SAVITRIBAI PHULE PUNE UNIVERSITY

A PRELIMINARY PROJECT REPORT ON

Image Captioning System using Deep Neural Network based on Encoder-Decoder Framework.

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

BACHELOR OF ENGINEERING
(Computer Engineering)(SEM-I)

SUBMITTED BY

Group ID : A19

Mr. Mayur Sopan Gadakh	Exam No: B190104240
Mr. Gaurav Bhima Chaudhari	Exam No: B190104219
Ms. Akanksha Bhausaheb Gaikwad	Exam No: B190104245
Ms. Shivanjali Anil Dhage	Exam No: B190104229

Under The Guidance of

Mr. R. S. Gaikwad

DEPARTMENT OF COMPUTER ENGINEERING
Amrutvahini College of Engineering, Sangamner
Amrutnagar, Ghulewadi - 422608

2023-24

AMRUTVAHINI COLLEGE OF ENGINEERING,SANGAMNER
DEPARTMENT OF COMPUTER ENGINEERING

CERTIFICATE

This is to certify that the Project Entitled

Image Captioning system using deep neural network based on encoder-decoder framework.

Submitted by

Group ID: A19

Mr. Mayur Sopan Gadakh	Exam No: B190104240
Mr. Gaurav Bhima Chaudhari	Exam No: B190104219
Ms. Akanksha Bhausaheb Gaikwad	Exam No: B190104245
Ms. Shivanjali Anil Dhage	Exam No: B190104229

are bonafide students of this institute and the work has been carried out by them under the supervision of Mr. R. S. Gaikwad and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of Bachelor of Engineering (Computer Engineering).

Mr. R. S. Gaikwad
Internal Guide
Dept. of Computer Engg.

Dr. R. G. Tambe / Dr. D. R. Patil
Project Coordinator
Dept. of Computer Engg.

Dr. S. K. Sonkar
H.O.D.
Dept. of Computer Engg.

Dr. M.A. Venkatesh
Principal
AVCOE Sangamner

Acknowledgment

Achievement is Finding out what you have been doing and what you have to do. The higher is submit, the harder is climb. The goal was fixed and We began with the determined resolved and put in a ceaseless sustained hard work. Greater the challenge, greater was our determination and it guided us to overcome all difficulties. It has been rightly said that we are built on the shoulders of others. For everything We have achieved, the credit goes to who had really help us to complete this project and for the timely guidance and infrastructure. Before we proceed any further, we would like to thank all those who have helped us in all the way through. We are thankful to our project guide **Mr. R. S. Gaikwad** for their guidance guidance care and support,which they offered whenever we needed it. We would like to thanks to project coordinator **Dr. R. G. Tambe** and **Dr. D. R. Patil** and also the respected Head of Department **Dr. S. K. Sonkar**.We would also thankful to Honourable Principal **Dr. M. A. Vankatesh** for his encouragement and support.

Abstract

This project is dedicated to advancing the field of image captioning through the creation of a sophisticated deep neural network framework. The architecture incorporates a robust "Convolutional Neural Network (CNN)" serving as the encoder by using EfficientNetV2 and a powerful "Recurrent Neural Network (RNN)" by using GRU (Gated Recurrent Unit) as the decoder. The model stands out in its ability to intricately capture visual details and comprehend the contextual relationships among objects within images. Within this neural network-based system, the CNN functions as an adept image feature extractor, adeptly preserving spatial information and facilitating object recognition. Complementarily, the RNN decoder is harnessed to predict words and seamlessly generate coherent, contextually relevant sentences based on the extracted image features. The primary objective of this project is to elevate the quality of image captions while enhancing the model's capability to navigate and interpret complex visual contexts. Through the fusion of EfficientNetV2 and GRU components, the framework is aimed to unlock new potentials in image understanding, paving the way for advancements in computer vision and natural language processing. The project's emphasis on improving caption quality and handling intricate visual scenarios underscores its commitment to pushing the boundaries of contemporary image captioning systems. The proposed neural network architecture, with its dual encoder-decoder structure, represents a significant leap forward in the quest for more sophisticated and context-aware image description generation. By addressing the intricacies of visual information and fostering a deeper understanding of contextual relationships, this project aspires to contribute to the evolution of image captioning methodologies and foster breakthroughs in the intersection of computer vision and natural language understanding.

Synopsis

AMRUTVAHINI COLLEGE OF ENGINEERING,
SANGAMNER

DEPARTMENT OF COMPUTER ENGINEERING

2023-2024

Project Synopsis

on

"Image Captioning system using deep neural network based on encoder-decoder framework"

BE Computer Engineering

BY

Group Id-B04

Mr. Mayur Gadakh (4136)

Mr. Gaurav Chaudhari (4117)

Ms. Akanksha Gaikwad (4141)

Ms. Shivanjali Dhage (4126)

Prof. R. S. Gaikwad
Project Guide
Dept. of Computer Engineering

12/9/23

Dr. D. R. Patil/ Dr. R. G. Tambe
Project Coordinator
Dept. of Computer Engineering

Prof. R. L. Paikrao
H.O.D
Dept. of Computer Engineering

Abbreviation

CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
GRU	Gated Recurrent Unit

List of Figures

List of Tables

INDEX

CHAPTER 1

INTRODUCTION

The field of computer vision and natural language processing has undergone a remarkable transformation with the emergence of automatic image captioning. This intricate process involves generating concise and contextually relevant textual descriptions for digital images, posing a challenge in discerning crucial objects, understanding their properties and interrelations, and articulating this understanding in coherent linguistic representation. Bridging this semantic gap requires a harmonious fusion of sophisticated computer vision techniques and robust language models derived from Natural Language Processing (NLP).

Traditionally, the synergy of machine learning and NLP has autonomously deciphered the content within images. Deep neural network strategies, notably Convolutional Neural Networks (CNNs) and advanced Recurrent Neural Networks (RNNs) like Long Short-Term Memory (LSTM), have addressed multifaceted challenges associated with image captioning, aiming to provide articulate natural language explanations that capture the essence of depicted scenes.

This study advances the state-of-the-art by proposing an innovative image captioning framework. At its core, the model leverages the robust feature extraction capabilities of "EfficientNetV2" as the CNN encoder, coupled with the sequential processing prowess of a Gated Recurrent Unit (GRU). Notably, our approach diverges from convention by introducing an attention mechanism tailored for images, enabling the model to selectively focus on specific regions, enriching the contextual depth of generated captions.

While preceding studies have explored diverse methodologies, our proposed model distinguishes itself through the strategic integration of EfficientNetV2 and GRU, coupled with a bespoke attention mechanism. The deliberate choice of the Flickr8k dataset for training and evaluation ensures a focused and cohesive analysis aligned with the intricacies of our model. The conventional encoder-decoder pipeline undergoes a paradigm shift, with EfficientNetV2 pre-trained on the dataset and a GRU serving as a decoder to construct meaningful image descriptions.

In contrast to traditional architectures like CNNs and LSTMs, our approach strategically leverages the unique strengths of EfficientNetV2 and GRU. The project's experimental phase encompasses rigorous training and meticulous testing on the Flickr8k

dataset, underpinned by comprehensive performance evaluations utilizing metrics such as BLEU scores. Furthermore, we draw inspiration from related datasets such as MS COCO, Flickr30k, and pertinent local datasets, ensuring a holistic assessment of the proposed image captioning generator.

A Fully Connected (FC) layer, also known as a Dense layer, is a fundamental building block in neural networks. In this layer, each neuron is connected to every neuron in the previous and next layers. These connections are characterized by weights, which are learned during training, and bias terms. The layer computes a weighted sum of its input, adds the bias, and applies an activation function to produce the output. FC layers are crucial for capturing complex relationships in data and are often used in various neural network architectures for tasks like classification and regression.

In the expansive realm of deep learning for image captioning, the synergistic marriage of EfficientNetV2 and Gated Recurrent Unit (GRU) architecture opens up a frontier of possibilities. EfficientNetV2, renowned for its superior efficiency in image feature extraction, seamlessly intertwines with the sequential processing prowess of GRU, creating a holistic model capable of unraveling the intricate narrative within visual content. As we embark on this journey, the aim is to transcend conventional boundaries, pushing the envelope of image captioning capabilities. This exploration not only represents a convergence of cutting-edge technologies but also holds the promise of fostering a deeper, more nuanced comprehension of visual data, ushering in a new era in the symbiotic relationship between deep learning, computer vision, and natural language understanding.

1. Image Captioning : Image captioning is a computer vision and natural language processing task where a model generates textual descriptions for given images. Utilizing neural networks, particularly encoder-decoder architectures, the model extracts features from the image using an encoder and then decodes these features into a coherent and descriptive sentence. Image captioning combines visual understanding and language generation, making it a multimodal task. This technology finds applications in accessibility tools, aiding visually impaired individuals, and enhances image indexing and retrieval systems. The

task involves teaching models to recognize objects, activities, and relationships within images, contributing to a more comprehensive understanding of visual content.

2. Encoder- decoder Architecture : An encoder-decoder model is a neural network architecture designed for sequence-to-sequence tasks. The encoder processes input sequences and compresses them into a fixed-size context or latent representation. The decoder then takes this representation and generates an output sequence step by step. It is commonly employed in tasks such as machine translation, where the input and output sequences can vary in length. The encoder captures the input's semantic information, enabling the decoder to produce meaningful outputs. This architecture is pivotal in natural language processing and image captioning, providing a structured approach to handle sequential data. It facilitates the transfer of information from input to output while accommodating varying lengths of sequences.

3. EfficientNetV2 : At the heart of our image captioning framework lies the EfficientNetV2, a state-of-the-art Convolutional Neural Network (CNN) architecture. EfficientNetV2 is renowned for its efficiency in terms of both computational resources and model parameters while maintaining superior performance in image classification tasks. Developed as an evolution of the EfficientNet architecture, version 2 incorporates novel techniques such as efficient scaling to balance model depth, width, and resolution. Its robust feature extraction capabilities make it an ideal candidate for the initial stage of our image captioning process, enabling the model to grasp salient features within images efficiently.

