RAJIV GANDHI INSTITUTE OF TECHNOLOGY

GOVERNMENT ENGINEERING COLLEGE

KOTTAYAM-686 501

DEPARTMENT OF COMPUTER APPLICATIONS

20MCA246 - MAIN PROJECT REPORT

MATURITY PREDICTION OF BITTER GOURD USING BITTER-NET

Submitted By

SREERAG N V

(KTE22MCA-2054)

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

THIRUVANANTHAPURAM

APRIL 2023

DECLARATION

I, undersigned hereby declare that the project report entitled **"MATURITY PREDICTION OF BITTER GOURD USING BITTER-NET"**, submitted for partial fulfillment of the requirements for the award of the degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under the supervision of **Prof. Sreejith V P**. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to the ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed as the basis for the award of any degree, diploma, or similar title of any other University.

Place

Date **SREERAG N V**

ACKNOWLEDGMENT

I want to express my gratitude to everyone who has supported me throughout the endeavour. First and foremost, I give thanks to God Almighty for His mercy and blessings, for without His unexpected direction, this would still be only a dream.

I sincerely thank **Dr. PRINCE A** Principal, Rajiv Gandhi Institute of Technology, Kottayam, for providing the environment in which this project could be completed.

I owe a huge debt of gratitude to **Dr. Reena Murali** Professor and Head of the Department of Computer Applications, for granting permission and making available all of the facilities needed to complete the project properly.

I am grateful to **Prof. Sreejith V P**, Assistant Professor, the Department of Computer Applications and Project Coordinator, for his helpful criticism of my thesis, as well as for his constructive suggestions and inspiration throughout the project.

Finally, I'd like to take this chance to express my gratitude to the Department of Computer Applications' entire teaching and technical team.

SREERAG N V

ABSTRACT

This research addresses the need for efficient and accurate classification of bitter gourd images into three distinct classes Immature, Partially matured, and Matured. The current lack of automated solutions blocks agricultural automation and effective crop management. To overcome this challenge, a comparative approach employing several Convolutional Neural Network (CNN) models is proposed. The selected models, including VGG16, MobileNet, InceptionNet, AlexNet, ResNet and DenseNet, offer diverse architectures, allowing for an in-depth evaluation of their effectiveness in bitter gourd classification. VGG16 is recognized for its simplicity and complexity, while MobileNet's lightweight architecture makes it exceptionally efficient. DenseNet promotes dense connectivity for increased feature reuse, AlexNet is known for its groundbreaking design that pushed forward deep learning, ResNet is famous for its ability to train very deep networks effectively, and InceptionNet offers inception modules for improved feature extraction. The evaluation criteria consist of three factors: recall, precision, and F1-score. These factors together provide information about how well the models can classify images of bitter gourds at different stages of development. Furthermore, the model's predictive accuracy is quantitatively evaluated by the Root Mean Square Error (RMSE), which measures the average divergence between the expected and actual maturity phases. Additionally, the classification results are visualized using the confusion matrix, which allows for a thorough analysis of the model's abilities to distinguish between the stages of bitter gourd. The solution involves training these models on a selected dataset of bitter gourd images, with labels corresponding to different maturity stages. The anticipated outcome of this research is to identify the most accurate and efficient deep-learning model for bitter gourd classification. And fine-tuning that model to a new proposed model called Bitter-Net. By doing so, I contribute to the advancement of agricultural automation and crop management.

Keywords: VGG16, MobileNet, InceptionNet, AlexNet, ResNet, DenseNet

Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Abbreviations	**Definition**
CNN	Convolutional Neural Network
VGG16	Visual Geometry Group 16
VGG19	Visual Geometry Group 19
InceptionV3	Inception Version 3
ROC curve	Receiver operating characteristic curve

Chapter 1

INTRODUCTION

This chapter introduces the project's core motivations, highlighting its aim to to develop an Image Caption Generator using deep learning techniques to bridge the gap between visual perception and linguistic expression.

1.1 Need for the project

The need for the Image Caption Generation project arises from the growing importance of bridging visual understanding with linguistic expression in artificial intelligence and computer vision domains. In today's digital landscape, where visual content proliferates across online platforms and digital repositories, the significance of bridging visual understanding with linguistic expression through Image Caption Generation becomes increasingly pronounced. This project addresses the imperative need for machines to interpret and articulate visual content in human-readable language, thereby enhancing accessibility, content organization, and user engagement in computer vision domains.

In content management systems and image databases, automated caption generation streamlines content retrieval and organization, amplifying usability and searchability for users navigating vast repositories of visual data. Furthermore, Image Caption Generation enriches multimedia understanding by empowering machines to interpret and express visual content in natural language. This capability extends its utility across diverse fields such as content-based image retrieval, social media content enrichment, and automated image annotation, augmenting the efficiency and effectiveness of various applications.

