B. Tech Project

On

“GENERATING VIDEO DESCRIPTIONS WITH ATTENTION-DRIVEN LSTM MODELS IN HINDILANGUAGE”

By

Name Of the Students	**Roll No.**
DHRUV	2020UEA6564
VANSH GUPTA	2020UEA6576
HARSH NAGAR	2020UEA6588
NAMAN	2020UEA6598

Under the Supervision

Of

Prof. R. K. Sharma

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NETAJI SUBHAS UNIVERSITY OF TECHNOLOGY(NSUT), NEW DELHI

MAY 2024

CANDIDATE(S) DECLARATION

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

I/We, Dhruv (2020UEA6564), Vansh Gupta(2020UEA6576), Harsh Nagar(2020UEA6588) and Naman (2020UEA6598) students of B. Tech., Department of Electronics & Communication Engineering, hereby declare that the Project-Thesis titled "Generating Video Descriptions With Attention-Driven LSTM Models In Hindi Language" which is submitted by me/us to the Department of Electronics & Communication Engineering, Netaji Subhas University of Technology (NSUT) East Campus, Geeta Colony in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology is my/our original work and not copied from any source without proper citation. The manuscript has been subjected to plagiarism check by Turnitin software. This work has not previously formed the basis for the award of any other Degree.

Place: New Delhi

Date: 03/05/2024

DHRUV	HARSH NAGAR	VANSH GUPTA	NAMAN
(2020UEA6564)	(2020UEA65688)	(2020UEA6576)	(2020UEA6598)

CERTIFICATE OF DECLARATION

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

This is to certify that the project entitled, "**Generating Video Descriptions With Attention-Driven LSTM Models In Hindi Language**" submitted by **Dhruv, Harsh Nagar, Vansh Gupta, Naman** is a record of bonafide work carried out by them, in the Division of Electronics & Communication Engineering (Artificial Intelligence & Machine Learning), Netaji Subhas University of Technology, New Delhi, under my supervision and guidance in partial fulfillment of requirements for the award of the degree of **Bachelor of Technology (B.Tech.)** in **Electronics & Communication Engineering (Artificial Intelligence & Machine Learning)**, in the academic year **2023-2024**.

Prof. RK Sharma
Netaji Subhas University of Technology (NSUT)

ACKNOWLEDGEMENT

We express sincere gratitude to our esteemed academic guide and mentor, **Prof. RK Sharma**, whose continuous guidance and unwavering support have been pivotal throughout our **B.Tech.** journey at **Netaji Subhas University of Technology (NSUT)**, influencing not only this research but also various other course works.

Special appreciation is extended to the dedicated professors and encouraging friends at **NSUT**, whose support has significantly contributed to the success of our project.

Last but not least, our heartfelt thanks go to our parents and siblings for their boundless love and steadfast support. Their encouragement has been the driving force that made the completion of this work possible.

ABSTRACT

This research addresses the existing gap in **video descriptions** for **regional languages**, with a particular emphasis on **Hindi**. Motivated by a thorough review of available literature, it was observed that languages like Hindi are inadequately represented in this domain. Consequently, we initiated the project titled **"Generating Video Descriptions with Attention-Driven LSTM Models in Hindi Language"** to enhance accessibility and inclusion of Hindi multimedia content. Leveraging advanced **LSTM** models and utilizing the **VATEX** dataset, our objective is to pioneer advancements in regional narrative video production. By venturing into unexplored terrain, we not only contribute to the **promotion of Indian language** and **culture** but also establish a precedent for exploring narrative films in other regional languages. This research is strategically designed to foster diversity, integration, and propel broader advancements at the intersection of natural language processing and multitasking.

Our findings demonstrate that our approach yields **competitive performance** when compared to **state-of-the-art video captioning** baselines such as **BLEU** and **METEOR**. This signifies the efficacy of our methodology in enhancing the quality of video descriptions, thereby contributing significantly to the field of regional language video captioning.

Keywords: video descriptions, VATEX, attention-based, LSTM, Hindi, BLEU.

LIST OF CONTENTS

LIST OF FIGURES

AI Generated

Authored

Referenced

LIST OF TABLES

LIST OF ABBREVIATIONS

CNN: Convolution Neural Network

LSTM: Long Short-Term Memory

BiLSTM: Bidirectional Long Short-Term Memory

RNN: Recurrent Neural Network

BPTT: Backpropagation Through Time

ML: Machine Learning

ReLu: Rectified Linear Unit

CCTV: Closed-Circuit Television

VGG: Visual Geometry Group

SCVC: Syntax Customized Video Captioning

ViSE: Visual-Semantic Embedding

EMVC: Event-Centric Multi-Modal fusion approach for dense Video Captioning

GRU: Gated Recurrent Unit

BLEU: Bilingual Evaluation Understudy Score

I. INTRODUCTION

We start off things as we first delve into a brief ***1.1 Overview*** *of video captioning and try to understand what it is. It then becomes crucial to understand the history of origination of our topic to better understand its breadth and depth and draw a solid* ***1.1.1 Background*** *to lay our foundation. After that, with a focus on foundation we take a brief look at various* ***1.2 Types of Video Captioning*** *so as to better aware the reader of our scope of area. As we synthesize the information gathered, we conclude this chapter by taking a quick look at various* ***1.3 Applications of Video Captioning/Descriptions.***

1.1 Overview

Video description has gained attention due to advances in computer vision, NLP, and machine learning. It involves describing video content with natural language sentences, serving various purposes like human-robot interaction and accessibility for the visually impaired.

Two main approaches exist: template-based language models and sequence learning. Template models use predefined templates to structure sentences, ensuring consistent syntax; sequence learning directly translates video content into phrases by extracting features and generating subtitles based on data. An encoder-decoder network offers an efficient solution. The encoder processes the input video, producing a fixed-dimensional vector fed into the decoder, which generates words sequentially.

1.1.1 Background

In the early 2010s, the field of video captioning and description generation witnessed a significant evolution, driven by advancements in computer vision and natural language processing. This transformative shift began with the introduction of large-scale datasets and neural network-based models. Key papers such as “Show and Tell: A Neural Image Caption Generator” by Vinyals et al. in 2015 played a pivotal role in pioneering the fusion of image analysis and language generation, setting the stage for modern video captioning. This progression from basic audio captioning in the 1990s to the sophisticated video captioning of the 2010s highlights the rapid growth of technology in enhancing media accessibility and usability. This literature survey explores the key developments that have shaped this dynamic field.

1.2 Types of Video Captioning

Captioning is the process of adding text to visual content, providing information about the content for various purposes. There are several types of captioning, each serving different needs and audiences. Here are some common types of captioning:

1. Closed Captions (CC):

Primarily made for those who are hard of hearing or deaf. able can be turned on or off by the observer. includes sound effects, conversation, and other pertinent audio data.

2. Open Captions:

Similar to closed captions but are permanently embedded in the video and cannot be turned off. Always visible during playback. Includes text for dialogue, sound effects, and other audio elements.

3. Subtitles:

Intended for viewers who may not be familiar with the language spoken in the video. Usually provided as an option, can be turned on or off. Translates spoken words into the viewer's language.

5. Live Captions:

Generated in real-time for live events, broadcasts, or streaming content. Displayed as the content is happening. Captures spoken words and audio information as it occurs.

6. Descriptive Video Service (DVS) or Audio Description:

Designed for individuals with visual impairments. Describes key visual elements, actions, and scene changes during pauses in dialogue.

1.3 Applications of Video descriptions

- **Security Surveillance**

  CCTV cameras are passive and lack the ability to analyze their surroundings. However, integrating video description technology into these cameras can empower them to detect suspicious activities. For instance, in a crowded mall, monitoring every individual is challenging for security personnel using traditional CCTV. Video description can help identify potential shoplifters or thieves by analyzing and alerting authorities, improving security.

Figure 1.1: Security Surveillance (AI*)

- **Improved indexing for Search Engine Optimizer**

  Existing algorithms are limited to matching text with available online content. Video titles are often too brief to fully describe video content. Video description can improve this by indexing videos based on captions, enabling search engines to provide more relevant video results and search for similar videos based on captions.

Figure 1.2: Improved indexing for Search Engine Optimizer (AI*)

- **Text-to-speech of Captions for visually impaired**

  Visually impaired individuals often rely on touch or hearing, leading to a lower quality of life and the need for guided assistance. Video description technology can enhance their lives by integrating with text-to-speech software on smart phones. This conversion of video-generated text descriptions into speech provides a personal guide, helping visually impaired individuals navigate their surroundings independently.

Figure 1.3: Text-to-speech of captions for visually impaired (AI*)

II. LITERATURE REVIEW

*With a comprehensive understanding of **video captioning**, encompassing its **definition**, **types**, **historical evolution**, and diverse **applications**, we embark on a **literature survey** to delve deeper into the nuanced developments and scholarly contributions in this dynamic field. Our exploration aims to unravel key research findings, methodologies, and insights that have shaped the landscape of video description. Through this survey, we seek to illuminate the forefront of knowledge, identify trends, and critically evaluate the advancements that have played a pivotal role in refining video captioning technologies. By synthesizing existing literature, we aim to provide a comprehensive overview that contributes to the ongoing discourse, fostering a deeper appreciation for the intersection of computer vision, natural language processing, and machine learning in the realm of video description.*

2.1 Base Research Papers

Our study and investigation of this project work based on:-

[1]. Attention-based Densely Connected LSTM for Video Captioning

[2]. Dense video captioning based on local attention

[3]. An attention-based hybrid deep learning approach for Bengali video captioning

[4]. Hybrid Architecture using CNN and LSTM for Image Captioning in Hindi Language

[5]. VATEX2020: pLSTM framework for video captioning

[6]. Video Captioning: a comparative review of where we are and which could be the route

[7]. An attention based dual learning approach for video captioning

[8]. Video Captioning with Attention-based LSTM and Semantic Consistency

The above papers summaries outline several innovative approaches to video and image captioning, integrating advanced neural network architectures like LSTM, attention mechanisms, and CNNs, aimed at enhancing the accuracy and relevance of generated captions across differentlanguages and modalities. A synthesized narrative incorporating the mentioned papers:

In recent advancements across the domain of video and image captioning, researchers have introduced a plethora of methodologies to tackle the inherent complexities of generating coherentand contextually accurate descriptions. Among these, the Attention-based Densely Connected LSTM (Dense LSTM) marks a significant leap forward. Unlike traditional LSTM models, whichpredominantly rely on the last generated hidden state, Dense LSTM innovatively leverages all previous states. This ensures a comprehensive flow of information, enabling the model to capturelong-range dependencies effectively, thereby enhancing the accuracy of predicted words by considering the overall context rather than being overly dependent on recent inputs [1].

The domain of dense video captioning has seen notable progress with the introduction of the Dense Video Captioning Model Based on Local Attention (DVCL). Addressing the challenge of establishing multimodal feature relationships between video frames and captions, DVCL employs a combination of 2D temporal differential CNN and a deformable transformer. An LSTM with a local attention mechanism further refines the model, aligning each generated word with the most relevant video frame, thus significantly improving caption accuracy [2].

Turning our attention to language-specific innovations, the field of Bengali video captioning has been enriched by an attention-based hybrid deep learning approach. This approach integrates LSTM, BiLSTM, and GRU models with various CNN architectures to process video frame features. Evaluated across standard metrics like BLEU, METEOR, and ROUGE, this hybrid model sets a new benchmark for Bengali video captioning, outshining its predecessors in capturing the essence of video content and translating it into accurate Bengali captions [3].

In a similar vein, the development of a multi-layered CNN-LSTM neural network model for image captioning in Hindi demonstrates the potential of hybrid architectures in breaking linguistic barriers. By fine-tuning hyperparameters and layer configurations, this model has shown remarkable improvements in BLEU scores over existing models. This advancement not only enhances the accessibility of technology for Hindi speakers but also opens new avenues for application in various societal domains [4].

The pLSTM framework, introduced for the VATEX-2020 video captioning challenge, representsanother stride in the field. Utilizing a sequential method that combines 3D CNN for visual feature encoding with LSTM for decoding, this framework adeptly condenses video information into precise captions. Such methodologies are pivotal for applications in sentiment analysis, machine translation, and more, showcasing the versatility and potential impact of advanced videocaptioning techniques [5].

A comparative review highlights the ongoing challenges and achievements within video captioning. By analyzing various methods through a performance metric-based ranking system, this review sheds light on the most effective techniques and suggests future directions for research. This encapsulates the dynamic and evolving nature of video captioning, emphasizing the need for continual innovation to address the complexity of translating visual content into descriptive text [6].

The attention-based dual learning approach (ADL) introduces a novel encoder-decoder-reconstructor architecture that leverages dual learning. By adopting a multi-head attention mechanism, ADL enhances the model's focus on salient information in both videos and captions.This approach fine-tunes performance through a dual learning mechanism, promising improved accuracy and relevance in generated captions [7].

Lastly, the integration of an attention mechanism with LSTM underscores the importance of semantic consistency in video captioning. By dynamically weighting CNN representations andemploying LSTM decoders for word generation, this framework ensures that the resulting sentences are not only semantically consistent with the video content but also enriched with relevant details. Such advancements underscore the critical role of multimodal embedding in bridging the gap between visual content and textual descriptions, pushing the boundaries of what's achievable in video captioning [8].

These summaries encapsulate the frontier of research in video and image captioning, highlighting the shift towards more integrated, context-aware, and linguistically diverse models.The emphasis on attention mechanisms, dense connections, and hybrid architectures across different languages points to a future where technology can seamlessly translate visual content into accurate, meaningful captions for a global audience.

2.2 Key Takeaways from Our Review

Contextual Information for Accuracy: Techniques like Attention-based Densely Connected LSTM and Dense Video Captioning Locally (DVCL) are pivotal in utilizing contextual detailsand dense frame connections to enhance the precision of video captioning.

Linguistic Diversity in Captioning: The field is witnessing notable advancements in linguistic diversity, with the development of hybrid models for Bengali and CNN-LSTM architectures forHindi captioning, showcasing the expanding abilities of AI to generate captions across differentlanguages.

Innovative Frameworks and Benchmarks: The introduction of the pLSTM framework in theVATEX2020 challenge represents a significant benchmark in video captioning, pushing the boundaries of current methodologies.

Comprehensive Field Reviews: Reviews offer valuable insights into the video captioning domain, outlining both the progress made and the potential avenues for future research anddevelopment.

Dual Learning and Semantic Consistency: New approaches that combine dual learning with attention-based LSTM models and emphasize semantic consistency are leading to more nuancedand contextually accurate video captions.

Evolution of AI in Visual Content Understanding: These developments reflect the continuousevolution and increasing potential of AI technologies in more effectively understanding and describing visual content, marking a forward leap in the capabilities of video and image captioning systems.

In this chapter, we delved into the multifaceted realm of video captioning and systematically examined different captioning types encountered throughout our literature review, focusing on the foundational base papers. Concluding our literature exploration, we highlighted the transformative journey of video captioning, showcasing significant breakthroughs in model architectures and linguistic diversity. In the upcoming chapter, our focus shifts to our core focus, where we will outline the ***research objectives****, delve into the* ***motivation*** *behind the study, and articulate the specific problem statement driving our investigation.*

III. MOTIVATION & OBJECTIVE

Our research project, titled **“Generating Video Descriptions with Attention Driven LSTM Models in Hindi”** is motivated by several compelling reasons:

- **Addressing a Gap in Research:** The motivation stems from the observation of a dearth of research in the field of video description generation, particularly in regional languages like Hindi. By recognizing this gap, our project aims to contribute valuable insights and solutions to an unexplored area.

- **Regional Language Focus:** Our focus on Hindi acknowledges the significance of regional languages in multimedia content. Hindi is one of the most widely spoken languages globally, and there is a clear need to enhance accessibility and inclusivity in this language, catering to a vast and diverse audience.

- **Leveraging Advanced Technology:** The incorporation of advanced LSTM models signifies a commitment to cutting-edge technology. By employing attention driven models, our research aims to enhance the accuracy and effectiveness of video descriptions in Hindi, showcasing a dedication to innovative approaches in natural language processing and multimedia content understanding.

- **Utilizing the VATEX Dataset:** The decision to leverage the VATEX dataset demonstrates a strategic choice in utilizing a comprehensive and widely recognized resource. This dataset allows our research to build upon a solid foundation, fostering reproducibility and comparability with other studies, while specifically tailoring the approach to the Hindi language.

- **Enhancing Accessibility and Inclusivity:** The overarching goal of our project is to enhance accessibility and inclusivity in Hindi multimedia content. By generating accurate and contextually relevant video descriptions, our research contributes to making multimedia content more accessible to individuals who are visually impaired or face language barriers, fostering inclusivity in the digital space.

Setting a Precedent for Regional Language Exploration: Beyond its impact on Hindi, our project aims to set a precedent for the exploration of other regional languages. This ambitious approach not only broadens the scope of our research but also encourages further studies in diverse linguistic landscapes, ultimately contributing to a more inclusive multimedia environment.

- **Fostering Diversity in the Multimedia Landscape:** Our research project aligns with the broader goal of fostering diversity in the multimedia landscape. By focusing on regional languages and paving the way for similar endeavors, our work contributes to a richer and more diverse representation of cultures and languages in digital content.

OBJECTIVE

Based on our review, we were motivated to launch the project **"Generating Video Descriptions with Attention-Driven LSTM Models in Hindi."** We observed a dearth of research in regional languages like Hindi, especially in the realm of video description generation. Recognizing this uncharted territory, we aim to leverage advanced LSTM models and VATEX dataset to enhance accessibility and inclusivity in Hindi multimedia content. Our project not only contributes to the Hindi language but also sets a precedent for the exploration of other regional languages, ultimately fostering diversity and inclusivity in the multimedia landscape.

In summary, our research project is motivated by a commitment to addressing research gaps, leveraging advanced technology, utilizing relevant datasets, enhancing accessibility, and fostering diversity in the multimedia landscape, with a particular focus on regional languages like Hindi.

IV. METHODOLOGY

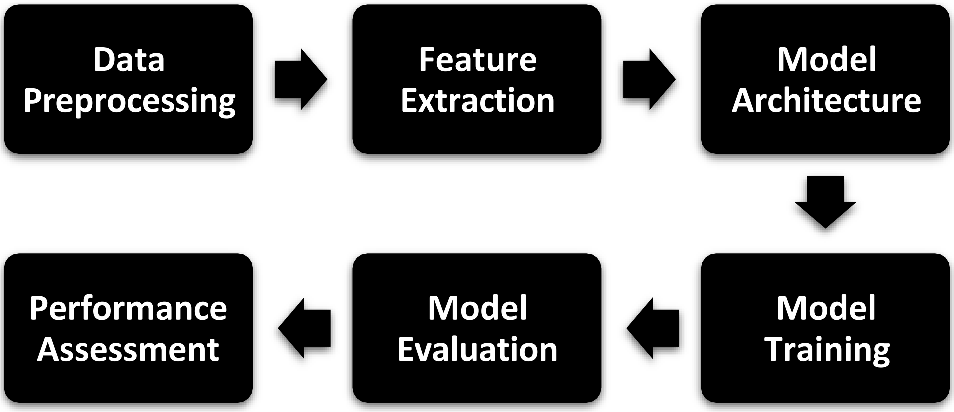


Figure 4.1: Methodology

- **Data Preprocessing:** Tokenization and Text Preprocessing: The textual descriptions associated with videos were tokenized and preprocessed to prepare them for model training. This step involved converting text to lowercase, removing special characters, and adding start and end tokens for caption generation.

- **Feature Extraction using VGG-16:** Video Frame Feature Extraction: Videos were processed frame by frame, and features were extracted using the VGG-16 pre-trained model. The features were used as input to the subsequent steps of the model.

- **Model Architecture:**

  - **Encoder:** The video features were fed into a Bidirectional LSTM (Bi-LSTM) layer, serving as the Encoder. The Bi-LSTM captured temporal information from both forward and backward directions.

  - **Attention Mechanism:** An attention mechanism was employed to dynamically weight the importance of different frames of the video during the caption generation process.