4. Gated Recurrent Unit (GRU) : Complementing the CNN encoder, our image captioning framework employs a Gated Recurrent Unit (GRU) as the sequential processing component. GRU is a type of Recurrent Neural Network (RNN) that excels in capturing long-term dependencies in sequential data. Unlike traditional RNNs, GRUs feature gating mechanisms that enhance their ability to retain essential information and discard irrelevant details. This makes

GRUs well-suited for the task of decoding and generating meaningful natural language descriptions of images based on the features extracted by the CNN encoder.

5. Teacher Forcing : Teacher forcing is a training approach in sequence-to-sequence models. During training, the model is fed true output sequences as input. This accelerates learning by guiding the model with correct sequences. In contrast, during inference, the model generates outputs based on its own predictions, potentially leading to exposure bias. Teacher forcing helps in capturing dependencies between input and output sequences. It is widely used in tasks like language translation and image captioning. The method aids faster convergence during training. However, the model might face challenges during testing due to discrepancies between training and inference conditions. To address this, a combination of teacher forcing and other techniques is often employed.

6. Attention Mechanism : A pivotal innovation in our image captioning framework is the incorporation of an attention mechanism tailored for images. Attention mechanisms have proven invaluable in natural language processing tasks, allowing models to focus on specific parts of input sequences when generating corresponding outputs. In the context of image captioning, our attention mechanism enables the model to selectively attend to relevant regions of the input image during the decoding phase. This adaptive focus enriches the contextual depth of the generated captions, ensuring that the model attends to the most pertinent visual features when constructing textual descriptions.

As the project unfolds, substantial contributions to the field are envisaged. By addressing the limitations of existing models and introducing novel components like EfficientNetV2 and GRU with an attention mechanism, our endeavor strives to set new benchmarks for the performance of automatic image captioning technology. This introduction lays the groundwork for a rigorous exploration into the nuances of image comprehension and caption generation, propelling the field towards enhanced capabilities and broader applicability.

1.1 PROJECT IDEA

The project endeavors to create an advanced image captioning system, leveraging cutting-edge deep learning techniques, including EfficientNetV2 and GRU. By combining these state-of-the-art models, the system aims to produce rich and contextually relevant textual descriptions for images. This venture sits at the intersection of computer vision and natural language processing, addressing the complex challenge of bridging visual understanding and linguistic expression. With a focus on automation, the system seeks to enhance accessibility for visually impaired individuals and improve image indexing for efficient retrieval. The utilization of EfficientNetV2 ensures robust feature extraction from images, while the GRU facilitates sequential information processing for fluent caption generation. This project contributes to the evolving landscape of multimodal AI systems, fostering a deeper integration of visual and textual understanding. The exploration of EfficientNetV2 and GRU techniques aligns with the forefront of deep learning, promising innovative strides in the field of image captioning.

1.2 MOTIVATION OF THE PROJECT

To bridge the gap between computer vision and natural language understanding through the development of an advanced image captioning system, this project is driven by the recognition that seamlessly integrating visual and linguistic capabilities holds immense potential for enhancing various applications in artificial intelligence and deep learning. The emphasis on image captioning stems from its status as a highly valuable skill, presenting substantial opportunities for career growth in the dynamically evolving fields of AI and deep learning. The project's motivation extends beyond technical innovation to address the practical implications of creating systems that can interpret and articulate visual content. By advancing image captioning techniques, the project aims to contribute to the broader landscape of AI applications, empowering individuals and industries with more intuitive and comprehensive tools for image analysis and understanding.

CHAPTER 2

LITERATURE SURVEY

2.1 LITERATURE SURVEY

1. This study introduces a deep neural network framework for automatic image captioning. It utilizes a Convolutional Neural Network (CNN) encoder to grasp spatial details and recognize objects in images, extracting features for creating a descriptive vocabulary. A Long Short-Term Memory (LSTM) decoder then predicts words and forms coherent sentences. The VGG-19 model serves as an image feature extractor, and the LSTM model processes sequences, producing a fixed-length output vector. The system is trained and tested on various open-source datasets like Flickr 8k, Flickr 30k, and MS COCO using Python, Keras, and TensorFlow. Performance is evaluated using the BLEU metric.[1]

2. Image captioning is evolving as an interesting area of research that involves generating a caption or describing the content in the image automatically. The idea behind image captioning is to make the computer perceive a given image like a human mind leading to automatic description. Image captioning is a challenging task that involves capturing semantically correct information and expressing in a simple sentence. A large number of methods have been proposed in the recent past, and we aim to do a comprehensive survey in the different deep learning algorithms used in image captioning based on the method framework.[5]

3. Image captioning is one of the most recent challenges that caught the interest of the computer vision community as well as the Natural Language Processing community. Recently, the tedious task of image captioning has attained quite notable progress by using numerous techniques. The primary goal of this paper is to study existing Deep Learning techniques for Image Captioning. We have discussed a convolutional neural network-based Image Caption generation model and the salient steps involved in it. We have also discussed dataset and evaluation metrics widely used in fundamental systems.[3]

4. This research explores machine learning algorithms for image and natural language processing, integrating existing packages. It implements an algorithm

generating comprehensive sentences from images. After requirements analysis, a bibliographic study informed model selection. A composite model, employing transfer learning in a deep convolutional neural network for feature extraction and a recurrent neural network for descriptions, was designed using Keras with TensorFlow. The result is a trained model capable of describing images in natural language.[2]

5. Image captioning is a process of automatically describing an image with one or more natural language sentences. In recent years, image captioning has witnessed rapid progress, from initial template-based models to the current ones, based on deep neural networks. This paper gives an overview of issues and recent image captioning research, with a particular emphasis on models that use the deep encoder-decoder architecture. We discuss the advantages and disadvantages of different approaches, along with reviewing some of the most commonly used evaluation metrics and datasets.[4]

Sr. No.	Paper Title	Year of Publication	Method Algorithm Used
1	Implementing Deep Neural Network Based Encoder-Decoder Framework for Image Captioning	2022	CNN and RNN
2	Comprehensive Comparative Study on Several Image Captioning Techniques Based on Deep Learning Algorithm	2022	CNN and RNN
3	An empirical study of image captioning using deep learning	2021	RNN or Transformers
4	Deep Learning Techniques for Automated Image Captioning	2020	CNN and RNN
5	An overview of image caption generation methods	2019	CNN

Table 2.1: Comparative Analysis

CHAPTER 3

PROBLEM DEFINITION AND SCOPE

3.1 PROBLEM STATEMENT

Implementing Deep Neural Network Based Encoder-Decoder Framework for Image Captioning using EfficientNetV2 as encoder and GRU as decoder.

3.1.1 Goals and objectives

Goal and Objectives:

- To study the deep learning techniques like CNN and RNN.
- To develop a deep-learning based image caption generator that can accurately describe the contents of an image in natural language.
- To create a user-friendly interface for interacting with the image captioning system.

3.1.2 Statement of scope

- Automatic image captioning using deep neural network encoder-decoder frameworks has extensive potential.
- It can enhance accessibility, content indexing, e-commerce, healthcare, education, and more by generating descriptive image captions.
- This technology streamlines processes, improves user experiences, and finds applications across diverse domains.

3.2 SOFTWARE CONTEXT

- The project will involve the development of software components for image captioning using EfficientNetV2 and GRU.
- The software will include modules for image feature extraction, natural language processing, and model integration.
- Additionally, the project will utilize relevant libraries, frameworks, and tools for deep learning, image processing, and natural language generation.

3.3 MAJOR CONSTRAINTS

- **Computational Resources :** Limited computational power and hardware may impact the speed and scalability of the model training and image caption generation processes, especially when dealing with large datasets like Flickr8k.
- **Data Availability :** In Flickr8k dataset the quality and diversity of these datasets may still pose challenges in terms of data preprocessing, management, and ensuring they are suitable for your specific deep learning models.
- **Time Frame :** Meeting project deadlines within the academic semester, while working with multiple datasets and conducting rigorous evaluations, is a crucial constraint, as it may affect the extent of research, development, and testing possible.

3.4 METHODOLOGIES OF PROBLEM SOLVING AND EFFICIENCY ISSUES

3.4.1 Methodologies of Problem Solving

- **Feature Extraction with EfficientNetV2 :** Utilize EfficientNetV2 as a feature extractor to represent images effectively. This methodology involves fine-tuning the pre-trained model's layers and extracting high-level features from the images.
- **Caption Generation with GRU :** Implement a GRU-based sequence-to-sequence model for generating captions. This methodology includes training the model to learn the language structure and generate coherent and contextually relevant descriptions.
- **Data Augmentation :** Apply data augmentation techniques to enhance dataset diversity and reduce overfitting. This involves techniques like image cropping, flipping, and color jittering to improve model generalization.

3.4.2 Efficiency Issues

- **Computational Resources :** Address efficiency issues by optimizing model architecture and training procedures to make the best use of available hardware. This includes considering batch sizes, model parallelization, and hardware acceleration (e.g., GPUs).

- **Memory Management :** Efficiently manage memory during training and inference to handle large datasets and avoid memory bottlenecks. Techniques such as batch loading, memory-efficient data structures, and model quantization can be explored.

- **Real-time Inference :** Optimize the caption generation process for real-time applications by improving model inference speed. This can involve quantization, model compression, and deployment on hardware suitable for real-time processing.

3.5 SCENARIO IN WHICH MULTI-CORE, EMBEDDED AND DISTRIBUTED COMPUTING USED

- **Multi-Core Utilization :** Leveraging multi-core processing is essential for parallelizing image processing tasks, enabling simultaneous feature extraction and caption generation for multiple images. This approach optimizes computational efficiency, reducing the time taken to process large volumes of images and generate captions.

- **Embedded Computing :** Implementing the image captioning system on embedded devices, such as edge computing platforms and IoT devices, allows for on-device processing without relying heavily on external computing resources. This approach ensures the availability of real-time captioning capabilities in applications where network connectivity may be limited or unstable.

- **Distributed Computing :** Utilizing distributed computing allows the system to distribute computational tasks across multiple nodes, optimizing the processing of extensive datasets and complex neural network models. By utilizing

a distributed architecture, the image captioning system can handle large-scale image processing and caption generation, catering to the demands of high-throughput applications.

