understanding visual elements within each frame, deep learning architectures like Convolutional Neural Networks (CNNs) are applied. At the same time, Recurrent Neural Networks (RNNs) or Transformers are used to understand the temporal sequences and transitions between frames. Traditional frame-by-frame analysis would be limiting and could not capture the continuity and narrative flow of video content. To deal with the challenges, researchers are exploring sequence-to-sequence models that process the holistic view of video streams much better and provide event captioning.

Yan et al. in his paper [8], tackle the nuanced challenge of video captioning by developing a structure that refreshes the portrayal of video contents through a refined spotlight on both global and local visual parts. The authors introduce a novel architecture called Global-Local Representation Granularity (GL-RG), which strategically combines these features to produce a richer, more detailed linguistic output. This technique watches out for the impediments of earlier models that either disregard the unpretentious between outline changes or deficiently model the visual setting, inciting less specific or stirred-up video captions. Through a predictable preparation procedure, the GL-RG structure does not just show a common show on benchmark datasets like MSR-VTT and MSVD, what's more, gives snippets of data to work on the collaboration between various video frame representations. The examination is maintained by a general assessment against cutting-edge techniques, showcasing significant improvements in caption exactness and detail, which presents

a convincing protection for the development's sensibility and potential applications in confirmed conditions like assistive advances and mechanized video evaluation.

The authors, [9] offer a new framework for identifying and describing vague activities in video clips while using common-sense hierarchies and zero-shot recognition at the same time. In contrast, the paper offers to generate short-word summaries of videos without relying on the same data distribution videos. The researchers propose the use of semantic trees leaning on data for the correct level of generalization and introduce the "weirdness" of compositions of actors, actions, and object nouns using priors learned from web-scale language models. In a nutshell, the semantic prior extends the dataset when the web-scale language model can, besides the common verbs, also propose a present for the unseen ones. Our present paper focuses on the semantic prior and web-scale language models to generate the correct description. In particular, the model is observed to perform very well for the competing approaches for the sake of generating short, coherent sentence descriptions about the video content. The presented essay is valuable as it gives hierarchical semantic and language models, which are used relative to the tremendous scale of a large dataset.

In their paper, [10] present a novel approach for action recognition in videos using a spatio-temporal descriptor based on 3D gradients. The authors fostered a memory-efficient algorithm for figuring 3D gradients through fundamental records, which considers conflicting scales. They correspondingly proposed a 3D orientation quantization framework using standard polyhedrons, reviving determination to illuminate changes. Their descriptor was rigorously tested on several datasets: KTH, Weizmann, and Hollywood. On the KTH dataset, it accomplished a standard accuracy of 91.4%, beating many existing systems. The Weizmann dataset, which highlights ten action classes, saw a typical precision of 84.3%. For the Hollywood dataset, which joins a blend of action classes from films, the descriptor showed steady execution, at this point with some instability across various classes. Despite its success, the model has limitations. Its show predominantly depends on ideal limit settings, which can separate across datasets. Besides, the computational different arrangement of using key records and 3D gradients presents a challenge. Nonetheless, the proposed descriptor shows tremendous potential for improvement in affirmation endeavors, highlighting the reasonableness of 3D gradients in capturing spatio-temporal features. The disclosures of [Klaser, Marszalek, and Schmid] feature the utility of 3D gradient-based descriptors in video evaluation, opening new pathways for future assessment, in all actuality, affirmation, and related areas.

In their paper, [11] investigate the integration of recurrent neural networks (RNNs) with convolutional neural networks (CNNs) for tasks involving sequences, like video recognition and picture description. They develop a sporadic convolutional orchestration that is useful from start to finish, utilizing the potential gains of both spatial and standard importance. The model gets both spatial parts and temporal dynamics by being compositional from these points of view. The creators offered their model an open-door benchmark dataset for video interest and picture depiction. Their method outsmarted existing models that depend on fixed spatio-temporal open fields or clear common averaging, showing central updates in these under-

takings. Anyway, the model's intricacy and computational referencing are higher isolated from less irritating models. Moreover, redesigning the coordinated organization can be challenging. Despite these issues, the outcomes show that getting CNNs along with RNNs offers evident benefits for visual attestation and depiction undertakings. Donahue et al.'s exposures feature the restriction of excess convolutional models in supervising befuddling ordinary parts and moving the field of visual certification and depiction.

As described by Das et al. in [12], an approach to video annotation has been devised, employing bottom-up along with top-down hybrid approaches, and a three-level system view on the problem poses as another solution. It does so using low-level sentences represented by multimodal latent topic models that are passed through the keyword annotation process and high-level sentences produced naturally, selecting the most relevant features of the video. The fake sentence is more similar to the human description, only the output of the complex system can show such an amazing oeuvre. This signifies descriptions creating links between bottom-level visual characteristics and the top-level abstract descriptions.

Rohrbach et al. investigate the possibility of generating natural language descriptions for video by mixing technology of visual recognition with statistical machine translation [13]. They create a rich semantic representation of the visual content, including object labels and activities, by using Conditional Random Fields to model relationships between those components of the visual input. The natural language generation is considered translation, with the semantic representation (SR) as the source and sentences as the target. To achieve this translation, they use a parallel corpus of videos and textual descriptions. The distribution models of SR allow for this transformation. Their approach is fully automatic and does not require manually refined instructions. Their work is significant and widely recognized by the community. Their method outperforms baseline ones according to the quantitative metrics which are BLEU and human judgments, and it is noticeable by the overview audience – their work is among the top researchers in the area of image description.

[14] introduce a novel sequence-to-sequence model, S2VT, for generating video captions. In the proposed model, LSTMs are used for handling the video frames' temporal dynamics and generating descriptive sentences. In the S2VT model, a stacked LSTM architecture is used, each connected to CNNs. The model processes sequences of frames and generates sequences of words. On the evaluation of the S2VT model using the Microsoft Video Description corpus, the MPII Movie Description Corpus, and the Montreal Video Annotation Dataset, state-of-the-art performance is achieved with a METEOR score of 29.8% on the MSVD. However, the model was not that successful with the MPII-MD and M-VAD datasets, which have very high diversity in activities and visual content. The authors underline that their model outperforms the traditional methods that map video frames to sentences without an explicit attention mechanism, while integrating optical flow features showed lower performance, denoting the necessity for a better way to integrate with appearance features.

The paper [15] introduces a sophisticated methodology for converting video content into natural language annotations. This approach combines computer vision and natural language processing by first performing a fine translation of visual data using conditional random fields (CRF) and then determining this representation using a statistical machine translation method down into writing. The results show that the CRF model predicts comprehension levels well, resulting in more accurate and slower descriptions compared to theory-based retrieval methods, unlike proposed algorithms achieves not only high accuracy and quality in identification but also provides flexibility in processing the symbol enhancement of various optical data as well as accessible to visually impaired individuals.

[16] Venugopalan et al. present the first approach for direct video-to-sentence generation by using a deep neural network that is designed under convolutional and recurrent layers. This approach builds upon the capacities of these pre-trained networks on large datasets of images to improve the performance on the video description tasks, a domain with scarceness of datasets and with large vocabularies. Their model adopts a convolutional network for extracting features from video frames and LSTM for word-by-word generation of descriptions. The conducted experiment based on the MSR Video Description Corpus indicates the system's effectiveness in terms of BLEU and METEOR compared to similar works. This makes it a step forward in the field of video-to-text conversion since it can capture the spatial features and temporal characteristics of videos without passing through a semantic space. Based on this fully end-to-end setup, this paper also shows the potential of deep learning architectures for direct mapping from raw video to coherent natural language outputs for the improved understanding of temporal visual information.

In their paper, [17] propose a technique for making video depictions by utilizing both local and global temporal structures. They maintained a structure that works with a 3D Convolutional Neural Network (3D CNN) to get present second conventional dynamics and a temporal attention mechanism to pick enormous temporal segments for making Text with a Recurrent Neural Network (RNN). The 3D CNN was prepared on video action recognition undertakings to give highlights sensitive to human emotion and behavior. The producers assessed their model utilizing the YouTube2Text dataset, achieving cutting-edge execution in BLEU and METEOR evaluations. They equivalently offered the model an open door to a new, more fundamental, and more testing dataset of matched video and natural language descriptions, showing the model's adaptability and believability. Regardless of its prosperity, the model's extravagant and computational arrangements present difficulties, and its show-off could move with additional stick-out and unstructured information. All around, [Yao et al.] feature the normal augmentations of cementing common plans in video depiction attempts, driving the field generally.

The Steered Gaussian Attention Model introduced by [18] for better video understanding, thereby setting novel state-of-the-art automatic video captioning performance. This new model introduced a method in which dynamic attention of the Gaussian parametric attention, beyond the fixed-duration constraint of the inputs, was integrated with the temporal steering mechanism. At the core of their approach is the use of Video2Vec latent encoding to capture strong temporal and spatial fea-

tures of a video that strongly guide the attention model through the video duration. This is in contrast to current practice: methods where static attention models, conditioned by the temporal dynamics of the training data, often generalize poorly to new content. We empirically evaluate the effectiveness of the proposed model on a set of benchmark datasets, including MSVD, MSR-VTT, and M-VAD, to demonstrate that our proposed Steered Gaussian Attention Model systematically improves over the existing state-of-the-art in BLEU and METEOR scores. This is mainly attributed to the capability of the model to provide localized attention in a salient way, allowing adaptiveness to video length and complexity. A major contribution of Sah et al. is designing and implementing length-agnostic attention models that can fit smoothly into a hierarchical video representation. This will increase the adaptability of the model across a wide range of video content types and subsequently open new research directions in video captioning and other related problems of video understanding. The advanced model can, therefore, fill the gap between the features of video content and natural language descriptions, setting up a robust framework for translating complex video information into coherent and contextually relevant captions

In their idea, [19] focused on attracting a model for making natural language captions for videos in Bengali. They achieved this by proposing an encoder-decoder architecture that joins 2D Convolutional Neural Networks (2D-CNN) and 3D Convolutional Neural Networks (3D-CNN) as the encoder, and Bidirectional Long Short-Term Memory (Bi-LSTM) as the decoder. The makers worked with and surveyed their model using the Microsoft Video Description (MSVD) dataset. Their model achieved a BLEU score of 30% and a CIDEr score of 20%. These results are significant given that there could have been no previous models for Bengali video recording. Regardless, the central hardships were the multifaceted arrangement of the Bengali language and the absence of close benchmarks. Despite these challenges, their work spreads out the assistance for future assessment in Bengali video captioning, showing the obstacle of joining CNN and LSTM structures for this task.

[20] Shaha et al. proposed a new architecture for video captioning while handling the typical challenges, like the shortage of datasets for this language. They created a fully human-annotated dataset for captioning Bengali videos, with the Microsoft Video Description Corpus for benchmarking. Their new end-to-end architecture introduces an attention-based decoder on the spatial and temporal features of the video through bidirectional Gated Recurrent Units into the architecture. This will have more capacity to dig deeper into the dependencies that exist between visual elements and their textual descriptions. The attention mechanism in their model is quite noteworthy in terms of developing the output sentences semantically and grammatically relevant for Bengali, one of the complex, nuanced languages. We infer highly from this paper how to combine advanced techniques such as VGG-19 for static feature extraction with capturing temporal dynamics using ResNeXt-101 to fully capture very subtle and intricate fine details needed for an effective description of video content. Their model results in performance evaluations with a margin of improvement over current methods and outperforms state-of-the-art results on BLEU-4, CIDEr, and ROUGE metrics. This is a further reason that raises attention to approaches that are effective both in dealing with the linguistic features typical

for Bengali and in terms of raising the general quality of video captions, produced by a machine learning model. The importance of the work by Shaha et al. is that it drives the field of video captioning for under-represented languages further and sets an excellent example for future research work in applying multimedia AI to regional languages. This work presents a strong framework for further exploration in the domain of bilingual or multilingual video captioning systems, building upon the rich semantic interplay between visual content and language.

Chapter 3

Datasets

3.1 Data Collection

To prepare the video dataset for Bangla captioning, we combined single-action videos from [21] [22] [23] different open-source platforms to make sure that our collection does not infringe on any copyright laws. We used the MSVD (Microsoft Research Video Description Corpus) dataset structure as our main structure for our dataset. Each single-action video in our dataset is uniquely defined and identified by a video ID. For each video ID, there are corresponding captions, which are more or less 20 Bangla captions. We added accurate and grammatically correct captions during the dataset-making. We also made it bias-free. One group observed the single-action video and generated ten captions, meaning the video dataset was free from biases. Afterward, another group cross-checked it and came up with another ten captions, after which the last group performed the cross-validation job. A proper way to manage our data is to name and categorize the single-action videos based on activities and then place them in a folder so that it is easy to locate and work with them. such is the case for captions. Such an organized approach helps in working with our data, and research possibilities increase in an automated Bangla single-action video-captioning scenario. After all this process the total amount of single-action videos collected was 2119 and the number of captions was 42406. We have given our new dataset the name "BSAVD - Bangla Single Action Video Dataset" to provide a concise and descriptive name for this first-ever dataset created specifically for video captioning in the Bangla language.