By addressing the pressing need for automated image captioning, this project contributes significantly to advancing human-computer interaction paradigms. It renders visual content more accessible and comprehensible to a broader audience, fostering innovation and exploration in artificial intelligence and computer vision research, and paving the way for enhanced human-machine collaboration in navigating and understanding the ever-expanding visual landscape of the digital era.

1.2 Objective

The primary objective of this project is to develop an Image Caption Generator using deep learning techniques. By leveraging convolutional neural networks (CNNs) for image feature extraction and recurrent neural networks (RNNs) like Long Short-Term Memory (LSTM) for language modeling, the system aims to automatically generate coherent and contextually relevant captions for a wide array of images.

1.3 Scope of the project

The scope of this project encompasses the development and implementation of an Image Caption Generator using deep learning techniques, with a focus on robustness and contextual relevance in caption generation. The project involves leveraging large-scale image-caption datasets like Flickr8k for model training and validation, encompassing a diverse range of visual content.

As I conclude this chapter, the groundwork has been laid for understanding the pivotal need and objectives of the Image caption generator project. The subsequent chapters embark on a comprehensive exploration, beginning with a literature survey aimed at identifying critical gaps in current research and technology related to image caption generator. This survey will provide insights into existing methodologies, pinpointing areas where innovation and refinement are imperative. Subsequent sections will delve into the specific methodologies employed, from dataset collection to model training using deep learning techniques, providing an in-depth understanding of the project's methodology.

Chapter 2

LITERATURE SURVEY

The literature survey chapter offers a comprehensive overview of existing research and studies relevant to the image caption generator stages,providing providing insights into current methodologies and challenges in the field.

2.1 Existing System

In the current landscape of image caption generation, the reliance on manual annotation methods remains common. Despite advancements in machine learning and neural networks, automated image captioning using CNN and LSTM remains limited. Research has progressed in understanding image features and natural language processing, but deploying scalable systems for accurate captions is lacking. Manual annotation introduces errors and hampers efficiency, necessitating the development of automated solutions to bridge this gap.

2.2 Study on existing system

M. Israk Ahmed et al. proposed Context-based Image Caption using Deep Learning. utilizes Resnet101 for feature extraction and context coding for image processing. The model incorporates SCST and experimental LSTM for captioning. They have achieved an accuracy of 78.3% with 113,287 images as a dataset.

Xu et al proposed Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. This paper introduces an attention-based model for generating image descriptions, focusing on relevant image regions while generating captions. use GoogLeNet model. They have achieved an accuracy of 96.88% with 600 images as a dataset.

Anderson et al proposed Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering paper. they use CNN and LSTM model.This work explores a combined bottom-up and top-down approach for attention in image captioning, enhancing the model's ability to focus on relevant visual features. They have achieved an accuracy of 70.3% with 600 images as a dataset.

You et al proposed Image Captioning with Semantic Attention. They have used GoogleNet model. This study delves into semantic attention mechanisms, allowing the model to attend to semantically meaningful regions when generating image captions. They have achieved an accuracy of 75.06% accuracy with 30000 images as a dataset.

Table 2.1: Literature Survey

Sl No.	Title	Author	Model	Accuracy	No. of images in dataset
1	Context-based Image Caption using Deep Learning	M. Israk Ahmed et al.	Resnet101	78.3%	113,287
2	Show, Attend and Tell: Neural Image Caption Generation with Visual Attention	Fu yuesheng et al	GoogLeNet	96.88%	600
3	Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering	Anderson et al	CNN	70.3%	5100
4	Image Captioning with Semantic Attention	You et al	GoogleNet	75.06%	30000

2.3 Gap Identification

Through comprehensive literature analysis, I noticed a significant need for a better way for image caption generation. Existing studies on image caption generation reveal gaps in scalability, consistency, and reliability of models utilizing CNN and LSTM. Addressing these gaps is crucial for advancing the field and enhancing captioning accuracy and effectiveness.

The literature survey chapter concludes by summarizing important findings and pointing out gaps in the current state of research on image caption generator. These insights inform the project's methodology and direct future research into deep learning-based techniques for precise image caption generation.

Chapter 3

PROPOSED SYSTEM

The proposed system is a comprehensive solution designed to address the challenges of image caption generation. Leveraging advanced deep learning techniques, including state-of-the-art Convolutional Neural Network (CNN) architectures, such as VGG16 and LSTM. The system aims to develop accurate captions on images. This chapter outlines the key features of the proposed system and the tools and technologies employed in its development.

