THE AUTOMATIC MYANMAR IMAGE CAPTIONING USING CNN AND BIDIRECTIONAL LSTM-BASED LANGUAGE MODEL

SAN PA PA AUNG

UNIVERSITY OF COMPUTER STUDIES, YANGON

NOVEMBER, 2022

The Automatic Myanmar Image Captioning Using CNN and Bidirectional LSTM-based Language Model

San Pa Pa Aung

University of Computer Studies, Yangon

A thesis submitted to the University of Computer Studies, Yangon in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

November, 2022

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

8.11.2022

Date

San Pa Pa Aung

ACKNOWLEDGEMENTS

First of all, I would like to thank the Union Minister, the Ministry of Science and Technology, for full facilities support during the Ph.D. course at the University of Computer Studies, Yangon.

Secondly, I would like to express very special thanks to Dr. Mie Mie Khin, Rector of the University of Computer Studies, Yangon, for allowing me to develop this thesis and for giving valuable suggestions for this research.

I extend my sincere gratitude to Dr. Mie Mie Thet Thwin, former Rector of the University of Computer Studies, Yangon, for giving the opportunity to do this research.

My greatest pleasure and the deepest gratitude to my supervisor Dr. Win Pa Pa, Professor, Course-coordinator of our Ph.D. 12th batch, University of Computer Studies, Yangon, for her kindly support, encouragement, excellent mentorship, invaluable guidance, practical advice throughout the research period.

I would like to express my deepest appreciation to the external examiner, Dr. Hla Myo Tun, Pro-Rector, Yangon Technological University, for his patience in critical reading, valuable comments and suggestions in the preparation of the dissertation.

I would like to extend my sincere thanks to Dr. Khin Than Mya, my former Course-coordinator of Ph.D. 12th batch, for her kindness, memorable experiences during Ph.D. course and valuable suggestions for doing research.

Moreover, I would also like to mention my special appreciation and thank to Dr. Khin Mar Soe, Professor, Head of Natural Language Processing Lab, University of Computer Studies, Yangon, for her excellent advice and suggestions, and providing me with the institutional resources.

I would like to extend my special thanks to Dr. Win Lelt Lelt Phyu, Professor, University of Computer Studies, Yangon, for helpful advice in the preparation of the dissertation.

I would like to express my respectful gratitude to all my teachers for their encouragement and recommending the thesis. To the reading committee teachers,

especially Daw Aye Aye Khine, Associate Professor, Head of English Department, I would like to thank her for valuable supports and editing my thesis from the language point of view.

I also thank my friends from the Ph.D. 12th batch for their co-operation and encouragement.

Last but not least, I am very much indebted to my family for always believing in me, for their endless love and support. They are always supportive of me during my period of studies, especially for this Doctorate Course.

ABSTRACT

Image captioning is one of the most challenging tasks in Artificial Intelligence which combines Computer Vision and Natural Language Processing (NLP). Computer vision is for detecting salient objects or extracting features of images as an encoder, and Natural Language Processing (NLP) is for generating correct syntactic and semantic image captions as decoder. Describing the contents of an image is a very complex task for machine without human intervention. Computer Vision and Natural Language Processing are widely used to tackle this problem. Although many image caption datasets such as Flickr8k, Flickr30k and MSCOCO are publicly available, most of the datasets are captioned in English language. There is no image caption corpus for Myanmar language. Therefore, Myanmar image caption corpus is created and annotated over 50k sentences for 10k images, which are based on Flickr8k dataset and 2k images are selected from Flickr30k dataset.

In this dissertation, for the purpose of achieving better performance, two different types of segmentations such as word and syllable segmentation level are studied in text pre-processing step. Furthermore, the investigation on segmentation level affects the Myanmar image captioning system performance. The experimental results reveal that the syllable level segmentation gives significantly better performance for Myanmar image description compared with the word level segmentation.

Additionally, this research also constructed its own GloVe vectors for both segmented corpora. As far as being aware and by means of this, this is the first attempt of applying syllable and word vector features in neural network-based Myanmar image captioning system and then compared with one-hot encoding vectors on various different models. Furthermore, the effect of applying GloVe vectors features in language modelling of EfficientNetB7 and Bi-LSTM based image captioning system are investigated in this work.