3.6 OUTCOME

The outcomes of implementing image captioning using EfficientNetV2 and GRU are multifaceted:

- **Enhanced Accessibility :** It makes visual content more accessible to individuals with visual impairments, as it provides descriptions for images, enabling a richer online experience.
- **Automated Tagging :** The technology automates the process of tagging and categorizing images, which can significantly improve content organization and retrieval.
- **Content Recommendation :** It enables more intelligent content recommendation systems by understanding the visual content of images and associating them with user preferences.
- **Improved Human-Computer Interaction :** The ability to generate captions for images enhances human-computer interactions, making it easier for users to interact with and search for visual content.
- **Advancements in AI :** It exemplifies the progress in AI and deep learning, showcasing how neural networks can bridge the gap between visual and textual information.

3.7 APPLICATIONS

- **Social Media :** Image captioning is commonly used on social media platforms to automatically generate captions for user-uploaded images, making content more engaging and informative.

- **Content Recommendation :** Image captions can be used to personalize content recommendations by analyzing the textual descriptions and user preferences, improving user engagement and retention.
- **E-commerce :** Image captioning can provide product descriptions and details for e-commerce websites, enhancing the shopping experience by offering detailed information about products.
- **Education :** Image captioning can be applied in educational materials to provide additional context and information for images in textbooks, online courses, and educational websites.

3.8 HARDWARE RESOURCES REQUIRED

Sr. No.	Parameter	Minimum Requirement	Justification
1	CPU Speed	2 GHz	Required for efficient processing of image data
2	RAM	8 GB	Necessary for handling the computational load during image captioning
3	GPU	2 GB	Required for fast processing

Table 3.1: Hardware Requirements

3.9 SOFTWARE RESOURCES REQUIRED

Platform :

1. Operating System : Windows 10 or Ubuntu 20.04 LTS
2. IDE : PyCharm or Jupyter Notebook
3. Programming Language : Python 3.7 or higher, with libraries such as TensorFlow, NLTK, and NumPy.

CHAPTER 4

SOFTWARE REQUIREMENT SPECIFICATION

4.1 INTRODUCTION

4.1.1 Purpose and Scope of Document

The purpose of this document is to outline the design and implementation of a Deep Neural Network-based Encoder-Decoder framework for Image Captioning. This framework combines Convolutional Neural Networks (CNNs) to encode images and Recurrent Neural Networks (RNNs) to generate descriptive captions.
The scope includes explaining the architecture, training process, and evaluation metrics for image captioning tasks. It provides a comprehensive guide for researchers and developers interested in creating image captioning systems, enabling them to understand the key components, techniques, and considerations involved in this field. The document aims to bridge the gap between theory and practical implementation in the domain of computer vision and natural language processing.

4.1.2 Overview of responsibilities of Developer

The developer's responsibilities in implementing a Deep Neural Network-based Encoder-Decoder framework for Image Captioning include

- **Data Preprocessing :** Collect, clean, and preprocess image and caption data, ensuring it's suitable for training the model.
- **Architecture Design :** Design the neural network architecture, combining CNNs and RNNs, specifying the number of layers, units, and activation functions.
- **Model Implementation :** Code the framework using deep learning libraries (e.g., TensorFlow, PyTorch) and integrate the encoder-decoder structure.
- **Hyperparameter Tuning :** Optimize hyperparameters like learning rates, batch sizes, and sequence lengths for better model performance.
- **Training :** Train the model on the prepared dataset, monitoring loss and performance metrics.

4.2 FUNCTIONAL REQUIREMENTS

4.2.1 System Feature 1 (Image Preprocessing Module)

The system must have a module for preprocessing input images to make them suitable for the image captioning model. The image preprocessing module in image captioning prepares raw images for analysis by resizing, normalizing pixel values, and potentially augmenting data. It enhances feature extraction by ensuring consistent image dimensions and colour ranges, facilitating subsequent deep learning model input. This module improves model performance and consistency in generating accurate image captions.

4.2.2 System Feature 2 (Encoder-Decoder Architecture)

The core of the system is the Encoder-Decoder architecture, where the image is encoded using a Convolutional Neural Network (CNN) and the encoded information is decoded into a caption using a Recurrent Neural Network (RNN).The Encoder-Decoder architecture is a fundamental framework used in image captioning. The encoder processes the input image, typically using convolutional neural networks (CNNs) to extract image features. The decoder, often using recurrent neural networks (RNNs) or transformer models, generates a textual caption based on the extracted features. This approach combines computer vision and natural language processing to produce coherent and contextually relevant image captions.
This feature includes:

- **Image Encoding :** The system implements the EfficientNetV2 model to effectively encode images, enabling comprehensive feature extraction and representation from input images for subsequent processing in the caption generation process.
- **Caption Generation :** The system utilizes the GRU (Gated Recurrent Unit) model for generating natural language captions from the encoded image information. This process involves leveraging the GRU's sequential processing capability to produce coherent and contextually relevant captions for the provided images.

- **Connection Between Encoder and Decoder :** The system establishes an efficient connection between the image encoder, implemented with EfficientNetV2, and the caption decoder, implemented with GRU. This integration ensures smooth information flow from the encoded image representation to the caption generation module, facilitating a seamless and accurate translation of visual features into descriptive captions.

4.2.3 System Feature 3 (Model Training and Fine-Tuning)

This feature involves the training and fine-tuning of the neural network model to ensure it can effectively generate accurate and contextually relevant image captions. It includes the following components:

- **Training Data :** The system employs diverse dataset, including Flickr8k to train the neural network model. This comprehensive training data helps the model gain a robust understanding of various visual contexts and linguistic patterns, enabling it to generate accurate and contextually relevant captions.
- **Loss function :** The system incorporates a customized loss function tailored to the specific requirements of the EfficientNetV2 and GRU-based architecture. This specialized loss function optimizes the model's training process, ensuring efficient fine-tuning and improved caption generation performance.

These components collectively contribute to the effective training and fine-tuning of the neural network model, enhancing its capability to generate accurate and contextually relevant image captions.

4.3 EXTERNAL INTERFACE REQUIREMENTS (IF ANY)

4.3.1 User Interfaces

- **Image Upload/Selection :** Allow users to upload or select images for captioning. You can include an option for capturing images from a camera.
- **Caption Display :** Display the generated image captions in a clear and readable format.

- **Caption Customization :** Provide options for users to customize the generated captions, such as adjusting length, style, or language.
- **Caption Generation Button :** Include a button or action to trigger the caption generation process.
- **Image Preview :** Show a preview of the uploaded image(s) to confirm the selection.
- **Accessibility Features :** Make the interface accessible to users with disabilities, including alt text for images and keyboard navigation.
- **Error Handling :** Implement error messages and guidance for users in case of issues with image processing or caption generation.
- **Privacy and Security :** Clearly communicate how user data and images are handled and stored, addressing privacy and security concerns.

4.3.2 Hardware Interfaces

- **Input Devices :** For non-real-time or batch processing, users can upload images from various sources, including: Computer storage (hard drive, SSD) External storage devices (USB drives, SD cards)
- **Processing Unit (CPU/GPU) :** The image captioning model requires significant computational power for image analysis and caption generation. A CPU and/or GPU is used for this purpose. Modern deep learning models benefit greatly from GPUs due to their parallel processing capabilities.
- **Memory (RAM) :** Sufficient RAM is essential for loading and processing images, especially when dealing with large datasets or multiple concurrent users.
- **Storage :** The system should have storage capacity for storing images, model parameters, and generated captions. SSDs or large-capacity hard drives are common choices.

- **Network Interface :** A network connection is necessary for accessing image databases, model updates, and potentially sharing or storing captioned images online.
- **Display :** A monitor or screen is needed to display the user interface and the captioned images. It can be a desktop monitor, laptop screen, or the screen of a mobile device.
- **Peripheral Devices :** Input devices such as keyboards, mice, and touchscreens are necessary for user interaction with the system.

4.3.3 Software Interfaces

- **Image Input :** Allow users to upload or provide images for captioning. You can use file upload widgets or device cameras (for mobile apps).
- **Image Processing :** Preprocess the input image. This may involve resizing, normalizing, and enhancing the image to improve model performance.
- **Caption Generation :** Pass the image features through the NLP model to generate captions for the image. These captions can be single sentences or multiple sentences depending on the complexity of the image.

4.4 NONFUNCTIONAL REQUIREMENTS

4.4.1 Performance Requirements

Performance requirements for an image captioning project using deep learning are critical to ensure that the system meets user expectations and operates efficiently. Here are key performance requirements to consider:

- **Accuracy :** The image captioning model should provide accurate and meaningful captions for a wide range of images. Define a minimum accuracy threshold, such as BLEU, METEOR, or CIDEr scores, to evaluate model performance.

- **Speed :** Define the expected response time for generating captions. Users generally expect quick results, so set a maximum response time to meet user experience expectations.
- **Scalability :** Ensure that the system can handle an increasing number of users and images. Define performance requirements for both horizontal (adding more servers) and vertical (scaling a single server) scalability.
- **Latency :** Define maximum acceptable response times for generating captions for different image sizes. Ensure low-latency interactions with the user interface.

4.4.2 Safety Requirements

- **Privacy Protection :** Ensure that user data and uploaded images are protected and not shared with unauthorized parties. Comply with data protection regulations such as GDPR or HIPAA, as applicable.
- **Transparency and Explainability :** Make the image captioning process as transparent and explainable as possible. Users should understand how captions are generated, and there should be a way to provide explanations for generated captions when requested.

4.4.3 Security Requirements

- **Data Encryption :** Implement data encryption in transit (using HTTPS) and at rest to protect image data and captions from unauthorized access or interception.
- **Secure Data Storage :** Ensure that user data and uploaded images are securely stored with appropriate access controls to prevent unauthorized access.

4.4.4 Software Quality Attributes

- **Accuracy :** The accuracy of generated captions is crucial. Captions should closely match the content of the images.

- **Performance :** The system should generate captions quickly, especially for real-time or interactive applications.
- **Scalability :** The ability to handle a growing number of users and images without significant degradation in performance is essential.
- **Usability :** The user interface should be intuitive and user-friendly, making it easy for users to interact with the system.
- **Security :** User data and generated captions should be protected from unauthorized access and breaches.
- **Maintainability :** The codebase should be well-structured, documented, and easy to maintain, allowing for updates and improvements.
- **Reliability :** The system should be available and responsive, with minimal downtime.

4.5 SYSTEM REQUIREMENTS

4.5.1 Database Requirements

To implement image captioning, we select a dataset flickr8k in which 8k images and 8k captions included.