3.2 Dataset description

We have structured this dataset in a way that single-action videos are matched with strategic corresponding captions, making it easy for further deep research on Bangla video captioning automation. The dataset precisely consists of a carefully knit collection of videos, each one with a unique video ID, thus sourced from some of open-source platforms. For each video ID, there are corresponding captions, each having more or less 20 Bangla captions, thus presenting enriched possibilities for linguistic and general-context analysis. All of these are stored in a separate folder, under a specific folder named with its unique identifier, making the matching of the single-action video to its captions easy. This again is strongly motivated by the already established MSVD dataset, known through the very robust framework

that is adequate for any kind of machine learning application. On top of that, our dataset is further refined in the sense that the dataset deals with the Bangla language only, thus closing that gap with the already available resources for dealing with the Bangla language in video captioning. Apart from the raw single-action video files, the dataset includes a well-organized CSV file, specifying the video IDs against their captions. Such a file serves not only as a bridge between video content and its textual descriptions but also makes the access and manipulation of the data in the training process of models associated with understanding and generating single-action video captions. We renamed the videos into a format so that single-action videos can be easily processed, which further strengthens the thematic organization and helps in certain canons, such as action recognition and contextual understanding in videos. Since then, after the primary preparation, all videos and their respective captions were stored in one folder for each type to structure the dataset accordingly so that it is easy and convenient for both researchers and developers working on it. The dataset can importantly help to do further research in technology related to video captioning, especially in the Bangla language, alongside natural language processing and machine learning fields.

Figure 3.1: Sample Video 1

Captions:-

- গরু ধানক্ষেতের কাদার উপর হাঁটছে।
- ধানক্ষেতে গরু হাঁটছে গোলাপী শার্ট পরা বৃদ্ধের সঙ্গে।

Figure 3.2: Sample Video 2

Captions:-

- সাদা শার্টে লোকটি বাগানে পাতা সংগ্রহ করছে।
- সাদা শার্টে লোকটি গাছের কাছে পাতা তুলছে।

Figure 3.3: Sample Video 3

Captions:-

- ব্যক্তি হলুদ পোশাকে নাচছে।
- হলুদ পোশাক পরা একজন লোক নাচছে।

3.3 Data Augmentation

3.3.1 Caption Augmentation

Data augmentation may be executed using data duplication and interlacing the captions of the videos. This way, the data to be used may be trained to a machine learning model where the data multiplies. Several benefits of this approach during the process are that it loads a video identifier and the corresponding captions dataset. During augmentation, all entries linked by the identifier of a video are grouped, and this means that they can be viewed and considered as one in the eye. It will then follow up with duplicating all the groups of the entries that are linked to the aforementioned video, and then interlace the original entries and the duplicated entries inside a group. Hence, duplicating entries in a way such as this makes sure that effective data related to a certain video is doubled. This has been proven useful, as in such humongous models, the model learns from different examples but does not learn from the new data in its vicinity. The file is then augmented and stored back into a structured file, or this can be the augmented version, replacing the former dataset itself, and this can be the one to be used for further processing. This augmented dataset is used to expect the ability of the model to be robust. This dataset will be used to train the model on more samples and go through how it should make more predictions on new samples.

3.3.2 Video Augmentation

This process augments and organizes video files for machine learning training and test purposes. The process initiates with checking and making a target directory within which it is assumed that storage space specifically dedicated to the process is present. The process further iterates over each video file present within a source directory. It also proceeds to have two different operations for both training and testing, depending on whether the video identifier is in the list of either training identifiers or testing identifiers, respectively. For training videos, the process takes the horizontal flip and increased color intensity of 10% to get an augmented version of the video, and then saves it in the target directory by modifying its name to indicate that the file is an augmented video, and the original video is also copied to the same directory. For testing videos, the function simply copies the original file to the target directory without any modifications. This procedural nature allows for augmentation of the number of iterations in the training set to increase diversity, and simultaneously does not hamper the integrity of the augmentation set for consistency in evaluation, allowing efficient usage of closing video files post being processed.

3.4 Caption preprocessing

The goal is to prepare a dataset that will be beneficial for training machine learning models for video captioning. The text descriptions of the different videos are initially passed through the tokenization and pre-processing library developed for the Bangla language. We used the BNLTK module for tokenization. The dataset of video identifiers holding their respective descriptions is divided into training and testing subsets, for which the percentage considered is kept very low to test the model's

performance on unseen data. For each video in the training subset, a description is tokenized to a sequence of words and then encapsulated within markers denoting the beginning and end of a sentence. This helps the model familiarize itself with the boundaries of the sentence at training time. The processed captions are then filtered based on their lengths to maintain the uniformity and manageability of the train data. Shuffling randomizes the order of the data. Then, this further set is divided into a greater portion. The use of a validation set is to fine-tune the parameters of a model and to make sure the model doesn't overfit that much on train data. Forming a vocabulary list from the train captions and training a tokenizer on it. The tokenizer forms an integer sequence of the text, where each integer denotes a word or a token from the vocabulary. The numerical representation of text is what machine learning models work with. The tokenizer is configured in such a way that it models a set number of the most frequent words, keeping the complexity low, and in turn, the model should be efficient. This process on a dataset with definite video IDs and captions is important for preparing well-organized data for an effective video captioning model.

3.5 Video preprocessing

3.5.1 Video Scaling:

We used the downscaling method to compress the size of the video from higher quality to 1280 x 720. The video file folder is treated to the same resolution, usually predetermined, like 1280 x 720 pixels, applying it to the program. First, the orientation of the video is decided, whether it is kept in landscape or flipped to portrait so that it matches the resolution. In both cases, the script opens for each video stream, reads through frames sequentially, resizes each frame to new dimensions, and writes them into a new video file in the user-specified output directory. The entire process is repeated for every MP4 file in the source directory, incrementing how many times the loop has been iterated so far with the implementation of the progress bar. The output directory is created if it doesn't exist. Efficient libraries are used for the processing of videos frame by frame, which minimizes memory usage and increases efficiency. Some of the videos we collected were too small in resolution. So we have to use the upscaling method to make the videos 1280 x 720, so the data is kept constant for all videos. It first checks the existence of the provided output directory and creates it if it does not exist. Then, it lists all MP4 files in the input directory. For each read video file, the following operation is performed: read each file, frame by frame. For each frame read, the frame needs to be resized into the dimension of the desired resolution using linear interpolation, and this is the best method that can be used. With this, we save the new video file that contains all the resized frames to be put into the specified codec and frame rate in the output directory. We have error handling in case we fail to open a video stream. The progress bar keeps track and shows the advancement of the upscaling process with every video. Resources are released, and the system ensures that it closes all windows when it finishes processing all the frames of a video. When a processed video causes a failure, a message prompting a failure will be displayed. This will all be done for various videos on the list, allowing the user to upscale them all in quick time to higher resolutions.

1. Video to Frame:

The function retrieves a video from a designated location and extracts each frame in sequence. If the total number of frames is below the desired threshold of 25, it compensates by duplicating the last frame until the threshold is met. The formula used is:-

$$\text{Number of frames to add} = 25 - \text{current number of frames} \tag{3.1}$$

The function ensures a uniform distribution of visual content by dividing the total number of frames into 25 segments and selecting the first frame from each segment. This is accomplished by splitting the range of total frame indices into 25 parts, and the selection process involves picking the first index from each part. Total frames represent the total number of frames extracted from the video. The array split function is employed to divide the range of frame indices into 25 parts, ensuring each segment is as equal in size as possible. The first frame from each segment is then selected to represent that segment evenly across.

2. Resize Frame:

Each selected frame is resized to target dimensions of 224x224 pixels. The scale for resizing is determined by the smallest ratio between the target dimensions and the frame's current dimensions:-

$$\text{Scale ratio} = \min\left(\frac{224}{\text{frame height}}, \frac{224}{\text{frame width}}\right)$$

If the resized frame dimensions are smaller than the target, symmetric padding is added to meet the required size. The amount of padding on each side is computed using:-

$$\text{Padding width} = \frac{224 - \text{new width}}{2}$$

$$\text{Padding height} = \frac{224 - \text{new height}}{2}$$

Padding is applied using a black color to ensure consistency and avoid influencing the analysis.

3.5.2 Global Feature Extraction

We used global features extraction because the global features are video frame summarization features that aim at encapsulating the overall appearance of a frame. Such features represent the whole discrete components that are contained in a scene or set of scenes and give us an understanding of what is happening over time. In our model, global features were extracted from videos using a multi-step process. After doing the steps from above with the processed data the feature extraction method starts to execute. In preparation for training, we resized each image to have symmetric padding so that its dimensions would match those of the target size while preserving the aspect ratio. To align with the model's input requirements, frames were preprocessed. Features were extracted from frames using a pre-trained ResNet50 model which were then reshaped into a feature vector for each video.

These global features represented content from videos and were stored as NumPy arrays for future processing tasks down the line. Through this approach, global video features were efficiently extracted thus allowing full representation and examination of video contents.

3.5.3 Local Feature Extraction

Local features in the video look for specific regions within frames and provide detailed and robust descriptions for object identification and matching. They are often resistant to changes caused due to scale, orientation and illumination. Some applications include object recognition, frame retrieval or matching, etc. The frames are then pre-processed for resizing and padding before passing into a pre-trained VGG16 model that is initialized to do feature extraction after dropping the top layer and adding a new dense layer to it. Pre-processed frames are passed through the model to predict features. The function will save the characteristics into a folder provided in the form of NumPy array files for each video. It also keeps the file of the extracted features under the given directory and splits it by the name of the video so as to make retrieval easy for further analysis or training.

3.5.4 Motion Feature Extraction Using Optical Flow

Optical flow is a technique for motion estimation between two consecutive frames based on the change of pixel intensity. It is useful in cases, among many, where we need the analysis of the motion in videos. Padding will be done in cases where the number of feature optical flow feature vectors, which will be extracted from the video, is not able to hit the target set in specifications; in our case, it is 25. The algorithm will just copy the last computed feature vector until the number of frames to be sent to the next processing is equal to the number of computed feature vectors. This way, each processed video contributes the same number of 25 uniform feature vectors, which ensures a trainable and consistent analysis. This padding strategy is crucial to make the analysis comparable and consistent across the videos for training machine learning models, which require data of fixed length. Padding, in case the video does not have the requisite number of frames, is done by duplication of the last feature vector such that the number of required frames is passed on to the subsequent processing. In the feature extraction process, the frames are first converted to grayscale to simplify the process of calculation of motion. Optical flow is calculated using the Farneback method. The Farneback method is applied to get the estimation of the flow vectors in 2D. The magnitude and angle of each of the flow vectors are calculated, which denote the intensity and direction of the motion, respectively. These are flattened and concatenated to create a feature vector for each pair of frames. The n-length feature vector is padded with the values of the last frame of the video. After the feature vectors of all the desired frames are queried, they are accumulated and stored in a pre-defined directory for motion features in some structured format. This will let the data be easily stored and retrieved not only to keep good reporting with logs but also to trace the process and pinpoint issues that need to be resolved. A systematic procedure of the extraction and handling of optical flow features is paramount to having a good dataset for the enhancement of motion analysis and understanding in a video, which forms a basic block of huge

computer vision applications.