According to the evaluation results, EfficientNetB7 with Bi-LSTM using word and syllable GloVe vectors features outperforms than EfficientNetB7 and Bi-LSTM with one-hot encoding, other state-of-the-art- neural networks such as Gated Recurrent Unit (GRU), Bidirectional Gated Recurrent Unit (Bi-GRU), and Long Short-Term Memory (LSTM), VGG16 with Bi-LSTM, NASNetLarge with Bi-LSTM models as well as baseline models. The EffecientNetB7 with Bi-LSTM using GloVe vectors achieved the highest BLEU-4 score of 35.09%, 49.52% of ROUGE-L, 54.34% of

ROUGE-SU4 and 21.3% of METEOR score on word vectors, and the highest BLEU-4 score of 46.2%, 65.62% of ROUGE-L, 68.43% of ROUGE-SU4 and 27.07% of METEOR score on syllable vectors.

TABLE OF CONTENTS

4. DATASET PREPARATION

5. VGG16 AND LONG SHORT-TERM MEMORY FOR MYANMAR IMAGE DESCRIPTION

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1
INTRODUCTION

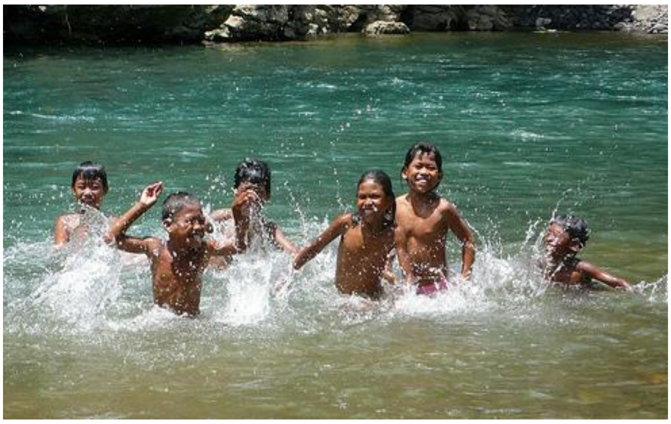
Image Captioning (IC) is one of the most challenging tasks in Artificial Intelligence which combines Computer Vision and Natural Language Processing (NLP). The main vital role of Computer Vision is to extract the key information in images and Natural Language Processing generates the corresponding descriptions.

Image captioning plays the important role for much more reasons. For example, automatic image captioning is useful for helping visually impaired person who can only feel the world by touch, intelligent human computer interactions, and developing image search engines. Social media networks like Facebook and Twitter that can directly generate captions from images. The exact information can be gained from these photos where are the places: (e.g., beach, cafe, and road), what are the people wearing and importantly what are they doing there [7]. The automatic generation of descriptions from the images with proper sentences are very difficult and challenging task for machine.

Myanmar language is semantically complicated and inadequacy of annotated resources than English. Hence, it is required to create a corpus that has to contain sufficient text data to predict the precise caption for automatic Myanmar image captioning. The dataset structure of single image with five distinct annotated Myanmar sentences is illustrated in Figure 1.1.

Typically, Myanmar image captioning system has two main components: image feature extraction as encoder and caption generation with natural language as decoder. In image feature extraction part, the feature vectors of given input images are extracted by using the pre-trained feature extraction models of Convolutional Neural Network such as VGG16, VGG19, InceptionV3, InceptionResNetV2, NASNetLarge and EfficientNetB7 models. In caption generation part, the language models such as Gated Recurrent Unit (GRU), Bidirectional Gated Recurrent Unit (Bi-GRU), and Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (Bi-LSTM) are used to predict the caption with Myanmar language based on the use of previous feature vectors and entire vocabulary in the corpus.

This paper is dedicated to enhance the automatic Myanmar captions by learning the contents of images and generate captions in Myanmar language. Myanmar image captions corpus is created over 50k sentences for 10k images which are based on Flickr8k dataset and 2k images are selected from Flickr30 dataset. The investigation of the text preprocessing steps such as word and syllable segmentation are done and which segmentation level affects the Myanmar image captioning system accuracy. Furthermore, the researcher's own syllable and word GloVe vectors are built that are used in Bi-LSTM language model to improve the efficiency of image captioning system.