4.5.2 Software Resources Required

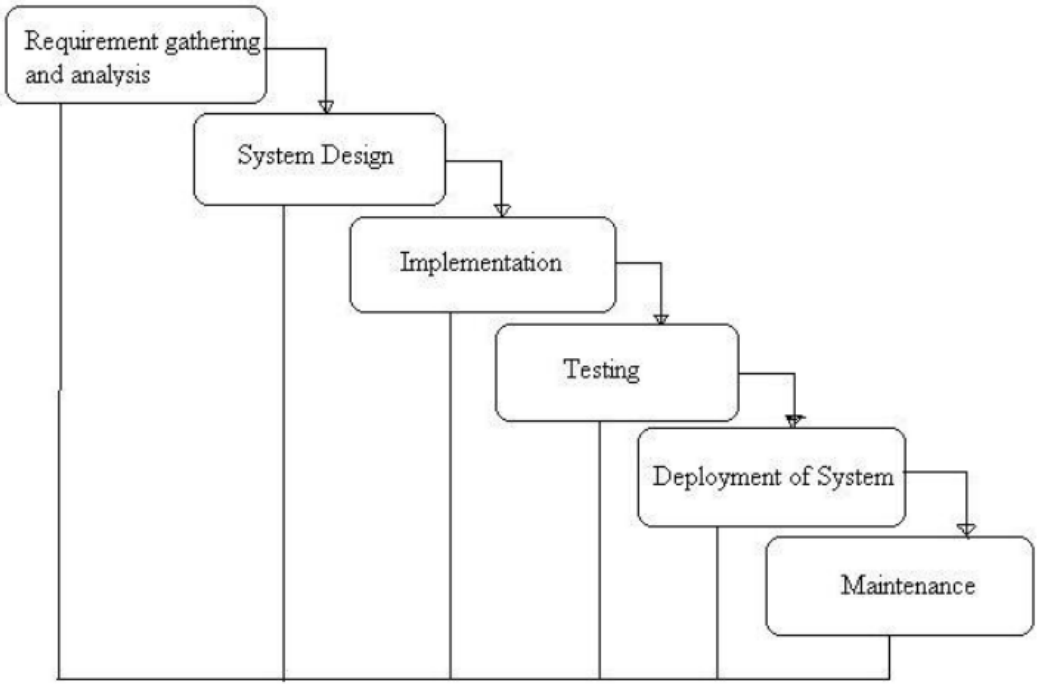
Platform :

1. Operating System : Windows 10 or Ubuntu 20.04 LTS
2. IDE : PyCharm or Jupyter Notebook
3. Programming Language : Python 3.7 or higher, with libraries such as TensorFlow, NLTK, and NumPy.

4.5.3 Hardware Resources Required

1. CPU : Speed 2 GHz Required for efficient processing of image data.
2. RAM : 8 GB Necessary for handling the computational load during image captioning.
3. GPU : Minimum 2 GB dedicated GPU is required for less training & testing time.

4.6 ANALYSIS MODELS: SDLC MODEL TO BE APPLIED



The Waterfall Model is a traditional software development methodology that follows a linear and sequential approach, consisting of distinct phases such as requirements, design, implementation, testing, and maintenance. While the Waterfall Model is more commonly associated with traditional software development, it is not typically used for deep learning tasks like image captioning, which involve a more iterative and experimental process. Deep learning models are often developed using iterative approaches, where the model is trained, evaluated, and refined in multiple cycles.

The application of the waterfall model to image captioning using deep learning, specifically with EfficientNetV2 and Gated Recurrent Unit (GRU), provides a structured and sequential approach. The cascade of processes, from feature extraction with EfficientNetV2 to context modeling with GRU, exemplifies a systematic flow in the development lifecycle. This methodology ensures a step-by-step refinement, allowing for a thorough exploration of the interplay between visual and textual information. As we conclude this endeavor, the waterfall model's rigidity aligns with the meticulous integration of EfficientNetV2 and GRU, contributing to a comprehensive and well-defined framework for advancing image captioning capabilities within the realm of deep learning.

The process can be described as follows:

- **Requirements :** Define the requirements for the image captioning system, including input data specifications, desired captioning output, and performance metrics.
- **Design :** Design the architecture of the deep learning model, including specifying the type of neural network (e.g., CNN-RNN), deciding on model parameters, and planning data preprocessing steps.
- **Implementation :** Develop and implement the deep learning model according to the designed architecture using a framework like TensorFlow or PyTorch.
- **Testing :** Evaluate the model on a validation dataset to assess its performance. This involves measuring metrics such as accuracy, BLEU scores, or other evaluation criteria relevant to image captioning.
- **Deployment :** If the model meets the desired performance criteria, deploy it for real-world use. This may involve integrating the model into an application or system capable of accepting input images and generating captions.
- **Maintenance :** Periodically update and fine-tune the model based on new data or changing requirements. Maintenance may also involve addressing issues discovered during real-world usage.

4.7 SYSTEM IMPLEMENTATION PLAN :

Sr.No.	Activity	Plan Start	Plan Duration (weeks)	Aug	Sep	Oct	Nov	Dec	Jan	Feb
1	Literature Survey	1	2							
2	Identify Objectives	14	2							
3	Feasibility study	1	2							
4	Study of Scope	15	2							
5	Requirement Analysis	1	2							
6	System Architecture	16	1							
7	UML Diagram	25	2							
8	Implementation and Testing	15	8							
9	Conclusion and Report	19	2							

Figure 4.1: System Implementation

CHAPTER 5

SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

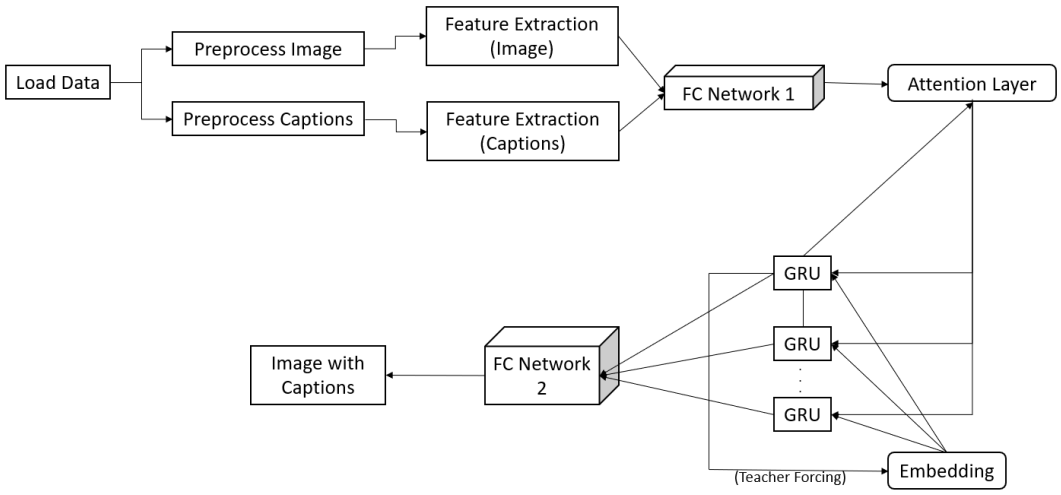


Figure 5.1: System Architecture

The image caption generation system begins with the pivotal step of loading data, encompassing a curated dataset of images and their corresponding captions. Subsequently, meticulous data preparation unfolds, involving resizing images, normalizing pixel values, and tokenizing captions for streamlined processing. The heart of the architecture lies in feature extraction, where Convolutional Neural Networks (CNNs) like EfficientNetV2 are employed to distill high-level visual features from the images. These extracted features serve as inputs for the image caption generation model, typically implemented using GRU. During the training phase, the model refines its parameters by predicting the next word in a sequence, learning the intricate associations between visual context and generated captions. The training data, consisting of image-caption pairs, fuels this iterative learning process through optimization algorithms like stochastic gradient descent. The integrated system encompasses components for user input, caption generation, and potential post-processing, culminating in a deployment-ready architecture that can generate descriptive captions for input images in real-world scenarios. This holistic architecture, combining EfficientNetV2, GRU, teacher forcing, and attention mechanisms, forms a sophisticated system for image captioning. It not only leverages the efficiency of EfficientNetV2 in visual feature extraction but also benefits from the sequential modeling capabilities of GRU, enhanced by attention mechanisms.