3.6 Data Analysis:

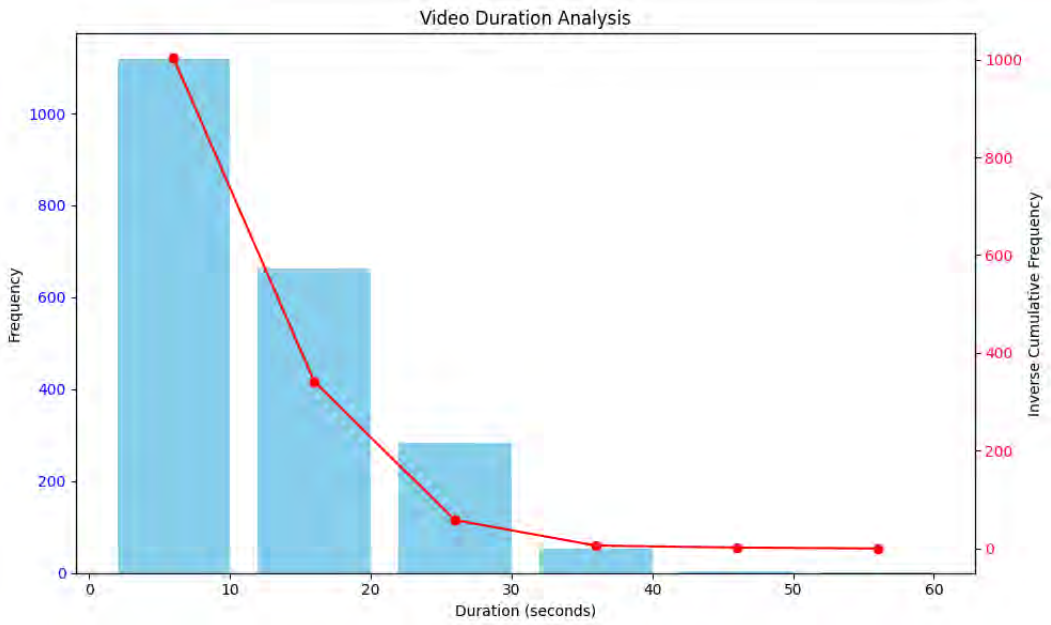


Figure 3.4: Video Duration Analysis

The graph indicates that videos are most frequent in the shorter length range of 0 to 10 seconds. The number of videos then falls off dramatically as time increases, dropping sharply at 10 seconds and even more so at 20 and 30. Very few videos are longer than 30 seconds with only one or two up to 60. This information can also be seen on an inverted cumulative distribution function graph like this. There is an initial steep decline followed by a levelling out which shows that there are progressively fewer increments in duration containing more videos beyond each new phase point. The data primarily contains more short videos rather than longer ones. It seems that this pattern shows us that there are a lot of very short videos in the dataset and not many long ones at all.

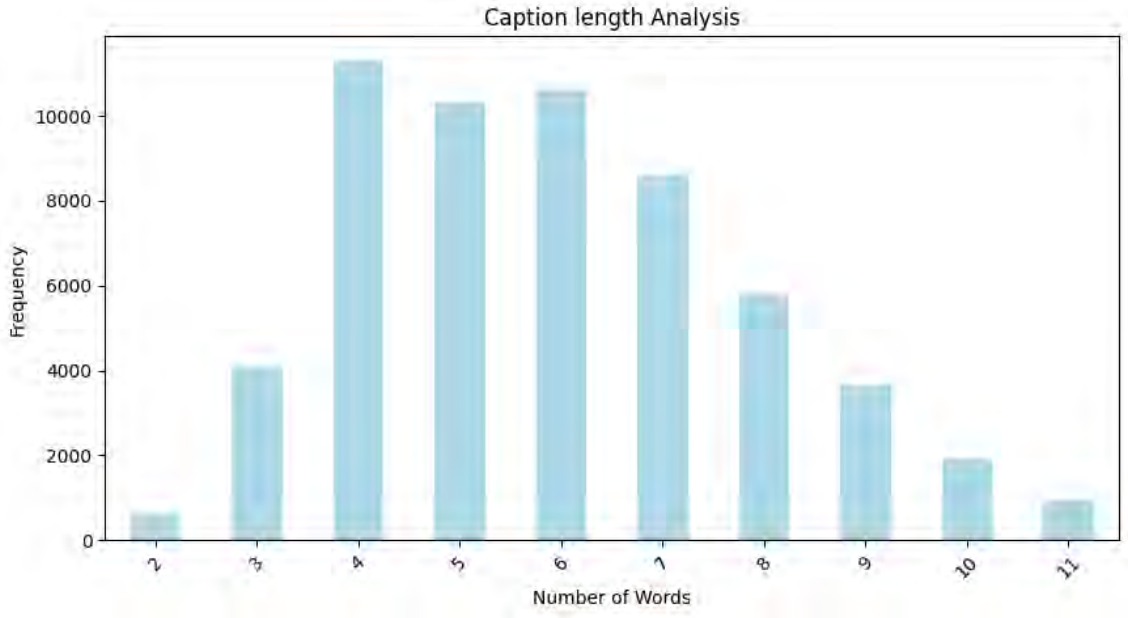


Figure 3.5: Caption Length Analysis

The graph illustrates a marked inclination towards the length of captions in the dataset. Captions having four, five, and six words are most common given that their frequencies are relatively high for each of them ranging from around 8000 to about 10000 instances. This means that most captions are succinct and meant for short, impactful descriptions. Captions less than four words and more than six words have much-reduced frequencies with two and eleven worded ones being least frequent. The picture created here is that very short or very long captions may not be popular or applicable in this dataset context due to either video nature or captioning task requirements.

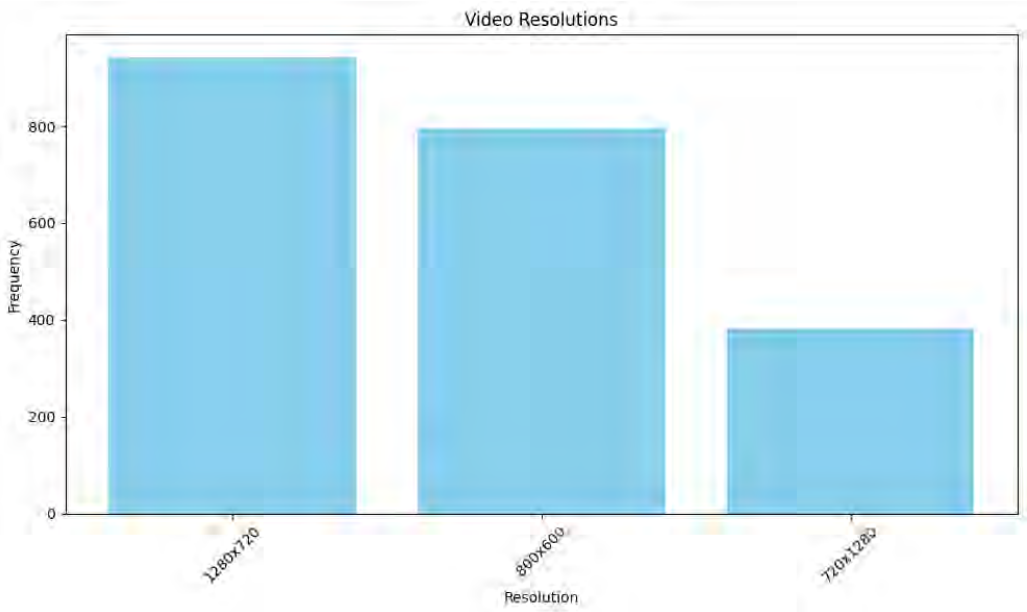


Figure 3.6: Video Resolution Analysis

In this graph, the slot gives the distribution of different video resolutions in a set of data. The main takeaway from the above graph is that almost two-thirds of the accessible resolutions are 1280x720, and it is also the most common resolution among all the videos, after which the 800x600 resolution ranks next. The 720x1280 resolution although contextual is less frequent compared to the other two. This indicates a probable '1280x720' & '800x600' resolutions standardization, which are usually tailored for 'HD videos.' When line 1280x720 is transposed vertically to become 720x1280, it might be that such a target is for the videos to be viewed in portrait on a mobile device.

Chapter 4

Proposed Model

The complete workflow is outlined as follows-

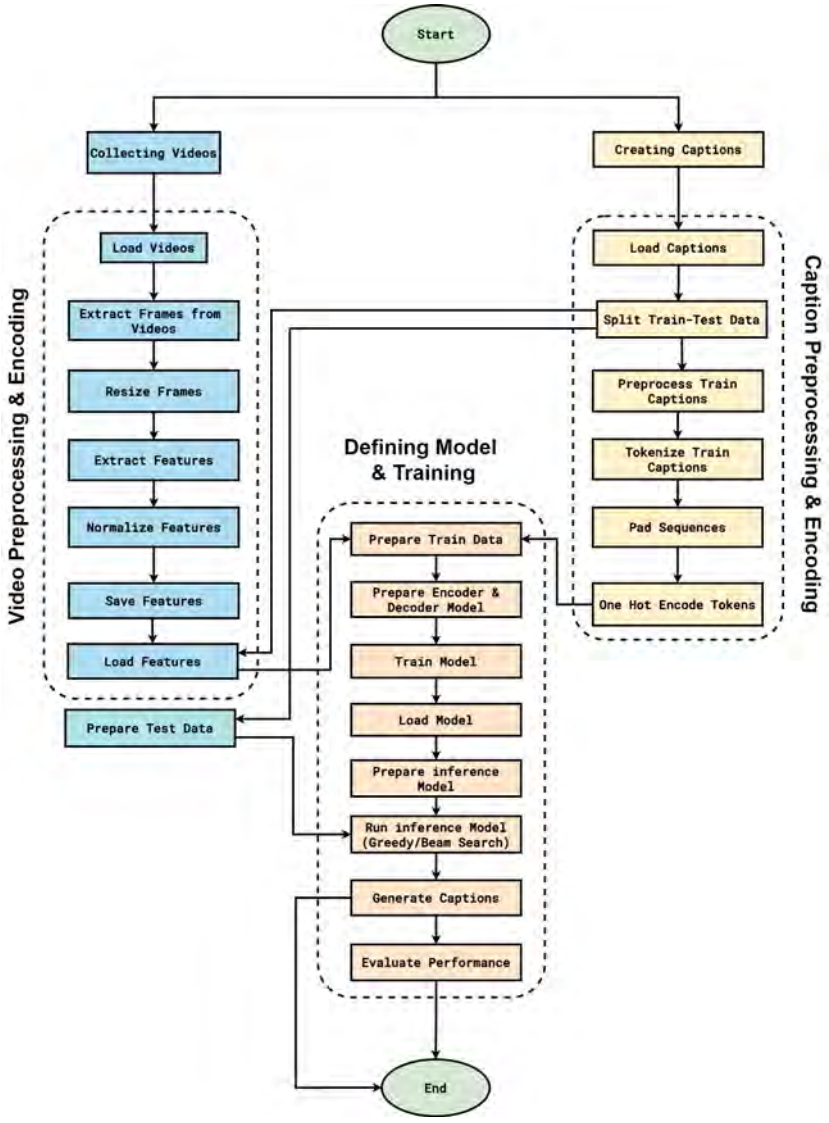


Figure 4.1: Top Level overview of our Proposed system

Our video captioning model employs a sophisticated sequence-to-sequence architecture, integrating multimodal feature extraction with advanced attention mechanisms to generate accurate and contextually relevant captions. The model comprises two main components: an encoder and a decoder. The encoder processes the input video data to generate a comprehensive context vector, which the decoder then uses to produce the output captions. Below, we delve into the specifics of the encoder, beginning with the process of input encoding. This includes the following types of encoding: video encoding, and sequence encoding.

4.1 Background on Recurrent Neural Networks and Attention Mechanisms

Sequence modeling tasks require the ability to process and understand sequential data effectively. Recurrent Neural Networks (RNNs) have emerged as a powerful class of neural architectures capable of capturing and modeling temporal dependencies within sequences. However, vanilla RNNs often struggle with capturing long-range dependencies due to the vanishing and exploding gradient problems. To address these limitations, advanced RNN architectures like Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks have been proposed, introducing gating mechanisms and memory cells to selectively retain and forget information.

While RNNs have demonstrated remarkable success, their performance can be further enhanced by incorporating attention mechanisms. Attention mechanisms allow the model to dynamically focus on the most relevant parts of the input sequence when generating the output. Several attention mechanisms, including Bahdanau Attention, Luong Attention, and Gaussian Attention, have been proposed and integrated into various sequence-to-sequence models, leading to significant performance improvements.

This section explores the theoretical foundations and practical implementations of RNNs, their variants (LSTM, BiLSTM, GRU, BiGRU), and attention mechanisms (Bahdanau, Luong, Gaussian), essential for understanding the architectural components employed in our proposed model.

4.1.1 LSTM in Video Captioning

LSTM or Long Short-Term Memory networks have been by and large utilized for video captioning undertakings considering their capacity to show model temporal sequences and maintain long-term dependencies. Concerning video captioning, LSTM networks are utilized to convey descriptive sentences for video outlines. The fundamental improvement of a LSTM unit solidifies entrances that control the improvement of data:

$$
\begin{aligned}
i_t &= \sigma(W_{iy}y_t + U_{ih}h_{t-1} + b_i) \\
f_t &= \sigma(W_{fy}y_t + U_{fh}h_{t-1} + b_f) \\
o_t &= \sigma(W_{oy}y_t + U_{oh}h_{t-1} + b_o) \\
g_t &= \phi(W_{gy}y_t + U_{gh}h_{t-1} + b_g) \\
m_t &= f_t \odot m_{t-1} + i_t \odot g_t \\
h_t &= o_t \odot \phi(m_t)
\end{aligned}
$$

These equations describe how the input gate (i_t), forget gate (f_t), output gate (o_t), and cell state (m_t) interact to manage the flow of information through the LSTM network.