3.1 Features of proposed system

The proposed system for image caption generator uses two different neural networks to generate the captions. The first neural network is Convolutional Neural Network(CNN), which is used to train the images as well as to detect the objects in the image with the help of various pre-trained models like VGG. The second neural network used is Recurrent Neural Network(RNN) based Long Short Term Memory(LSTM), which is used to generate captions from the generated object keywords. As, there is lot of data involved to train and validate the model, generalized machine learning algorithms will not work. Deep Learning has been evolved from the recent times to solve the data constraints on Machine Learning algorithms. GPU based computing is required to perform the Deep Learning tasks more effectively.

By providing appropriate, expressive, and fluid subtitles, Deep Neural Networks can tackle the problems and accelerate the creation of subtitles. Users of social media will no longer have to waste hours searching for subtitles on Google with the system we offer. This technology will provides an easy-to-use platform for social network users to upload selected photographs. Photos of any size can be uploaded and also can read the caption out in English. Tensor flows and algorithms can be used by neural networks to solve any problem and provide appropriate, expressive, and fluent subtitles. It is feasible to calculate automatic metrics efficiently. The time spent searching for captions can be minimized as they will be automatically generated.

Chapter 4

MATERIALS AND METHODS

4.1 Tools

1. **Google Colab :** This project was done using a cloud-based platform provided by Google that allows to write and execute Python code in a Jupyter Notebook environment directly on Google's servers. It offers free access to computing resources, including CPU, GPU, and TPU, as well as integration with Google Drive for storing and sharing notebooks.

2. **TensorFlow:** This is a deep learning frameworks that provide efficient implementations of CNNs and LSTMs, along with other neural network components. It offer high-level APIs for building and training complex models, making them popular choices for implementing image captioning systems.

3. **CPU: Intel(R) Core(TM) i3-10110U CPU @ 2.10GHz:** The computational power for this study was provided by an Intel(R) Core(TM) i3-10110U CPU running at 2.10GHz. The CPU played a vital role in model training, feature extraction, and the overall execution of the deep learning algorithms. The efficiency of the CPU contributes to the timely processing of image data and the training of complex neural network architectures.

4.2 System architecture

The system architecture for image caption generation using CNN and LSTM extract high-level features from input images via a pre-trained CNN, which captures spatial information about objects and textures. These features are then fed into an LSTM network that sequentially processes them to generate captions. Word embeddings represent the vocabulary and contextual information, enriching the caption generation process. During training, the model learns to minimize the discrepancy between predicted and ground truth captions using optimization techniques like Adam. In inference, captions are generated by sampling words from the LSTM output distribution. Evaluation metrics such as BLEU assess the quality and relevance of generated captions. This combined CNN-LSTM architecture achieves contextually relevant image captions, finding extensive use and demonstrating state-of-the-art performance in image captioning tasks.