5.2 DATA FLOW DIAGRAMS

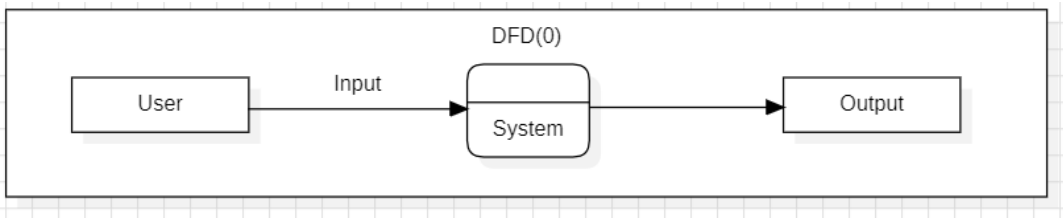


Figure 5.2: DFD0

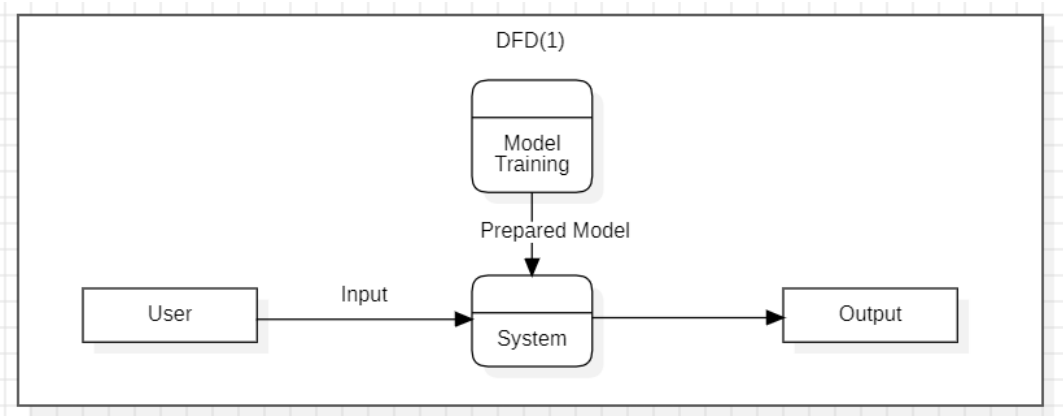


Figure 5.3: DFD1

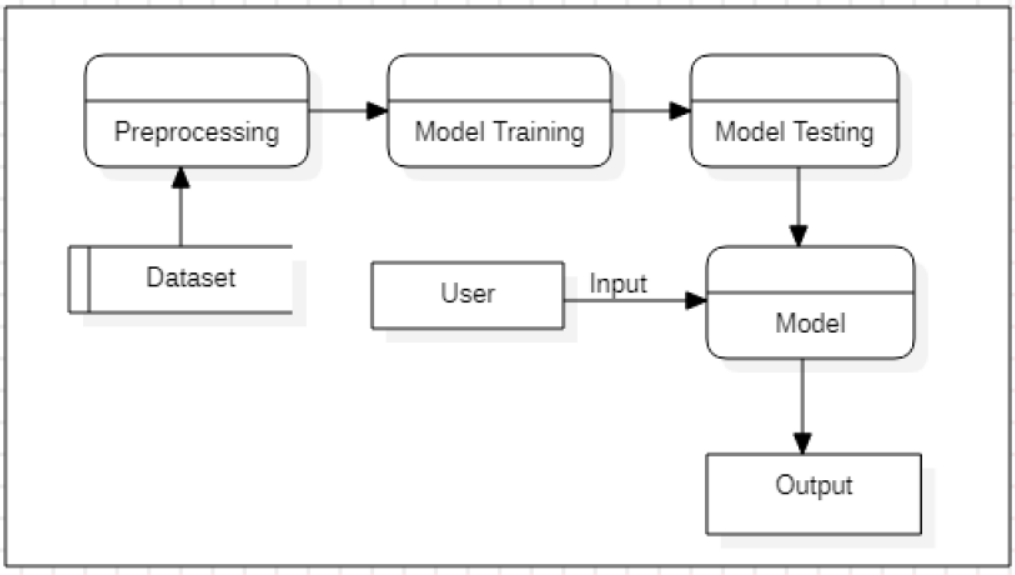


Figure 5.4: DFD2

A Data Flow Diagram (DFD) visually represents the flow of data within a system.

- External Entities : User: Initiates the image captioning process by uploading an image.
- Upload and Preprocess Image : Accepts the uploaded image from the user. Performs data preprocessing on the image (resizing, normalization, etc.).Outputs the preprocessed image data.
- Feature Extraction : Takes the preprocessed image data as input.Utilizes a feature extraction model (e.g., EfficientNetV2) to extract relevant features.Outputs the extracted features.
- Generate Caption : Takes the extracted features as input.Utilizes a caption generation model (e.g., GRU with attention) to generate captions.
- Display Image and Caption : Presents the original image and the generated caption to the user.
- Image Database : Stores information about images, including ImageID, FilePath, Timestamp.
- Caption Database : Stores generated captions, including CaptionID, CaptionText, Timestamp.This DFD provides an overview of the processes involved in image captioning, including data inputs, transformations, and outputs.

The system involves a user initiating image captioning by uploading an image, which undergoes preprocessing and is fed into a feature extraction model (e.g., EfficientNetV2). Extracted features are then used in a caption generation model (e.g., GRU with attention) to produce captions. The original image and generated caption are displayed to the user. Information about images is stored in an Image Database, and generated captions are stored in a Caption Database, both with timestamps for reference and organization. This integrated process provides a comprehensive framework for image captioning, encompassing user interaction, data processing, feature extraction, caption generation, and result presentation.

5.3 ENTITY RELATIONSHIP DIAGRAMS

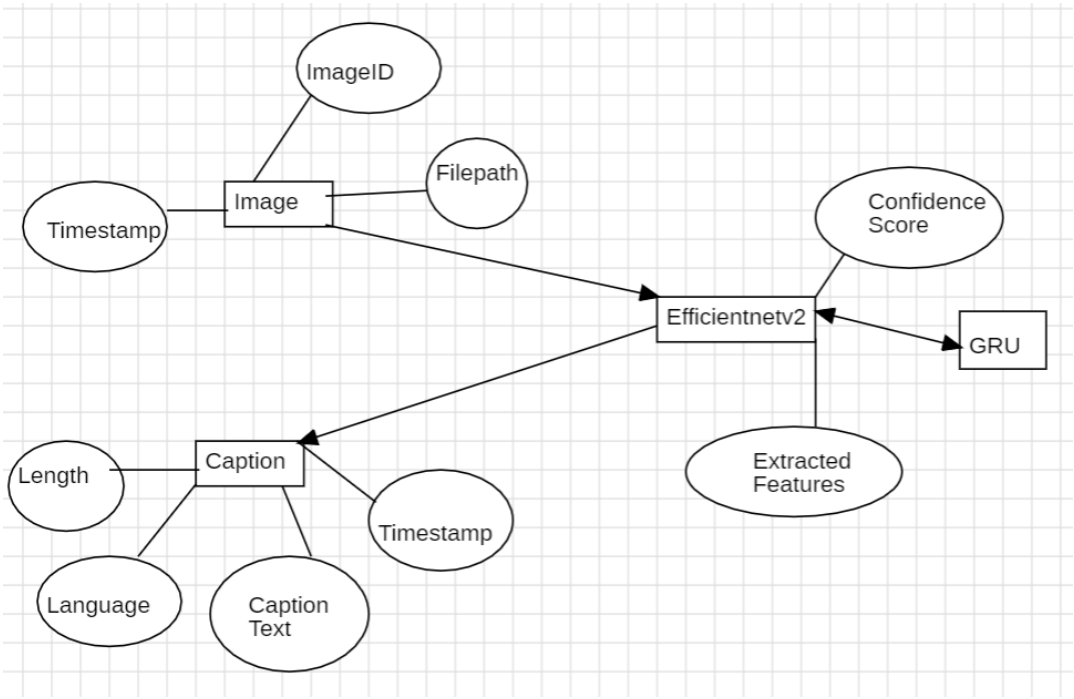


Figure 5.5: Entity Relationship Diagrams

An Entity-Relationship (ER) diagram typically represents the entities and relationships in a database. In image captioning with deep learning, the entities include caption, Image, EfficientNetV2 and GRU. The Entity-Relationship Diagram (ERD) illustrates the interconnected entities in the image captioning system. The “Image” entity encompasses attributes such as ImageID, FilePath, and Timestamp, forming a one-to-many relationship with the “Caption” entity, which consists of CaptionID, CaptionText, and Timestamp. Both entities are associated with the “EfficientNetV2” and “GRU” entities, representing many-to-one relationships, as multiple images share the same feature extraction and caption generation models. The “EfficientNetV2” entity includes ModelID, Architecture, and Parameters, while the “GRU” entity encompasses ModelID, Architecture, Attention, and Parameters. This structured ERD encapsulates the integral components, relationships, and attributes within the image captioning system, providing a comprehensive overview of its architecture and data flow.

5.4 UML DIAGRAMS

5.4.1 Use Case Diagram

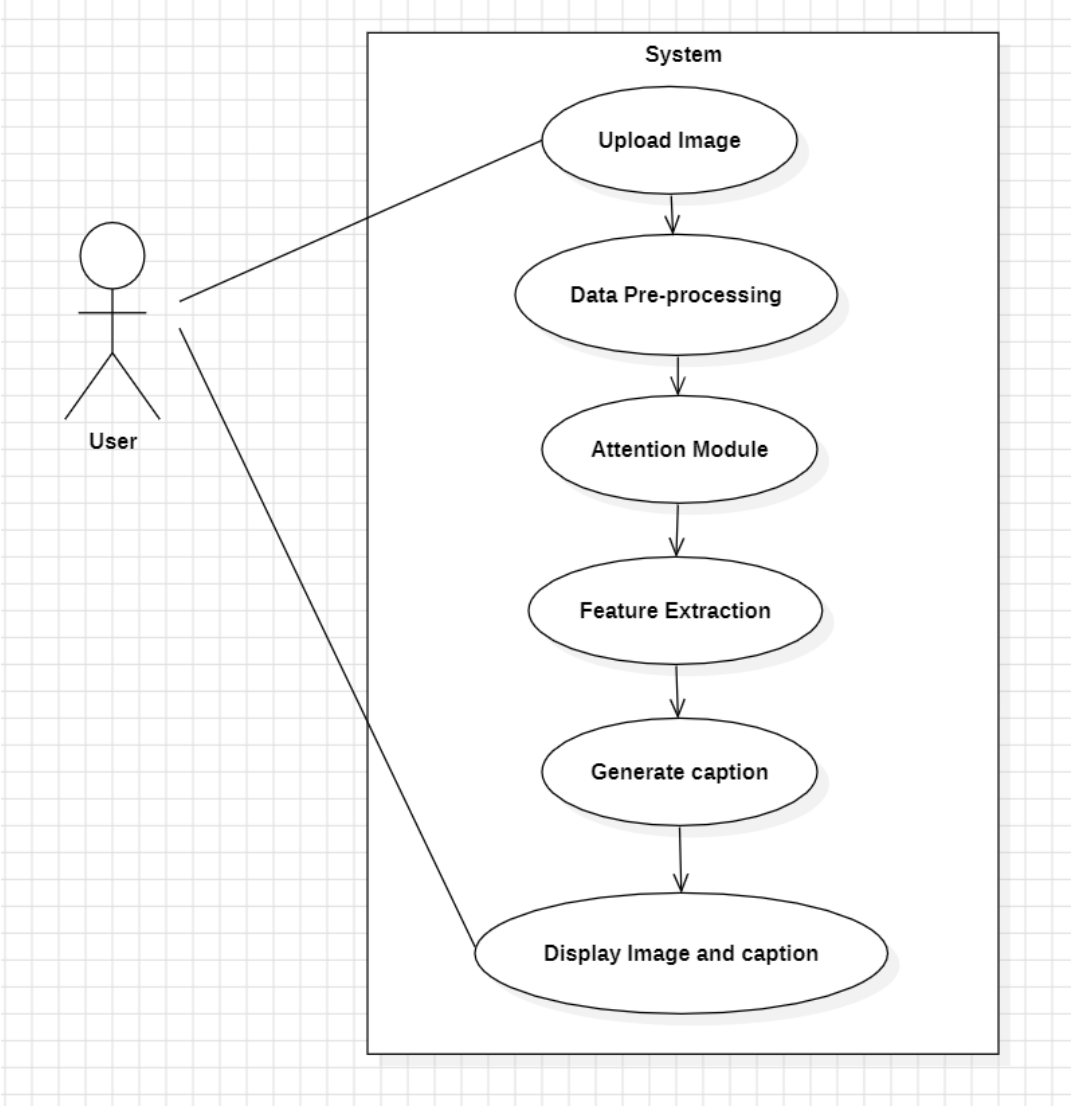


Figure 5.6: Use Case Diagram

A use case diagram for image captioning provides a high-level view of the system's functionalities and interactions with external actors. The actor represent entities that interact with the system, and use cases represent specific functionalities or features provided by the system. Here's a simplified use case diagram for an image captioning system.[6]