    - **Decoder:** The Decoder received the attention-refined context vector along with the tokenized captions as inputs. It generated captions word by word, taking the context vector into account.

- **Model Training:** The Sparse Categorical Cross-entropy loss was employed as the loss function to compute the difference between predicted and actual word indices in the captions.

- **Model Evaluation and BLEU Score Calculation:** Caption Generation: Captions were generated for test videos, utilizing the trained model and the extracted video features BLEU Score Calculation: The BLEU score, which measures the quality of the generated captions by comparing them to reference captions, was computed to evaluate the model's performance.

- **Performance Assessment:** BLEU Score Analysis: Precision at different n-gram levels (1 to 4) and the BLEU score were computed and analyzed to assess the quality and relevance of the generated captions.

V. PROBLEM STATEMENT

The lack of comprehensive video descriptions in Hindi impedes accessibility and inclusivity for a vast audience, hindering their engagement with multimedia content. To address this gap, our research focuses on developing and optimizing Attention-Driven LSTM models for generating accurate, coherent, and contextually relevant video descriptions in the Hindi language. This problem statement underscores the critical need to make multimedia content more accessible and engaging for Hindi-speaking audiences by employing advanced natural language processing techniques and attention mechanisms in the domain of video description generation.

Furthermore, our approach leverages the previously unexplored state-of-the-art VATEX dataset. By incorporating this rich and diverse dataset, we aim to enhance the robustness and effectiveness of our models in capturing the nuances of regional narratives and linguistic intricacies specific to Hindi. The utilization of the VATEX dataset not only enriches our research methodology but also positions our work at the forefront of addressing the unique challenges posed by regional language video description generation.

"The absence of detailed video descriptions in Hindi hinders accessibility, limiting engagement with multimedia content for a diverse audience. This research aims to bridge this gap by developing Attention-Driven LSTM models, optimizing their ability to generate accurate and contextually relevant video descriptions in Hindi, thereby enhancing inclusivity and user experience."

In the preceding chapter, we have meticulously detailed our ***research methodology****, established* ***clear objectives****, delved into the* ***motivation*** *behind our study, and precisely defined the* ***problem statement*** *driving our investigation. In the upcoming chapter, our focus will shift towards the practical aspects of our research as we discuss the* ***implementation phase****. This will be followed by an exploration of the* ***results*** *obtained through our* ***methodology****, providing insights into the outcomes of our study. Subsequently, we will engage in a thorough discussion, interpreting and contextualizing the results within the* ***framework*** *of our research objectives. This chapter aims to bridge the theoretical groundwork laid in earlier sections with the tangible* ***outcomes and their implications****, offering a comprehensive understanding of the study's execution and findings.*

VI. IMPLEMENTION, RESULT & DISCUSSION

In this chapter, we transition from the theoretical to the ***practical aspects*** *of our research. First, we explore the* ***implementation phase****, detailing the concrete steps taken to execute our methodology. Following this, we present the* ***results*** *obtained from our implementation, providing a comprehensive overview of the findings. The discussion section follows, where we analyze and interpret the results within the context of our research objectives. This chapter serves as a bridge between theory and practice, offering a hands-on exploration of our research process and its tangible outcomes, paving the way for a deeper understanding and meaningful insights into the subject matter.*

6.1 Existing Models

6.1.1 Convolutional Neural Network (CNN) Model

Deep, feed-forward artificial neural networks are under the category of convolutional neural networks (CNNs). Among deep learning's most widely used algorithms are CNNs. They have several uses in speech recognition, picture identification, and natural language processing and can produce cutting-edge outcomes. For image identification tasks including object detection, classification, face recognition, and other computer vision issues, convolutional neural networks, or CNNs, have been widely employed. The length k of a feature map in a CNN does not have to be fixed at a certain number; instead, it may be varied for various layers, and each feature map can then be weighted according to its receptive field size to determine the feature map size of the layer.

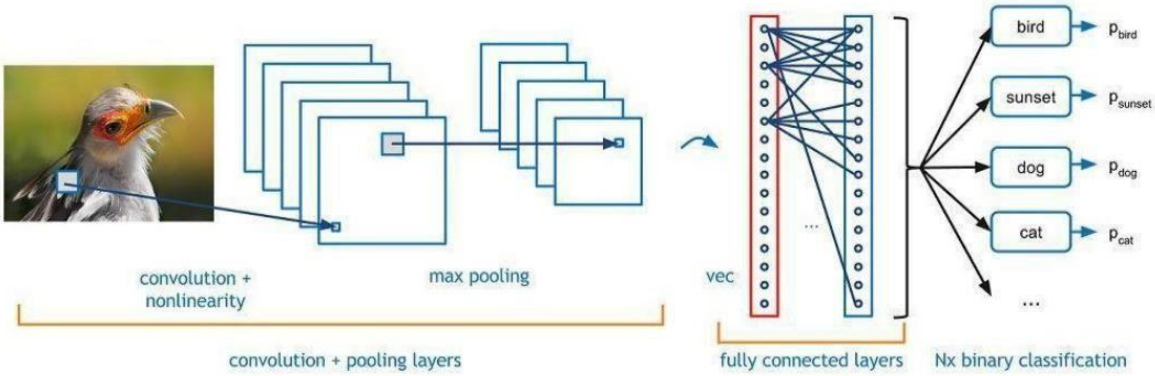


Fig 6.1.1: Comparison between Convolution layers (left) and fully connected layers(right).

The Convolution Operation extracts the input image's high-level features, such as edges. It is unnecessary to limit ConvNets to a single Convolutional Layer. Normally, the first ConvLayer is in charge of capturing low-level properties like edges, colour, gradient direction, and so on.

The architecture adjusts to the high-level properties by adding layers, giving us a network that understands the photos in the dataset in a way that is similar to how we would.

6.1.1.1 Terminologies Used in CNN

6.1.1.1.1 Input Volume

An picture with the following dimensions is the input layer, also known as the input volume: [width x height x depth]. It is a pixel-value matrix. Example: The depth here reflects the channels of R, G, B. The input layer should be repeatedly divisible by 2. Common numbers include 32, 64, 96, 224, 384,6 and 512. Fig 6.1.1.1.1 shows an image decomposed into a 3D matrix for the purpose of using convolution.

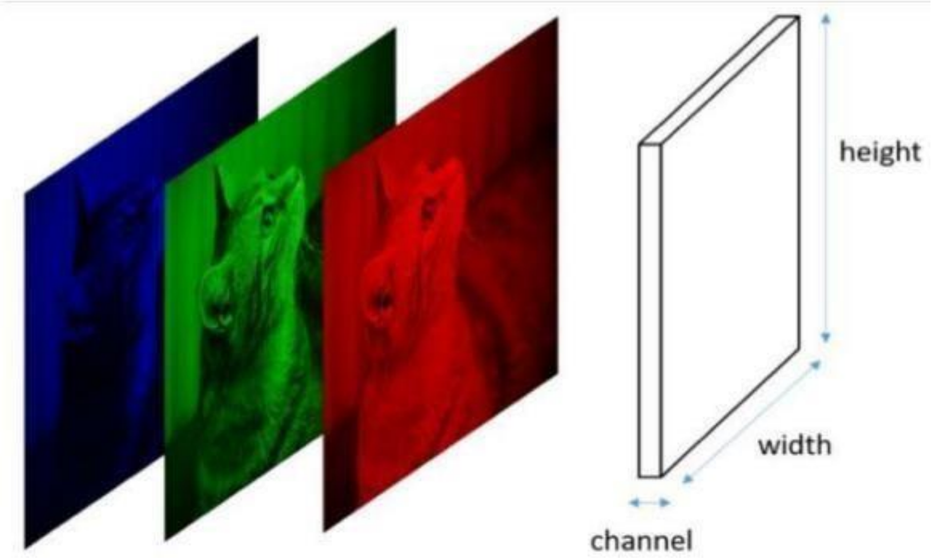


Fig 6.1.1.1.1: A colored image consisting of 3 channels (Red, Green, Blue).

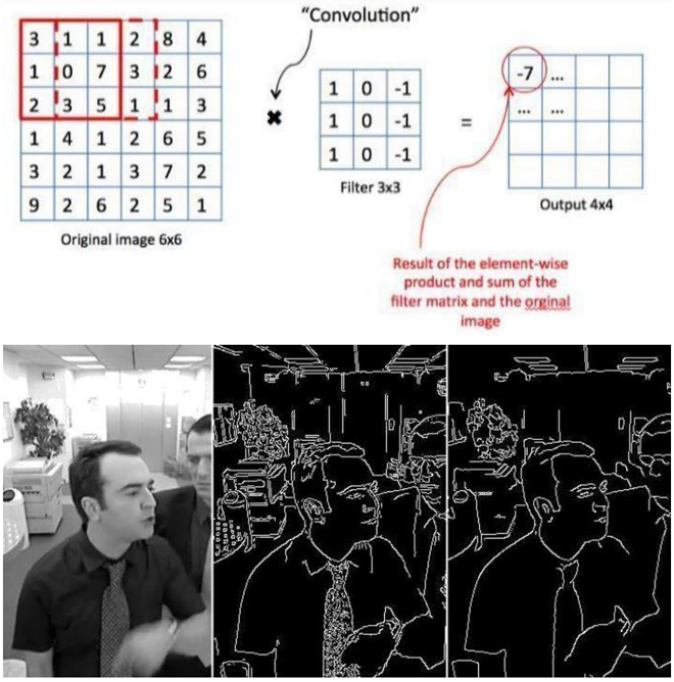
6.1.1.1.2 Features

Features are something using which a CNN can distinguish one class from another. They are an essential element used in training of the CNN, which is helpful for image analysis. For instance, a CNN can learn eyes as an element of the human face while training on an enormous arrangement of pictures of human appearances.

6.1.1.1.3 Filters

The filters in Convolutional Layers identify patterns. Different patterns in an image include multiple edges, shapes, textures, objects, and so on. Various detectors, such as Edge Detector, Corner Detector, and Shape Detector, can be utilized as filters. Our filters become more complicated as the network grows deeper, allowing us to recognize genuine items such as eyes, ears, feathers, fur, hair, scales, and beaks in subsequent layers rather than edges and basic shapes. Deeper layer filters will recognize whole dogs, cats, lizards, and birds, among other things. CNN can successfully capture spatial and temporal connections in images by using filters.

A filter is generally of size 3x3, 5x5, etc. Filters work with convolution operation. The convolution operation is a weighted sum of the product of pixel values with filters. A 3x3 filter will use a 3x3 portion of the image matrix to generate one value. The output image is smaller than the input image by 2 pixels. For mxn image and f x f filter size, output image will have (m-f+1) x (n-f+1) size. In a matrix with multiple channels, the filter will also have multiple channels, and the final output matrix will be (m-f+1) x (n-f+1). In Fig 6.1.1.1.3, the first element of the output image is the element-wise product of the 3x3 filter and the top 3x3 portion of the image. The filter will move 1 pixel right for the next element of output. After reaching the end of the image, it will move 1 pixel down and start from the first column again.



3	1	1	2	8	4
1	0	7	3	2	6
2	3	5	1	1	3
1	4	1	2	6	5
3	2	1	3	7	2
9	2	6	2	5	1

1	0	-1
1	0	-1
1	0	-1

-7	...		
...	...		

"Convolution"
✖
=
Original image 6x6
Filter 3x3
Output 4x4
Result of the element-wise product and sum of the filter matrix and the orginal image

Fig 6.1.1.1.3: a) 3x3 filter on a 2D Image matrix b) edge detection using filters

6.1.1.1.4 *Receptive Field*

To decrease the complexity that will be introduced due to the relation of input volume to the neurons, receptive fields are introduced. The receptive field binds the local pixels to it. On this layer, the neural network's function and build activation maps.

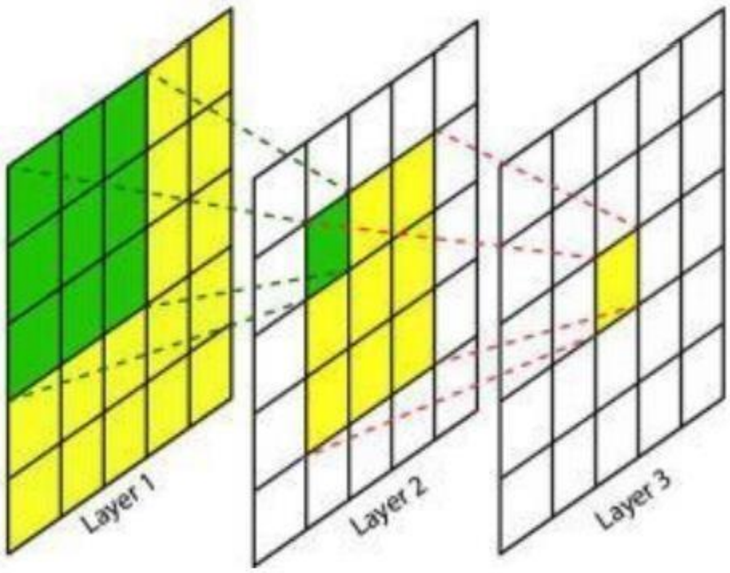
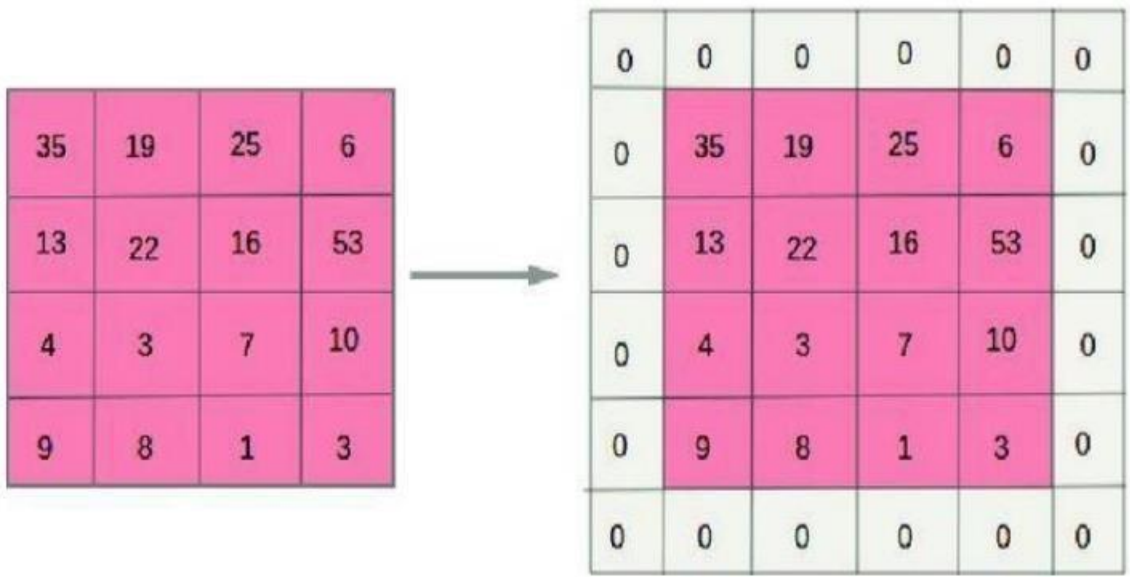


Fig 6.1.1.1.4: Receptive field

6.1.1.1.5 Padding – Zero

Zero paddings are used to restore the output's input volume dimensions. Zeros are symmetrically applied to the input volume in this process, and the size of the input is adjusted as needed.



35	19	25	6
13	22	16	53
4	3	7	10
9	8	1	3

0	0	0	0	0	0
0	35	19	25	6	0
0	13	22	16	53	0
0	4	3	7	10	0
0	9	8	1	3	0
0	0	0	0	0	0

Fig 6.1.1.1.5: Zero Padding

6.1.1.1.6 Hyperparameters

What is known as hyperparameters define the configuration of the different layers of the CNN network. Input volume dimension, zero paddings, stride duration, and receptive field are the CNN hyperparameters.

6.1.1.2 The CNN Architecture

The CNN architecture is made up of four kinds of layers. These are Convolutional Layer, RELU layers (RELU), Pooling layers (POOL), and Fully Connected layers (FC).
These layers help transform the input image into useful representations of the output.

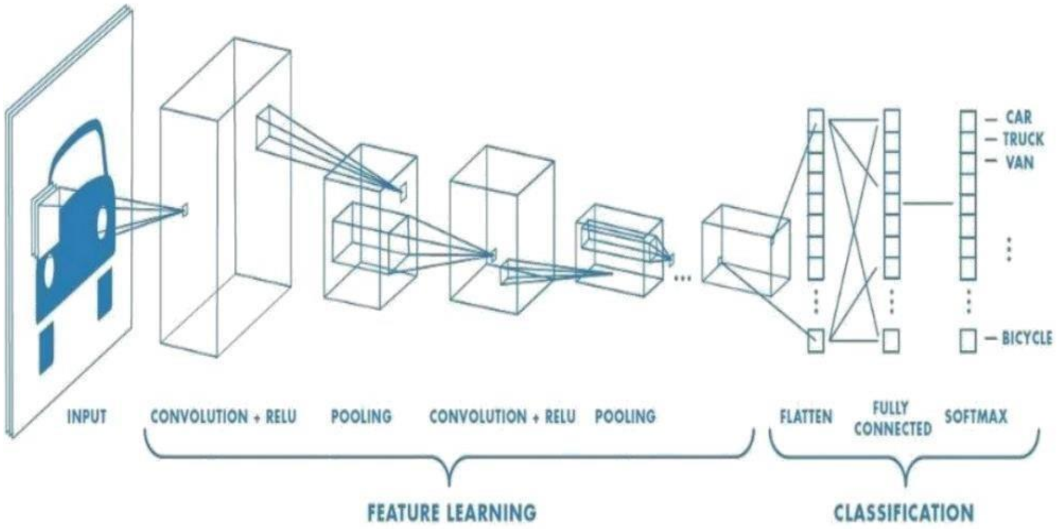


Fig 6.1.1.2: Convolution, pooling, relu, and fully connected layers are all included in this CNN.

6.1.1.2.1 Convolutional Layer

It performs one of the essential functions in the input image to detect the presence of the features. By dragging the filter onto the image, it performs the coevolutionary product that produces the activation map. The weights on the filter are initialized and updated using the gradient descent approach used during the training stage.

N filters can be used in a convolution layer. A single filter of size f x f x c (c for channels) transforms a m x n x c size matrix into a m x n matrix. The m*n*M size matrix will result from N filters. The image has shrunk in height and width in Fig. 6.1.1.2.1, but the number of channels as increased. These are the extracted features from the image, stacked one on top of the other.

In a fully connected layer with 100 neurons, the 640x480x3 image will have 100*640*480*3+100 = 92160100 weights, which is incredibly high. If we use 100 filters instead of 5x5x3, we will have 100x(5x5x3+1) (+1 for bias) = 7600 parameters in a convolution layer, which is much fewer than a fully linked layer.