In video captioning, LSTMs are frequently joined with attention mechanisms like self-attention and Bahdanau attention to improve performance. These attention mechanisms permit the model to zero in on unambiguous pieces of the video approaches that are generally pertinent to the ongoing word being produced. For instance, self-attention assists the model with thinking about the whole succession of casings and specifically centers around significant edges. Bahdanau attention computes a setting vector as a weighted amount of all encoder stowed away states, underscoring the most important casings for each word in the subtitle [24], [25].

Despite these improvements, LSTM-based models frequently display normal execution in video subtitling assignments. The essential difficulties include:

1. **Long-Reach Dependencies:** LSTMs at times struggle to capture long-range dependencies in video sequences, leading to less accurate captions for videos with complex temporal relationships.
2. **Computational Complexity:** The combination of LSTM networks with attention mechanisms increases computational complexity, which can be a bottleneck for real-time applications.
3. **Contextual Limitations:** LSTMs process sequences in a single direction (either forward or backward), which can limit their ability to fully understand the context of each frame.

4.1.2 BiLSTM in Video Captioning

To address these limitations, Bidirectional Long Short-Term Memory (BiLSTM) networks are used for video captioning. BiLSTM networks further develop the standard LSTM design by taking care of progressions in both forward and backward directions. This bidirectional dealing licenses the model to get a more complete setting of the video frames. The BiLSTM unit is portrayed by two equivalent LSTMs:

$$
\begin{aligned}
\overrightarrow{h_t} &= \text{LSTM}(y_t, \overrightarrow{h_{t-1}}) \\
\overleftarrow{h_t} &= \text{LSTM}(y_t, \overleftarrow{h_{t+1}}) \\
h_t &= [\overrightarrow{h_t}; \overleftarrow{h_t}]
\end{aligned}
$$

Here, $\overrightarrow{h_t}$ and $\overleftarrow{h_t}$ represent the forward and backward LSTM outputs, respectively, and h_t is the concatenated output that captures information from both directions [26], [27].
BiLSTM networks, when combined with attention mechanisms like self-attention and Bahdanau attention, can essentially furthermore cultivate video captioning execution. The idea instruments assist the BiLSTM with appearing to zero in on the principal pieces of the video frames, inciting more definite and clear subtitles. The advantages of utilizing BiLSTM include:

1. **Enhanced Setting Centered Understanding:** By managing sequences in both directions, BiLSTM networks can more readily capture the relationships between frames, resulting in more contextually accurate captions.

2. **Improved Performance:** BiLSTM models have been shown to outperform traditional LSTM models in various video captioning benchmarks, achieving higher BLEU and METEOR scores. This indicates better alignment with human-produced captions.

3. **Robustness to Normal Variations:** The bidirectional nature of BiLSTM allows the model to handle videos with complex temporal patterns more effectively, leading to improved reliability in diverse video captioning tasks.

All things considered, BiLSTM networks address a significant advancement in standard LSTM models for video captioning offering better execution and a more complete comprehension of video content.

4.1.3 GRU in Video Captioning

For captioning videos, GRU is used since recurrent systems like GRU are useful when dealing with sequential data such as frames in a video or text sentences to generate a corresponding video caption. The Gated Recurrent Units (GRUs) help eliminate some of the complexities of the gating mechanism of LSTM units while making them significantly faster to train and overcome the vanishing gradient problem. In their application, they are usually used in conjunction with attention structures including self-attention, Bahdanau attention, as well as Gaussian attention to shift the focus on parts of the video frames most useful for the task at hand. For example, Bahdanau attention allows the model to weigh the importance of different frames dynamically: For example, Bahdanau attention allows the model to weigh the importance of different frames dynamically:

$$
\begin{aligned}
z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]) \\
r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\
\tilde{h}_t &= \tanh(W \cdot [r_t \odot h_{t-1}, x_t]) \\
h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t
\end{aligned}
$$

Where z_t is the update gate, r_t is the reset gate, and $\tilde{h}_t$ is the hidden state. These gates control the flow of information, allowing the GRU to maintain long-term dependencies without the complexity of LSTMs.

In their application, GRUs are often combined with attention mechanisms such as self-attention, Bahdanau attention, or Gaussian attention to focus on relevant parts of the video frames. For example, Bahdanau attention allows the model to weigh the importance of different frames dynamically:

$$
\begin{aligned}
e_{tj} &= \text{score}(h_{t-1}, h_j) \\
\alpha_{tj} &= \frac{\exp(e_{tj})}{\sum_{k=1}^{n} \exp(e_{tk})} \\
c_t &= \sum_{j=1}^{n} \alpha_{tj} h_j
\end{aligned}
$$

In the video annotation test, this provides a small average gain to the GRU-based models but is unable to properly convey the temporal relationships in most videos. It appears that a more elaborate design is required to get more performance containing at the very least a BiGRU [28], [29].

4.1.4 BiGRU in Video Captioning

Another technique that incurs much resource expense is Bidirectional Gated Recurrent Units, commonly known as BiGRUs. This functionality is used in video captioning such that the system will give better outcomes. BiGRUs will do better than the standard GRUs, as they process in both the forward and backward directions of the sequences for better context. The working of the unit BiGRU is as follows:

$$
\begin{aligned}
\overrightarrow{h_t} &= \text{GRU}(x_t, \overrightarrow{h_{t-1}}) \\
\overleftarrow{h_t} &= \text{GRU}(x_t, \overleftarrow{h_{t+1}}) \\
h_t &= [\overrightarrow{h_t}; \overleftarrow{h_t}]
\end{aligned}
$$

When used in combination with other attention mechanisms such as self-attention modules or Bahdanau attention, BiGRU can perfectly pinpoint important segments of the video which can lead to more fluent and meaningful captions. The introduction of the attention mechanism gives the capacity of the model to give general prominence to relevant parts of the video frames as well as helps in improving the coherence and quality of the generated captions. For example, in a particular study,

the formulation of the BiGRU model, proposed by the authors, demonstrated improved BLEU and METEOR results compared to traditional GRU models, using the YouTube2Text and MSR-VTT datasets. This improvement further proves that BiGRU outperforms the previous models in terms of addressing the temporal relations in the video data and arriving at clearer descriptions of the video content. In captioning language generation, BiGRU has therefore shown to have higher performance than GRU due to bidirectional processing that involves attention mechanisms [30], [31].

In traditional encoder-decoder models, the hidden state of the decoder at each time step only depends on the hidden state of the encoder at the previous time step

which results in the cause of limiting the model's ability to capture long-range dependencies in the input sequence, as the information from earlier parts of the sequence is forgotten. This problem was mitigated using Bahdanau Attention and Gaussian Attention. These two focus on calculating the attention weights based on the similarity between the current hidden state of the decoder and all hidden states of the encoder, Resulting in the model capturing long-range dependencies more effectively, further improving the performance of video caption generation models.

4.1.5 Bahdanau Attention

Bahdanau attention, also known as additive attention, is a type of attention mechanism introduced by Dzmitry Bahdanau et al.[32]. It's a way to allow the model to focus on certain parts of the input sequence when generating the output sequence. The attention weights are calculated using a neural network with a softmax output layer. Bahdanau Attention can be used to allow the decoder to focus on different parts of the input sequence at each time step. It allows the decoder to attend to the entire input sequence, capturing context and addressing the issue of information loss in the fixed-length context vector as a result we get more accurate the generated captions, especially for long sequences. Bahdanau Attention is a popular choice for encoder-decoder architecture that uses GRU, BiGRU, LSTM, or BiLSTM as it allows the model to focus on specific frames while generating each word, which is beneficial for video captioning .

4.1.6 Gaussian Attention

Gaussian attention is a type of attention mechanism that uses a Gaussian distribution to compute the attention weights allowing the model to focus on specific regions of the input sequence. It's similar to Bahdanau attention, but instead of using a softmax output layer, it uses a Gaussian distribution to compute the weights when the attention weights need to be continuous as Gaussian Attention computes a soft window over the encoder states using a Gaussian kernel. However, even after its limited ability to handle complex temporal dependencies in video captioning, for exceptional reasons, gives the best results, such as Dataset characteristics when the video sequences have a predictable structure or if the relevant frames are usually within a certain window around the current frame, Gaussian Attention perform better. If the relevant information in your videos is typically concentrated within a specific temporal window, Gaussian Attention's localized focus might be more effective as this method uses attention over continuous spatial locations in images or frames.

In summary, Understanding the characteristics of your data and model, Bahdanau Attention is useful when combined with GRU/LSTM encoders for its fine-grained focus on relevant video frames, in addition, though Gaussian Attention is generally less common for video captioning, it can outperform other mechanisms in specific scenarios. It is necessary to add that Self-Attention used in Transformer models which is indeed computationally intensive, requires significant memory for complexity [33].

4.2 Model Architecture

After evaluating various combinations of these recurrent neural network architectures and attention mechanisms, we found the GRU-Gaussian model exhibited the best performance for our video captioning task. This model effectively balances computational efficiency and accuracy by combining the GRU's ability to capture temporal dependencies with the Gaussian attention mechanism's focused yet flexible attention.

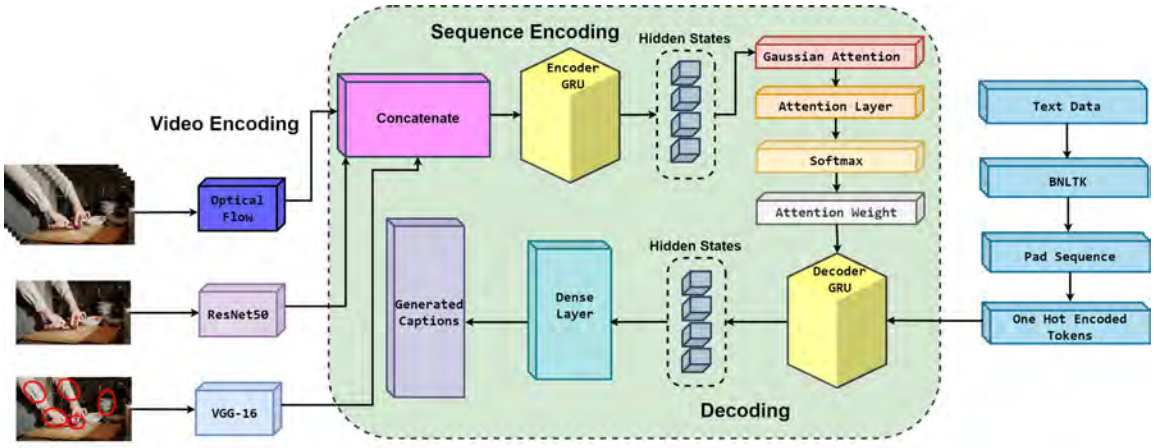


Figure 4.2: Model Architecture

The encoding part includes video encoding and recurrent encoding. Inception, ResNet50, and VGG16 are among the pre-trained models and also optical flow algorithms that have been used to extract features from video frames which is the main part of the video encoding section. Temporal information is captured using the GRU layer after extracting relevant features for the Recurrent Encoding phase.Encoding involves using an attention mechanism that creates context vectors after which captions are generated by the GRU decoder depending on what input has been fed and the encoded features.

4.2.1 Video Encoding

The process of video encoding involves taking detailed attributes from the video frames with pre-trained models and some advanced algorithms. Having earlier given insight on how the video frames were preprocessed in this paper, I will now identify how these pre-trained models like ResNet50 or VGG16 extract the features. Our global feature extraction is based on ResNet50 and InceptionNetV3, which help to gather information about the whole video segment, while local features are captured by VGG16 for more detailed analysis and the optical flow algorithm does the motion feature extraction.

InceptionV3

In our research, we are going in-depth into the Inception architecture, a creative design of CNN architecture proposed by Szegedy et al. in the year 2015. Inception proposes a new design of CNN with an inception module that captures multiscale

features within one layer by applying multiple convolution filters of different sizes at the same time.

[34] This remarkably raises the ability of the network to recognize complicated patterns and structures in the data, hence very useful in the case of detailed image representations. In contrast to the typical CNNs, the architecture of Inception makes it compute effectively and extract better features owing to parallel paths. In our video captioning task, we leveraged the Inception model for fine-tuning custom feature extraction. We removed the last layers of the pre-trained model and made the network fine-tune our dataset composed of video frames. We aimed to adapt the model in a manner that the extracted features would help generate exact video captions. We opted for the Inception model to extract global features because its inception modules effectively capture high-level, multiscale features from the video frames, giving an overall sense of the context.