(1) ရေ ထဲမှာ ကလေးငယ်များ ကစား နေကြတယ်

(2) ကလေးများ ရေ ကစား နေတယ်

(3) ကလေး ခြောက်ယောက် ရေ ထဲမှာ ကစား နေတယ်

(4) ကလေး အများအပြား ရေ ထဲမှာ ကစား နေကြတယ်

(5) ကလေးများ ရေ ထဲမှာ ကစား နေတယ်

Figure 1.1 Example Image and Five Different Annotated Myanmar Sentences

1.1 Problem Statements

Image captioning is a challenging and elaborated task that has particularly required sufficient amount of linguistic knowledge and resources in the form of images and annotated text data to predict the accurate captions. Most of the research in this image captioning area generated image captions in English while there are a lot of different languages exist in the world. With their distinctive languages, there is a specific need of research in those isolated language. Until now, there are no complete image captioning for different languages because it is difficult to identify the objects accurately in images such as color, gender, age and count of the objects. Image captioning task for Myanmar language is complex for many reasons.

As one of the distinct characteristics of Myanmar language, its morphology is extremely rich and complex and even ambiguous. The deficiency of natural language resources like annotated corpus is the major issue in resolving image captioning for Myanmar language. Currently, Myanmar Natural Language Processing (NLP) is struggling to be developed nonetheless the available lexical resources are very insufficient.

According to the lack of resource and its language nature, it can be said that how to carry out the task of understanding contents of images in Myanmar scripts automatically is still difficult to handle. Due to the mentioned problems, Myanmar image captioning is necessary to develop in the field of Myanmar NLP research and it should be accomplished by applying state-of-the-art methods.

1.2 Motivation of the Thesis

No research has been done for image captioning system in Myanmar language. Automatic caption generation from a given input image is not clear to comprehend for machine because it is a portion of deep learning technique that can be retrieved the information within an image and now a day it is one of the fundamental ambitions of computer vision.

One principal motivation of computational visual recognition models is to challenge wonderful human capacity to understand visual scenes and retrieve particularized information from them with surprising efficiency. A lot of complexed models have been grown to retrieve visual information from images based on visual classification of objects in the images.

The second one is to build GloVe embedding vectors for language modelling. GloVe vectors are needed to generate accurate captions for automatic Myanmar image captioning system. Nonetheless, insufficient amount of captioning data is accessible for Myanmar language to construct GloVe embedding model by implementing machine learning methods. In addition, there is no previously Myanmar image captioning system has been applied GloVe embedding features for language modelling. The impact of these features on language modelling of neural network-based Myanmar image captioning should be examined.

The third one is to generate captions directly from images on social media platforms such as Facebook and Twitter. The exact information can be gained from these photos where are the places: (e.g., beach, cafe, and road), what are the people wearing and importantly what are they doing there. It is very useful and has a great impact on visually impaired people who can only feel the world by touch.

The fourth one is language modelling technique used in Myanmar image captioning system. Retrieval based approach has part of constraint and one of the main factors of downgrading the value of image captioning system is the efficiency of language models.

Hence, Myanmar image captioning system should be accomplished by implementing state-of-the-art modelling techniques to advance image captioning system. As far as being aware and up to the knowledge, the initial effort to implement neural network architecture in Myanmar image captioning system is the fifth motivation.

1.3 Objectives of the Thesis

The core intention of this research is to improve Myanmar image captioning system that can predict the more accurate image captions with Myanmar language. For improving the accuracy of feature extraction, pre-trained feature extraction models of Convolutional Neural Network (CNN) such as VGG16, VGG19, InceptinV3, InceptionResNetV2, NASNetLarge and EfficientNetB7 models have been applied to extract the features of images as encoder and the most suitable feature extraction model has been investigated. To develop the encoder-decoder neural network model, encoder is the vital initial step of image captioning model that extracted all of the features in images and the extracted features are used as input to the decoder.

For language modelling, Gated Recurrent Unit (GRU), Bidirectional Gated Recurrent Unit (Bi-GRU), Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (Bi-LSTM) have been investigated to apply which are the best language modelling in Myanmar image captioning. The following are the other objectives:

In Myanmar image captions corpus, manually created sentences are not segmented precisely to get the quality improvement for image captioning. To advance the quality of Myanmar, two different kinds of segmentation: word and syllable segmentation are used in the first step of the system. One of the purposes of this research is to find out which segmentation level affects in Myanmar Image captioning system.

Construction of image caption corpus for Myanmar language is the most essential factor for implementing machine learning methods in Myanmar image captioning system. There is no publicly available Myanmar image caption dataset yet. Therefore, proposing image captions corpus for Myanmar language is one of the goals of this research.