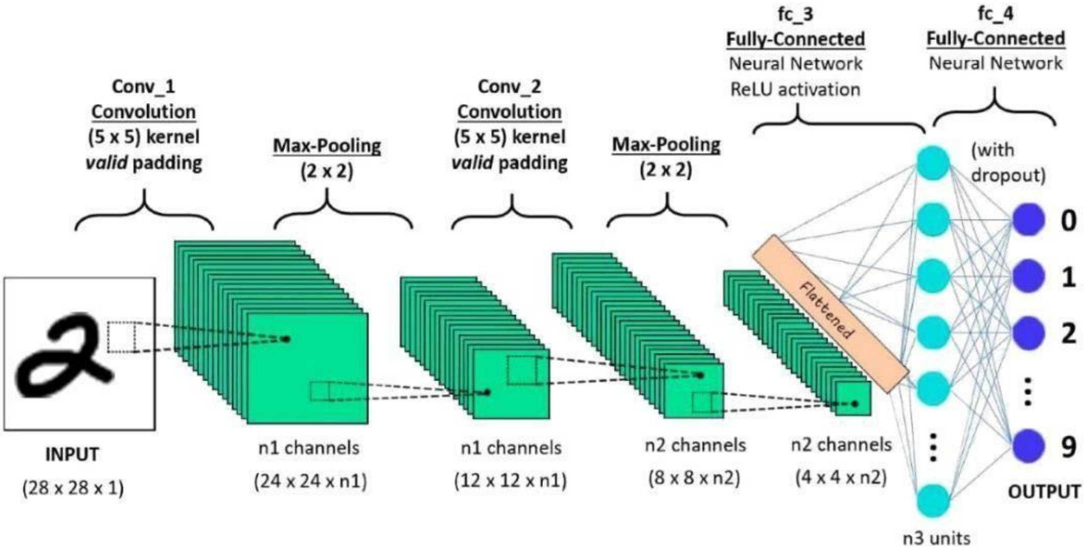


Fig 6.1.1.2.1: The first layer is convolution layer which uses n1 filters of 5x5. Valid Padding means that no padding is used and the output image will be smaller than input Image

Since the convolution operation is linear, it is common to introduce nonlinearity using a layer with an activation function. Earlier nonlinear functions like sigmoid and tanh were used, but it is found that ReLU works much better because of its computational efficiency, and the network can train much faster. It also helps eliminate the vanishing gradient problem in which the gradient decay exponentially with backpropagation due to sigmoid and tanh functions. Each value of the matrix is subjected to the function f(x) = max(0,x). As a result, the output shape remains unchanged. In Fig 6.1.1.2: ReLU layer is shown combined with Convolutional layer

6.1.1.2.3 Pooling Layer

A pooling layer is used for reducing image size. This layer has no parameters. Max pool is a common operation used which takes a f x f max filter and with f strides (strides is the number of the unit it moves after each convolution). Pooling operation works along all channels and does not reduce the number of channels.

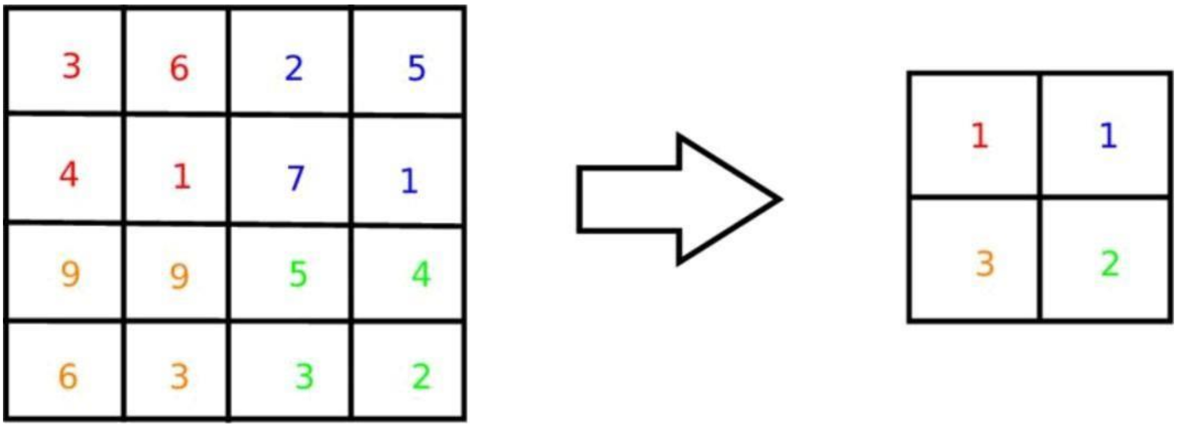


Fig 6.1.1.2.3: 2x2 max pooling operation

Another popular pooling method is average pooling, which uses the subregion’s average rather than the maximum value.

6.1.1.2.4 Fully Connected Layer

A fully linked layer is a feed-forward network that makes up the network’s last few layers. This layer’s input is a vector that has been flattened out.

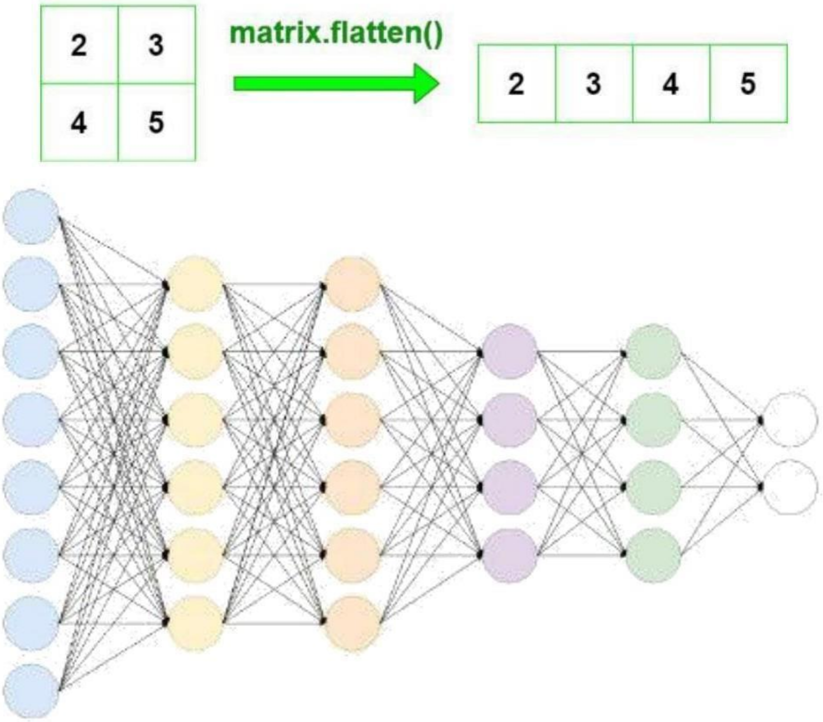


Fig 6.1.1.2.4: a) Flatten matrix into image b) fully connected layers

6.1.2 VGG-16

VGGNet-16 is made up of 16 convolutional layers and is particularly appealing due to its consistency. It is the most popular method for extracting features from photos at the moment. VGG can be achieved using transfer learning, in which the model is pre-trained on a dataset and the parameters are changed for improved precision.
The ImageNet dataset contains 14 million photos divided into 1000 classes, and this model has a top-5 test precision of 92.7 percent.

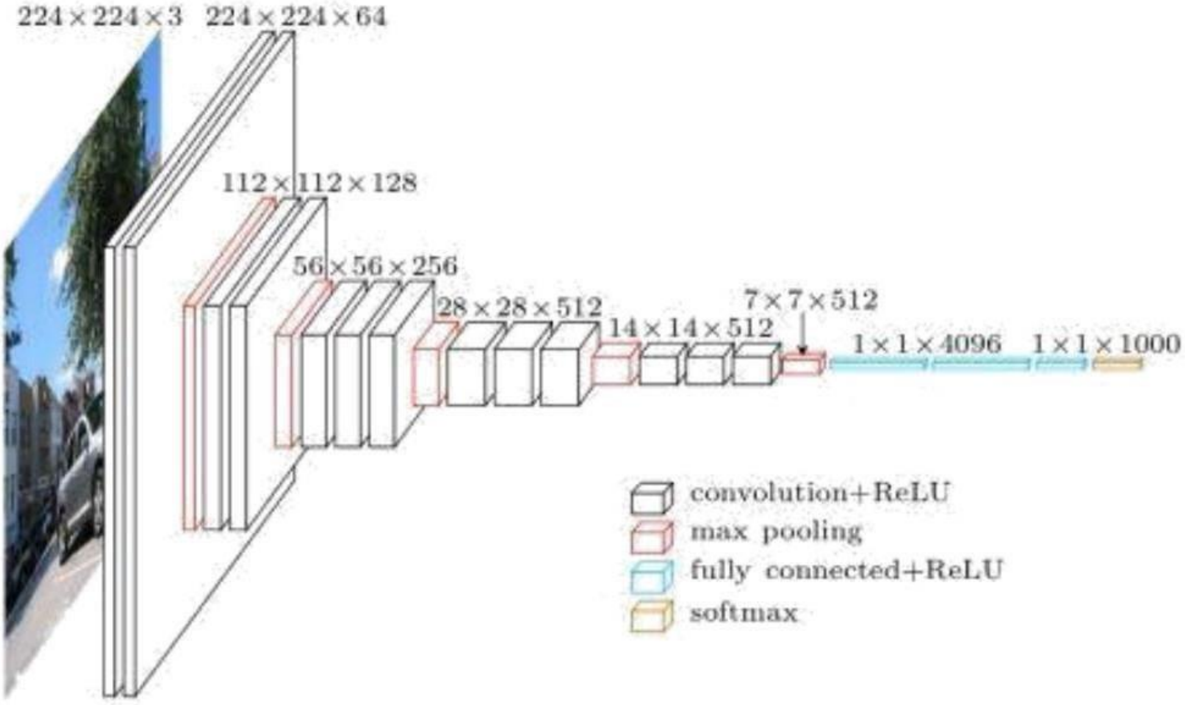
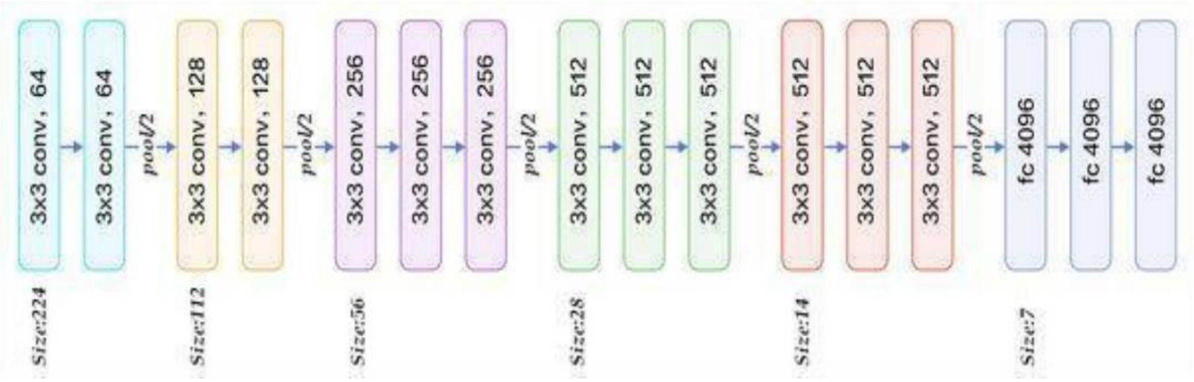


Fig 6.1.2: VGG-16 overview

6.1.2.1 Architecture



The network receives a three-dimensional image as input (224, 224, 3). The first two layers contain the same padding and 64 3*3 filter-size channels. Following a maximum stride pool layer (2, 2), two layers of 256 filter size convolution layers and filter size layers were added (3, 3). The max stride pooling layer (2, 2), which is comparable to the preceding layer, follows. Then there are 256 filters and two filter scale convolution layers (3, 3). After that, there are two sets of three convolution layers and a max pool layer. Each one has the same padding and 512 (3, 3) size filters. This image is subsequently transferred to a two-layer convolution stack.

In these convolution and max-pooling layers, we utilize a 3*3 filter instead of 11*11 in AlexNet and 7*7 in ZF-Net. Some of the layers used to modify the amount of input channels also require 1 pixel. After each convolution layer, a 1-pixel padding (identical padding) is applied to avoid the image's spatial feature.

After convolution and max-pooling layer stacking, we receive a (7, 7, 512) feature map. We flatten this output so that it becomes a (1, 25088) function vector. Following that, there are three fully connected layers: the first receives input from the last vector function and outputs a vector (1, 4096), the second layer outputs a vector size (1, 4096), and the output of the second fully connected layer is passed to the soft-max layer to utilize the vector classification.

6.1.2.2 Results and Challenges

VGG-16 was one of the top-performing architectures in the 2014 ILSVRC competition. With a 7.32 percent top-5 classification error, it came in second in the classification task (only behind GoogLeNet with a classification error of 6.66 percent). It also won the position error localization task with a score of 25.32 percent.

The following are some of VGG-16's challenges:

1. The training time for the original VGG-16 model was extended. On the Nvidia Titan GPU, it took 2-3 weeks of training.
2. The ImageNet dataset model is roughly 580 MB in size after training, therefore it consumes a lot of disc space and is inefficient.

6.1.3 XCEPTION

6.1.3.1 Original Depthwise Separable Convolution

Depth wise Separable CNN is a class of CNN that is widely used because they have fewer parameters than original CNN, which reduces overfitting and also, the model is faster to train and computationally cheaper.

Suppose there is an input image of size $m * m * c$ (m=height, n=width, c=channels). Suppose there are n filters of size $k * k * c$. The output size will be $p * p * n$.

Number of multiplications in one convolution operation = $k * k * c$ Total number of multiplications = $k * k * c * p * p * n = k2 * c * p2 * n$ In Original depth-wise separable convolutions, there are two operations:

There are two operations in original depth-wise separable convolutions:

1. Depthwise Convolution
2. Pointwise convolution

In Depthwise convolution, we use filters of size $k * k * 1$. It means for c channels, and we use c filters of size $k * k * 1$ where each filter is applied to 1 channel. Output shape will be $p*p*c$

Number of multiplications in one depth wise convolution operation = $k * k * 1$

Total number of multiplications = $k * k * 1 * p * p * c = k2 * c * p2$

After Depth wise convolution, we use pointwise convolution in which we use n filters of size 1x1xc. Output shape will be $p * p * n$

Number of multiplications in one pointwise convolution operation = $1 * 1 * n$

Total number of multiplications = $n * p * p * c = n * c * p2$

Total multiplications in Separable Depth wise Convolution = $k2 * c * p2 + n * c * p2$

Total multiplications in Separable Depth wise Convolution = $c * p2 * (k2 + n)$

Ratio of standard and Depth wise convolution operations = $c * p2 * (k2 + n) / k2 * c * p2 * n$

Let n = 100 filters and k = 5 so ratio = (1/100) + (1/25) = 0.05

It means depth wise convolution is 20 times faster than standard convolution.

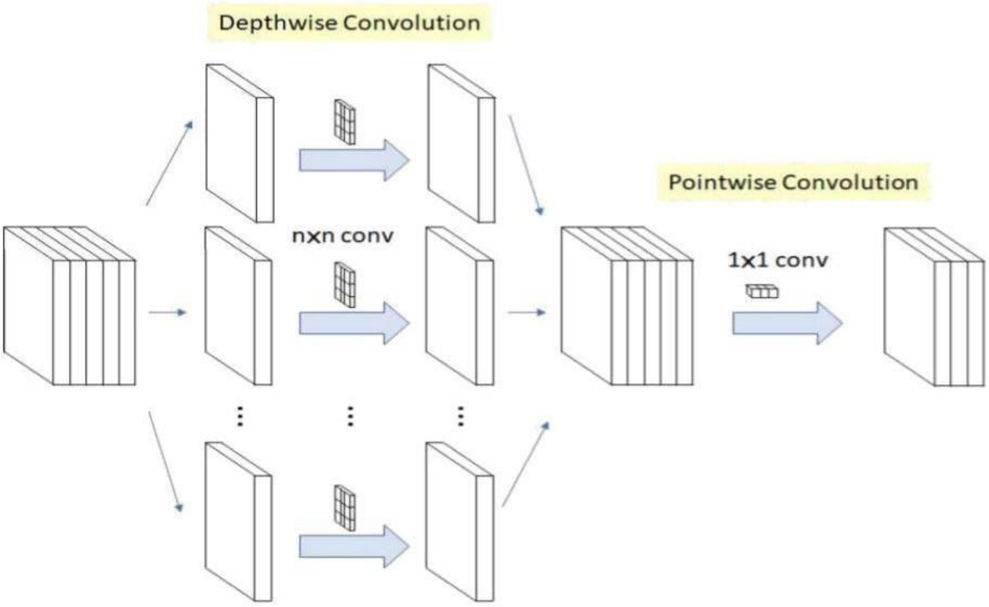


Fig 6.1.3.1: Original Depth wise Separable Convolution

6.1.3.2 Modified Depthwise Separable Convolution

Pointwise convolution is followed by depthwise convolution in the improved Depthwise separable convolution. With no intermediary non-linearity, Xception employs a modified Depthwise separable convolution.

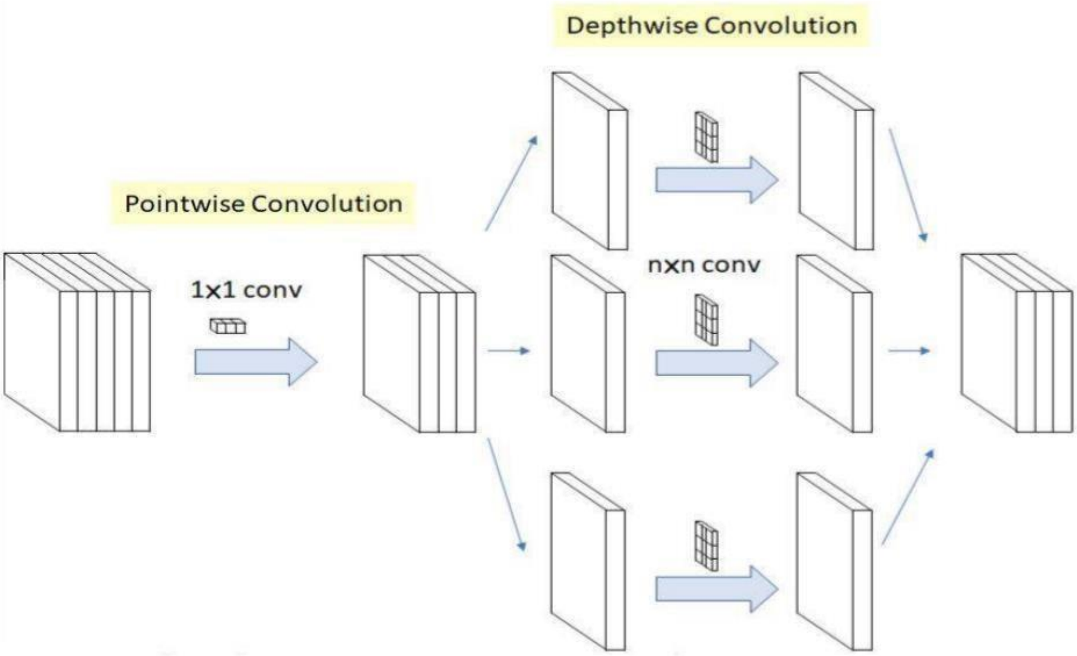


Fig 6.1.3.2: Modified Depth wise Separable Convolution

6.1.3.3 Architecture

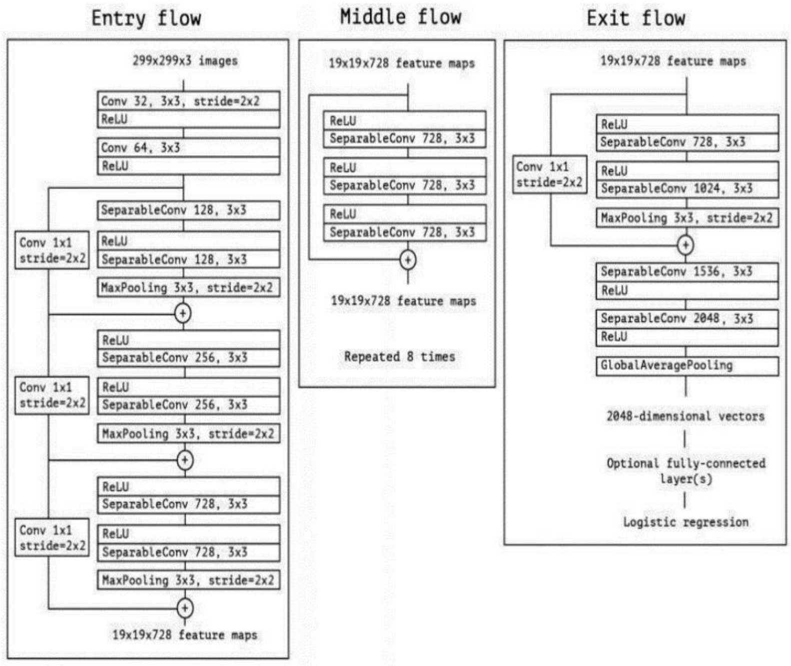


Fig 6.1.3.3: Xception Net

The Xception Model is divided into three sections, as shown in Fig. 6.1.3.3: Entry Flow, Middle Flow, and Exit Flow.