We added a Global Average Pooling layer on top of the base model to reduce the spatial dimensions of the feature maps, making the features more suitable for further processing in the pipeline of generating captions. Though Inception has one of the most sophisticated architectures and can potentially extract features efficiently, it performed dismally on our dataset. Evaluations using our metrics of choice showed that the results were not as expected and posed challenges in customizing complicated models for specific tasks.

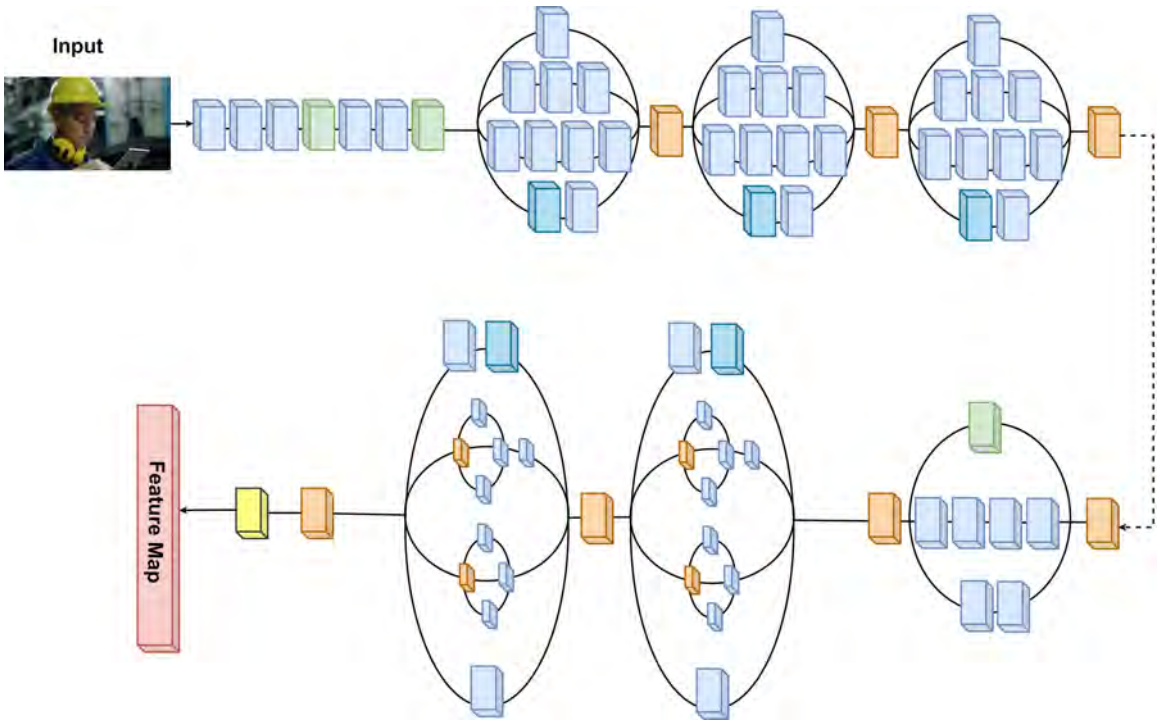


Figure 4.3: InceptionV3

ResNet50

Upon working with InceptionV3, we found many issues in getting the desired metric scores in video captioning. Looking for solutions, we tried other architectures and

fine-tuning techniques, finally settling upon ResNet50. Its high accuracy in feature extraction made it look good to use. Similar to InceptionV3, the pre-trained convolutional layers of ResNet50 produced a strong foundation for transfer learning, thus making us adapt the model to our needs and take advantage of the strong feature extraction power in this model.

ResNet50 is a model created for image classification, so this model was made to adapt to video captioning, which needs an attached language generator module [35]. We removed the last fully connected layers and replaced them with a global average pooling layer to decrease the spatial dimension of the feature maps. This modification then helped us to capture the full power of feature extraction from ResNet50 and to generate full and contextually relevant captions for video frames. These modifications helped match the extracted features well with our task of video captioning and enhanced the interpretation of video content.

We have chosen ResNet50 for global feature extraction, as it can capture high-level features in a much more efficient way through its bottleneck modules. These bottleneck modules help to represent the information well within fixed computational resources; hence, ResNet50 is quite beneficial in resource-constrained situations. Additionally, skip connections in ResNet50 allow for a direct flow of information. This helps accelerate the processing of information through the network. This is very essential for video captioning, where the main task is to capture the most minute and relevant details.

Despite these modifications and the advanced architecture of ResNet50, our results from our chosen metrics indicated that the challenges of adapting complex models to specific tasks are always there, and the results were below expectations. Nonetheless, the use of ResNet50 for global feature extraction is still a strong choice due to its ability for powerful and efficient feature representation.

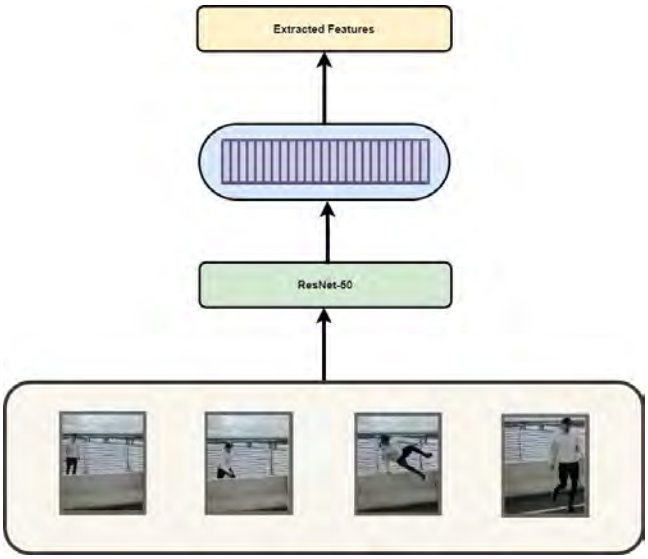


Figure 4.4: ResNet50

VGG16

The profound and hierarchical architecture of VGG16 makes it capable and appropriate for capturing local features in video captioning, being able to extract crucial visual details with high efficiency. VGG16 achieves efficient feature extraction using up to five consecutive 3x3 convolutional filters in each stage, which avoids larger and computationally expensive filters. Following each convolutional stage, the 2x2 max pooling shrinks the size of the feature maps and provides spatial invariance, thus making resilient features. The last layer of VGG16 consists of three fully connected layers, which usually predict diverse categories of images.

[36] However, we fine-tuned these layers for video captioning by discarding the last fully connected layers and attaching a Global Average Pooling layer followed by a dense layer of 4096 units and ReLU activation. It is modified to tune the model to extract local features precisely pertinent to our video captioning task. The VGG16 algorithm is good at capturing visual features, such as edges, textures, and shapes, that are comprehensive and of great value as a feature extractor for a lot of computer vision tasks other than classification tasks. It is also very apt for transfer learning because, for the same computer vision problem, we need to detect the object and generate captions. The final layers can be customized, whereas the rest of the layers can be retained as such. In the case of video captioning, local feature extraction by VGG-16 captures fine-grained details of video frames, exploiting knowledge from a large dataset by using the pre-trained weights for our task. We finetuned this model for our specific task. It will have an enhancement in the working of the model, leading to more efficient and accurate feature extraction in the frames and thus effectively dealing with the data constraints, with an increase in the model's efficiency in generating meaningful and contextually relevant captions. This process ensures that the rich details essential for generating high-quality video captions are taken up and used.

4.2.2 Sequence Encoder

The encoder begins with the processing of the three different types of input features: global, motion, and local. Each kind of feature is of fundamental importance and depicts other dimensions of the video data. The global features reflect high-level semantic information; the motion features reflect dynamic changes with time dimensions, while the local features focus on detailed information within the frames of the video. Each type of feature is fed to a different input layer and these are represented as tensors, denoted by 'encoder_inputs_local'. These inputs are represented as tensors, denoted by X_{global}, X_{motion}, and X_{local} respectively. Each type of feature is processed through separate input layers, ensuring proper formatting and shaping for subsequent computations.

The encoder uses a strategy that concatenates these three modalities. All three input feature tensors are joined along the feature axis, developing a unified representation of the input data. This can be mathematically described as follows:

$$X_{\text{concat}} = \text{Concatenate}(X_{\text{global}}, X_{\text{motion}}, X_{\text{local}})$$

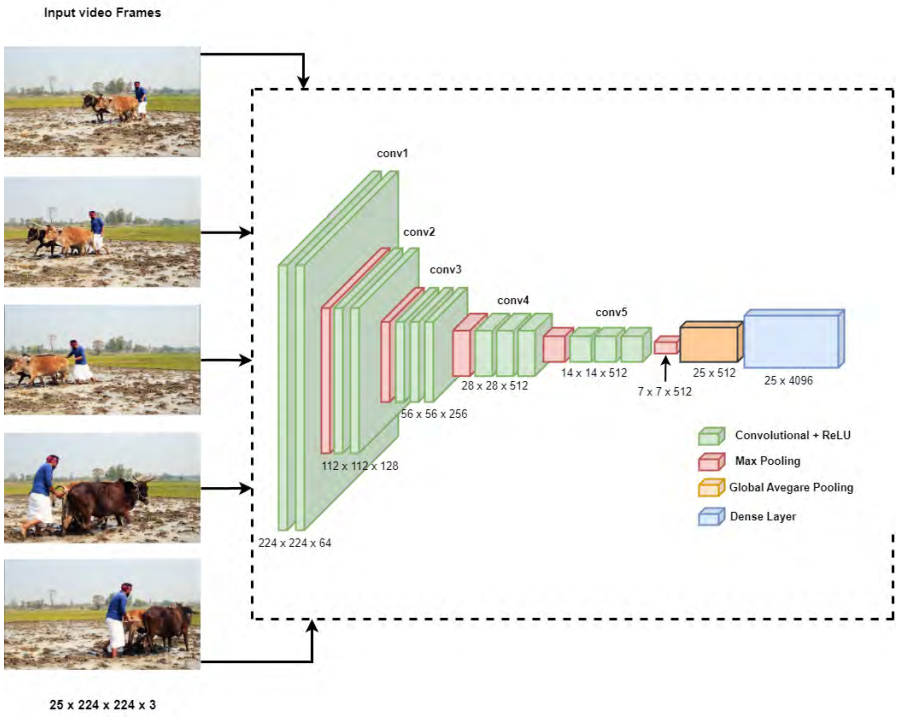


Figure 4.5: VGG16

This comprehensive tensor, X_{concat}, integrates diverse information, enabling the encoder to consider all relevant aspects of the video input. This tensor is then fed into a Gated Recurrent Unit (GRU) layer, a variant of the recurrent neural network (RNN) designed to capture temporal dependencies in sequential data while mitigating the vanishing gradient problem. This layer processes the input sequence and generates a sequence of hidden states H_{enc} and the final hidden state h_{enc}. This process can be formulated as where H_{enc} represents the sequence of hidden states, and h_{enc} is the final hidden state.

$$H_{\text{enc}}, h_{\text{enc}} = \text{GRU}(X_{\text{concat}})$$

The GRU operates through a series of gating mechanisms—update gate z_t, reset gate r_t, and candidate activation $\tilde{h}_t$—which control the flow of information and updates to the hidden states:

$$\begin{aligned} z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]) \\ r_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\ \tilde{h}_t &= \tanh(W \cdot [r_t \odot h_{t-1}, x_t]) \\ h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \end{aligned}$$

To improve the model's ability to focus on the most relevant parts of the input sequence, a Gaussian attention mechanism is applied to the sequence output H_{enc}. Unlike Bahdanau attention, which uses learned weights to compute alignment scores, Gaussian attention employs a probabilistic approach to model the focus points over the input sequence. The attention mechanism involves calculating a set of attention weights α_t, which are applied to the hidden states to produce a context vector C.

1. **Score Calculation:**
$$e_t = W \cdot H_{\text{enc}} + b$$
where W is a learned weight matrix, and b is a bias vector.

2. **Attention Weights:** For each time step t, the attention weights α_t are computed based on the Gaussian distribution centered around where e_t are the raw attention scores:
$$\alpha_t = \frac{\exp(e_t)}{\sum_k \exp(e_k)}$$

3. **Context Vector:** The context vector C is calculated as a weighted sum of the hidden states H_{enc}, using the Gaussian attention weights α_t:
$$C = \sum_i \alpha_{t,i} \cdot H_{\text{enc},i}$$

This would mean taking advantage of global, motion, and local features to encode, aided by the attention mechanism in video captioning. Video data on its own is complex and multifaceted; the model needs to capture a wide variety of patterns both in the time and space domains. Such combined capabilities of the GRU layer for modeling temporal dependencies and attention mechanisms focusing on relevant information help to make the encoder produce a rich and informative context vector. This is what the decoder needs for generating exact and relevant captions. The architecture of the encoder assures that the model processes input video data effectively and summarizes it, forming a sound base for the generation process.