4.2.1 Block diagram

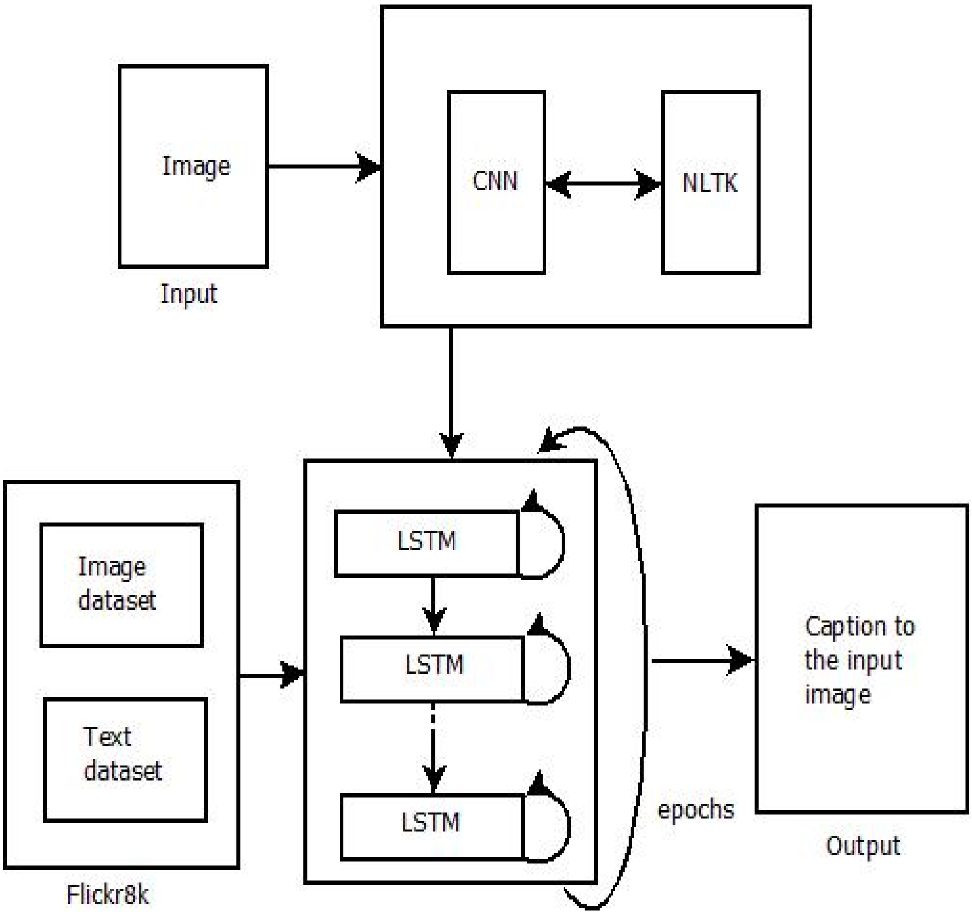

Figure 4.1: Block diagram

The project involves a sequential process of data collection, training, testing, and model evaluation.

Data Collection This project begins with the acquisition of a suitable dataset, focusing on high-quality images paired with descriptive captions. The Flickr 8K dataset is chosen for its comprehensive collection of images and corresponding human-written captions, providing a rich source for training and evaluation. The dataset is carefully curated to ensure diversity in visual content and linguistic expressions, essential for robust model development. Each image-caption pair serves as a training example, facilitating the learning process of the Image Caption Generator model.

Training During the training phase, the collected dataset is utilized to train the Image Caption Generator model. The Convolutional Neural Network (CNN) is employed to extract meaningful features from images, while the Long Short-Term Memory (LSTM) network learns to generate descriptive captions based on these extracted features. The model is trained iteratively using optimization techniques such as backpropagation and gradient descent, adjusting its parameters to minimize the discrepancy between generated and ground truth captions. This process allows the model to learn the intricate associations between visual content and textual descriptions, enabling it to produce accurate and coherent captions for unseen images.

Testing Following training, the performance of the Image Caption Generator model is assessed through rigorous testing procedures. A separate subset of the dataset, distinct from the training data, is reserved for testing purposes to ensure unbiased evaluation. The trained model is deployed to generate captions for unseen images in the test set, and the quality of the generated captions is evaluated against human-written references using metrics such as BLEU. This testing phase serves to validate the generalization capability of the model and assess its performance in real-world scenarios, providing insights into its effectiveness and areas for improvement.

Model Evaluation In the final stage of the project, the performance of the Image Caption Generator model is comprehensively evaluated. Evaluation metrics such as BLEU are employed to quantify the quality, fluency, and relevance of the generated captions compared to human-authored references. The model's ability to capture semantic meaning, linguistic diversity, and contextual relevance is assessed, providing valuable insights into its strengths and limitations. Additionally, qualitative analysis and user feedback may be solicited to gain further understanding of the model's performance and refine its capabilities. This evaluation process ensures that the Image Caption Generator meets quality standards and effectively bridges the gap between visual perception and linguistic expression.

4.3 Architectures employed for image caption generation

4.3.1 CNN

A pure rustic neural network, in whatever location all neurons in a single layer merge with all of the neurons in the subsequent layer is inefficient in regards to analyzing large pictures and video. For a normal size picture with many picture elements called pixels and 3-tone colors (RGB i.e., red color,green color,blue color), the range of restriction utilizing an accepted neural system will be in the tons, that can prompt overfitting. To constrain effective quantities of restrictions and recognition of the neural system on significant pieces of picture, CNN utilises a 3D arrangement in which each adjustment of neurons breaks down a little area or “highlight” of picture. Rather than all neurons to skip their selections to the next neural layer, each gathering of the neurons spends significant time in distinguishing one piece of picture, such as a nose, a left ear, mouth or a leg. The last yield is a point of scope, illustrating how reasonable every one of abilities is elected as part of the class.

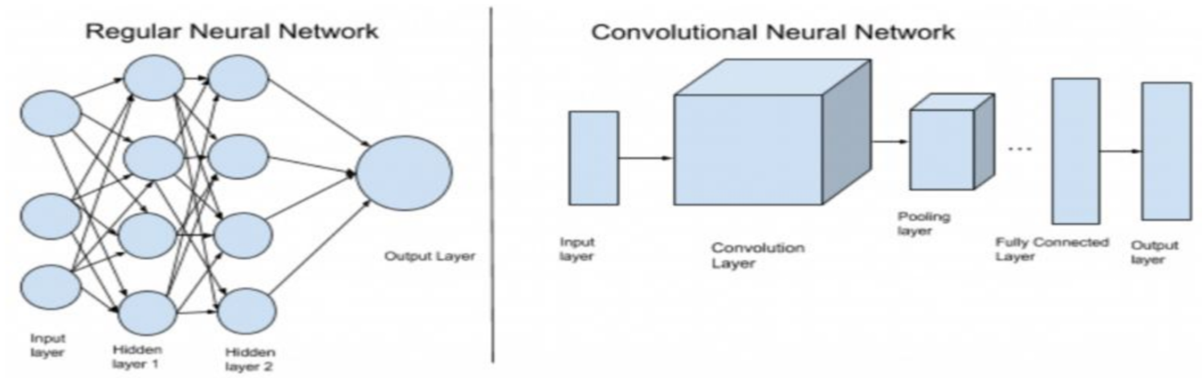

Figure 4.2: CNN architecture [9]