Two sets of convolution layers follow the ReLU layer in the Entry Block. Separate convolution layers, pooling layers, and skip connections are also included. The figure shows the filter size and stride of each layer. If we have knowledge from prior layers that may be relevant in subsequent layers, we employ skip connections.

Similarly, the middle block comprises three sets of ReLU, followed by eight repetitions of Separable Convolution. The figure depicts the input and output shapes. There is a worldwide average pooling of the exit flow (GAP). This layer replaces completely connected layers with GAP to decrease parameters. It reduces the output to 1x1xc for the input matrix of m x n x c. For each channel, it takes an average across the entire 2D matrix. The matrix is then flattened into a single vector and fed to a fully connected layer, which finally feeds it to the final layer, which uses logistic regression to identify classes. The model was trained on 1000 ImageNet Dataset classes, and the results are displayed in Figure 6.1.3.4.

Table 6.1.3.4: Comparison with other models

Models	Top-1 Accuracy	Top-5 Accuracy
VGG-16	0.715	0.901
ResNet-152	0.770	0.933
Inception V3	0.782	0.941
Xception	0.790	0.945

6.1.4 Recurrent Neural Network (RNN)

A RNN is a neural network in which the current step's output is used as an input for the following phase. It allows for a more complicated temporal activity. In typical neural networks, all inputs and outputs are independent of one another, but in circumstances like sequence generation, the prior words are necessary to predict the next word of a phrase, and so, the previous words must be remembered. When RNN was created, it was able to solve the issue with the aid of a Hidden Layer.

There are five different types of RNNs.

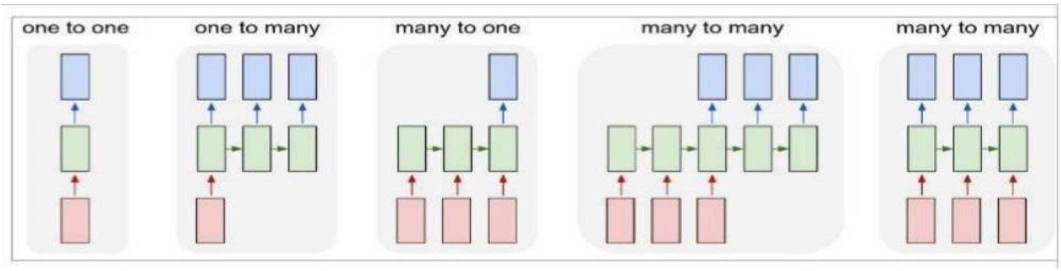


Fig 6.1.4: Different types of RNN

a) **One to One:** These are standard neural networks with fixed size input and fixed-size output

b) **One to Many:** Sequence output from a single input. Image captioning, for example, where the input is an image and the output is a variable-length sequence.

c) **Many to One:** Sequence Input and a single output. Example: Sentiment analysis where we determine whether a sentence is expressing positive comment or negative

d) **Many to Many:** Input and Output are both sequences, and their length can differ. Example: Video Captioning where input is video and output is a sentence

e) **Many to Many:** Input and Output are both sequences, and their length is the same. Example: Named Entity Recognition in which we have to locate and classify named entities in unstructured text.

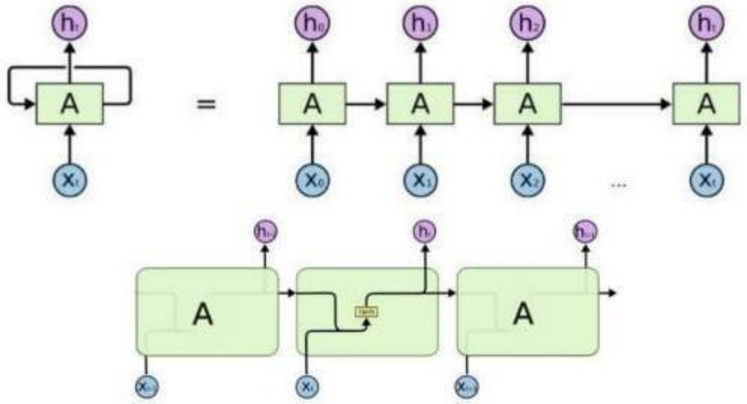


Fig 6.1.4: a) RNN Unit unfolded into different steps

b) A repeating module in one layer of RNN

As you can see in Fig 6.1.4 a), RNN is a single unit with input, output, and a self-input. RNN uses the hidden state and current input to produce the next hidden state and current output. The hidden state is fed into the unit only with the next input.

6.1.4.1 Terminologies Used in RNN

6.1.4.1.1 Hidden State

The Hidden state, which remembers certain information about a sequence, is the most important aspect of RNN. With each step, the hidden state is updated and 38tilize38 to forecast the next step's result. Variable-length input and output are possible in this concealed state. The output is generated using the concealed state.

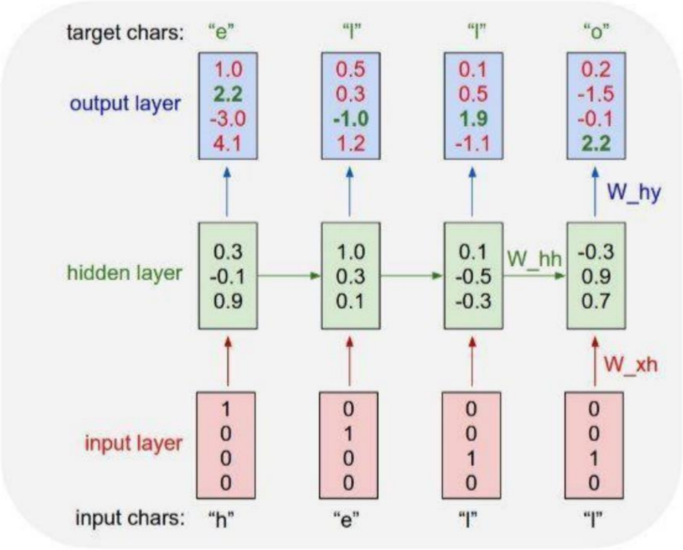


Fig 6.1.4.1.1: A RNN cell with 3 units in hidden layer which is used to predict next character in sequence

6.1.4.1.2 Timesteps

In Fig 6.1.4 a), there are multiple repetitions of one cell. We give some input x0, and it produces some state hidden h0. We then pass x1 and h0 to the same cell to produce h1 and so on. A timestep is a single occurrence of the cell. Timestep 40 means the cell will be repeated 40 times. Number of Timesteps in Input represented by Tx and output by Ty. For Tx = 1 and Ty = n, we have one to many types of RNN (Fig 6.1.4).

6.1.4.1.3 Loss Function

In RNN, we have a loss function that is comparable to that of a neural network. If we want a binary output, we employ binary cross-entropy in the output, just like in logistic regression. We employ categorical cross-entropy, which is similar to softmax loss, for multiclass classification. For number prediction, we 39tilize MAE (mean absolute error) or MSE (mean square error). Backpropagation using gradient descent is used to update the weights.

6.1.4.1.4 Backpropagation Through Time

In backpropagation, we calculate the derivative of cost with respect to weights and propagate it backward. Now, this derivative is used to update weights. In data like time series where we have timesteps, each timestep has one input and predicts one output. Error is calculated for each timestep and accumulated to update weights.

$$\frac{\partial E}{\partial W} = \sum_{t=1}^{T} \frac{\partial E_t}{\partial W}$$

$$W \leftarrow W - \alpha \frac{\partial E}{\partial W}$$

Each timestep can be visualized as a layer with the same parameters. It is backpropagation through time.

6.1.4.1.5 Vanishing Gradient

In Fig. consider input has 20 timesteps. It uses sigmoid and tanh activation, which is nonlinear, and their derivative is small values less than or equal to 1, which means the backpropagation signal will dampen while traversing backward for weight updation. The network will act like a 20-layer neural net with sigmoid and tanh activation. The gradient will reduce exponentially since it is multiplied at each layer and vanishes after a few timesteps. It is the vanishing gradient problem. The product of derivatives can explode if weights are initialized too large, but that can be identified quickly.

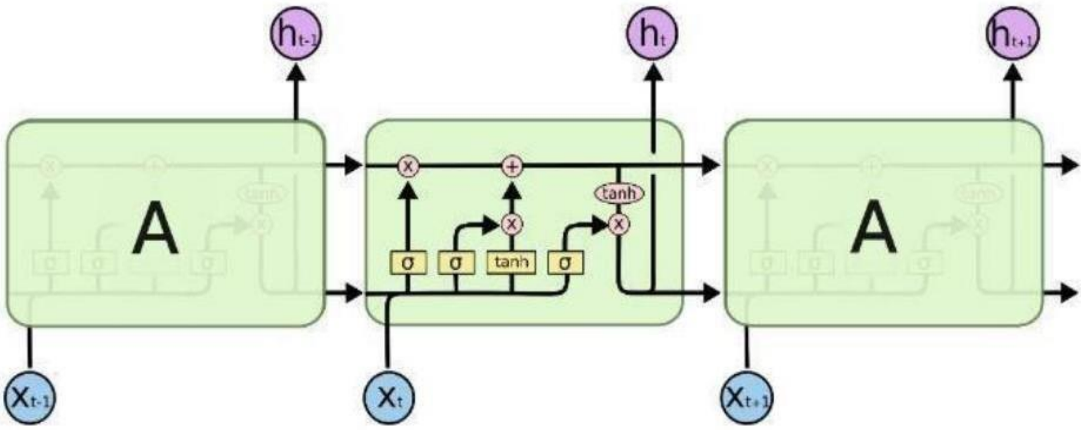
6.1.5 LSTM

RNN assists in connecting earlier information to current activities, such as understanding the context of the current sentence by using previous sentences. To complete a task, we may simply require recent knowledge. For example, in sentence generation, we may just require a few words to produce the next word, such as “I wake up at 6 a.m. and drink apple juice,” where the next word is “juice.” We only need a “drink apple” sentence to anticipate the word “juice.” RNN can learn to use current information in situations when the relevant information and the location where it is needed are both small.

In other circumstances, extra background is required to determine the final word. “I live in India and can write in,” for example. The following word could be “English” or “Hindi” or any of the other limited options, but it’s unlikely to be “German” or “Spanish.” The context “I live in India” is required in this scenario. It’s possible that the distance between the essential information and the location where it’s needed grows significantly. RNN is unable to learn to connect the information as the gap widens. The issue with RNN is this. They are incapable of dealing with long-term reliance. Short-term memory is another name for this.

6.1.5.1 Short Term Memory Solution

Long Short-Term Memory is the answer to short-term memory issues (LSTM). The cell state and numerous gates are key to the LSTM concept. The state of the cell is similar to a conveyor belt. Only small linear interactions run along the full chain of timesteps. The data can readily travel down the connection without interruption.



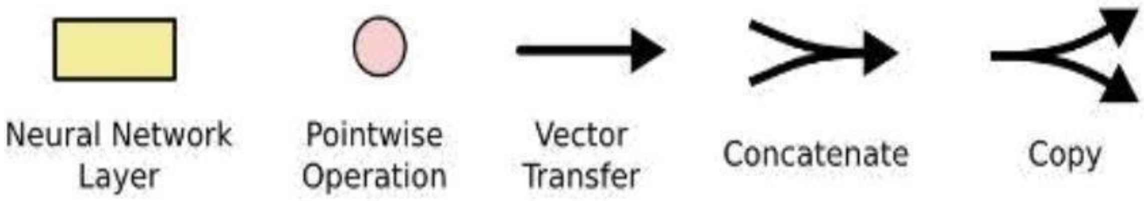


Fig 6.1.5.1: a) LSTM Unit b) Operations inside LSTM unit

6.1.5.2 Architecture

The cell state is the horizontal line that runs through the top of the cell. Using gates, the LSTM can add or remove information from the cell state. A neural net layer and pointwise multiplication make up gates. The sigmoid layer in gates outputs a value between 0 and 1, with 0 indicating that nothing should flow through the gates and 1 indicating that everything should.

Forget gate: The forget gate determines which information from the cell state should be discarded. It generates output between 0 and 1 for each number in the cell state using the previous concealed state and current input. One indicates that the value should be kept, whereas 0 indicates that it should be removed

Input gate: This gate determines what fresh data will be stored in the cell state. The value to update is determined by a sigmoid layer. One means vital, and zero means not necessary.
A tanh layer generates a vector of fresh candidate values that can be added to the state of the cell. We multiply the old cell state by ft (forget state output), then add it to the candidate it (input gate output).

Output gate: The next concealed state is determined by this gate. To output the sections we choose, we first transmit the prior hidden state and current input to the sigmoid layer, then multiply with the candidate passed by the tanh layer.

6.1.6 BASIC BLOCK ARCHITECTURE

6.1.6.1 Understanding the Architecture

The architecture used is a Sequence-to-Sequence Model. The process of generating captions from video will be splatted into the following phases:

Feature Extraction:

In this phase, frames are uniformly extracted from videos and fed into a CNN network to extract spatial features and encode them into vectors, which will be later fed into the encoder network. CNNs like VGG and Xception are used, which have high accuracy on the ImageNet dataset and are trained on 1000 different classes.

Encoding:

The next phase is encoding. LSTMs are used to encode a sequence of vectors extracted from frames into a single vector, which will be fed into the Decoder network. This Encoder network is responsible for extracting temporal features from the vectors. Several variants of the network can be used. Some of them include Multi-layer, Gated Recurrent Units (GRUs), Bidirectional Layers, Attention Layers, etc.

Decoding:

Decoding is the final step. The movie captions are generated using LSTMs, which encode one word at a time. Multi-layer, GRUs, and other versions of the decoding network are possible. The embedding layer uses a Glove Word encoding to encode each word into a dense vector.

Training:

The model is trained on a portion of VATEX Dataset with 100 videos with an average of 40 Multilingual Captions per video and a validation set of 20 videos. Only English captions are used for this purpose. This network allows encoding and decoding networks to learn simultaneously.

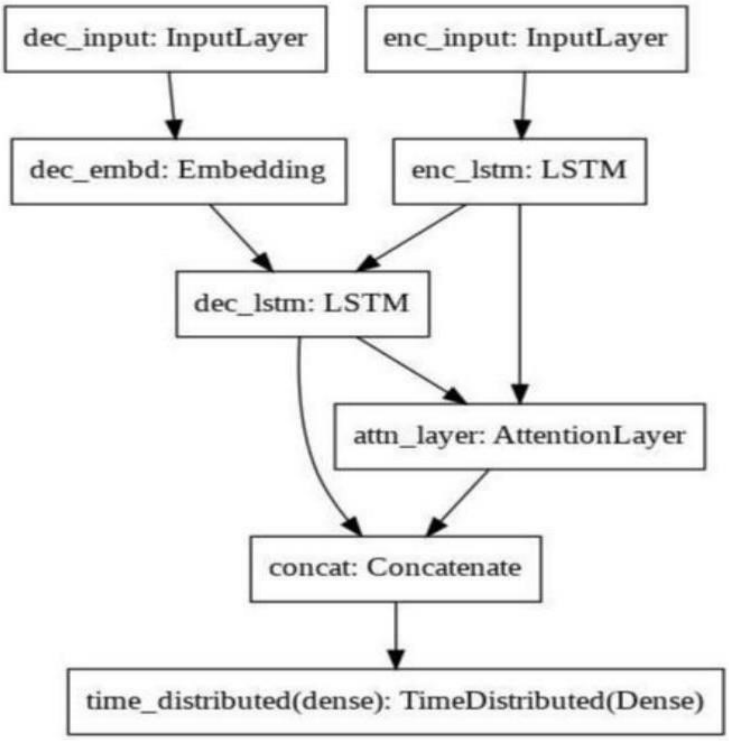
Generating Captions:

Two inference models are required to generate captions. They act like separate encoders and decoders for the above network. The first model’s output is one dense vector for each video and is input for the second model. The second model predicts the next word depending upon the current words predicted and the first model’s initial input. A newline character denotes the end of the sequence.

Sequence to sequence:

Frames are fed sequentially, and words are generated sequentially, allowing variable length frames as input and variable-length sentences as output. Sequence input of frames allows Encoder to learn temporal features of the video.

Network for training



Inference Models for generating captions

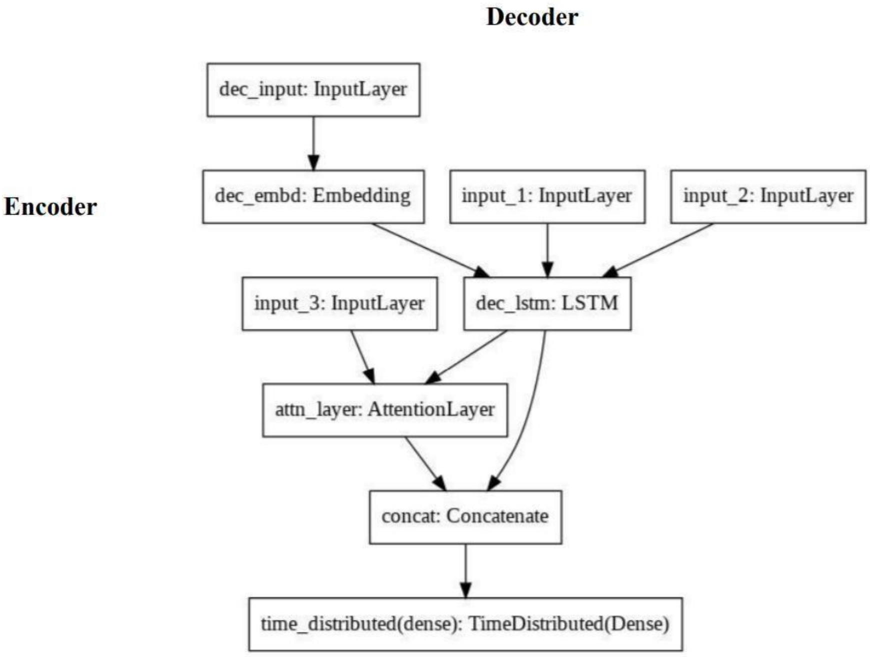


Fig 6.1.6.1: a) Sequence to Sequence Model b) Encoder Inference c) Decoder inference

6.1.7 TOOLS AND TECHNOLOGIES USED

6.1.7.1 Software Used

- Python 3.8
- Tensorflow 2.5.1
- Tensorflow.Keras
- Bash/Linux Terminal (Windows Terminal with Bash can also work)
- Skvideo 1.1.10
- OpenCV 4.4.0.42
- VSCode/PyCharm

6.1.7.2 Hardware Used

- Desktop/Laptop with Min Requirements (GPU, Screen, Modem)
- High Performance GPU with at least 4GB VRAM (Higher will provide faster results)
- (Future Scope) Security Surveillance Camera
- (Future Scope) Health Monitoring Device

6.1.7.3 Libraries Used

- **Cv2:**

The Python OpenCV library is used to perform necessary OpenCV loading and image reading operations. It also offers a range of other functions used in 38 this project, such as video capture, pre-processing and image transformation. Here it has been used to capture frames from video.

- **Numpy:**

This Python performance library deals with multidimensional arrays and scientific mathematics. It provides direct functions such as multiplication of the matrix, generation of a number sequence, etc. It was used here to serve as temporary storage for video frames, extracted features, etc. to aid in their processing.

- **Skvideo:**

Using Python, skvideo is designed for quick video processing. It provides an allin-one solution for video processing software at the research level. Skvideo.io is a plugin developed for using an Ffmpeg/LibAV backend to read and write images. A suitable probing tool (ffprobe, avprobe, or even media info) will be used to parse metadata from videos, depending on the backend available. Here it has been used to read video from the directory.