Briefly, the video encoder in this sequence-to-sequence model of video captioning is a very complex component, as it makes all these very different features of videos coalesce into one coherent, informative context vector that is able to produce high-quality and contextually plausible video captions. Hence, the architecture requires all kinds of feature combinations and advanced mechanisms, including attention, to handle the complexity of the video data.

4.2.3 Decoder

In the video captioning seq2seq model, the decoder will simply produce an output sequence, that is, the video caption set, using the context vector that was derived from the encoder. This section of the paper proceeds with an extensive exposition of the decoder architecture: the elaborate set of steps, mathematical formulations, and, finally, the importance of generating coherent and contextually relevant video captions.

Input Layer and Initial State The decoder begins with an input layer that accepts the decoder input sequences. These sequences, denoted as $Y_{\text{dec\_input}}$, typically consist of previously generated tokens from the output vocabulary and have been pre-processed to a fixed length using padding. The input layer is designed to handle tensors of the form (N, T,D), where N is the batch size, T is the sequence length, and D is the dimensionality of the input tokens.

A crucial aspect of the decoder is the initialization of its hidden state. The initial hidden state is set to the context vector C obtained from the encoder. This context vector encapsulates the essential information from the input sequence, providing the decoder with a rich summary to guide the generation process. Formally, the initial hidden state $h_{\text{dec},0}$ is given by:

$$h_{\text{dec},0} = C$$

The decoder also contains a Gated Recurrent Unit layer, which handles input sequences and gives off a sequence of hidden states. The GRU layer basically comprises a few gating mechanisms. Also, in our model, a Gated Recurrent Unit is implemented within the decoder, just like in the encoder. First, an initial hidden

state in the decoder GRU is formed by using the context vector of the encoder. It predicts the output sequence in a step-by-step manner. In every time step, it uses the previous output token fed into it and the current hidden state for the prediction of the next output token. Mathematically, the equations of the decoder GRU remain roughly the same as that of the encoder GRU, with slight modifications considering the context vector and the previous output token. The equations that govern the decoder GRU are as follows:

1. Update Gate:
$$u_t = \sigma(W_u \cdot y_{t-1} + R_u \cdot h_{t-1} + b_u)$$
2. Reset Gate:
$$r_t = \sigma(W_r \cdot y_{t-1} + R_r \cdot h_{t-1} + b_r)$$
3. Candidate Hidden State:
$$\tilde{h}_t = \tanh(W_h \cdot y_{t-1} + r_t \odot (R_h \cdot h_{t-1}) + b_h)$$
4. Hidden State:
$$h_t = u_t \odot h_{t-1} + (1 - u_t) \odot \tilde{h}_t$$

These are further broken down as: where Ru is the weight matrix for the update gate that helps compute the amount of past information to keep. Rr is the weight matrix for the reset gate that contributes to the computation of the amount of past information to forget; Rh is the weight matrix that interacts with the previous hidden state after modulation by the reset gate to make up the candidate hidden state. These are used in conjunction with the weight matrices Wu, Wr, Wh which make the connections from the current input y_{t-1} to each of the respective gates

and candidate states. In the case of a decoder, we set the architecture with a feedback loop as the output at each time step, which becomes the input at the next time step so that the model can predict sequences based on its previous outputs. This is fundamentally important in captioning tasks because, much of the time, the structure of the sequence in the target language has to be built over time. Note that the process we have described is conditional upon a stopping criterion, usually either a maximum sequence length or by producing an end-of-sequence token.

The decoder features a feedback loop where the output of each step is fed back into the model as input for the next step, facilitating sequence generation based on previous outputs. This loop is crucial in tasks like captioning where the sequence structure in the target language needs to be built incrementally. The output sequence is generated by repeating this process until a stopping criterion is met, such as a maximum sequence length or the generation of an end-of-sequence token.

The input sequence for GRU $Y_{\text{dec\_input}}$ and the hidden state sequence that is generated H_{dec} and the final hidden state h_{dec}. This process is formulated as:

$$H_{\text{dec}}, h_{\text{dec}} = \text{GRU}(Y_{\text{dec\_input}}, \text{initial\_state} = C)$$

Later on, we added a dropout layer after the output from the decoder GRU to regularize the network and avoid overfitting. The dropout layer blanks a fraction of the output values randomly to drive the model to learn much more robust representations. The transformed sequence is then fed into a dense layer containing the sigmoid activation of the output. Therefore, this dense layer serves as an FC layer, which maps the hidden states into the desired output space to arrive at a final output sequence. As a result, the output goes from 0 to 1 by the sigmoid activation and is thus proper for classification. The output of the dense layer can be formulated as:

$$Y_{\text{dec\_output}} = \text{Dense}(H_{\text{drop}}, \text{activation} = \text{sigmoid})$$

The decoder architecture is pivotal for generating accurate and contextually relevant video captions. By initializing with the context vector from the encoder, the decoder is equipped with a comprehensive summary of the input video data. The GRU layer's ability to capture temporal dependencies ensures that the generated captions maintain coherence over time. Additionally, the attention mechanism's focus on relevant parts of the input sequence enhances the contextual accuracy of the captions.

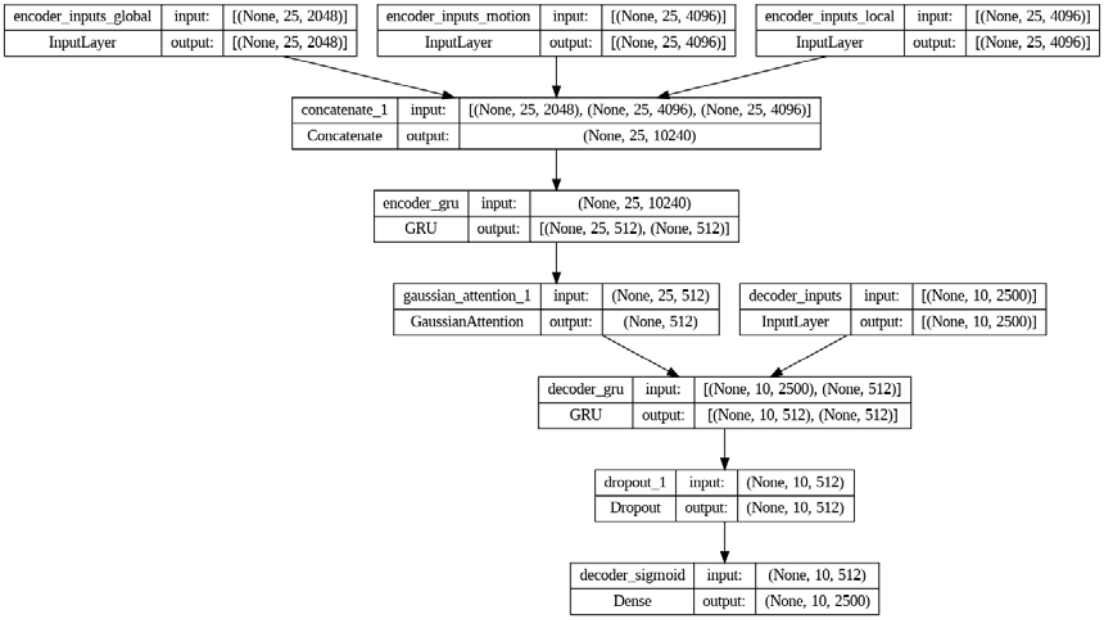


Figure 4.6: Layer Architecture of the Proposed Model

Chapter 5

Result Analysis

5.1 Experimental Setup

We performed all our experiments and model training on Ubuntu 22.04 and Python 3.9.19, utilizing an NVIDIA GeForce RTX 3070 GPU to accelerate computations and ensure efficient training. TensorFlow 2.15.1, Keras 2.15.0 were the main libraries for our video captioning model. Some other supporting libraries include numpy 1.25.2, scikit-learn, matplotlib, pandas and more. The github repository of the "Pycoco-evalcap" library were also used for the need to measure evaluation metrics such as BLEU, METEOR, and CIDEr scores which provided a comprehensive assessment of how well the model performed in generating accurate and contextually relevant captions

5.2 Hyperparameters

The only determined important major hyperparameters when training the video captioning model are batch size and epoch, where 64 and 25, respectively, are found to find a trade-off between training efficiency and model performance. The latent dim was set to 512, which gave the model the ability to represent complex patterns in data. Then, the number of encoder tokens was set to 2048 and the number of decoder tokens to 2500; that gives a robust vocabulary to process and generate captions. The encoder considered sequences of 25 time steps, and for the decoder, the number of time steps was 10. Learning rate hyper-tuning was done to find the

best setting for the training process. Various learning rates were tried out in order to tune them efficiently and set the rate to 0.0003. This rate provided a balance between the convergence speed and stability of training, ensuring that the model learned effectively without overfitting or underfitting. This was a very careful process involving iterative testing and evaluation in order to produce the best performance for video captioning.

5.3 Model Performance

.

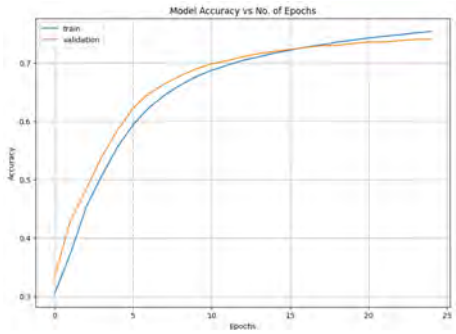


Figure 5.1: Model Accuracy vs. No. of Epochs for GRU+Gaussian

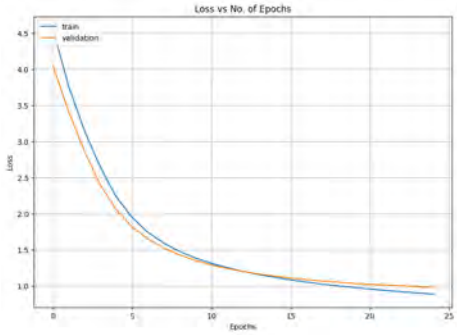


Figure 5.2: Loss During Training for GRU+Gaussian

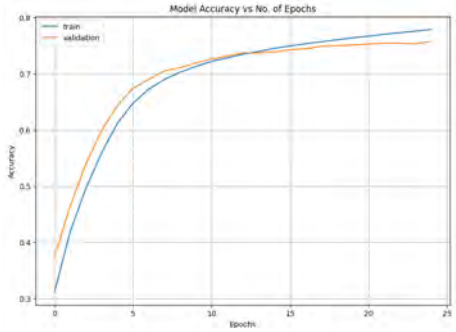


Figure 5.3: Model Accuracy vs. No. of Epochs for GRU+Bahdanau

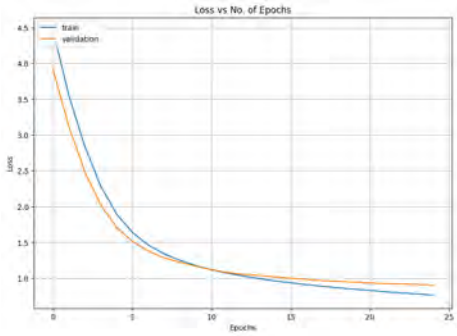


Figure 5.4: Loss During Training for GRU+Bahdanau

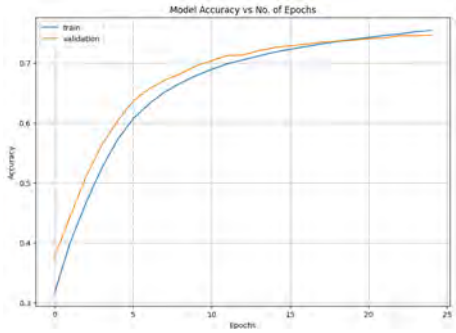


Figure 5.5: Model Accuracy vs. No. of Epochs for LSTM+Bahdanau

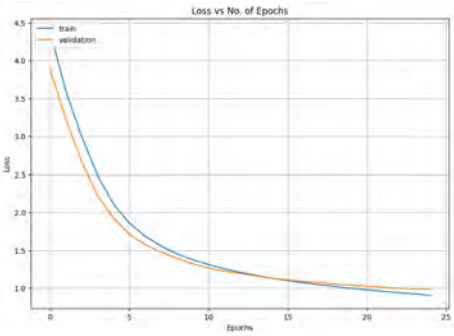


Figure 5.6: Loss During Training for LSTM+Bahdanau

.