5.4.2 Activity Diagram

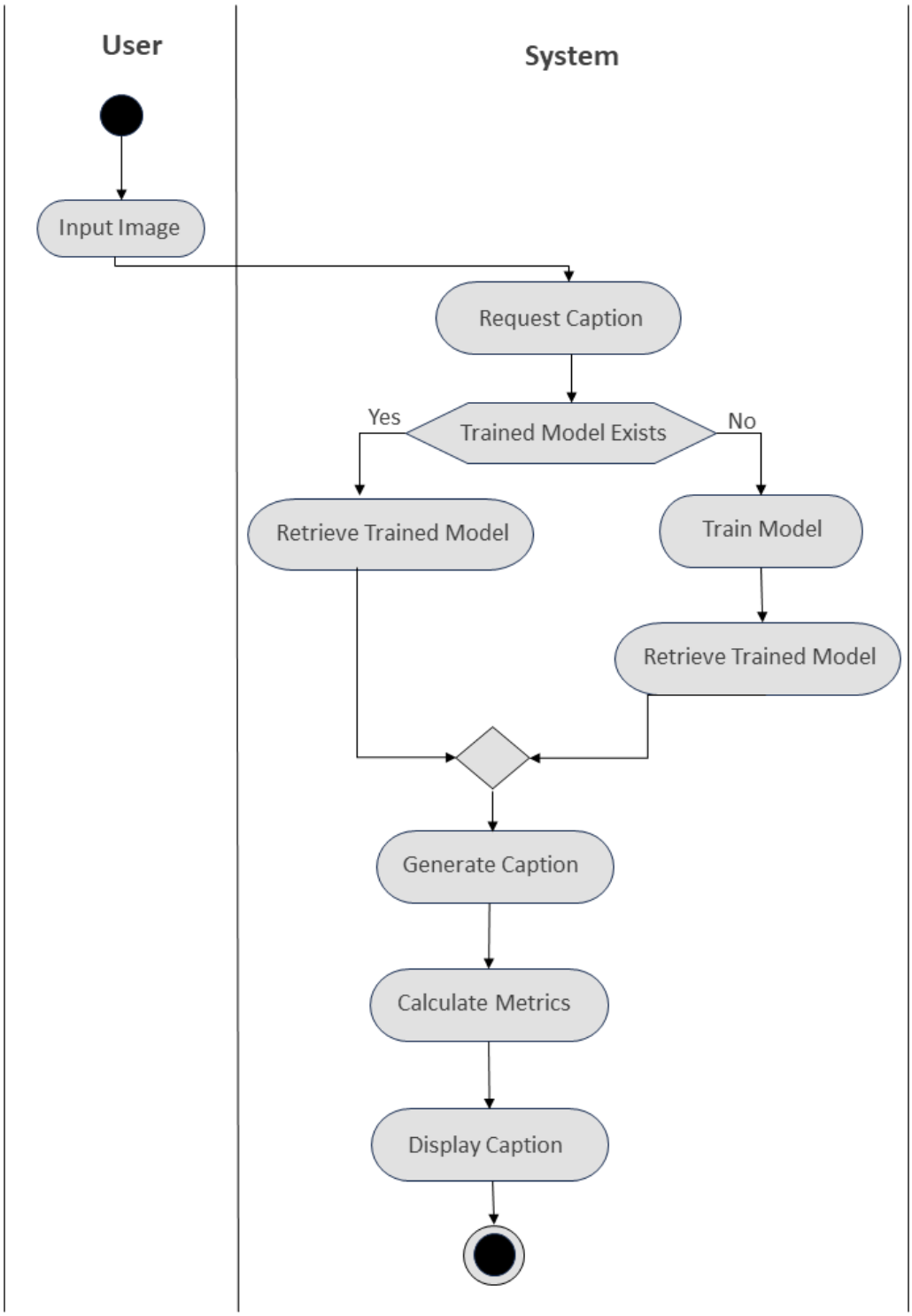


Figure 5.7: Activity Diagram

- An activity diagram for image captioning can illustrate the flow of activities involved in the process.
- “Image User” is an actor representing the user interacting with the system.
- “Upload Image” is a use case where the user uploads an image for captioning.
- “Image Captioning System” is a use case representing the overall image captioning process.
- “Receive Caption and Display” is a use case where the system receives the generated caption and displays it to the user.
- “Generate Caption for Image” is a use case responsible for actually generating the caption for the uploaded image.
- The “Image User” actor interacts with the system through “Upload Image” and receives the caption through “Receive Caption and Display.”
- This diagram illustrates the high-level interactions and functionalities involved in the image captioning The “Image User” actor starts by uploading an image.
- The “Captioning Requester” initiates the request for a caption.
- The “Image Captioning System” generates a caption for the uploaded image.
- The “Metrics Calculation System” calculates metrics for the generated caption.
- Arrows indicate the flow of activities, and each box represents an activity or a system component.
- This diagram provides a visual representation of the sequential steps involved in the image captioning process, including uploading an image, requesting a caption, generating the caption, and calculating metrics for evaluation.

5.4.3 Sequence Diagram

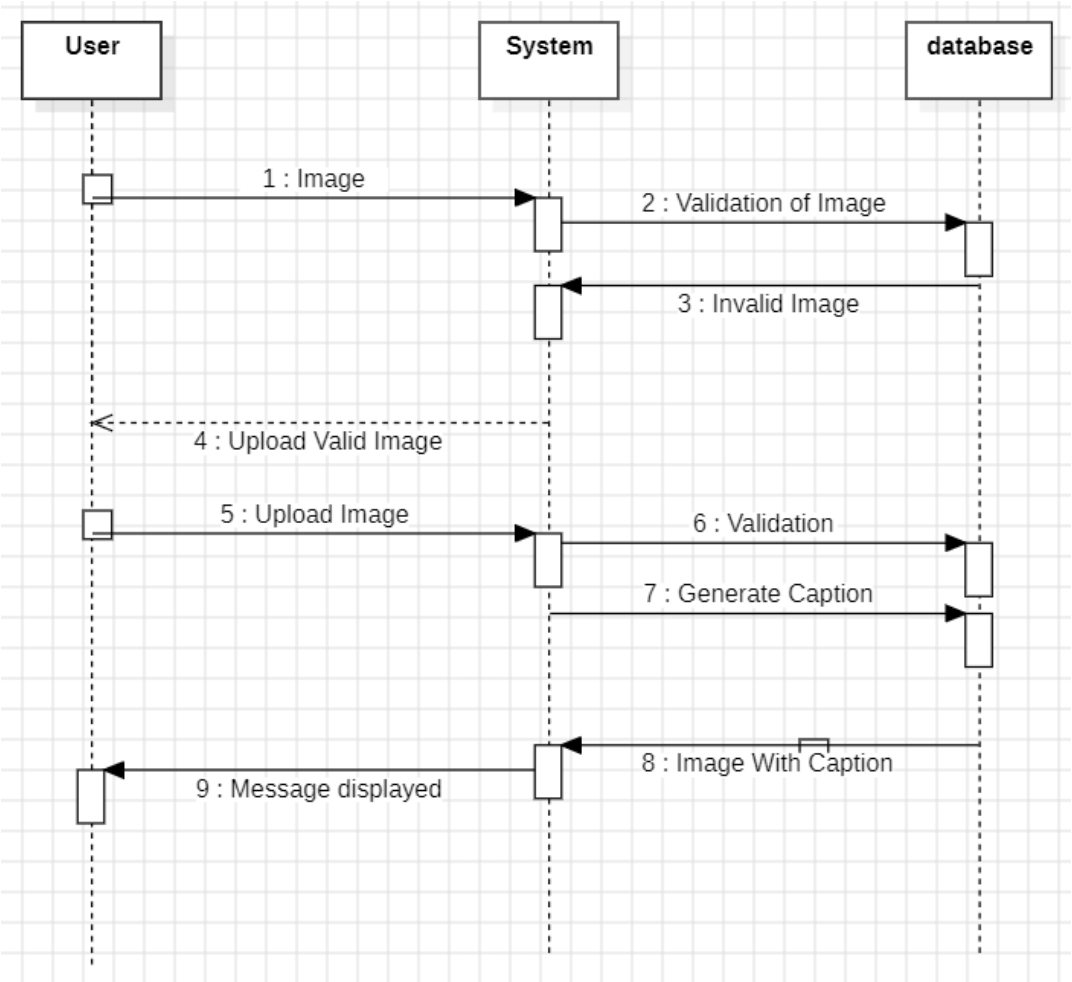


Figure 5.8: Sequence Diagram

- A sequence diagram for image captioning can help illustrate the interactions between different components or objects in a chronological order.
- In this sequence diagram:
- The “Image User” uploads an image.
- The “Image Validation” component validates the uploaded image.
- If the image is valid, it is uploaded to the “Image Captioning System.”
- The “Image Captioning System” generates a caption for the image.
- The final result is an “Image with Caption.”