- **Keras:**

Keras is an open-source library that offers artificial neural networks, a Python interface. Keras includes a variety of implementations of common neural network building blocks like layers, targets, activation functions, optimizers, and a slew of other tools for working with picture and text data and simplifying the coding required to construct deep neural network code. In addition to standard neural networks, Keras supports convolutional and recurrent neural networks. Other typical utility layers like dropout, batch normalization, and pooling are supported. It also allows for distributed deep-learning model training on GPU and Tensor Processing Unit (TPU) clusters. It was utilized to preprocess data and create models in this case.

- **Shutil:**

The shutil module in Python offers many features for high-level file operations and file collections. It falls under the standard utility modules of Python. This module helps to simplify file and directory copying and removal operations. This saves the steps of opening, reading, writing, and closing files while no processing is actually performed. It is a utility module that can perform tasks such as copy, transfer, or delete directory trees.

6.1.7.4 About the code

The code is hosted on Github, and it consists of 2 parts: dataset and utils. In the dataset directory, we have the VATEX videos, the CSV file containing the descriptions of those videos, and the word embedding file. The extracted video features are also stored in this directory. The utils folder contains the python scripts used to preprocess the data, create the model, and extract the features from the videos.

csv_cleaner.py:

It is used for cleaning the csv file which includes choosing only relevant fields, removing duplicates and deleting null entries.

data_processing.py:

It is used for pre-processing the given video descriptions so that it can be fed into the LSTM. This includes removing articles, auxiliary verbs etc.

train_test_splitter.py:

It is used to divide the dataset, i.e., both the videos and the csv file into testing and training groups.

video_utils.py:

This script is used to extract features from each frame of the video using a suitable CNN model and store it for future processing.

6.1.7.5 Important Functions Used

- **skvideo.setFFmpegPath():**

It sets the directory path that contains both ffmpeg and ffprobe. Ffmpeg is important for processing/reading the video frame by frame.

- **skvideo.io.vread():**

To load any video into memory as one ndarray, we have used skvideo.io.vread. This feature assumes that you have enough memory to load the video and should be used only for small videos.

- **cv2.resize():**

It is used for transformation of images which may include scaling, zooming, shrinking etc. Here it has been used to keep the frames uniform in size.

- **pd.read_csv():**

The ability to read and manipulate csv files is a key feature of pandas. Not only can you read a csv file locally, but you can also read csv from a URL, or you can choose which columns to export so you don’t have to update the array later.

- **shutil.move():**

The file or directory (source) is recursively transferred to another location (destination) and the destination is returned. If there is already a destination directory, then src will be moved within that directory. If the destination is already present but is not a directory, it can be overwritten.

6.1.8 VATEX – A Large-Scale, High-Quality Multilingual Dataset for Video-and-Language Research.

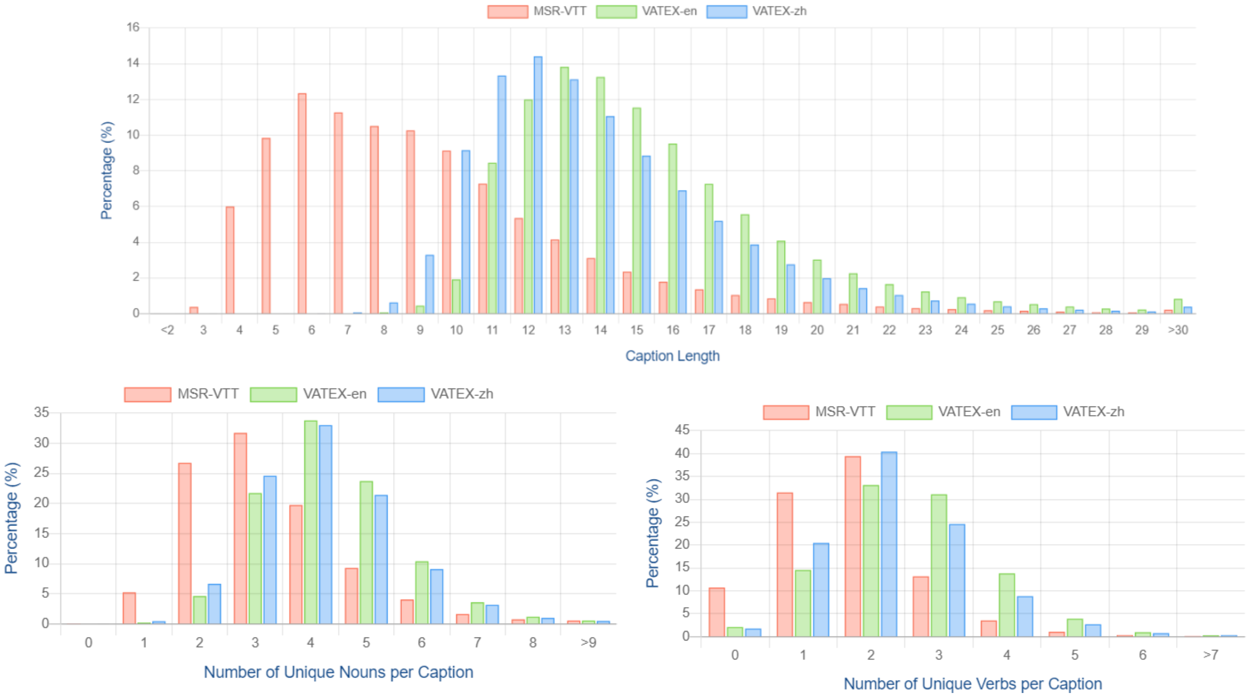
6.1.8.1 : Introduction

A brand-new, extensive multilingual video description dataset called VATEX has 825,000 captions in both Chinese and English for over 41,250 videos. The following are the main distinctive characteristics of VATEX. Firstly, it has descriptions in both Chinese and English at scale, which can help numerous multilingual research that are limited by datasets that are only available in one language. Second, VATEX has the highest number of clip-sentence pairs in the corpus. Each video clip has numerous distinct phrases attached to it, and each caption is distinct. Thirdly, with 600 total human activities covered, VATEX offers more thorough but representative video footage.

Additionally, they present two tasks for VATEX-based video-and-language research:

(1) Multilingual Video Captioning, which aims to describe a video in multiple languages using a compact, unified captioning model.

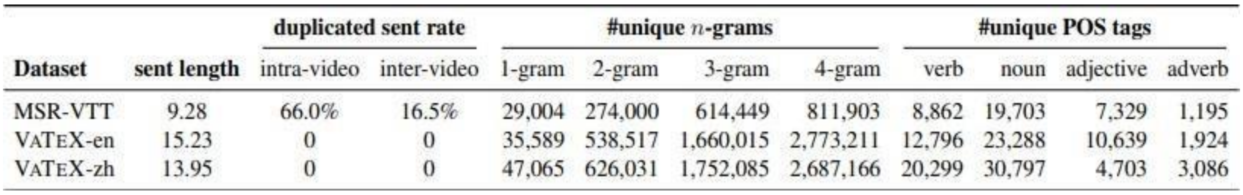
(2) Video-guided Machine Translation, which translates a description from the source language into the target language by utilising additional spatiotemporal context provided by the video.



[1] Figure 6.1.8.1: Statistical histogram distributions on MSR-VTT, VATEX-en, and VATEX-zh. Compared to MSR-VTT, the VATEX dataset contains longer captions, each with more unique nouns and verbs.

6.1.8.2 Dataset Analysis

Table 6.1.8.2 Average Data



		duplicated sent rate		#unique n-grams				#unique POS tags			
Dataset	sent length	intra-video	inter-video	1-gram	2-gram	3-gram	4-gram	verb	noun	adjective	adverb
MSR-VTT	9.28	66.0%	16.5%	29,004	274,000	614,449	811,903	8,862	19,703	7,329	1,195
VATEX-en	15.23	0	0	35,589	538,517	1,660,015	2,773,211	12,796	23,288	10,639	1,924
VATEX-zh	13.95	0	0	47,065	626,031	1,752,085	2,687,166	20,299	30,797	4,703	3,086

In Table 6.1.8.2, we briefly compare the overall statistics of the existing video description datasets. In this section, we conduct comprehensive analysis between our VATEX dataset and the MSR-VTT dataset, which is the widely-used benchmark for video captioning and the closest to VATEX in terms of domain and scale. Since MSR-VTT only has English corpus, we split VATEX into the English corpus (VATEX-en) and the Chinese corpus (VATEX-zh) for comparison. VATEX contains 413k English and 413k Chinese captions depicting 41.3k unique videos from 600 activities, while MSR-VTT has 200k captions describing 7k videos from 257 activities. In addition to the larger scale, the captions in both VATEX-en and VATEX-zh are longer and more detailed than those in MSR-VTT. The average caption lengths of VATEX-en, VATEX-zh, and MSR-VTT are 15.23, 13.95, and 9.28. To assess the linguistic complexity, we compare the unique n-grams and part-of-speech (POS) tags (e.g., verb, noun, adverb etc.) among MSR-VTT, VATEX-en and VATEX-zh which illustrates the improvement of VATEX over MSR-VTT and the difference between the English and Chinese corpora. Evidently, our VATEX datasets represent a wider variety of caption styles and cover a broader range of actions, objects, and visual scenes.

We also perform in-depth comparisons of caption diversity. First, as seen in Table 3, MSR-VTT faces a severe duplication issue in that 66.0% of the videos contains some exactly same captions, while our VATEX datasets are free of this problem and guarantee that the captions within the same video are unique. Not only within videos, but the captions in our VATEX datasets are also much more diverse even within the whole corpus, which indicates that our VATEX can also be a high-quality benchmark for video retrieval.

For a more intuitive measure of the lexical richness and caption diversity, we then propose the Type-Caption Curve, which is adapted from the type-token vocabulary curve but specially designed for the caption corpora here. The total number of captions and the number of distinct vocabulary words (types) are computed for each corpus. So, we plot the number of types against the number of captions for MSR-VTT, VATEX-en, and VATEX-zh (see Figure 4 where we choose 4-grams as the types). From these type-caption curves, inferences are drawn about lexical style or caption diversity (vocabulary use), as well as lexical competence (vocabulary size), so our VATEX datasets are shown to be more linguistically complex and diverse.

6.2 Requirement Analysis

A terminal or a windows cmd allows us to install the missing software requirements in our operating system that are required for the smooth functioning of the program. As this software is not yet optimized to be a stand-alone, the most optimal criteria for running this program effectively and smoothly are still high-end. Some of the hardware requirements are discussed in the table below:

Table 6.2.1: Hardware Requirements

Hardware Specific Requirements	**Version or Edition**
GPU (Graphic Processing Unit)	GeForce RTX 4090 Ti
Processor	Intel Core i7-12500H (12th gen H)
RAM	32GB+ (Recommended LPDDR5X 4267MHz)
Disk Space	512GB (NVMe BC711 NVMe SK Hynix)
Camera	Preferably I.P. Camera

In order to run this program, because we have met the hardware requirements, it is important to concentrate on the additional software requirements. The terminal uses the python package manager (pip) to install packages directly. The following will be needed for the smooth functioning of the software, as indicated in the table:

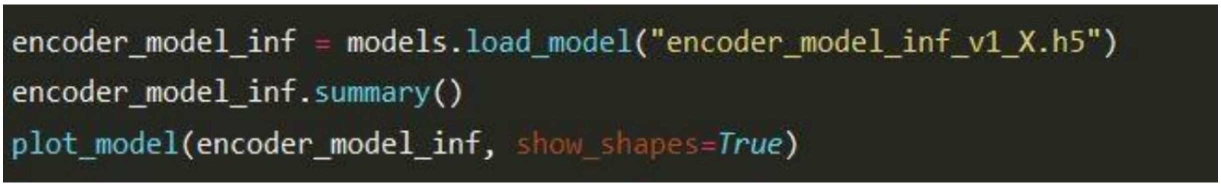
Table 6.1.2: Software Requirements

Software Specific Requirements	Version or Edition
Python	3.6.1+
Open CV	4.2.0+
Keras	Direct downloadable package
Shutil	Direct downloadable package
VATEX dataset	Direct downloadable package
Imutils	Direct downloadable package
Argparse	Direct downloadable package
Numpy	Direct downloadable package
Cv2	Direct downloadable package
pip	Pip3 or Updated package
Text Editor or IDE	PyCharm or VS code

6.3 Setting Up the Model

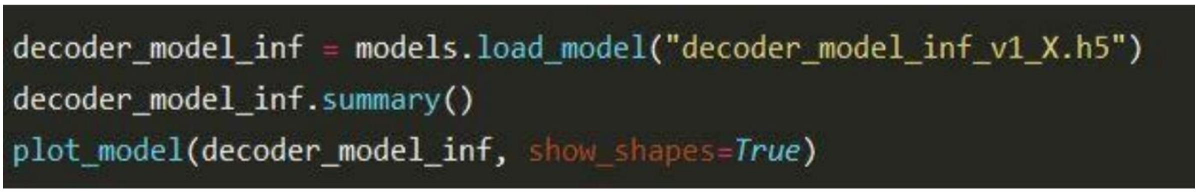
6.3.1 Loading the model

We've already trained the model (encoder and decoder) and stored it as a.h5 file for future usage. An encoder model and a decoder model make up the overall model. As a result, we must first load these two models:



```
encoder_model_inf = models.load_model("encoder_model_inf_v1_X.h5")
encoder_model_inf.summary()
plot_model(encoder_model_inf, show_shapes=True)
```

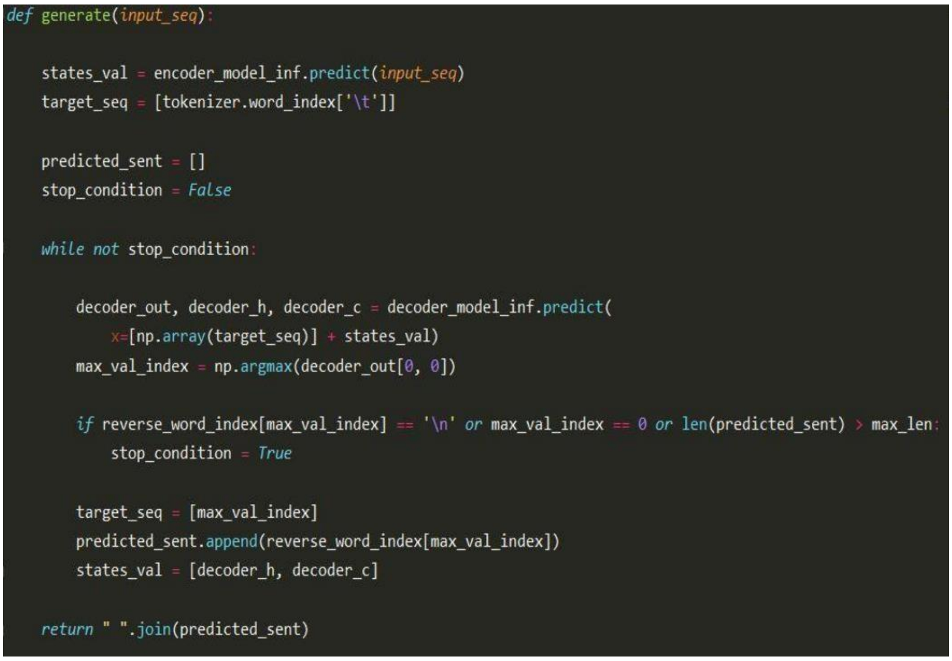
Fig 6.3.1.1: Loading the model (Encoder)



```
decoder_model_inf = models.load_model("decoder_model_inf_v1_X.h5")
decoder_model_inf.summary()
plot_model(decoder_model_inf, show_shapes=True)
```

Fig 6.3.1.2: Loading the model (Decoder)

Once these two models have been loaded successfully, we can use them to generate captions for the given videos. We have a generate () function that receives an input sequence as an argument and returns the predicted sentence. The function body of the generate function is displayed below:

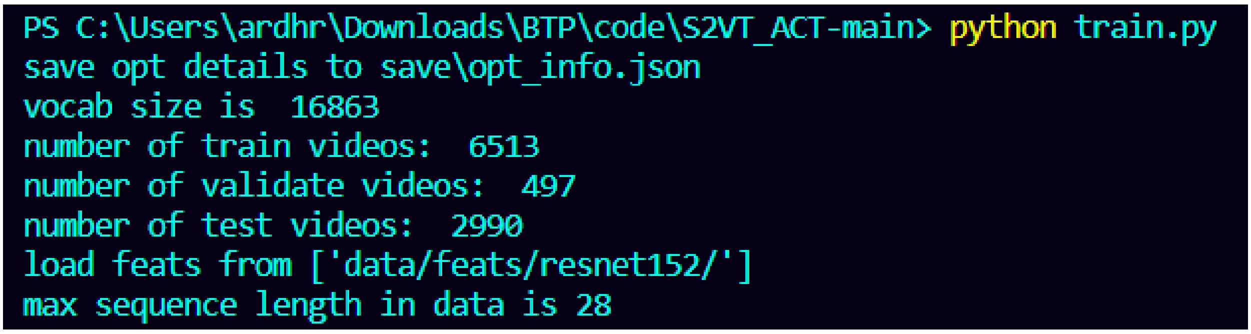


```
def generate(input_seq):

    states_val = encoder_model_inf.predict(input_seq)
    target_seq = [tokenizer.word_index['\t']]

    predicted_sent = []
    stop_condition = False

    while not stop_condition:

        decoder_out, decoder_h, decoder_c = decoder_model_inf.predict(
            x=[np.array(target_seq)] + states_val)
        max_val_index = np.argmax(decoder_out[0, 0])

        if reverse_word_index[max_val_index] == '\n' or max_val_index == 0 or len(predicted_sent) > max_len:
            stop_condition = True

        target_seq = [max_val_index]
        predicted_sent.append(reverse_word_index[max_val_index])
        states_val = [decoder_h, decoder_c]

    return " ".join(predicted_sent)
```

Fig 6.3.1.3: Setting up the model

6.3.2 Running the Model using Command Line

Navigate to the root directory of the project using windows cmd/ powershell / terminal



```
PS C:\Users\ardhr\Downloads\BTP\code\S2VT_ACT-main> python train.py
save opt details to save\opt_info.json
vocab size is  16863
number of train videos:  6513
number of validate videos:  497
number of test videos:  2990
load feats from ['data/feats/resnet152/']
max sequence length in data is 28
```

Fig 6.3.2.1: Running the model

6.4 Results Obtained

Some of the results that we have achieved so far are:

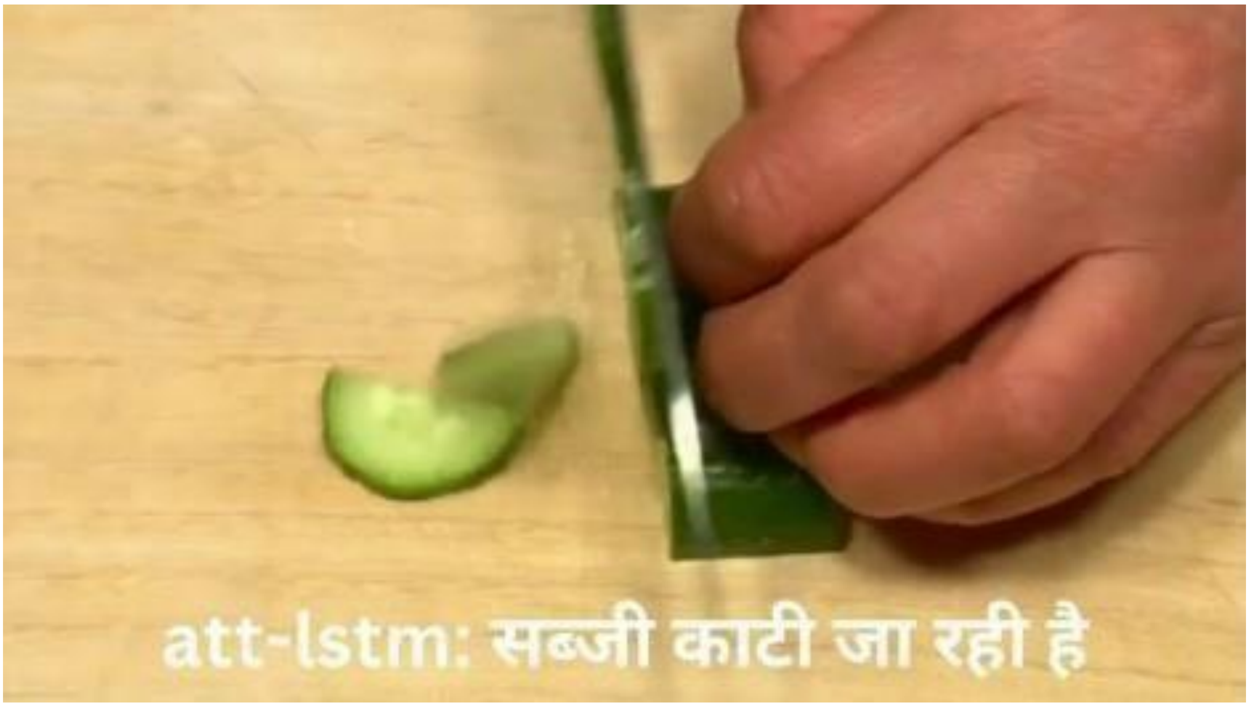
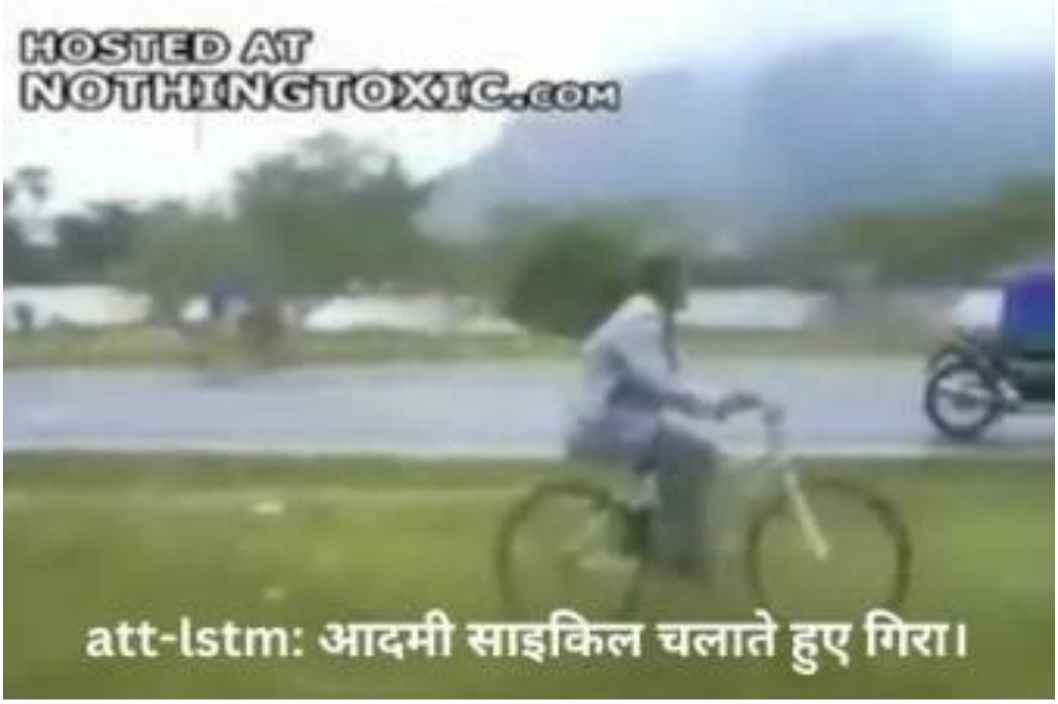


Fig 6.4.1: Output 1



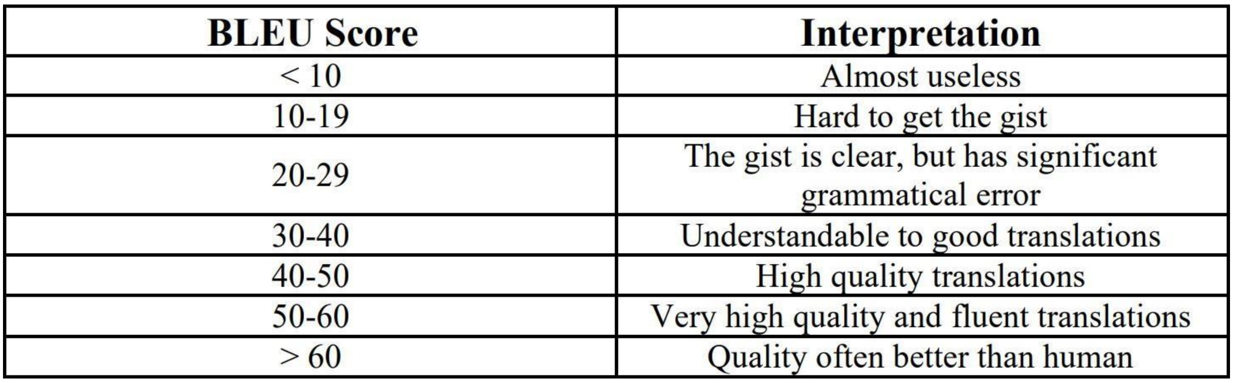
att-lstm: आदमी साइकिल चलाते हुए गिरा।

Fig 6.4.2: Output 2

6.5 Performance Analysis and Bleu Score

BLEU stands for Bilingual evaluation understudy. It is used to evaluate text output. Table 6.5.1 shows how Bleu Score is interpreted and based on the Bleu score; we analyses the performance of the model. It can be observed that we desire a higher bleu score. Our model does not account for grammatical errors but the gist of the caption so our desired Bleu score is any number greater than 30.

Table 6.5.1: Bleu Score interpretation



BLEU Score	Interpretation
< 10	Almost useless
10-19	Hard to get the gist
20-29	The gist is clear, but has significant grammatical error
30-40	Understandable to good translations
40-50	High quality translations
50-60	Very high quality and fluent translations
> 60	Quality often better than human

Bleu Score is calculated using the following formulas:

N-Gram Precision

$$p_n = \frac{\sum_{n-gram\,\epsilon\,hyp} count_{clip}(n-gram)}{\sum_{n-gram\,\in hyp} count\ (n-gram)}$$

$$Bleu = Be^{\frac{\sum_{n=1}^{N} p_n}{N}}$$

N grams refer to N words combinations next to each other in a sentence. In unigram precision, we take individual words. Brevity penalty is used for shorter sentences than the reference.

Our model resulted in a Bleu score of 24 which is significantly good for a simpler model and smaller dataset. This means that our model can understand the content of the video but has some grammatical errors.

Since our model does not account for grammatical errors, our model will evaluate to a low Bleu score if we consider bigram or trigram precision. Also, Bleu score does not take into consideration the synonyms of words.

In this penultimate chapter, we have delved into the practical facets of our research journey, covering implementation, results, and engaging in a thoughtful discussion. As we approach the final chapter, we will consolidate our insights and findings in the conclusion. Here, we aim to distill the key takeaways from our study, summarizing its contributions and implications. Additionally, we will outline potential avenues for future work, identifying areas that warrant further exploration and development. This concluding chapter serves as the culmination of our research, providing a comprehensive reflection on the undertaken study while paving the way for continued inquiry into the dynamic field we have explored.

VII. CONCLUSION & FUTURE WORK

In this concluding chapter, we bring our research journey to a culmination by summarizing key findings and insights in the conclusion. Reflecting on the implications of our study, we provide a cohesive understanding of the contributions made and the significance of our research. Furthermore, we outline potential directions for future work, identifying areas that warrant further exploration and development. This final chapter serves as a comprehensive endpoint, offering a synthesis of our research outcomes and paving the way for continued advancements in the field through the proposed avenues of future research.

- In conclusion, our research endeavors to address the evident gap in video descriptions for regional languages, with a specific focus on Hindi. Through a meticulous examination of existing literature, it became apparent that languages like Hindi are underrepresented in the realm of video descriptions. In response to this observation, we initiated the project titled "Generating Video Descriptions with Attention-Driven LSTM Models in Hindi Language" with the aim of augmenting accessibility and inclusivity of Hindi multimedia content.

- By harnessing advanced LSTM models and leveraging the VATEX dataset, our research sought to pioneer advancements in regional narrative video production. The strategic design of our approach not only contributes to the promotion of Indian language and culture but also establishes a precedent for exploring narrative films in other regional languages. Our work is driven by a commitment to fostering diversity, integration, and catalyzing broader advancements at the intersection of natural language processing and multitasking.

- The findings of our research underscore the effectiveness of our methodology, as evidenced by competitive or superior performance compared to state-of-the-art video captioning baselines such as BLEU and METEOR. This validation signifies a significant stride forward in enhancing the quality of video descriptions for regional languages, particularly Hindi, thereby making a noteworthy contribution to the field of regional language video captioning.

In essence, our research not only fills a critical void but also establishes a framework for future exploration in the domain of regional language video production. As we move forward, we anticipate that our findings will serve as a catalyst for continued innovation and improvement in the realm of natural language processing and multimedia accessibility, ultimately fostering a richer and more inclusive digital landscape.

PREVIOUSLY DONE WORK	**OUR CHANGES**
MSR-VTT Dataset	VATEX Dataset
LSTM	Attention-Based LSTM
Captions Outside Video	Burnt-in Caption
Less Human Activities are covered	More Human Activities Covered
Low Performance (BLEU Score)	High Performance (BLEU Score)

Table 7.1: Comparison Table

In this culminating chapter, we draw the curtain on our research report by encapsulating the journey undertaken. The conclusion succinctly summarizes the key findings and insights gleaned throughout our study, providing a cohesive understanding of the implications and contributions made to the field of investigation. Looking forward, we outline potential avenues for future work, identifying areas that beckon further exploration and development. As we bring our report to a close, this final chapter serves as a formal reflection on the culmination of our research efforts, offering both a summary of our achievements and a guide for prospective investigations in the dynamic landscape we have explored.

VIII. REFERENCES

[1]. Xin Wang, Jiawei Wu, Junkun Chen, Lei Li2=, Yuan-Fang Wang, William Yang Wang (2020) VATEX: A Large-Scale, High-Quality Multilingual Dataset for Video-and-Language Research, University of California, Santa Barbara, CA, USA, Byte Dance AI Lab, Beijing, China, arXiv:1904.03493v3.

[2]. Yongqing Zhu, Shuqiang Jiang (2019) Attention-based Densely Connected LSTM for Video Captioning, Key Lab of Intelligent Information Processing, Institute of Computing Technology, CAS, Beijing, 100190, China University of Chinese Academy of Sciences, Beijing, 100049, China, MM '19, October 21–25, 2019, Nice, France.

[3].Yong Qian, Yingchi Mao, Zhihao Chen, Chang Li, Olano Teah Bloh, Qian Huang (2023) Dense video captioning based on local attention, Key Research and Development Program of China, Grant/Award Number: 2022YFC3005401; Key Research and Development Program of Yunnan Province, Grant/Award Numbers: 202203AA080009, 202202AF080003; the Key Technology Project of China Huaneng Group, Grant/Award Number: HNKJ20-H46, DOI: 10.1049/ipr2.12819.

[4]. Md. Shahir Zaoad, M.M. Rushadul Mannan, Angshu Bikash Mandol, Mostafizur Rahman, Md. Adnanul Islam, Md. Mahbubur Rahman (2023) An attention-based hybrid deep learning approach for Bengali video captioning, Department of Computer Science and Engineering, Military Institute of Science and Technology, Dhaka 1216, Bangladesh.

[5]. Ayush Kumar Poddara, Dr. Rajneesh Rani (2023) Hybrid Architecture using CNN and LSTM for Image Captioning in Hindi Language, Dr B R Ambedkar National Institute of Technology, Jalandhar, Punjab, India, Peer-review under responsibility of the scientific committee of the International Conference on Machine Learning and Data Engineering 10.1016/j.procs.2023.01.049.

[6]. Alok Singh, Salam Michael Singha, Loitongbam Sanayai Meetei, Ringki Das, Thoudam Doren Singh, Sivaji Bandyopadhyay, (2023)] VATEX2020: pLSTM framework for video captioning, Department of Computer Science and Engineering, National Institute of Technology Silchar Assam, India, Center for Natural Language Processing, National Institute of Technology Silchar Assam, India.

[7].Daniela Moctezuma, Tania Ram´ırez-delReal, Guillermo Ruiz, Oth´on Gonz´alez-Ch´avez1 (2022) Video Captioning: a comparative review of where we are and

which could be the route, Centro de Investigaci´on en Ciencias de Informaci´on Geoespacial AC, Circuito Tecnopolo II , Aguascalientes, 20313, Mexico, Consejo Nacional de Ciencia y Tecnolog´ıa (CONACyT), Av. Insurgentes Sur 1582, Ciudad de Mexico, 03940, Mexico.

[8].Wanting Ji a, Ruili Wang b, Yan Tian b, Xun Wang (2021) An attention based dual learning approach for video captioning, School of Information, Liaoning University, Shenyang, China, School of Computer Science and Information Engineering, Zhejiang Gongshang University, Hangzhou, China.

[9]. Lianli Gao, Zhao Guo, Hanwang Zhang, Xing Xu and Heng Tao Shen, Senior Member, IEEE (2017) Video Captioning with Attention-based LSTM and Semantic Consistency, School of Computer Science and Engineering, University of Electronic Science and Technology of China, 611731. Hanwang Zhang is with Department of Computer Science, Columbia University, USA. Heng Tao Shen is the correspondence author, Citation information: DOI 10.1109/TMM.2017.2729019, IEEE.

[10]. Olivastri, Silvio & Singh, Gurkirt & Cuzzolin, Fabio. (2019). End-to-End Video Captioning. 1474-1482. 10.1109/ICCVW.2019.00185

[11]. Nayyer Aafaq, Ajmal Mian, Wei Liu, Syed Zulqarnain Gilani, and Mubarak Shah. 2019. Video Description: A Survey of Methods, Datasets, and Evaluation Metrics. ACM Comput. Surv. 52, 6, Article 115 (January 2020), 37 pages. DOI:https://doi.org/10.1145/3355390

[12]. Lee, Sujin & Kim, Incheol. (2018). Multimodal Feature Learning for Video Captioning. Mathematical Problems in Engineering. 2018. 1-8.

[13]. JX. Hua, X. Wang, T. Rui, F. Shao and D. Wang, "Adversarial Reinforcement Learning With Object-Scene Relational Graph for Video Captioning," in IEEE Transactions on Image Processing, vol. 31, pp. 2004-2016, 2022, doi: 10.1109/TIP.2022.3148868.

[14].Iashin, Vladimir, and Rahtu, E. 2020. Multi-modal dense video captioning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops.

[15]. J. Deng, L. Li, B. Zhang, S. Wang, Z. Zha and Q. Huang, "Syntax-Guided Hierarchical Attention Network for Video Captioning," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 32, no. 2, pp. 880-892, Feb. 2022, doi: 10.1109/TCSVT.2021.3063423

[16]. S. Liu, Z. Ren and J. Yuan, "SibNet: Sibling Convolutional Encoder for Video Captioning," in IEEE Transactions on Pattern Analysis and Machine Intelligence, v vol. 43, no. 9, pp. 3259-3272, 1 Sept. 2021, doi: 10.1109/TPAMI.2019.2940007.

[17]. Harsh Agrawal, Karan Desai, Xinlei Chen, Rishabh Jain, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. nocaps: novel object captioning at scale. arXiv preprint arXiv:1812.08658, 2018.

[18] Ozan Caglayan, Lo¨ıc Barrault, and Fethi Bougares. Multimodal attention for neural machine translation. arXiv preprint arXiv:1609.03976, 2016.

[19] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense- captioning events in videos. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), pages 706–715, 2017.

[20] Gunnar A Sigurdsson, Gul Varol, Xiaolong Wang, Ali ¨ Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In Proceedings of the 2016 European Conference on Computer Vision (ECCV), pages 510–526, 2016.

[21] Wajdi Zaghouani, Nizar Habash, Ossama Obeid, Behrang Mohit, Houda Bouamor, and Kemal Oflazer. Building an arabic machine translation post-edited corpus: Guidelines and annotation. In Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC), pages 1869–1876, 2016.

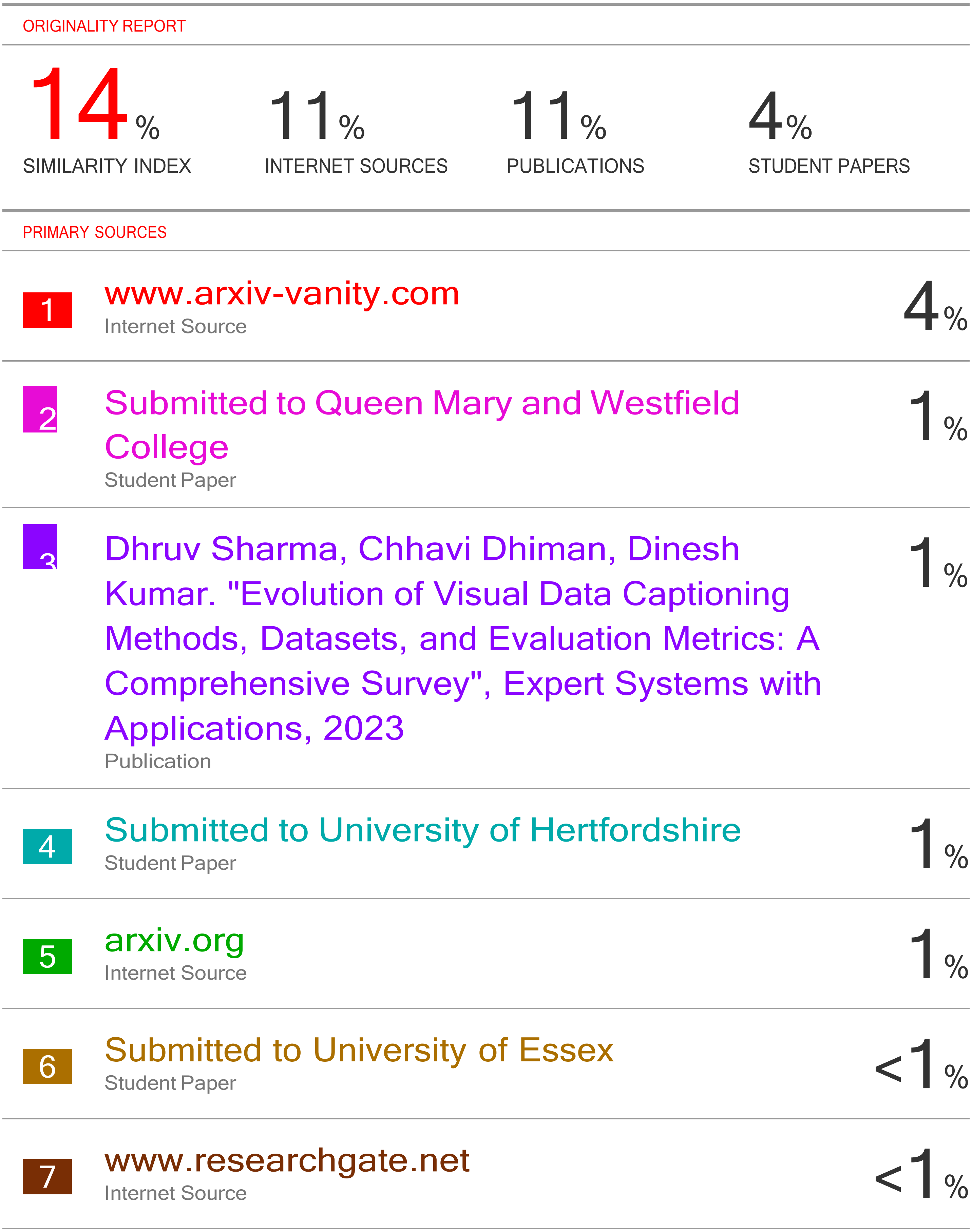
[22] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. Proceedings of the 28th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 4566–4575, 2015.