Another objective is creating word and syllable vectors for Myanmar language which can provide vast coverage and satisfying achievement, and implementing these vectors as the input features to language modelling of Myanmar image captioning system to examine their effectiveness.

1.4 Contributions of the Thesis

There are five major contributions in this thesis:

The very first contribution of this thesis is constructing a Myanmar image captions corpus and using that corpus for caption description in language modelling part of automatic Myanmar image captioning system. This is the initial published work designed for Myanmar image captioning.

The second contribution is exploring text preprocessing for Myanmar text segmentation by the study of Myanmar image captions corpus and implementing for both word and syllable segmentation corpus and applying these corpora in language modelling of Myanmar image captioning.

The third contribution is investigating feature extraction model for learning objects of given images and applying these best feature extraction model in image understanding part of image captioning system as encoder. EfficientNetB7 feature extraction model is used for Myanmar image caption generation, which can correctly recognize the objects in the images compared with other different feature extraction models.

Unsupervised learning is used to achieve the word vectors from huge amount of raw text data and can be utilized as the input features to language modelling of image captioning system. Constructing monolingual corpus for the ambition of creating word and syllable vectors for Myanmar language and implementing these vectors features as the additional input features to language modelling of Bi-LSTM based Myanmar image captioning system is the fourth contribution.

Recently, word vector approach presented the powerful achievement over one-hot encoding approach. The fourth contribution is using neural network-based architecture in Myanmar image captioning system for advancing the efficiency of language model. Enhancing of various input features including word embedding features and syllable embedding features in language modelling of neural network-based Myanmar image captioning system is the fifth contribution of this thesis. The appropriate network architecture for Myanmar image caption generation is examined by doing various investigations on GRU, Bi-GRU, LSTM, and Bi-LSTM based language models.

1.5 Organization of the Thesis

This dissertation is organized with eight chapters including introduction of image captioning system, problem statements, motivations, objectives, and contributions of this research.

The literature review on image captioning techniques, related work of this research and evaluation metrices of Image captioning are described in Chapter 2. In Chapter 3, the computational models of ImageToText have been discussed. The detailed process of building Myanmar image captions corpus and construction of GloVe vectors for both word and syllable segmented corpus are described in Chapter 4. Moreover, image preprocessing step is also illustrated. The detailed implementation and experimental results of VGG16 and LSTM based Myanmar image description is reported in Chapter 5. Chapter 6 describes the comparison of four different types of encoder decoder pairs and also reports the implementation results for Myanmar image description. In chapter 7, two distinct kinds of segmentation such as word and syllable are studied and also reported the detailed implementation and experimental results with different input features. Moreover, the proposed model is compared with baseline models and word embedding features are also explored and investigated in this chapter. Finally, Chapter 8 summarizes with the research work and depicts the advantages and limitation of the system, and the further directions to progress the image captioning system.

CHAPTER 2
LITERATURE REVIEW AND RELATED WORKS

This chapter presents the literature review on image captioning, related work of this research and some former research of image captioning on different languages.

The insufficient amount of image annotation dataset for morphological complicate language rather than English is an issue to obtain the precise results. The image description generation is especially divided into two approaches such as retrievable based approaches and constructive-based approaches. The initial approach is applied in the former experiments to solve image captioning which has the difficulty as a retrieval task. A database is created according to the image features extraction and caption generation for given images and then the most suitable captions are retrieved [19]. This approach is not powerful to predict correct captions and the generation of captions are limited with the feature size of images and the size of database. Hence, retrieval-based methods are not suitable for current requests.

Currently, constructive-based methods have developed and well-known for the reason that current development in automatic image captions description and neural machine translation. A constructive-based method constantly generates the accurate caption for individual image [10][2]. The authors [54] applied this method that is additionally split into two modules as deep convolutional neural network for encoding image attributes and Long Short-Term Memory network for decoding to predict a grammatically accurate description.

Nowadays, a lot of images are found in many different ways such as the Web site, news articles, social media and commercial. Humans easily understand to transform these images and to demonstrate a natural language. Nonetheless, machines are difficult to demonstrate textual captions from an image because the machines require to comprehend the semantic and the contents of the images. A long-standing purpose in the domain of Artificial Intelligence is to make possible machines to observe and comprehend the image of the encompassing.