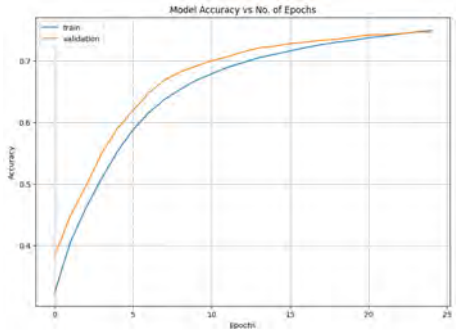


Figure 5.7: Model Accuracy vs. No. of Epochs for LSTM + Gaussian

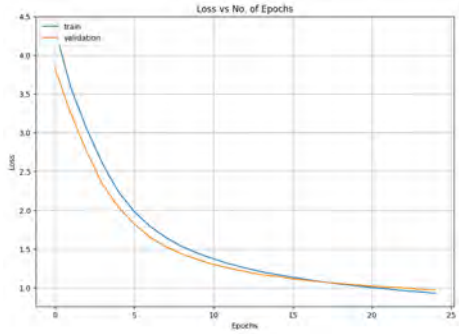


Figure 5.8: Loss During Training foe LSTM + Gaussian

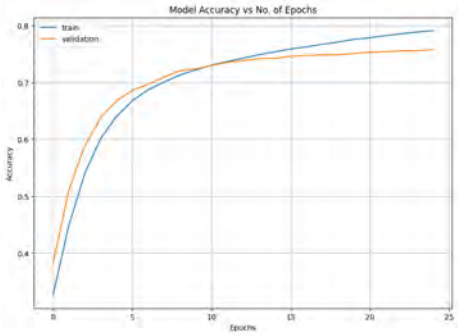


Figure 5.9: Model Accuracy vs. No. of Epochs for BiGRU+Gaussian

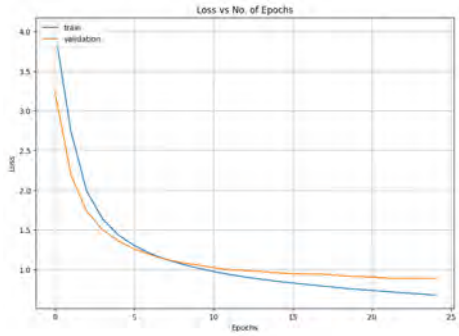


Figure 5.10: Loss During Training for BiGRU+Gaussian

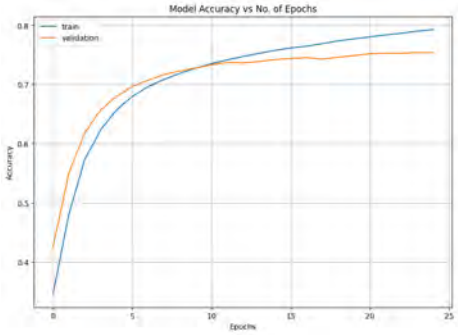


Figure 5.11: Model Accuracy vs. No. of Epochs for BiLSTM + Gaussian

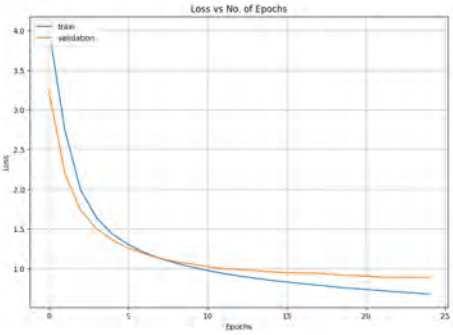


Figure 5.12: Loss During Training for BiLSTM + Gaussian

5.3.1 Overview of the graphs

Model accuracy vs. No. of Epochs graph indicates that the graph is plotting the accuracy of a model against the number of training epochs demonstrating a typical training process where the model improves its performance on both training and validation sets.The x-axis represents the number of epochs, which are iterations over the entire training dataset. The range is from 0 to 25 epochs. The y-axis represents the accuracy of the model, which measures how often the model's predictions are correct. The range starts from around 30% to close to 80%.

It is used in tasks such as video captioning by most major generative AI models. Accuracy here is obtained by comparing the model's predictions to the real data during training. It turns the token streams into a one-hot-encoded vector of target distributions for each point in time during training. It predicts these distributions and measures the difference between the categorical cross-entropy loss function priority and the model's output probability distributions with the real target distributions. It derives in this respect accuracy from the determining factor, whether the class at the time with the highest probability predicted from the model is equal to the actual class or not. This accuracy per batch is averaged across samples, thus giving a measure on how well the model is able to predict veracious sequences. This guarantees a measure of accurate text appropriateness in context during the generation of coherent captions based on input data.

Firstly, Training Accuracy increases steadily throughout the epochs, indicating that the model is learning from the training data. When it does not decrease suggests there is no overfitting occurring within this epoch range. Next, the validation accuracy initially increases faster than the training accuracy, which is a good sign of the model generalizing well. The closeness of the training and validation accuracy curves implies that the model is not overfitting significantly. Here, in the accuracy graph of Bi-GRU and Bi-LSTM, overfitting is noticeable as the lines are moving away. In addition to that, the slight lead of validation accuracy over training accuracy in the early epochs suggests a robust model, but as training continues, the training accuracy surpasses the validation accuracy slightly, indicating potential overfitting if training continues further. Moreover, here, the plateau of the validation accuracy indicates the model's maximum performance given the current data and model configuration.

5.4 Generating Caption

To generate a caption, the essential components include the model, video feature vector, tokenizer, and the maximum length of the caption.

5.4.1 Beam search

The key part of beam search is that the sequences are produced by exploring several optional results at each time step, and all but those with higher probabilities are discarded at all but the highest probability retaining states. It initializes an initial target sequence and states, iteratively predicts the next token, and updates the

sequence based on probabilities. It keeps the multiple paths active using the top nodes with the highest probabilities at every point and further uses a recursive exploration of each node formed until a whole sequence is produced or the maximum length is reached. The technique explores at a deeper level the likely outputs, hence making it most likely to generate more correct and contextually relevant descriptions.

5.4.2 Greedy search

Greedy search's primary is it passes a target sentence, which is initialized to the beginning of a sentence, and then iterates to predict the next token such that the target sequence is appended with the token each time it observes a token with the maximum probability. It does so with the help of a predetermined number of tokens so that it does not go on for an infinite duration or when a predetermined end token is reached. The difference is that beam search will be able to follow more paths at a time, while greedy search will just follow a single course based on immediate probabilities, and that is why it is faster to perform yet perhaps less accurate. The method then returns the generated description for the input video in the form of the decoded sequence.

Lastly, the test method takes care of the overall testing of the system. It loads the test data by concatenating the global, motion, and local features of each test video. Beam search or greedy search only one is called for generating text for every test instance but for our advantage we used greedy search. Then the predictions are stored in an output file in such a way that every line contains one video and its corresponding generated description. This would be a comprehensive approach to evaluate the performance of the developed decoding strategies on unseen data, giving insight into the capabilities of the model to generate meaningful and correct descriptions of video content.

5.5 Generated Captions

Figure 5.13: Sample Result 1

Ground Captions:-

- কচ্ছপ সমুদ্রের নিচে একটি প্রবালে বসে ছিল।
- সাঁতার কেটে কচ্ছপ প্রবাল ছেড়ে গেল।

Generated Caption:-

- সাঁতার কাটছে অনেকগুলো কচ্ছপ ও তার উপর আলো।

Figure 5.14: Sample Result 2

Ground Captions:-

- পিচের রাস্তায় সাদা-কালো বিড়াল সাদা বাটি থেকে খাবার খাচ্ছে।
- রাস্তায় সাদা বাটি থেকে খাচ্ছে বিড়াল।

Generated Caption:-

- সাদা বাটি থেকে খাবার খাচ্ছে সাদা বিড়াল।

Figure 5.15: Sample Result 3

Ground Captions:-

- পানিতে সাদা টুপি পরা লোক ফসল লাগাচ্ছে।
- সাদা টুপি পরা একজন লোক পানির মধ্যে দাঁড়িয়ে ফসল রোপণ করছে।

Generated Caption:-

- সবুজ ঘাসের মাঝে দাঁড়িয়ে আছে কৃষক।

Figure 5.16: Sample Result 4

Ground Captions:-

- সাদা জামা পরা ছেলে সাইকেল নিয়ে র‍্যাম্প থেকে লাফ দিয়ে কৌশল দেখাচ্ছে।
- সাইকেল র‍্যাম্পের উপর থেকে লাফ দেখাচ্ছে।

Generated Caption:-

- কালো শার্ট পরে পুরুষটি সাইকেল চালাচ্ছেন।

5.6 Evaluation Metrics

After training a model it's necessary to evaluate the model to understand if it needs to be improved or if it's working correctly or performing well or not, depending on the problem we choose evaluation metrics. There are several metrics used to evaluate the quality of image or video caption generation models. Here are some of the most commonly used ones:

5.6.1 BLEU

When evaluating how well a model-generated caption matches up with a reference caption, BLEU is a common tool. This will tell us, on average, how many details were retained by the model from the reference and also how accurate it is overall in textual generations. The introductory accuracy of BLEU score ranging between 0-1 claimed that almost 99% human judgment is tracked by it. [37] One of the key benefits accrued to BLUE is the fact that comparisons can be made between the various models or even the captions. However, The BLEU score has some limitations when used for evaluating video captions. One limitation is that it may not accurately reflect the quality of the generated captions if they are semantically correct but use different wording-like similarities than the reference captions. Additionally, the BLEU score may penalize generated captions that are longer than the reference captions, even if they are more descriptive and informative. This can be a particular issue in video captioning, where the generated captions may need to be longer to accurately describe the content of the video. It emphasizes word-for-word resemblance; therefore, it may miss out on some of the more delicate aspects of language, like sentence structure and meaning. Here, c= length of the candidate translation and r = reference corpus length, brevity penalty,BP=penalty:

$$\text{BLEU} = \text{BP} \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \text{[37]}$$

5.6.2 METEOR

METEOR handles the weakness of BLEU by introducing recall with precision and finding correlation with human judgments. It gives the best score based on using word-to-word matching between the machine translation and each reference translation individually independently by taking account of unigram precision and recall, with additional penalties for incorrect word order. Additionally, METEOR considers the order of the words in the generated and reference captions, which can be particularly important in video captioning where the "temporal dynamics" of the input video need to be focused, as METEOR takes into account word frequency, synonyms, and parts of a speech in a bid to give us better indications of semantic similarity, hence its accuracy in the assessment of caption quality is important.

$$\text{METEOR} = F_{\text{mean}} \cdot (1 - p)[38]$$

where F_{mean} is the harmonic mean of precision and recall, and p is a penalty factor.

5.6.3 ROUGE_L Metric

ROUGE_L is a metric used for evaluating the quality of automatic summarization and machine translation, including video captioning. A longer shared sequence indicates more similarity between the two sequences, the machine-generated and given caption. The number of overlapping units such as n-grams, order of single and paired words are evaluated to determine the quality. ROUGE_L is a useful metric for video captioning because it compares the structure similarity and also the longest co-occurring in sequence n-grams. It can provide a more accurate evaluation of the generated captions than metrics that only consider exact word matches.

$$P = \frac{\text{LCS}(X, Y)}{|X|}, \quad \text{where } P \text{ is Precision,}$$
$$R = \frac{\text{LCS}(X, Y)}{|Y|}, \quad \text{where } R \text{ is Recall,}$$
$$F1 = \frac{2 \cdot P \cdot R}{P + R}, \quad \text{where } F1 \text{ is the F1 Score.[39]}$$

5.6.4 CIDEr

A metric that measures the similarity between the machine-generated captions and the human-made captions based on the consensus among human raters. In simple words, It gives higher scores to generated captions that are similar to multiple reference captions rather than just one. It is the weighting of n-grams, which gives more weight to n-grams that are rare in the corpus but frequent in the reference captions which changes the story of the caption. CIDEr is best in the case of correlating well with human judgments, for its claim that it was introduced by keeping in mind how we people describe a frame [40]. Hence, This can be particularly useful for evaluating video captions where the generated captions need to accurately describe the content of the video considering both semantic relevance and diversity.

$$\text{CIDEr}_n(c_i, S_i) = \frac{1}{m} \sum_j \frac{g^n(c_i) \cdot g^n(s_{ij})}{\|g^n(c_i)\| \|g^n(s_{ij})\|}$$

$$\text{CIDEr}(c_i, S_i) = \sum_{n=1}^{N} w_n \text{CIDEr}_n(c_i, S_i)[40]$$

For a video caption generation model, each metric has its pros and cons Therefore, a combination of matrices gives a more comprehensive evaluation.