4.3.2 LSTM

The architecture of LSTM is very simple, it consists of 3 major gates, which store the data for a longer period of time and help in solving the difficulties which RNNs couldn’t solve. The 3 major gates of the LSTM covers are Forget gate which is used to filter the data, i.e. to delete all that data which is not needed in the future to solve a particular task. This gate is responsible for the overall performance of the LSTM, it optimizes the data. · Then input gate which is starting of LSTM . This gate takes input from the user and supplies the input data to other gates. Finally, Output gate that is responsible for showcasing the desired result in a proper manner.

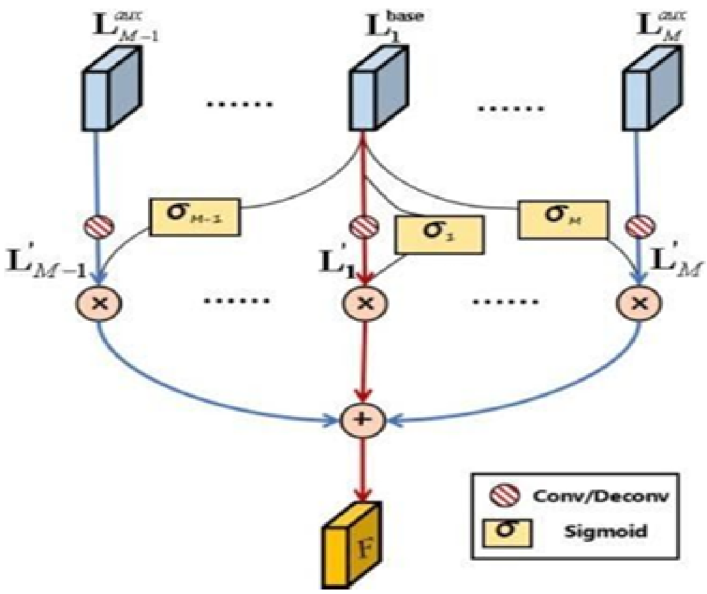

Figure 4.3: Gates in LSTM

4.3.3 CNN-LSTM Model

In order to prepare an image caption generation model, we will be summing up the two different architectures. It is further called as CNN-LSTM model. So, in this we will be using these two architectures to get the caption for the input pictures.

CNN - it's been used to extract the important features from the input picture. To do this, we have taken a pre-trained model for our consideration named Xception.

LSTM - its been used to store the data or the features from the CNN model and further process it and to support in the generation of a good caption for the picture.

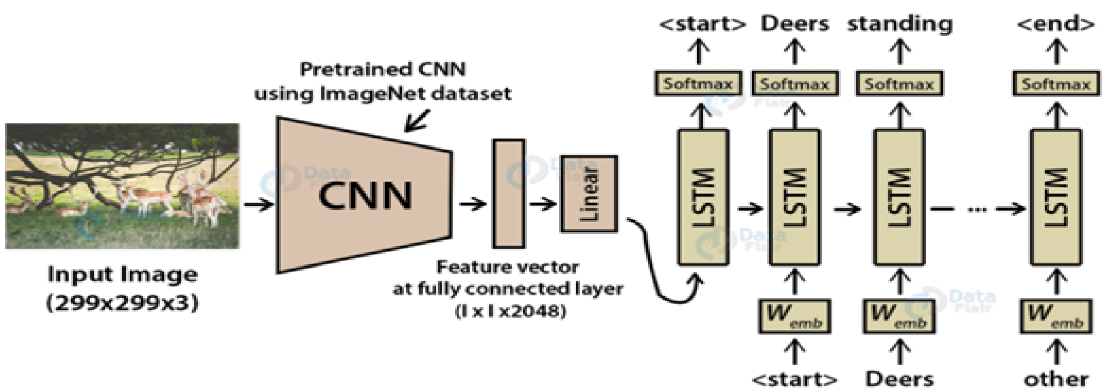

Figure 4.4: 7 CNN-LSTM model

Chapter 5

RESULT AND ANALYSIS

5.1 Results

Actual Captions:

- startseq child playing on rope net endseq
- startseq little girl climbing on red roping endseq
- startseq little girl in pink climbs rope bridge at the park endseq
- startseq small child grips onto the red ropes at the playground endseq
- startseq the small child climbs on red ropes on playground endseq

Predicted Caption:

- startseq little girl grips the red ropes endseq

Figure 5.1: Image 1

5.2 Discussion

The reported BLEU-1 and BLEU-2 scores of 0.540643 and 0.310481, respectively, provide insights into the performance of the image caption generation model. BLEU scores serve as metrics for evaluating the similarity between the generated captions and the reference captions. A BLEU-1 score of 0.54 indicates relatively good performance in capturing individual words within the generated captions. However, the lower BLEU-2 score suggests there's room for improvement in accurately generating longer phrases that effectively reflect the content depicted in the images. Further analysis of these scores can help identify specific areas for enhancement and refinement in the model architecture and training process.

The project widely-used Flickr8K dataset as a benchmark for both training and evaluating the image caption generation model. A notable aspect of the approach was the generation of five captions for each image in the dataset. It would be beneficial to elaborate on the methodology employed for caption generation. Additionally, providing insights into the criteria used for selecting the presented captions, such as ranking based on evaluation metrics or random selection, would offer valuable context regarding the diversity and representativeness of the generated captions.

Chapter 6

CONCLUSION

This project was dedicated to employing deep learning methodologies, specifically utilizing Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks, for the purpose of generating captions for images. Automatic image captioning remains a challenging task in the realm of research due to its complexities. Despite considerable efforts directed towards refining image feature extraction and sentence formulation, the project successfully implemented a model capable of generating accurate descriptions.

Operating within the constraints of computational resources, it utilized a smaller dataset, Flickr8k, for training the model. The combination of CNN for effective feature extraction and LSTM for generating semantically correct output in various languages was pivotal in achieving our objectives. Additionally, developed a dictionary based on words extracted from existing captions in the training dataset, thereby enhancing the model's ability to generate relevant sentences corresponding to the provided image.

Model's performance was evaluated through testing on the Flickr8k training images, yielding promising results. These outcomes were further validated through assessment using metrics such as the Bilingual Evaluation Understudy (BLEU). However, it acknowledge that the model's precision could be further enhanced by training on larger datasets, a direction for future improvement.

Moreover, the project shed light on the potential of integrating unsupervised data from isolated contexts and images to bolster image explanation techniques. This suggests promising avenues for future research, with the aim of advancing AI-driven image understanding and accessibility.

In summary, this project represents a significant stride forward in the field of image processing, paving the way for improved content retrieval and accessibility in various applications.

Chapter 7

FUTURE SCOPE

Future work involves refining models, exploring diverse datasets, and adapting the approach for real-world agricultural scenarios. The study acknowledges the need for continuous improvement, particularly in model generalization and robustness. Exploring larger and more diverse datasets will contribute to enhancing the models' ability to handle variations in real-world conditions. Additionally, adapting the approach for practical agricultural applications involves considerations such as real-time processing and integration with on-field sensors.

These future implications align with the evolving landscape of deep learning in agriculture and underscore the ongoing efforts to make automated crop stage identification more adaptable and applicable in real-world scenarios.

Bibliography

[1] M. I. Ahmed, S. Mahmud Mamun, and A. U. Zaman Asif, “DCNN-Based Vegetable Image Classification Using Transfer Learning: A Comparative Study,” *2021 5th International Conference on Computer, Communication and Signal Processing (ICCCSP),* Chennai, India, 2021, pp. 235-243, doi: 10.1109/ICCCSP52374.2021.9465499.

[2] Yuesheng, Fu, Song Jian, Xie Fuxiang, Bai Yang, Zheng Xiang, Gao Peng, Wang Zhengtao, and Xie Shengqiao. “Circular Fruit and Vegetable Classification Based on Optimized GoogLeNet.” *IEEE Access,* vol. 9, 2021, pp. 113599-113611.

[3] Aelchimminut, “Fruits 262,” Kaggle, [Online]. Available: https://www.kaggle.com/datasets/aelchimminut/fruits262. [Accessed: October]

[4] Steinbrener, Jan, Konstantin Posch, and Raimund Leitner. “Hyperspectral fruit and vegetable classification using convolutional neural networks.” *Comput. Electron. Agric.,* vol. 162, 2019, pp. 364-372.

[5] Ye, Haijian, Han, Hang, Zhu, Linna, Duan, Qingling. “Vegetable Pest Image Recognition Method Based on Improved VGG Convolution Neural Network.” *Journal of Physics: Conference Series,* vol. 1237, 2019, 032018. doi: 10.1088/1742-6596/1237/3/032018.