[23] Xirong Li, Xiaoxu Wang, Chaoxi Xu, Weiyu Lan, Qijie Wei, Gang Yang, and Jieping Xu. Coco-cn for cross-lingual image tagging, captioning and retrieval. IEEE Transactions on Multimedia, 2019.

[24] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402, 2012.\

[25] Ramon Sanabria, Ozan Caglayan, Shruti Palaskar, Desmond Elliott, Lo¨ıc Barrault, Lucia Specia, and Florian Metze. How2: a large-scale dataset for multimodal language understanding. In Proceedings of the Workshop on Visually Grounded Interaction and Language (ViGIL), 2018

IX. PLAGIAIRISM REPORT



ORIGINALITY REPORT

14%	11%	11%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.arxiv-vanity.com Internet Source	4%
2	Submitted to Queen Mary and Westfield College Student Paper	1%
3	Dhruv Sharma, Chhavi Dhiman, Dinesh Kumar. "Evolution of Visual Data Captioning Methods, Datasets, and Evaluation Metrics: A Comprehensive Survey", Expert Systems with Applications, 2023 Publication	1%
4	Submitted to University of Hertfordshire Student Paper	1%
5	arxiv.org Internet Source	1%
6	Submitted to University of Essex Student Paper	<1%
7	www.researchgate.net Internet Source	<1%

8	deepai.org Internet Source	<1 %
9	Atharva Shirode, Tejas Rathod, Parth Wanjari, Aparna Halbe. "Car Damage Detection and Assessment Using CNN", 2022 IEEE Delhi Section Conference (DELCON), 2022 Publication	<1 %
10	www.coursehero.com Internet Source	<1 %
11	origin.geeksforgeeks.org Internet Source	<1 %
12	savedelicious.com Internet Source	<1 %
13	"Computer Networks and Inventive Communication Technologies", Springer Science and Business Media LLC, 2021 Publication	<1 %
14	"The Fourth Industrial Revolution and Beyond", Springer Science and Business Media LLC, 2023 Publication	<1 %
15	Tanzila Kehkashan, Abdullah Alsaeedi, Wael M.S. Yafooz, Nor Azman Ismail, Arafat Al-Dhaqm. "Combinatorial Analysis of Deep Learning and Machine Learning Video	<1 %

	Captioning Studies: A Systematic Literature Review", IEEE Access, 2024 Publication	
16	Submitted to Imperial College of Science, Technology and Medicine Student Paper	<1 %
17	dokumen.pub Internet Source	<1 %
18	Hamza Haruna Mohanned, Selim Surucu, Roya Choupani. "Lung Inflammatory Classification of Diseases using X-ray Images", 2021 6th International Conference on Computer Science and Engineering (UBMK), 2021 Publication	<1 %
19	Arunaben Prahladbhai Gurjar, Shitalben Bhagubhai Patel. "chapter 1 Fundamental Categories of Artificial Neural Networks", IGI Global, 2022 Publication	<1 %
20	Submitted to Liverpool John Moores University Student Paper	<1 %
21	Wanting Ji, Ruili Wang, Yan Tian, Xun Wang. "An attention based dual learning approach for video captioning", Applied Soft Computing, 2021 Publication	<1 %

22	Ghazala Rafiq, Muhammad Rafiq, Gyu Sang Choi. "Video description: Acomprehensive survey of deep learning approaches", Artificial Intelligence Review, 2023 Publication	<1%
23	ir.uitm.edu.my Internet Source	<1%
24	www.shs-conferences.org Internet Source	<1%
25	Daniela Moctezuma, Tania Ramírez-delReal, Guillermo Ruiz, Othón González-Chávez. "Video captioning: A comparative review of where we are and which could be the route", Computer Vision and Image Understanding, 2023 Publication	<1%
26	e-pres.di.uoa.gr Internet Source	<1%
27	www.slideshare.net Internet Source	<1%
28	www.mdpi.com Internet Source	<1%
29	"Biometric Recognition", Springer Science and Business Media LLC, 2017 Publication	<1%
	Submitted to King's College	

30	Student Paper	<1%
31	Aditya D, R.G. Manvitha, C.L Revanth Mouli. "Detect-O-Thon: Identification of Infected Plants by using Deep Learning", Global Transitions Proceedings, 2021 Publication	<1%
32	Konstantinos Demertzis, Lazaros Iliadis, Panagiotis Kikiras. "Chapter 18 A Lipschitz - Shapley Explainable Defense Methodology Against Adversarial Attacks", Springer Science and Business Media LLC, 2021 Publication	<1%
33	Submitted to University of Leeds Student Paper	<1%
34	dspace.dtu.ac.in:8080 Internet Source	<1%
35	home.simula.no Internet Source	<1%
36	ds.inflibnet.ac.in Internet Source	<1%
37	link.springer.com Internet Source	<1%
38	Chiranjeevi Karri, Omar Cheikhrouhou, Ahmed Harbaoui, Atef Zaguia, Habib Hamam. "Privacy Preserving Face Recognition in Cloud	<1%

	Robotics: A Comparative Study", Applied Sciences, 2021 Publication	
39	deepvision.data61.csiro.au Internet Source	<1%
40	paperswithcode.com Internet Source	<1%
41	scholarworks.rit.edu Internet Source	<1%
42	Khaled Bayoudh, Fayçal Hamdaoui, Abdellatif Mtibaa. "Hybrid-COVID: a novel hybrid 2D/3D CNN based on cross-domain adaptation approach for COVID-19 screening from chest X-ray images", Physical and Engineering Sciences in Medicine, 2020 Publication	<1%
43	www.lrec-conf.org Internet Source	<1%
44	"Computational Intelligence for Engineering and Management Applications", Springer Science and Business Media LLC, 2023 Publication	<1%
45	"Computer Vision - ECCV 2022", Springer Science and Business Media LLC, 2022 Publication	<1%
46	Submitted to University of West London Student Paper	<1%

47	www.collegedekho.com Internet Source	<1%
48	Alok Singh, Thoudam Doren Singh, Sivaji Bandyopadhyay. "Attention based video captioning framework for Hindi", Multimedia Systems, 2021 Publication	<1%
49	Md. Shahir Zaoad, M.M. Rushadul Mannan, Angshu Bikash Mandol, Mostafizur Rahman, Md. Adnanul Islam, Md. Mahbubur Rahman. "An Attention-Based Hybrid Deep Learning Approach For Bengali Video Captioning", Journal of King Saud University - Computer and Information Sciences, 2022 Publication	<1%
50	Siddhant J. Buchade, Sachin Bhoite. "Comparative Study of ML Algorithms for Garbage Classification", Research Square Platform LLC, 2024 Publication	<1%
51	Zhi Chang, Dexin Zhao, Huilin Chen, Jingdan Li, Pengfei Liu. "Event-centric multi-modal fusion method for dense video captioning", Neural Networks, 2021 Publication	<1%
52	ijircce.com Internet Source	<1%

53	landbot.io Internet Source	<1%
54	pubmed.ncbi.nlm.nih.gov Internet Source	<1%
55	qubixity.net Internet Source	<1%
56	www.ijert.org Internet Source	<1%
57	www2.mdpi.com Internet Source	<1%
58	Muhammad Rafiq, Ghazala Rafiq, Gyu Sang Choi. "Video Description: Datasets & Evaluation Metrics", IEEE Access, 2021 Publication	<1%
59	Hamayun A. Khan. "DM-L Based Feature Extraction and Classifier Ensemble for Object Recognition", Journal of Signal and Information Processing, 2018 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		

X. IJISRT Accepted and IEEE Discover submitted

- IEEE Mangalore Subsection (R0011901) is organizing 8th IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (2024 IEEE DISCOVER).

  Paper ID – 40

- IJISRT (International Journal of Innovative Science and Research Technology) Paper ID - IJISRT24APR2695 (certified)

GENERATING VIDEO DESCRIPTIONS WITH ATTENTION-DRIVEN LSTM MODELS IN HINDI LANGUAGE

Naman*1, Harsh Nagar#2, Dhruv#3, Vansh Gupta#4

#Eelectronics & Communication (AI & ML),
Netaji Subhas University of Technology, New
Delhi, India - 110031

1naman.new15@gmail.com
2harshknagar@gmail.com
3ar.dhruv29@gmail.com
4vanshgupta434@gmail.com

Abstract - This research addresses the existing gap in video descriptions for regional languages, with a particular emphasis on Hindi. Motivated by a thorough review of available literature, it was observed that languages like Hindi are inadequately represented in this domain. Consequently, we initiated the project titled "Generating Video Descriptions with Attention-Driven LSTM Models in Hindi Language" to enhance accessibility and inclusion of Hindi multimedia content. Leveraging advanced LSTM models and utilizing the VATEX dataset, our objective is to pioneer advancements in regional narrative video production. By venturing into unexplored terrain, we not only contribute to the promotion of Indian language and culture but also establish a precedent for exploring narrative films in other regional languages. This research is strategically designed to foster diversity, integration, and propel broader advancements at the intersection of natural language processing and multitasking. Our findings demonstrate that our approach yields competitive performance when compared to state-of-the-art video captioning baselines such as BLEU and METEOR. This signifies the efficacy of our methodology in enhancing the quality of video descriptions, thereby contributing significantly to the field of regional language video captioning.

Keywords – Video description, attention-based LSTM, VATEX, Hindi language.

I. INTRODUCTION

Video description has gained attention due to advances in computer vision, NLP, and machine learning. It involves describing video content with natural language sentences, serving various purposes like human-robot interaction and accessibility for the visually impaired.

Two main approaches exist: template-based language models and sequence learning. Template models use predefined templates to structure sentences, ensuring consistent syntax; sequence learning directly translates video content into phrases by extracting features and generating subtitles based on data. An encoder-decoder network offers an efficient solution. The encoder processes the input video, producing a fixed-dimensional vector fed into the decoder, which generates words sequentially.

In the early 2010s, the field of video captioning and description generation witnessed a significant evolution, driven by advancements in computer vision and natural language processing. This transformative shift began with the introduction of large-scale datasets and neural network-based models. Key papers such as "Show and Tell: A Neural Image Caption Generator" by Vinyals et al. in 2015 played a pivotal role in pioneering the fusion of image analysis and language generation, setting the stage for modern video captioning. This progression from basic audio captioning in the 1990s to the sophisticated video captioning of the 2010s highlights the rapid growth of technology in enhancing media accessibility and usability. This literature survey explores the key developments that have shaped this dynamic field.

Captioning is the process of adding text to visual content, providing information about the content for various purposes. There are several types of captioning, each serving different needs and audiences. Here are some common types of captioning: closed captions, open captions, subtitles, live captions, *descriptive video service (DVS) or audio descriptions. There are sone major applications of videos captioning, i.e. security surveillance, improved indexing for search engine optimizer (SEO), text-to-speech of captions for visually impaired.

The landscape of video and image captioning is evolving with remarkable innovations highlighted in recent studies. The "Attention-based Densely Connected LSTM for Video Captioning" [2] underscores the pivotal role of contextual integration via Dense LSTM for enriched narrative construction in video captioning. In parallel, "Dense Video Captioning Based on Local Attention" [3] introduces a paradigm shift towards Dense Video Captioning Locally (DVCL), focusing on global feature dependence and enhancing word-frame correlation for precision. The realm of linguistic diversity in captioning is expanded through "Attention-Based Hybrid Deep Learning for Bengali Video Captioning" [4], setting new benchmarks in Bengali language video captioning with a hybrid deep learning approach. Similarly, "Hybrid Architecture using CNN and LSTM for Image Captioning in Hindi Language" [5] leverages the synergy between CNN and LSTM to achieve significant improvements in Hindi image captioning. The "pLSTM Framework for Video Captioning in VATEX2020" [6] marks a milestone with its pioneering framework, establishing a new benchmark in the VATEX2020 challenge. A broad perspective on the field's evolution and future directions is provided by "Video Captioning: A Comparative Review of Progress and Opportunities" [7], detailing the progress and identifying potential opportunities in video captioning. The "Attention-Based Dual Learning Approach for Video Captioning" [8] delves into a dual-learning paradigm, enriching the video captioning process by synergizing insights from both videos and generated captions. Lastly, "Video Captioning with Attention-based LSTM and Semantic Consistency" [9] enhances captioning accuracy by marrying attention mechanisms with LSTM, underscored by a focus on semantic consistency, showcasing the cumulative advancements and diverse methodologies propelling the field forward.

II. PREVIOUS RELATED WORK

A. Convolution Neural Network (CNN)

One type of deep, feed-forward artificial neural network is the convolutional neural network (CNN). CNNs are one of the most popular algorithms in deep learning. They are capable of achieving state-of-the-art results and have a wide range of applications in image recognition, speech recognition, and natural language processing. Convolutional Neural Networks (CNNs) have been widely used in image recognition tasks like object detection, classification, face recognition, and other computer vision problems. In CNN’s, we are not required to fix the length k of a feature map as a given value; instead, we can vary it for different layers and then weight each feature map by its receptive field size to define a feature map size of the layer. The Convolution Operation extracts the input image's high-level features, such as edges. It is unnecessary to limit ConvNets to a single Convolutional Layer. Typically, low-level features like edges, colour, gradient direction, and so on are captured by the first ConvLayer. The architecture adjusts to the high-level properties by adding layers, giving us a network that understands the photos in the dataset in a way that is similar to how we would.

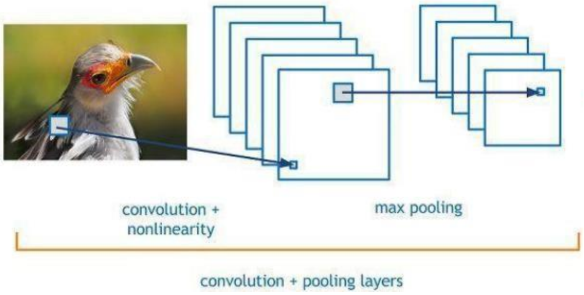


Fig. 1: Convolution layers

B. VGG-16

VGGNet-16 is made up of 16 convolutional layers and is particularly appealing due to its consistency. It is the most popular method for extracting features from photos at the moment. VGG can be achieved using transfer learning, in which the model is pre-trained on a dataset and the parameters are changed for improved precision. The ImageNet dataset contains 14 million photos divided into 1000 classes, and this model has a top-5 test precision of 92.7 percent. VGG-16 was one of the top-performing architectures in the 2014 ILSVRC competition. With a 7.32 percent top-5 classification error, it came in second in the classification task (only behind GoogLeNet with a classification error of 6.66 percent). It also won the position error localization task with a score of 25.32 percent.

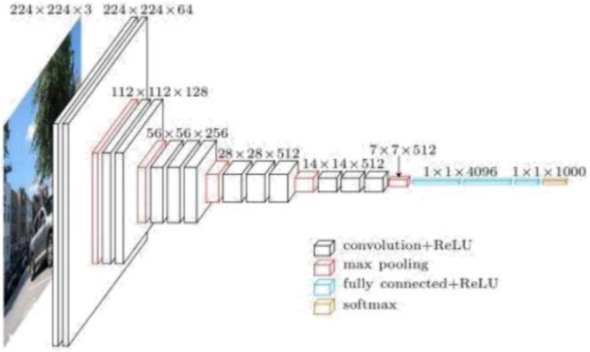


Fig. 2: VGG-16 overview

C. XCEPTION

The Xception Model is divided into three sections: Entry Flow, Middle Flow and Exit Flow.

Two sets of convolution layers follow the ReLU layer in the Entry Block. Separate convolution layers, pooling layers, and skip connections are also included. The figure shows the filter size and stride of each layer. If we have knowledge from prior layers that may be relevant in subsequent layers, we employ skip connections. Similarly, the middle block comprises three sets of ReLU, followed by eight repetitions of Separable Convolution. The figure depicts the input and output shapes. There is a worldwide average pooling of the exit flow (GAP). This layer replaces completely connected layers with GAP to decrease parameters. It reduces the output to 1x1xc for the input matrix of m x n x c. For each channel, it takes an average across the entire 2D matrix. The matrix is then flattened into a single vector and fed to a fully connected layer, which finally feeds it to the final layer, which uses logistic regression to identify classes. The model was trained on 1000 ImageNet Dataset classes.

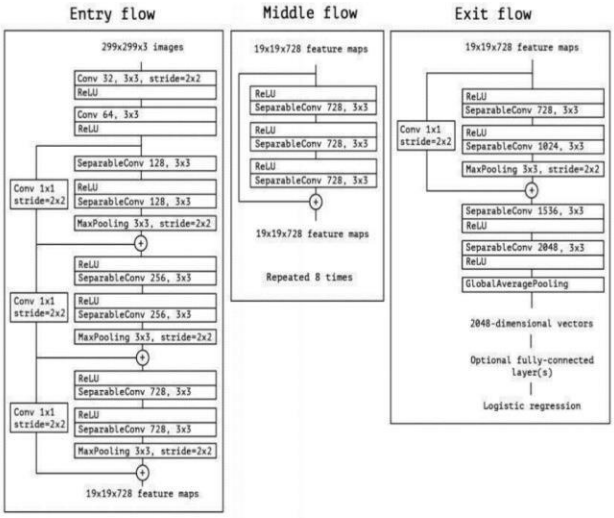


Fig. 3: XCEPTION net

D. Recurrent Neural Network (RNN)

A RNN is a neural network in which the current step's output is used as an input for the following phase. It allows for a more complicated temporal activity. In typical neural networks, all inputs and outputs are independent of one another, but in circumstances like sequence generation, the prior words are necessary to predict the next word of a phrase, and so the previous words must be remembered. RNN was born, and with the help of a Hidden Layer, it was able to address the problem. There are five different types of RNNs.

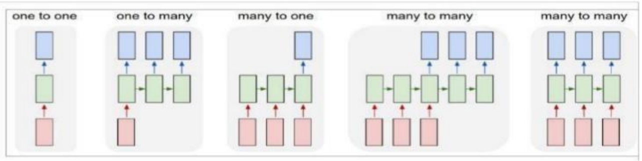


Fig. 4: Types of RNNs

E. LSTM

RNN assists in connecting earlier information to current activities, such as understanding the context of the current sentence by using previous sentences. To complete a task, we may simply require recent knowledge. For example, while creating sentences, we sometimes only need a few words to generate the following word. For example, in the sentence "I awake at 7 a.m. and drink banana shake," the word that comes after is "juice." All it takes is a phrase beginning with "drink banana" to predict the word "shake." When the relevant information and the place where it is needed are both tiny, RNN can learn to use the existing information. In other cases, further context is needed to decide which is the last word. For instance, "I can write in and live in India." "English," "Hindi," or any of the other few possible words may be the one that comes next, but "German" or "Spanish" are improbable choices. In this case, the context "I live in India" is necessary. It's possible that the distance between the essential information and the location where it's needed grows significantly. RNN is unable to learn to connect the information as the gap widens. The issue with RNN is this. They are incapable of dealing with long-term reliance. Short-term memory is another name for this.

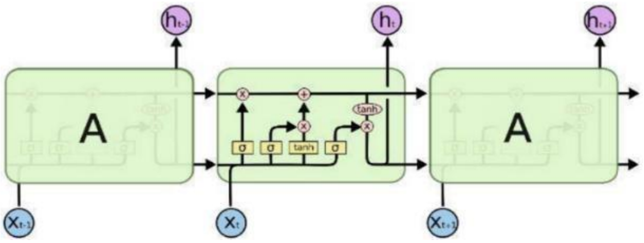


Fig. 5: LSTM unit

III. PROPOSED APPROACH & EXPERIMENNT

We were motivated to launch the project "Generating Video Descriptions with Attention-Driven LSTM Models in Hindi." We observed a dearth of research in regional languages like Hindi, especially in the realm of video description generation. Recognizing this uncharted territory, we aim to leverage advanced LSTM models and VATEX dataset to enhance accessibility and inclusivity in Hindi multimedia content. Our project not only contributes to the Hindi language but also sets a precedent for the exploration of other regional languages, ultimately fostering diversity and inclusivity in the multimedia landscape.