5.7 Performance Analysis

Table 5.1: Performance Analysis

Models	**BLEU 1**	**BLEU 4**	**Meteor**	**Rouge**	**Cider**
LSTM + Bahdanau	0.43	0.124	0.298	0.336	0.346
LSTM + Gaussian	0.445	0.131	0.31	0.332	0.365
BiLSTM + Bahdanau	0.47	0.139	0.311	0.34	0.41
BiLSTM + Gaussian	0.472	0.14	0.33	0.322	0.42
GRU+ Bahdanau	0.457	0.136	0.332	0.356	0.402
GRU+ Gaussian	**0.516**	**0.154**	0.322	0.386	**0.492**
BiGRU+ Bahdanau	0.472	0.147	**0.356**	0.369	0.45
BiGRU+ Gaussian	0.476	0.15	0.342	**0.385**	0.503

Based on the provided summary of training results, the highest scores for each metric: The GRU + Gaussian model achieves the highest scores in BLEU 1 (0.516), BLEU 4 (0.154), and Cider (0.492). The BiGRU + Bahdanau Attention model achieves the highest score in Meteor (0.356). The BiGRU + Gaussian model achieves the highest score in Rouge (0.385). This indicates that GRU-based models, particularly with Gaussian attention, perform exceptionally well across most metrics, indicating superior performance in generating text that closely matches reference text. While BiGRU models excel in specific metrics like Meteor and Rouge.

The reason for such an exception is the complexity of architecture. To elaborate, GRU architecture is simpler than LSTM and GRUs have fewer gates compared to LSTMs. This simplicity allows GRUs to be more efficient and faster in training and inference, which can be crucial for handling the large amount of sequential data in video captioning. While LSTMs are powerful and effective in capturing long-term dependencies, their complexity can sometimes lead to overfitting, especially with smaller datasets. With them, adding Gaussian attention mechanisms helps the model focus on relevant parts of the input sequence. That means, if the dataset carries the video sequences that have a predictable structure or if the relevant frames are usually within a certain window this attention performs better than in- Single action videos. While both GRU and LSTM can use Gaussian attention, the combination of GRU with Gaussian attention seems to better to work better with the strengths of GRUs in handling sequential but concentrated data on relevant video frames, leading to better performance, whereas LSTMs stayed back due to its inherent complexity and potential training difficulties. At the same time, we expected that BiGRU would perform well due to its fame for capturing richer contextual information by considering both past and future contexts, but again, the increased complexity and computational requirements for its bidirectional processing and the specifics of the video dataset lead to lower performance in video captioning tasks as model fail to reach its full potentials compared to simpler models like unidirectional GRUs.

Chapter 6

Conclusion

In summary, video captioning systems mark another milestone in the effort to close the gap between the explosive growth of digital video content and the felt need for accessible media. We have tried hard in this thesis to cope with a number of challenges in the process of video captioning, particularly in the Bangla language context, which is mostly neglected by the linguistics community as well as significant difficulties due to limited computational resources for video processing, impacting the decision to test different models. Ranging from innovative steps in creating robust datasets and feature extraction techniques, was introduced to further advance the system that will change the user experience. This will have tremendous implications for the improvement in the accessibility of video contents to allow a greater audience to benefit from the media, specifically, the blind. Furthermore, video captioning technology will change the nature of education, entertainment, and information dissemination by making video content more interactive and revealing. Following the best score from our research, combining GRUs with Gaussian attention mechanisms for video captioning offers a powerful approach to improving the generation of coherent and contextually accurate captions. GRUs are advantageous in video captioning due to their simplified architecture compared to the LSTM networks, offering faster training times and reduced computational complexity while maintaining effective performance in managing sequence data. The GRU's ability to mitigate issues such as vanishing gradients and efficiently capture temporal dependencies makes it a better choice for sequential data processing.Furthermore, allowing Gaussian attention mechanisms further enhances this approach by allowing the model to have a continuous and precise focus on specific parts or on key elements within the frames. Thus, for this paper, the combination of GRUs and Gaussian attention leads to significant improvements in video captioning performance due to computational efficiency and the ability to generate contextually rich captions. Clearly, the adoption of video captioning technologies will be crucial to make digital content accessible and enjoyable to all in the future.

Continuing our efforts toward addressing the linguistic and technical challenges of video captioning, we help to bring into existence a more inclusive and knowledgeable world where all can benefit from the knowledge and cultural richness which video content has to offer. This will help us to promise directions for the future by experimenting with transformer-based architectures and bring forth the architec-

tures that rely on video data multimodal learning techniques combined with audio. With that, the scale and diversity of training datasets could be further increased, which would further enhance the capability of video captioning models. We believe that such works could prove very useful to enable a wide range of applications, from increasing the scope of video content summarization resulting in improving the human-machine relationship.

Bibliography

[1] C. Yan, Y. Tu, X. Wang, *et al.*, "Stat: Spatial-temporal attention mechanism for video captioning," *IEEE Transactions on Multimedia*, vol. 22, no. 1, pp. 229–241, 2020. DOI: 10.1109/TMM.2019.2924576.

[2] J. Deng, L. Li, B. Zhang, S. Wang, Z. Zha, and Q. Huang, "Syntax-guided hierarchical attention network for video captioning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 2, pp. 880–892, 2022. DOI: 10.1109/TCSVT.2021.3063423.

[3] Y. Tu, C. Zhou, J. Guo, S. Gao, and Z. Yu, "Enhancing the alignment between target words and corresponding frames for video captioning," *Pattern Recognition*, vol. 111, p. 107 702, 2021, ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2020.107702. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320320305057.

[4] L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen, "Video captioning with attention-based lstm and semantic consistency," *IEEE Transactions on Multimedia*, vol. 19, no. 9, pp. 2045–2055, 2017. DOI: 10.1109/TMM.2017.2729019.

[5] S. Chen, Q. Jin, J. Chen, and A. G. Hauptmann, "Generating video descriptions with latent topic guidance," *IEEE Transactions on Multimedia*, vol. 21, no. 9, pp. 2407–2418, 2019. DOI: 10.1109/TMM.2019.2896515.

[6] M. A. Gernsbacher, "Video captions benefit everyone," *Policy Insights from the Behavioral and Brain Sciences*, vol. 2, no. 1, pp. 195–202, 2015, PMID: 28066803. DOI: 10.1177/2372732215602130. eprint: https://doi.org/10.1177/2372732215602130. [Online]. Available: https://doi.org/10.1177/2372732215602130.

[7] X. Long, C. Gan, and G. de Melo, "Video Captioning with Multi-Faceted Attention," *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 173–184, Mar. 2018, ISSN: 2307-387X. DOI: 10.1162/tacl_a_00013. eprint: https://direct.mit.edu/tacl/article-pdf/doi/10.1162/tacl_a_00013/1567598/tacl_a_00013.pdf. [Online]. Available: https://doi.org/10.1162/tacl%5C_a%5C_00013.

[8] L. Yan, Q. Wang, Y. Cui, *et al.*, "Gl-rg: Global-local representation granularity for video captioning," in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, ser. IJCAI-2022, International Joint Conferences on Artificial Intelligence Organization, Jul. 2022. DOI: 10.24963/ijcai.2022/384. [Online]. Available: http://dx.doi.org/10.24963/ijcai.2022/384.

[9] G. Guan, Z. Wang, K. Yu, S. Mei, M. He, and D. Feng, "Video summarization with global and local features," in *2012 IEEE International Conference on Multimedia and Expo Workshops*, 2012, pp. 570–575. DOI: 10.1109/ICMEW.2012.105.

[10] A. Klaser, M. Marszałek, and C. Schmid, "A spatio-temporal descriptor based on 3d-gradients," in *BMVC 2008-19th British Machine Vision Conference*, British Machine Vision Association, 2008, pp. 275–1.

[11] H. Wang, A. Kläser, C. Schmid, and C.-L. Liu, "Dense trajectories and motion boundary descriptors for action recognition," *International journal of computer vision*, vol. 103, pp. 60–79, 2013.

[12] S. Guadarrama, N. Krishnamoorthy, G. Malkarnenkar, *et al.*, "Youtube2text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition," Dec. 2013, pp. 2712–2719. DOI: 10.1109/ICCV.2013.337.

[13] P. Das, C. Xu, R. F. Doell, and J. J. Corso, "A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2013.

[14] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, *Sequence to sequence – video to text*, 2015. arXiv: 1505.00487 [cs.CV].

[15] M. Rohrbach, W. Qiu, I. Titov, S. Thater, M. Pinkal, and B. Schiele, "Translating video content to natural language descriptions," Dec. 2013, pp. 433–440, ISBN: 978-1-4799-2840-8. DOI: 10.1109/ICCV.2013.61.

[16] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, *Translating videos to natural language using deep recurrent neural networks*, 2015. arXiv: 1412.4729 [cs.CV].

[17] L. Yao, A. Torabi, K. Cho, *et al.*, *Describing videos by exploiting temporal structure*, 2015. arXiv: 1502.08029 [stat.ML].

[18] S. Sah, T. Nguyen, M. Dominguez, F. P. Such, and R. Ptucha, "Temporally steered gaussian attention for video understanding," in *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 2208–2216. DOI: 10.1109/CVPRW.2017.274.

[19] F. M. Shah, "Natural language video captioning in bengali using deep learning," Ph.D. dissertation, Ahsanullah University of Science and Technology, 2020.

[20] T. Das and A. Majumdar, *Video captioning in bengali with visual attention*, Dec. 2022. [Online]. Available: https://www.researchgate.net/publication/366634326_Video_Captioning_in_Bengali_With_Visual_Attention.

[21] *Pexels: Free stock photos videos.* [Online]. Available: https://www.pexels.com.

[22] *Mixkit: Free stock video clips.* [Online]. Available: https://mixkit.co.

[23] *Pixabay: Free images and videos.* [Online]. Available: https://pixabay.com.

[24] J. Song, Z. Guo, L. Gao, W. Liu, D. Zhang, and H. T. Shen, *Hierarchical lstm with adjusted temporal attention for video captioning*, 2017. arXiv: 1706.01231 [cs.CV].

[25] L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen, "Video captioning with attention-based lstm and semantic consistency," *IEEE Transactions on Multimedia*, vol. 19, no. 9, pp. 2045–2055, 2017. DOI: 10.1109/TMM.2017.2729019.

[26] M. S. Zaoad, M. R. Mannan, A. B. Mandol, M. Rahman, M. A. Islam, and M. M. Rahman, "An attention-based hybrid deep learning approach for bengali video captioning," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 1, pp. 257–269, 2023, ISSN: 1319-1578. DOI: https://doi.org/10.1016/j.jksuci.2022.11.015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157822004128.

[27] K. Khurana and U. Deshpande, "Video question-answering techniques, benchmark datasets and evaluation metrics leveraging video captioning: A comprehensive survey," *IEEE Access*, vol. 9, pp. 43 799–43 823, 2021. DOI: 10.1109/ACCESS.2021.3058248.

[28] L. Gao, X. Wang, J. Song, and Y. Liu, "Fused gru with semantic-temporal attention for video captioning," *Neurocomputing*, vol. 395, pp. 222–228, 2020, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2018.06.096. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S092523121930904X.

[29] R. Khan, M. S. Islam, K. Kanwal, M. Iqbal, M. I. Hossain, and Z. Ye, *A deep neural framework for image caption generation using gru-based attention mechanism*, 2022. arXiv: 2203.01594 [cs.CL].

[30] J. Perez-Martin, B. Bustos, and J. Pérez, "Improving video captioning with temporal composition of a visual-syntactic embedding," in *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2021, pp. 3038–3048. DOI: 10.1109/WACV48630.2021.00308.

[31] Y. Song, Y. Zhao, S. Chen, and Q. Jin, "RUC$_A$$IM3atTRECVID2019$: $Videototext$.," *TRECVID*, Jan. 2019. [Online]. Available: https://www-nlpir.nist.gov/projects/tvpubs/tv19.papers/ruc_aim3.pdf.

[32] D. Bahdanau, K. Cho, and Y. Bengio, *Neural machine translation by jointly learning to align and translate*, 2016. arXiv: 1409.0473 [cs.CL].

[33] L. Zhang, J. Winn, and R. Tomioka, *Gaussian attention model and its application to knowledge base embedding and question answering*, 2016. arXiv: 1611.02266 [stat.ML].

[34] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, *Rethinking the inception architecture for computer vision*, 2015. arXiv: 1512.00567 [cs.CV].

[35] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition*, 2015. arXiv: 1512.03385 [cs.CV].

[36] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large-scale image recognition*, 2015. arXiv: 1409.1556 [cs.CV].

[37] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: A method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[38] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," in *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, J. Goldstein, A. Lavie, C.-Y. Lin, and C. Voss, Eds., Ann Arbor, Michigan: Association for Computational Linguistics, Jun. 2005, pp. 65–72. [Online]. Available: https://aclanthology.org/W05-0909.

[39] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[40] R. Vedantam, C. L. Zitnick, and D. Parikh, *Cider: Consensus-based image description evaluation*, 2015. arXiv: 1411.5726 [cs.CV].