5.4.4 Class Diagram

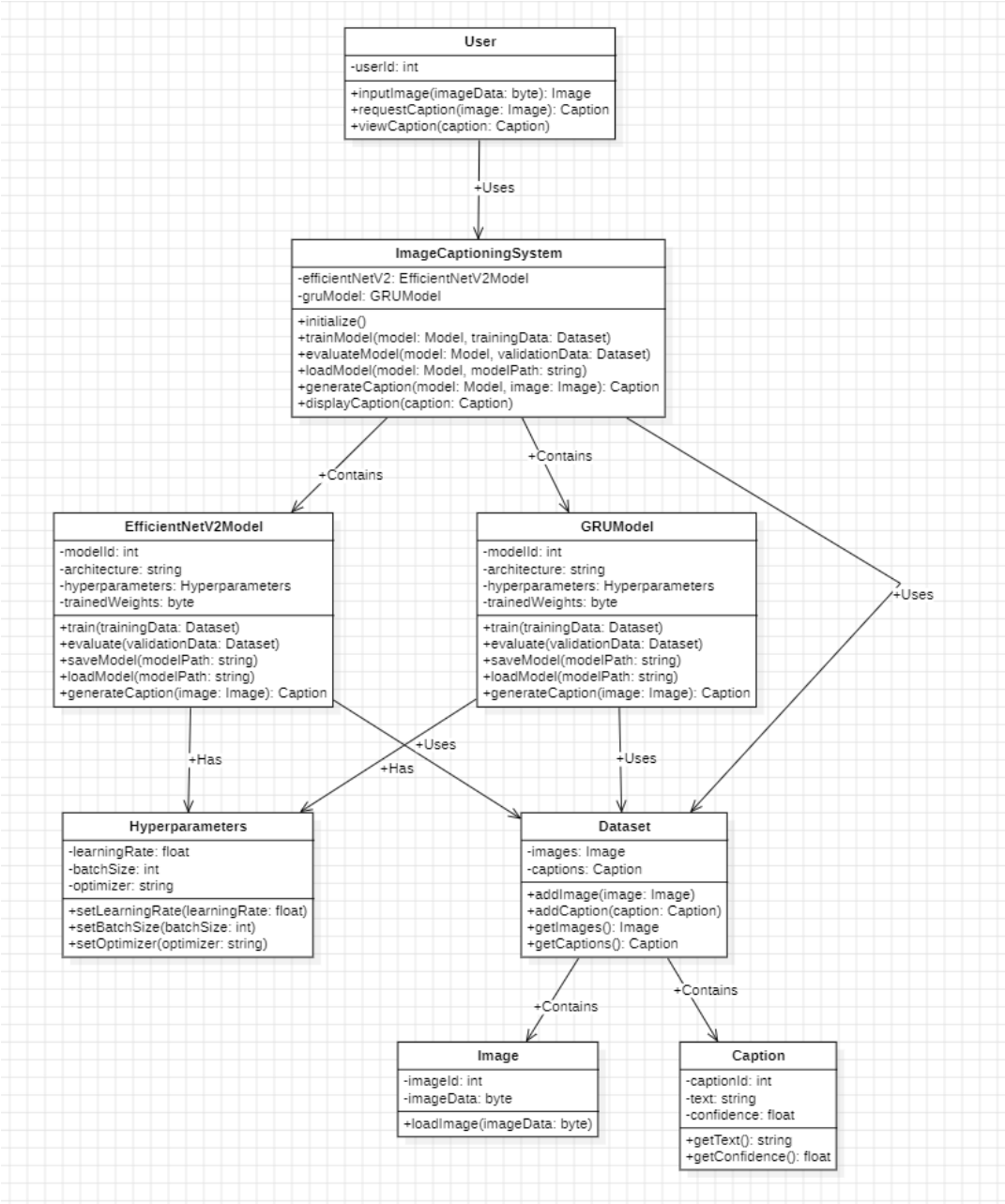


Figure 5.9: Class Diagram

In a class diagram for image captioning with deep learning, you can represent classes, their attributes, and methods. The central class in this system is the ImageCaptioningSystem, which encapsulates the entire image captioning functionality. It is associated with two main components: the ImageProcessor and the CaptionGenerator. The ImageProcessor class is responsible for handling image-related operations.[6]

5.4.5 Object Diagram

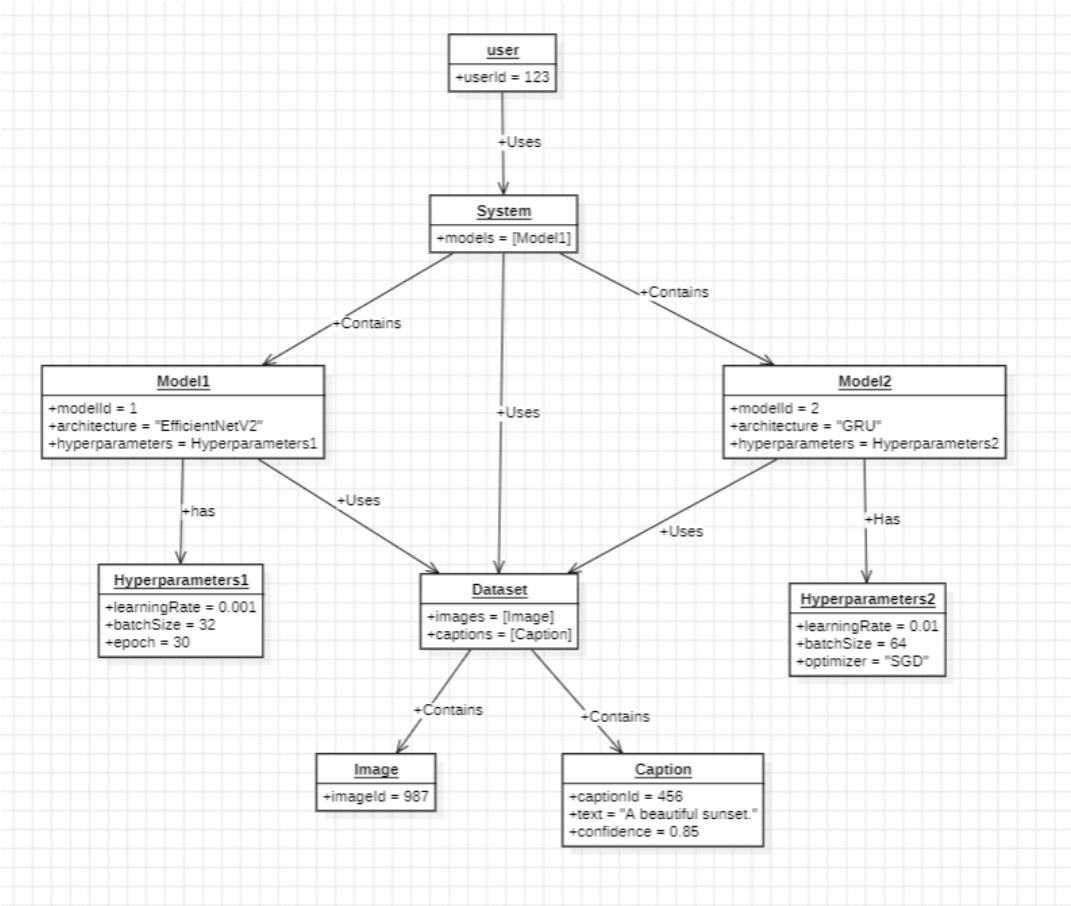


Figure 5.10: Object Diagram

An object diagram illustrates instances of classes and their relationships at a specific point in time. In the context of image captioning with deep learning, we have instances of the classes mentioned in the class diagram. imageCaptionSystemInstance contains a composition relationship with imageProcessorInstance, indicating that the ImageProcessor is a vital part of the ImageCaptioningSystem. It also has an association with captionGeneratorInstance. imageProcessorInstance is associated with imageInstance, showcasing that it processes this particular image. The instances of various classes represent the specific components involved in image captioning. The relationships and associations illustrate how these instances collaborate within the system[6]

CHAPTER 6

OTHER SPECIFICATION

6.1 ADVANTAGES

- **Automated Description :** Deep learning models can automatically generate descriptive captions for images, reducing the need for manual annotation and providing textual information that can be used for various purposes.
- **Accessibility :** Image captions can make visual content more accessible to individuals with visual impairments by providing descriptions of the content in a textual format.
- **Content Retrieval :** Image captions can improve content retrieval in image databases or search engines. Users can search for images using text-based queries, making it easier to find specific images or content.
- **Personalized Content :** Image captions can be tailored to the preferences or needs of the viewer. This personalization can enhance the user experience, especially in content recommendation systems.

6.2 LIMITATIONS

- **Accuracy and Quality :** Deep learning models for image captioning are not always perfect in generating accurate and high-quality captions. They can make mistakes, misinterpret images, or produce captions that do not accurately describe the content.
- **Overfitting :** Deep learning models may overfit to the training data, which means they perform well on the training data but struggle with new or diverse images, leading to incorrect or irrelevant captions.
- **Ambiguity Handling :** Images can be inherently ambiguous, and it can be challenging for deep learning models to handle ambiguity in image content and provide contextually appropriate captions.
- **Lack of Common Sense Understanding :** Deep learning models often lack common sense reasoning abilities, which can lead to captions that make factual errors or provide implausible interpretations of images.

6.3 APPLICATIONS

- **Social Media :** Image captioning is commonly used on social media platforms to automatically generate captions for user-uploaded images, making content more engaging and informative.
- **Content Recommendation :** Image captions can be used to personalize content recommendations by analyzing the textual descriptions and user preferences, improving user engagement and retention.
- **E-commerce :** Image captioning can provide product descriptions and details for e-commerce websites, enhancing the shopping experience by offering detailed information about products.
- **Education :** Image captioning can be applied in educational materials to provide additional context and information for images in textbooks, online courses, and educational websites.

CHAPTER 7

SUMMARY AND CONCLUSION

7.1 SUMMARY

Image captioning with EfficientNetV2 and a GRU is an advanced deep learning technique that enables automatic generation of descriptive text for images. EfficientNetV2, a convolutional neural network, extracts meaningful visual features from the input images. These features are then processed by a GRU, a type of recurrent neural network, to produce coherent and contextually relevant captions. This technology has a wide range of applications, including enhancing image accessibility for the visually impaired, automating image tagging, and improving content recommendation systems. It showcases the power of deep learning in bridging the gap between visual and textual information, ultimately facilitating more effective human-computer interactions and information retrieval.