[6] Mimma NE, Ahmed S, Rahman T, Khan R. “Fruits Classification and Detection Application Using Deep Learning.” *Scientific Programming,* Nov 17, 2022.

[7] Li, Zhenbo, Li, Fei, Zhu, Ling, Yue, Jun. “Vegetable Recognition and Classification Based on Improved VGG Deep Learning Network Model.” *International Journal of Computational Intelligence Systems,* vol. 13, 2020, pp. 559. doi: 10.2991/ijcis.d.200425.001.

[8] L. Zhang, J. Jia, G. Gui, X. Hao, W. Gao, and M. Wang. “Deep Learning Based Improved Classification System for Designing Tomato Harvesting Robot.” *IEEE Access,* vol. 6, 2018, pp. 67940-67950. doi: 10.1109/ACCESS.2018.2879324.

[9] ”VGG16 - A Comprehensive Guide,” DataGen Tech, [Online]. Available: https://datagen.tech/guides/computer-vision/vgg16/.

[10] ”Inception-v3 on Papers with Code,” Papers with Code, [Online]. Available: https://paperswithcode.com/method/inception-v3.

ANNEXURE

Sample Code

VGG16

```
from keras.preprocessing import image
from keras.applications.vgg16 import VGG16, preprocess_input,
    decode_predictions
import numpy as np

# Load the pre-trained VGG16 model
model = VGG16(weights='imagenet')

# Load and preprocess an image
img_path = 'path/to/your/image.jpg'
img = image.load_img(img_path, target_size=(224, 224))
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array, axis=0)
img_array = preprocess_input(img_array)

# Get model predictions for the image
predictions = model.predict(img_array)

# Decode and print the top-3 predicted classes
decoded_predictions = decode_predictions(predictions, top=3)[0]
print("Predictions:")
for i, (imagenet_id, label, score) in enumerate(decoded_predictions):
    print(f"{i + 1}: {label} ({score:.2f})")
```

InceptionV3

```
    from keras.preprocessing import image
from keras.applications.inception_v3 import InceptionV3,
   preprocess_input, decode_predictions
import numpy as np

# Load the pre-trained InceptionV3 model
model = InceptionV3(weights='imagenet')

# Load and preprocess an image
img_path = 'path/to/your/image.jpg'
img = image.load_img(img_path, target_size=(299, 299))  # InceptionV3
    requires input size (299, 299)
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array, axis=0)
img_array = preprocess_input(img_array)

# Get model predictions for the image
predictions = model.predict(img_array)

# Decode and print the top-3 predicted classes
decoded_predictions = decode_predictions(predictions, top=3)[0]
print("Predictions:")
for i, (imagenet_id, label, score) in enumerate(decoded_predictions):
    print(f"{i + 1}: {label} ({score:.2f})")
```