2.1 Overview of Image Captioning

Automatic generation of image descriptions need to understand both images understanding and a language generation for that image. Image understanding is the main difficulty of Computer Vision. Language generation consists of the portion of Natural Language Understanding (NLU). A conventional image captioning groundwork contains an image encoder to find out features from an image and a language decoder to produce captions for that image.

2.1.1 Image Understanding

Computer vision is the capability of machines to observe and comprehend what is in their encompassing. There are many different ways to extract needed information from the images. A powerful chunk of this field is performed in computer vision; specifically visual recognition and visual understanding. Visual recognition consists of identifying, localizing, and classifying objects of an image. Visual comprehending is necessary objects recognition as well extracting the complete detail of particular object and their associated relationship. An image captioning approach requires to accurately identify various objects. An image object can have numerous features rather than one attribute. In deep learning-based methods, features are automatically learned. Convolutional Neural Networks (CNNs) are deep neural network architectures that are considered for functioning on images, videos, sound spectrograms in speech processing, character sequences in text and so on.

2.1.2 Natural Language Understanding

In accordance with the NLU point of view, producing text contains a sequence of steps. Firstly, the visible feature of the input which is also known as content selection have to understand and the content of text planning require to arrange. Finally, surface realization requires to communicate. Surface realization needs lexicalization that means to choose the correct words, referential expression generation utilizing suitable pronouns, and then bringing together correlated information termed as aggregation. Recurrent Neural Network (RNN) [57] and Long Short-Term Memory (LSTM) [51], Gated Recurrent Unit (GRU) [29] and Bidirectional Long Short-Term Memory [4] have commonly used deep learning-based language models that presented better achievement in many natural language understanding tasks including image captioning.

2.2 Image Captioning for Different Languages

In this chapter, several papers will be presented which discussed about the building image captioning with different models and datasets. The used dataset is different languages such as English, Indonesian, Hindi, Arabic and Chinese.

2.2.1 Image Captioning for English Language

Due to the success of deep neural network, most of the researchers have proposed to use the encoder-decoder framework for machine translation as well as image captioning.

Neural network for image captioning was proposed early by Vinyals et al [43], which is an encoder-decoder system. Convolutional Neural Network (CNN) is used for image encoding and Recurrent Neural Network for decoding captions. BLEU scores evaluated on both MSCOCO dataset and Flickr30k dataset.

The authors [65] proposed some modification based on the original model to explore a new language model to generate more accurate descriptions for input image. Long-Term Recurrent Merge Networks (LTRMN) model with a double layer LSTM designed and added a merge layer into the middle of double-layer LSTM, and then the image content vector is added in the merge layer, so that the generated image captions are more consistent with the contents of image.

In paper [64], the end-to-end architecture to generate the image captions is implemented. This architecture generated better textural captions from the image representation by replacing the CNN encoder in the place of RNN encoder. The proposed model is also called Neural Image Caption model. The output of last hidden layer CNN is fed into the RNN decoder to predicts the textual captions for the image.

In paper [15], VGG16 and Alexnet are used as an encoder and Bi-LSTM model is used as a decoder. Experiments are done on three popular benchmark datasets: Flickr8k, Flickr30k and MSCOCO datasets. Alexnet visual model is inferior effective than VGG16. The authors obtained the maximum BLEU-1 scores 65.5%, BLEU-2 scores of 46.8%, BLEU-3 scores of 32.0% and BLEU-4 scores of 21.5% respectively by applying VGG16 with Bi-LSTM model on Flickr8k dataset.

In [59], deep convolutional neural network was applied to find out the image contents and two different LSTM network is used to discover long-term visual-language interactions and make prediction by the applying of history and future context information at high-level semantic space. Then, the deep multimodal bidirectional models also examined, in which the depth of nonlinearity transition is expended in numerous approaches to identify hierarchical visual-language embeddings. The performance of proposed models is measured on four popular benchmark datasets such as Flickr8K, Flickr30K, MSCOCO, and Pascal1K datasets. The authors obtained the highest BLEU-N (N=1, 2, 3, 4) scores 66.7%, 48.3%, 33.7% and 23% respectively and 19.1% of METEOR score using VGG16 with Bi-LSTM model on Flickr8k dataset. The authors observed that the model performance on small-scale dataset Flickr8K is not great as huge dataset likes Flickr30K and MSCOCO. It is unsuccessful to recognize the objects in complicated background images as a result of feature extraction model is not state-of-the-art model that has only 16 layers deep. The deeper networks earnings the better comprehending the contents of the images. Furthermore, their method does not take into account word embedding in language modelling that make generation image captions task better.