A. Dataset

VATEX - A Large-Scale, High-Quality Multilingual Dataset for Video-and Language Research [1]

A brand-new, extensive multilingual video description dataset called VATEX has 825,000 captions in both Chinese and English for over 41,250 videos. The following are the main distinctive characteristics of VATEX. Firstly, it has descriptions in both Chinese and English at scale, which can help numerous multilingual research that are limited by datasets that are only available in one language. Second, VATEX has the highest number of clip-sentence pairs in the corpus. Each video clip has numerous distinct phrases attached to it, and each caption is distinct. Thirdly, with 600 total human activities covered, VATEX offers more thorough but representative video footage.

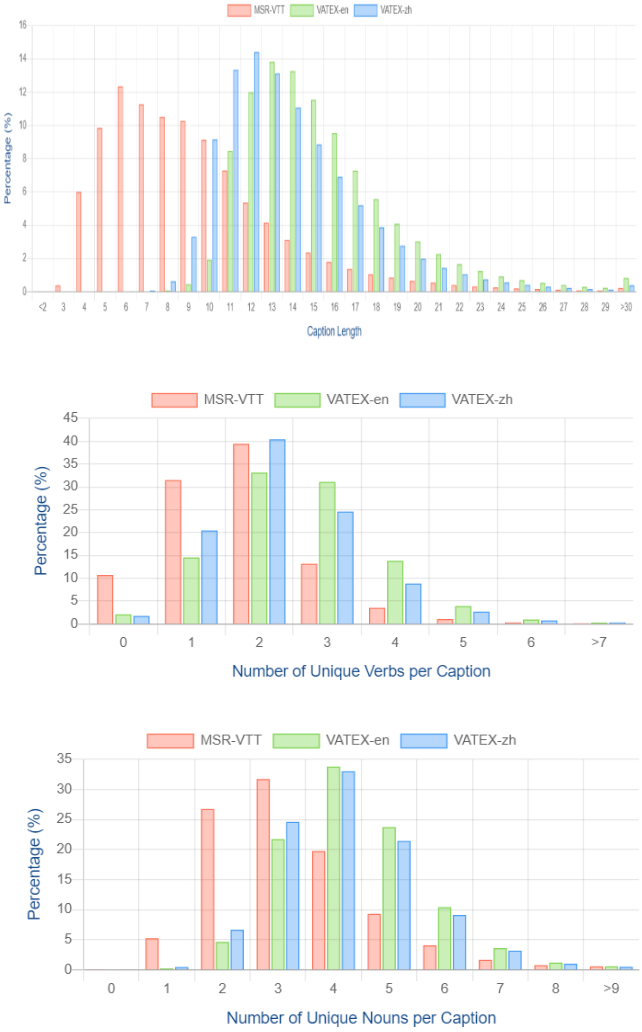
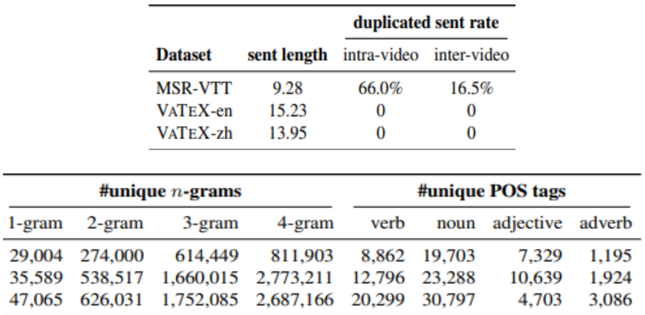


Fig. 6: Statistical histogram distributions on MSR-VTT, VATEX-en, and VATEX-zh. Compared to MSR-VTT, the VATEX dataset contains longer captions, each with more unique nouns and verbs.

B. Dataset analysis

Table 1: Average Data



Dataset	sent length	duplicated sent rate	
		intra-video	inter-video
MSR-VTT	9.28	66.0%	16.5%
VATEX-en	15.23	0	0
VATEX-zh	13.95	0	0

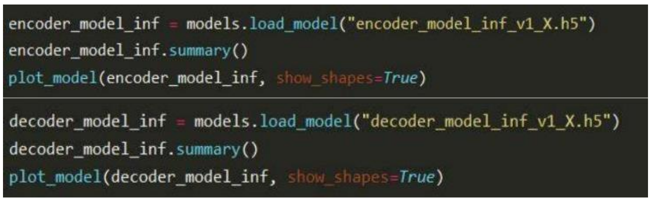
#unique n-grams				#unique POS tags			
1-gram	2-gram	3-gram	4-gram	verb	noun	adjective	adverb
29,004	274,000	614,449	811,903	8,862	19,703	7,329	1,195
35,589	538,517	1,660,015	2,773,211	12,796	23,288	10,639	1,924
47,065	626,031	1,752,085	2,687,166	20,299	30,797	4,703	3,086

In Table 1, We do a quick comparison of the overall statistics of the available datasets for video descriptions. We perform a thorough examination in this part between our VATEX dataset and the MSR-VTT dataset, which is the most similar to VATEX in terms of scale and domain and is the most used benchmark for video captioning. We divided VATEX into the English corpus (VATEX-en) and the Chinese corpus (VATEX-zh) for comparison because MSR-VTT only contains an English corpus. VATEX contains 413k English and 413k Chinese captions explaining 41.3k unique videos from 600 activities, whereas MSR-VTT has 200k captions explaining 7k films from 257 activities. In comparison to MSR-VTT, the subtitles in VATEX-en and VATEX-zh are larger, longer, and more comprehensive. VATEX-en, VATEX-zh, and MSR-VTT had average caption lengths of 15.23, 13.95, and 9.28, respectively. We compare the unique n-grams and part-of-speech (POS) tags (such as verb, noun, adverb, etc.) between MSR-VTT, VATEX-en, and VATEX-zh in order to evaluate the linguistic complexity. This comparison highlights the advantages of VATEX over MSR-VTT as well as the distinctions between the English and Chinese corpora. Our VATEX datasets encompass a greater range of activities, objects, and visual settings, and thus reflect a bigger diversity of caption styles.

We also carry out detailed analyses of caption diversity comparisons. First, as Table 1 illustrates, MSR-VTT suffers from a serious duplication problem because 66.0% of the movies have some identical duplicate captions. In contrast, our VATEX datasets are free from this issue and ensure that each caption inside a single video is distinct. Our VATEX datasets' captions show more variability both inside and throughout the corpus, indicating that our VATEX may be used as an improved benchmark for video retrieval. Next, as a more intuitive measure of the lexical richness and caption diversity, we propose the Type-Caption Curve, which is derived from the type-token vocabulary curve but particularly designed for the caption corpora here. The total number of captions and the number of distinct vocabulary items (types) are computed for each corpus. We therefore plot the number of sorts vs the number of captions for MSR-VTT, VATEX-en, and VATEX-zh (see Figure 4, where we use 4-grams as the types). These type-caption curves show how our VATEX datasets are more linguistically diverse and sophisticated, with information on lexical competency (vocabulary size) and lexical style or caption variation (vocabulary usage) discernible via inference.

C. Implementation

We've already trained the model (encoder and decoder) and stored it as a.h5 file for future usage. An encoder model and a decoder model make up the overall model. As a result, we must first load these two models:



```
encoder_model_inf = models.load_model("encoder_model_inf_v1_X.h5")
encoder_model_inf.summary()
plot_model(encoder_model_inf, show_shapes=True)

decoder_model_inf = models.load_model("decoder_model_inf_v1_X.h5")
decoder_model_inf.summary()
plot_model(decoder_model_inf, show_shapes=True)
```

Fig. 7: Loading the model (encoder + decoder)

Once these two models have been loaded successfully, we can use them to generate captions for the given videos. We have a generate () function that receives an input sequence as an argument and returns the predicted sentence. Navigate to the root directory of the project using windows cmd/ PowerShell / terminal.

D. Results Obtained

Among the outcomes we have attained are:

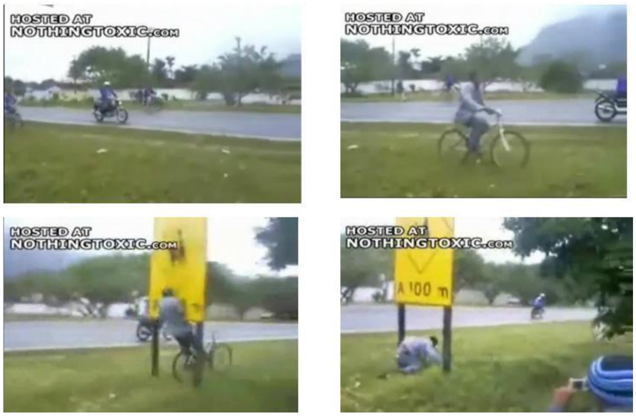
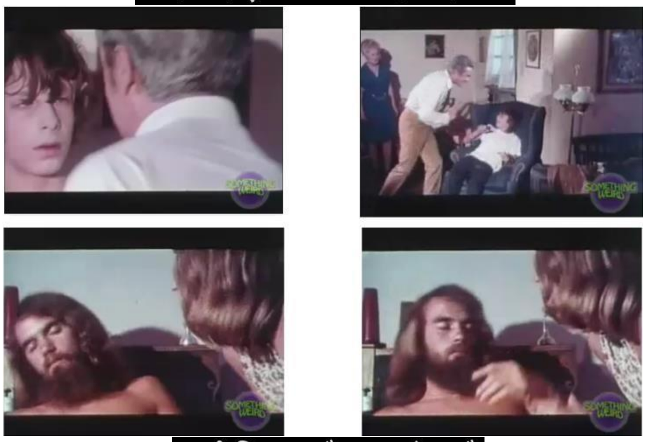

HOSTED AT NOTHINGTOXIC.com
HOSTED AT NOTHINGTOXIC.com
HOSTED AT NOTHINGTOXIC.com
HOSTED AT NOTHINGTOXIC.com

Fig. 8: Output 1 & 2

आदमी साइकिल चला कर गिर गया



आदमी गिर गया और दूसरा सो रहा है

E. Performance Analysis & BLEU Score

BLEU stands for Bilingual evaluation understudy. It is used to evaluate text output. Table 6.5.1 shows how Bleu Score is interpreted and based on the Bleu score; we analyze the performance of the model. It can be observed that we desire a higher bleu score. Our model does not account for grammatical errors but the gist of the caption so our desired Bleu score is any number greater than 30.

Table 2: Bleu Score interpretation

BLEU Score	Interpretation
<10	Almost useless
10-19	Hard to get the gist
20-29	The gist is clear but has grammatical error
30-39	Understandable to good translation
40-49	High quality translation
50-59	Very high quality and fluent translation
≥60	Quality often better than human

Bleu Score is calculated using the following formulas:

N-Gram Precision

$$p_n = \frac{\sum_{n-gram\, \epsilon\, hyp} count_{clip}(n-gram)}{\sum_{n-gram\, \in hyp} count\ (n-gram)}$$

$$Bleu = Be^{\frac{\sum_{n=1}^{N} p_n}{N}}$$

N grams refer to N words combinations next to each other in a sentence. In unigram precision, we take individual words. Brevity penalty is used for shorter sentences than the reference. Our model resulted in a Bleu score of 24 which is significantly good for a simpler model and smaller dataset. This means that our model can understand the content of the video but has some grammatical errors. Since our model does not account for grammatical errors, our model will evaluate to a low Bleu score if we consider bigram or trigram precision. Also, Bleu score does not take into consideration the synonyms of words.

IV. CONCLUSION

The findings of our research underscore the effectiveness of our methodology, as evidenced by competitive or superior performance compared to state-of-the-art video captioning baselines such as BLEU and METEOR. This validation signifies a significant stride forward in enhancing the quality of video descriptions for regional languages, particularly Hindi, thereby making a noteworthy contribution to the field of regional language video captioning. We bring our research journey to a culmination by summarizing key findings and insights in the conclusion. Reflecting on the implications of our study, we provide a cohesive understanding of the contributions made and the significance of our research. In essence, our research not only fills a critical void but also establishes a framework for future exploration in the domain of regional language video production. As we move forward, we anticipate that our findings will serve as a catalyst for continued innovation and improvement in the realm of natural language processing and multimedia accessibility, ultimately fostering a richer and more inclusive digital landscape. It can be observed that we desire a higher bleu score. Our model does not account for grammatical errors but the gist of the caption so our desired Bleu score is any number greater than 30. Our model resulted in a Bleu score of 24.

V. ACKNOWLEDGEMENT

We express sincere gratitude to our esteemed academic guide and mentor, Prof. RK Sharma, whose continuous guidance and unwavering support have been pivotal throughout our B.Tech. journey at Netaji Subhas University of Technology (NSUT), influencing not only this research but also various other course works. Special appreciation is extended to the dedicated professors and encouraging friends at NSUT, whose support has significantly contributed to the success of our project. Last but not least, our heartfelt thanks go to our parents and siblings for their boundless love and steadfast support. Their encouragement has been the driving force that made the completion of this work possible.

REFERENCES

[1]. Xin Wang, Jiawei Wu, Junkun Chen, Lei Li2=, Yuan-Fang Wang, William Yang Wang (2020) VATEX: A Large-Scale, High-Quality Multilingual Dataset for Video-and-Language Research, University of California, Santa Barbara, CA, USA, Byte Dance AI Lab, Beijing, China, arXiv:1904.03493v3.

[2]. Yongqing Zhu, Shuqiang Jiang (2019) Attention-based Densely Connected LSTM for Video Captioning, Key Lab of Intelligent Information Processing, Institute of Computing Technology, CAS, Beijing, 100190, China University of Chinese Academy of Sciences, Beijing, 100049, China, MM '19, October 21–25, 2019, Nice, France.

[3]. Yong Qian, Yingchi Mao, Zhihao Chen, Chang Li, Olano Teah Bloh, Qian Huang (2023) Dense video captioning based on local attention, Key Research and Development Program of China, Grant/Award Number: 2022YFC3005401; Key Research and Development Program of Yunnan Province, Grant/Award Numbers: 202203AA080009, 202202AF080003; the Key Technology Project of China Huaneng Group, Grant/Award Number: HNKJ20-H46, DOI: 10.1049/ipr2.12819.

[4]. Md. Shahir Zaoad, M.M. Rushadul Mannan, Angshu Bikash Mandol, Mostafizur Rahman, Md. Adnanul Islam, Md. Mahbubur Rahman (2023) An attention-based hybrid deep learning approach for Bengali video captioning, Department of Computer Science and Engineering, Military Institute of Science and Technology, Dhaka 1216, Bangladesh.

[5]. Ayush Kumar Poddara, Dr. Rajneesh Rani (2023) Hybrid Architecture using CNN and LSTM for Image Captioning in Hindi Language, Dr B R Ambedkar National Institute of Technology, Jalandhar, Punjab, India, Peer-review under responsibility of the scientific committee of the International Conference on Machine Learning and Data Engineering 10.1016/j.procs.2023.01.049.

[6]. Alok Singh, Salam Michael Singha, Loitongbam Sanayai Meetei, Ringki Das, Thoudam Doren Singh, Sivaji Bandyopadhyay, (2023)] VATEX2020: pLSTM framework

for video captioning, Department of Computer Science and Engineering, National Institute of Technology Silchar Assam, India, Center for Natural Language Processing, National Institute of Technology Silchar Assam, India.

[7]. Daniela Moctezuma, Tania Ram´ırez-delReal, Guillermo Ruiz, Oth´on Gonz´alezCh´avez1 (2022) Video Captioning: a comparative review of where we are and 59 which could be the route, Centro de Investigaci´on en Ciencias de Informaci´on Geoespacial AC, Circuito Tecnopolo II , Aguascalientes, 20313, Mexico, Consejo Nacional de Ciencia y Tecnolog´ıa (CONACyT), Av. Insurgentes Sur 1582, Ciudad de Mexico, 03940, Mexico.

[8]. Wanting Ji a, Ruili Wang b, Yan Tian b, Xun Wang (2021) An attention based dual learning approach for video captioning, School of Information, Liaoning University, Shenyang, China, School of Computer Science and Information Engineering, Zhejiang Gongshang University, Hangzhou, China.

[9]. Lianli Gao, Zhao Guo, Hanwang Zhang, Xing Xu and Heng Tao Shen, Senior Member, IEEE (2017) Video Captioning with Attention-based LSTM and Semantic Consistency, School of Computer Science and Engineering, University of Electronic Science and Technology of China, 611731. Hanwang Zhang is with Department of Computer Science, Columbia University, USA. Heng Tao Shen is the correspondence author, Citation information: DOI 10.1109/TMM.2017.2729019, IEEE.

[10]. Olivastri, Silvio & Singh, Gurkirt & Cuzzolin, Fabio. (2019). End-to-End Video Captioning. 1474-1482. 10.1109/ICCVW.2019.00185.

[11]. Nayyer Aafaq, Ajmal Mian, Wei Liu, Syed Zulqarnain Gilani, and Mubarak Shah. 2019. Video Description: A Survey of Methods, Datasets, and Evaluation Metrics. ACM Comput. Surv. 52, 6, Article 115 (January 2020), 37 pages. DOI: https://doi.org/10.1145/3355390.

[12]. Lee, Sujin & Kim, Incheol. (2018). Multimodal Feature Learning for Video Captioning. Mathematical Problems in Engineering. 2018. 1-8.

[13]. JX. Hua, X. Wang, T. Rui, F. Shao and D. Wang, "Adversarial Reinforcement Learning with Object-Scene Relational Graph for Video Captioning," in IEEE Transactions on Image Processing, vol. 31, pp. 2004-2016, 2022, doi: 10.1109/TIP.2022.3148868.

[14].Iashin, Vladimir, and Rahtu, E. 2020. Multi-modal dense video captioning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops.

[15]. J. Deng, L. Li, B. Zhang, S. Wang, Z. Zha and Q. Huang, "Syntax-Guided Hierarchical Attention Network for Video Captioning," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 32, no. 2, pp. 880-892, Feb. 2022, doi: 10.1109/TCSVT.2021.3063423.

[16]. S. Liu, Z. Ren and J. Yuan, "SibNet: Sibling Convolutional Encoder for Video Captioning," in IEEE Transactions on Pattern Analysis and Machine Intelligence, v vol. 43, no. 9, pp. 3259-3272, 1 Sept. 2021, doi: 10.1109/TPAMI.2019.2940007.

[17]. Harsh Agrawal, Karan Desai, Xinlei Chen, Rishabh Jain, Dhruv Batra, Devi Parikh, Stefan Lee, and Peter Anderson. nocaps: novel object captioning at scale. arXiv preprint arXiv:1812.08658, 2018.

[18] Ozan Caglayan, Lo¨ıc Barrault, and Fethi Bougares. Multimodal attention for neural machine translation. arXiv preprint arXiv:1609.03976, 2016.

[19] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense- captioning events in videos. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), pages 706–715, 2017.

[20] Gunnar A Sigurdsson, Gul Varol, Xiaolong Wang, Ali ¨ Farhadi, Ivan Laptev, and Abhinav Gupta. Hollywood in homes: Crowdsourcing data collection for activity understanding. In Proceedings of the 2016 European Conference on Computer Vision (ECCV), pages 510–526, 2016.

[21] Wajdi Zaghouani, Nizar Habash, Ossama Obeid, Behrang Mohit, Houda Bouamor, and Kemal Oflazer. Building an arabic machine translation post-edited corpus: Guidelines and annotation. In Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC), pages 1869–1876, 2016.

[22] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus- based image description evaluation. Proceedings of the 28th IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 4566–4575, 2015.

[23] Xirong Li, Xiaoxu Wang, Chaoxi Xu, Weiyu Lan, Qijie Wei, Gang Yang, and Jieping Xu. Coco-cn for cross-lingual image tagging, captioning and retrieval. IEEE Transactions on Multimedia, 2019.

[24] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402, 2012.

[25] Ramon Sanabria, Ozan Caglayan, Shruti Palaskar, Desmond Elliott, Lo¨ıc Barrault, Lucia Specia, and Florian Metze. How2: a large-scale dataset for multimodal language understanding. In Proceedings of the Workshop on Visually Grounded Interaction and Language (ViGIL), 2018.