Project Screenshots

File seclection

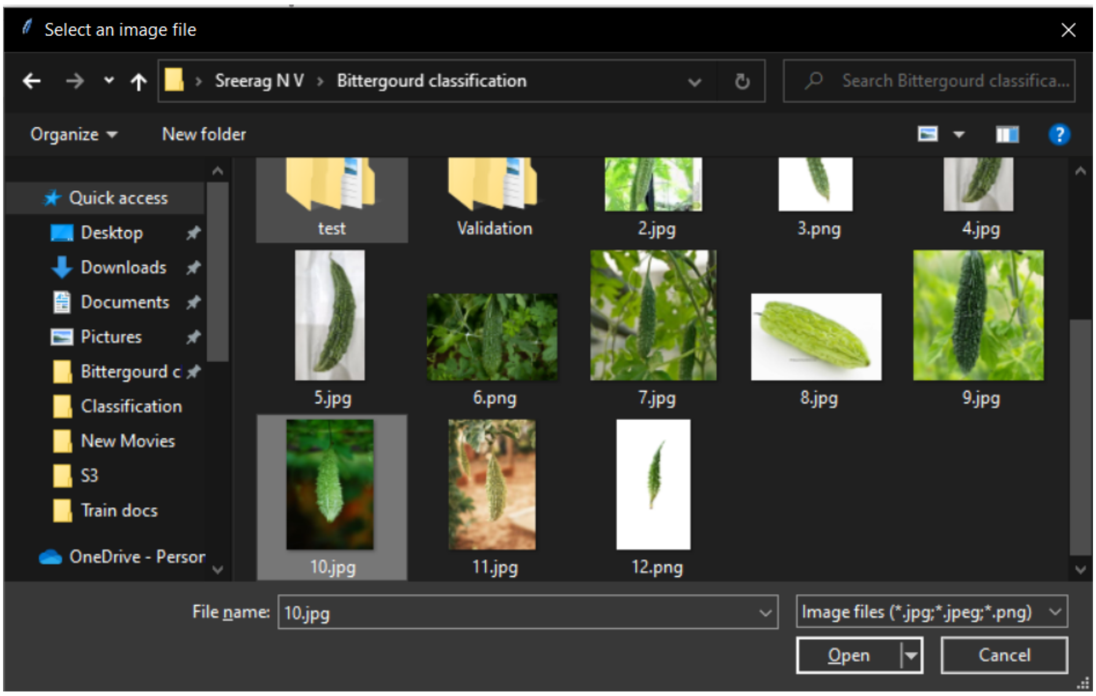

Figure 7.1: File seclection

Stage Prediction

```
1/1 [==============================] - 1s 1s/step
Predicted Label: Matured
```

Figure 7.2: Stage Prediction

GitHub History

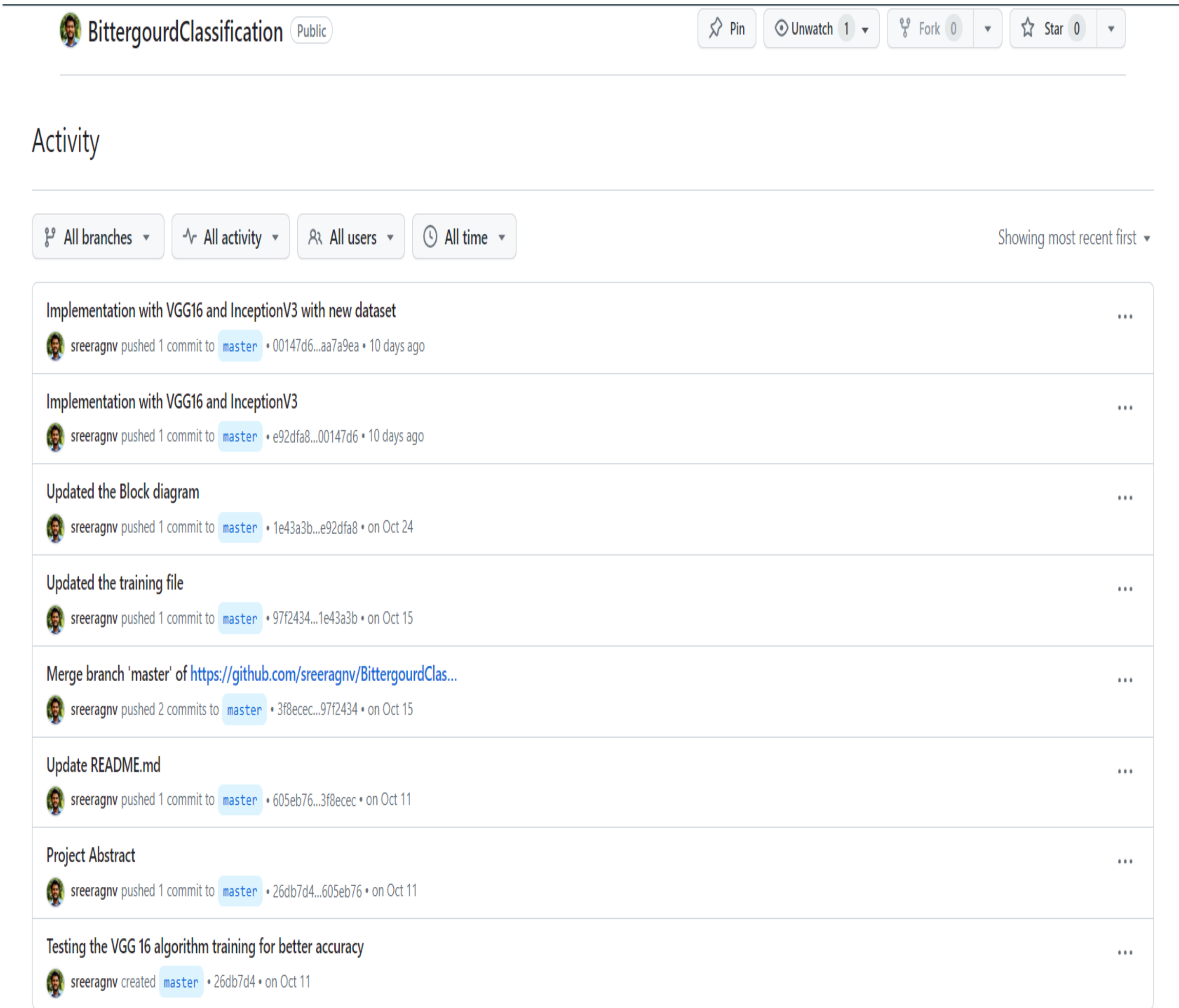

Figure 7.3: GitHub History