In paper [60], ResNet101 was applied as feature extraction model and Standard LSTM with one cell is used to make prediction in the sentences as decoder. The pretrained vector representations as Word2vec and GloVe embedding are compared on the MSCOCO dataset. The model performance with GloVe vectors achieved better results than the model with Word2vec because image captioning is more suitable with co-occurrence of word pairs in the entire corpus. Moreover, it did not state the generated captions results, and they used only the pre-trained word vectors with English language. In this article, the results of the models were compared by calculating the quality of the image captioning task with BLEU, METEOR, ROUGE-L, CIDEr and SPICE metrics which are widely used algorithms.

2.2.2 Image Captioning for Indonesian Language

In this subsection, there are at least 3 papers which have discussed about generating Indonesian image captioning, in reference [1][39][22]. In [1], the authors were proposed generating image description on Indonesian Language by using pre-trained Inception-V3 and Gated Recurrent Unit. To obtain Indonesia image captioning dataset from Flickr30k dataset, google translator is used and manually repaired some of these results by checking the captions one by one. The proposed model obtained

BLEU scores 36, 17, 6 and 2, respectively. These BLEU scores are not good because some of the translated results are manually checked none for all translated sentences, therefore, the dataset is not clean. The clean dataset is very important to improve the quality of the system in training machine learning models.

In [39], the authors investigated the attention-based image captioning model by utilizing ResNet101 feature extraction model of CNN as the encoder and LSTM with adaptive attention as the decoder for Indonesian image captioning task. In these work, MSCOCO and Flickr30k dataset are used to train the model. To construct the Bahasa Indonesian corpus, both of these datasets are translated into Bahasa by using Google Translate and manually checked by human. They achieved the BLEU-1 score of 0.678, BLEU-2 score of 0.512, BLEU-3 score of 0.375, BLEU-4 score of 0.274 and CIDEr score of 0.990.

In [22], Convolutional Neural Network was used to extract the content of the images and Long Short-Term Memory is used to predict the captions with Indonesian language. The authors used the FEE-ID dataset which are taken from Flickr. Google Translate and professional English-Indonesian translator are used to transform English captions to Indonesian captions. This dataset contains the total 8099 images and single picture is equipped with 5 Indonesian sentences. This model achieved the BLEU-1score of 50.0 %, BLEU-2 score of 31.4%, BLEU-3 score of 23.9% and BLEU-4 score of 13.1%.

2.2.3 Image Captioning for Hindi Language

Rijul et al. [49] implemented the Hindi Language image caption generation using RESNET 101 Convolutional Neural Network for understanding the content of images and Gated Recurrent Unit for caption generation. MSCOCO dataset is used to build the Hindi corpus. In the first step, google translator is utilized to transform English captions to Hindi captions, and then the agreement of 84% is taken from two annotators who manually checked and corrected the translated sentences on sample data of 400 captions. These annotators confirm and correct the output results acquired from Google translator. Experiments results described that the proposed model attained BLEU-1 score of 57.0%, BLEU-2 score of 39.1%, BLEU-3 score of 26.4% and BLEU-4 score of 17.3 on this dataset. In that work, some of the generated captions occurred the errors because non-presence of certain words in the training dataset. If the size of dataset increases, these kinds of errors will reduce.

In [8], VGG16 pretrained feature extraction model of CNN was used to learn the visual features of the images and LSTM is used to predict the captions in Hindi language. The authors constructed the Hindi image captions dataset which is based on Flickr8k dataset. Google cloud translator is used to translate English captions to Hindi captions, which is called “Flickr8k-Hindi Dataset”. This dataset contains four different types of data based on a number of descriptions for each image according to clean or unclean descriptions. The experimental results stated that training the model with a single clean description per image generates higher quality caption rather than a model trained with five uncleaned descriptions per image.