7.2 CONCLUSION

In conclusion, image captioning using EfficientNetV2 and GRU represents a significant advancement in the field of deep learning. It demonstrates the ability of neural networks to comprehend and describe visual content with textual precision. The technology's applications in accessibility, image organization, and content recommendation underline its practical importance. As it continues to evolve, it has the potential to revolutionize how we interact with and understand the vast amount of visual data in today's digital world. Image captioning with EfficientNetV2 and GRU is a promising example of how artificial intelligence can enhance our relationship with visual media.

CHAPTER 8

REFERENCES

[1] M. M. Rahman, A. Uzzaman and S. I. Sami, “Implementing Deep Neural Network Based Encoder-Decoder Framework for Image Captioning,” 2021 *IEEE International Conference on Signal Processing, Information, Communication & Systems (SPICSCON)*, Dhaka, Bangladesh, 2021, pp. 26-31.

[2] Kavitha, P. V., and V. Karpagam, “Deep Learning Techniques for Automatic Image Captioning .” *Disruptive Technologies for Big Data and Cloud Applications: Proceedings of ICBDCC* 2021 (2022): pp 167-175.

[3] M. S. Alam, V. Narula, R. Haldia and G. Nikam Ganpatrao,“An Empirical Study of Image Captioning using Deep Learning,” 2021 *5th International Conference on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India, 2021, pp. 1039-1044.

[4] I. Hrga and M. Ivašić-Kos, “An overview of image caption generation methods.” 2019 *42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), Opatija, Croatia*, 2019, pp. 995-1000.

[5] Ningthoujam, Chitrapriya, and Tejbanta S. Chingtham, “Comprehensive Comparative Study on Several Image Captioning Techniques Based on Deep Learning Algorithm.” *In Contemporary Issues in Communication, Cloud and Big Data Analytics: Proceedings of CCB 2020, pp. 229-240. Springer Singapore*, 2022.

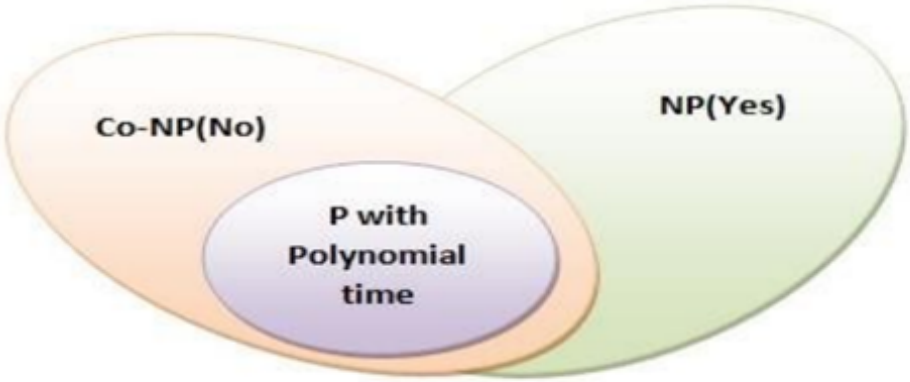
[6] A. A. A. Jilani, A. Nadeem, T. -H. Kim and E. -S. Cho, ”Formal Representations of the Data Flow Diagram: A Survey,” 2008 *Advanced Software Engineering and Its Applications, Hainan, China*, 2008, pp. 153-158.

ANNEXURE A

PROBLEM STATEMENT FEASIBILITY

- What is P?

  P is set of all decision problems which can be solved in polynomial time by a deterministic. Since it can be solved in polynomial time, it can be verified in polynomial time. Therefore P is a subset of NP.

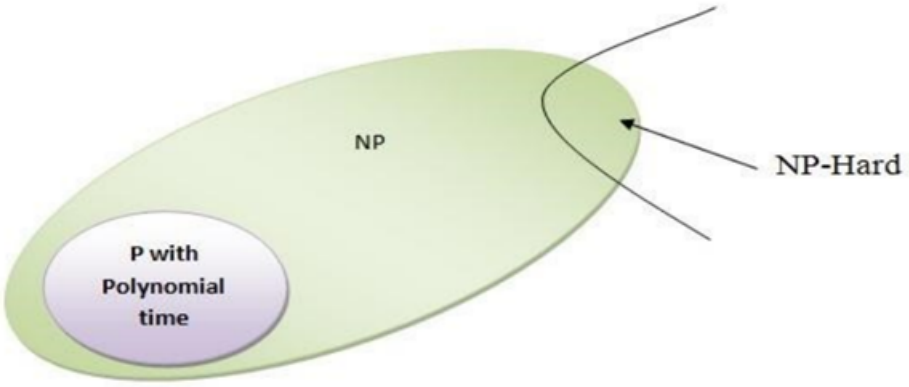


- What is NP?

  “NP” means we can solve it in polynomial time if we can break the normal rules of step-by-step computing.

- What is NP Hard?

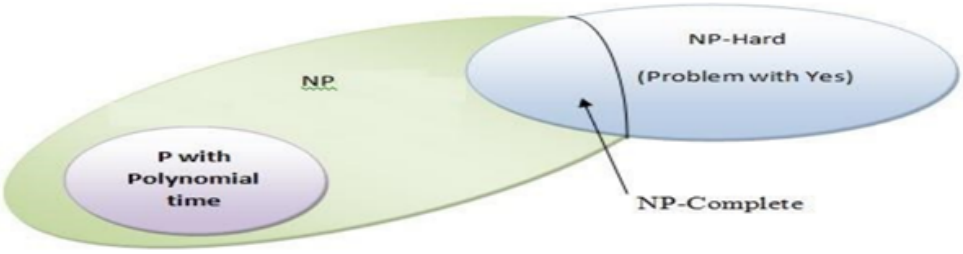
  A problem is NP-hard if an algorithm for solving it can be translated into one for solving any NP-problem (non deterministic polynomial time) problem. NP hard therefore means “at least as hard as any NP-problem,” although it might, in fact, be harder. NP-hard, or nondeterministic polynomial-time hard, is a classification in computational complexity theory. A problem is NP-hard if it is at least as hard as the hardest problems in NP (nondeterministic polynomial time) in terms of computational difficulty. If you could solve an NP-hard problem efficiently (in polynomial time), you could also solve any problem in NP efficiently.NP-hard problems may not necessarily be in NP, meaning there’s no known efficient algorithm to solve them. The concept is often used in discussions related to algorithmic problem-solving and complexity classes.



- What is NP-Complete?

  Since this amazing computer can also do anything a normal computer can, we know that problems are also in NP. So, the easy problems are in P (and NP), but the really hard onesare only in NP, and they are called NP complete. It is like saying there are things that People can do (P), there are things that Super People can do (SP), and there are things only Super People can do (SP-complete).

  NP-Complete: As our system is in developing state so we can't say that our system is currently in NP complete state Ideas of pattern-growth in uncertain environment.



ANNEXURE B

DETAILS OF THE PAPERS REFERRED

1. Rahman, Md Mijanur, Ashik Uzzaman, and Sadia Islam Sami. “Implementing Deep Neural Network Based Encoder-Decoder Framework for Image Captioning.” In 2021 IEEE International Conference on Signal Processing, Information, Communication & Systems (SPICSCON), pp. 26-31. IEEE, 2021.

2. Alam, Mohammad Shahnawaz, Vaishali Narula, Ruchika Haldia, and Gitanjali Nikam Ganpatrao. “An empirical study of image captioning using deep learning.” In 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), pp. 1039-1044. IEEE, 2021.

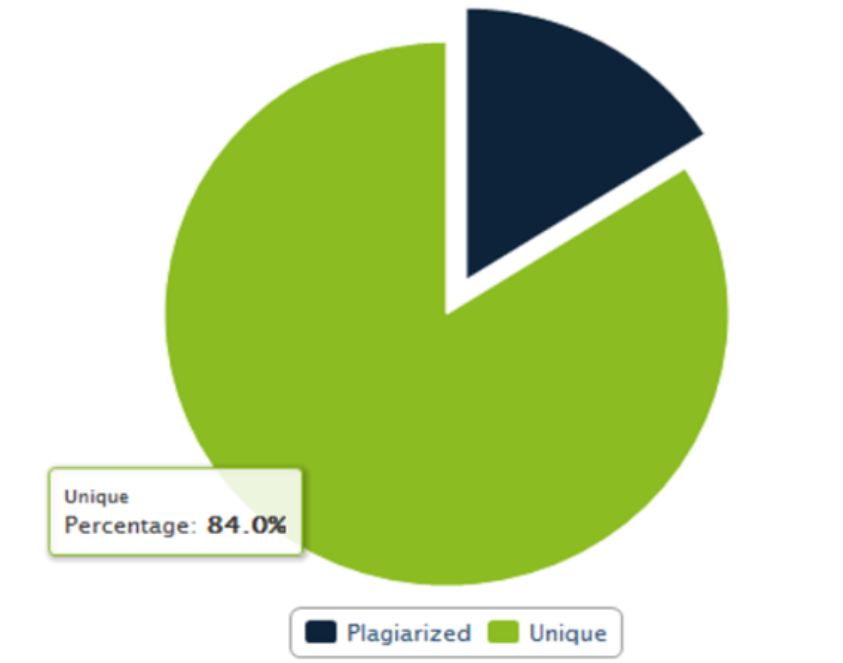
3. Hrga, Ingrid, and M. Ivašić-Kos. “An overview of image caption generation methods.” In 2019 42nd International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 995-1000. IEEE, 2019.

4. Ningthoujam, Chitrapriya, and Tejbanta S. Chingtham. “Comprehensive Comparative Study on Several Image Captioning Techniques Based on Deep Learning Algorithm.” In Contemporary Issues in Communication, Cloud and Big Data Analytics: Proceedings of CCB 2020, pp. 229-240. Springer Singapore, 2022.

5. Puscasiu, Adela, Alexandra Fanca, Dan-Ioan Gota, and Honoriu Valean. “Deep Learning Techniques for Automated Image Captioning.” In 2020 IEEE international conference on automation, quality and testing, robotics (AQTR), pp. 1-6. IEEE, 2020

ANNEXURE C

PLAGIARISM REPORT FOR THIS REPORT



Date	Sunday, November 19, 2023
Words	1287 Plagiarized Words / Total 7878 Words
Sources	More than 220 Sources Identified.
Remarks	Low Plagiarism Detected - Your Document needs Optional Improvement.