RESNET101 pre-trained feature extraction model of CNN was used to extract the features from the images. It is working as an encoder to encode the images into a fixed-length vector representation. Gated Recurrent Unit (GRU) is used with different types of attention-based architectures such as spatial attention, visual attention, Bahdanau attention, and Luong attention that help to make prediction in caption generation with Hindi language. The authors used the MSCOCO dataset which contains the total 84405 images with different five captions per image to implement the Hindi image captioning model. To build Hindi corpus, all of the English captions are translated into Hindi by using Google translator, but the translated sentences lost the meaning of the captions and some of the sentences are grammatically incorrect. So, to avoid these mistakes, the translated corpus is corrected manually by human annotators. The authors compared the obtained results with several baselines models in terms of BLEU scores [52].

2.2.4 Image Captioning for Arabic Language

There are few papers that discussed generating Arabic image captioning system. The authors [5] used Deep Learning Technique to implement the automatic image captioning for Arabic language. Visual Geometry Group (VGG) OxfordNet 16-layer of CNN is used for the image encoder. For the language model, Long Short-Term Memory is used to predict the sentences with Arabic language. The authors have taken the 1166 images from MSCOCO and 2261 images from Flickr8k dataset. The total images from both datasets (COCO and Flickr) are 3427. The authors built the Arabic corpus by collecting various ways 5358 descriptions for 1176 images utilizing Crowd-Flower Crowdsourcing service, and 750 descriptions for 150 images were achieved from a human translator. A professional English-Arabic translator is used to translate the rest of the image's captions to Arabic captions and then, the translated sentences are checked by Arabic native speakers.

V. Jindal [61] applied Region Convolutional Neural Network (RCNN) to detect the objects in images and root-word based Recurrent Neural Networks with LSTM memory cell is used to generate the most appropriate words for an image. Moreover, dependency tree relations of the obtained words are used to check the word order of the RNN to form the sentences in Arabic. The datasets are collected from various Middle Eastern newspaper websites. And then, the author compared the results of his proposed model with BLEU score captions generated in English and translated into Arabic.

In [27], three different models have demonstrated: the first model is multi-object-based captioning that can handle one or multiple detected objects. In this model, image is preprocessed according to the object detector requirements. The preprocessed images are fed forward into the object detector which are extracted the detected objects. The second one is a combined pipeline that utilized both object detector and attention-based captioning. The third model is applied soft attention mechanism. In this work, the authors used the MSCOCO 2014 dataset which contains 123,287 images and 5 different captions per image. All of three models' performance is evaluated by using multi-lingual semantic sentence similarity techniques.

2.2.5 Image Captioning for Chinese Language

In [24], Visual Geometry Group (VGG) 16 layers of CNN was used to learn the objects of the images and Recurrent Neural Network (RNN) is applied to predict the sentences in Chinese language with two different segmentations methods such word level and character level. Experiments is done on Flickr30k dataset, which contains 31,000 images and each image with 5 Chinese sentences. The authors obtained the Chinese captions by using Google Translation API. According to their experiments results, character level segmentation achieved much better results than word-level for Chinese sentences.

In [63], Pool5 layer of a pre-trained GoogLeNet was utilized to extract the features of the images and Long Short-Term Memory is used to predict the sentences based on the previous features extraction model and entire vocabulary in the image captions corpus. To build the Chinese captions corpus, the authors used the Flickr8k dataset which contains 8000 images and five annotated English captions. Firstly, they used the English-Chinese translation services to transform English captions to Chinese captions, but the translation results are not performed well while the sentences become longer and consists of the ambiguous words. That is why, a number of native speakers of Chinese performed the annotations, where they have written the sentences from their own point of views that describing salient objects and scenes in every images.

In [69], the authors used the Inception V4 image feature encoding algorithm which is the fourth-generation CNN model of the Google Net series. Long Short-Term memory network is used in language decoder. Experiment is done on Chinese dataset also called the ICC which contains 7000 images. The performance of the system is measured by using BLEU and METEOR algorithm. The authors stated that the overall test results are more accurate, but some details results are flawed because of the equipment limitations, and the lack of the training time leads to low recognition rates for some details.

2.3 Evaluation Metrics

The performance of the system is measured using BLEU-N(N=1,2,3,4) metrics [33], ROUGE-L, ROUGE-SU4[17] and METEOR [50] metric which are mostly used to measure the accuracy of image description generation.

2.3.1 Bilingual Evaluation Understudy (BLEU)

Bilingual Evaluation Understudy (BLEU) is a matric that is applied to evaluate the quality of machine generated texts. BLEU compares the N-gram (N=1,2,3,4) of the candidate translation with N-gram of the reference translation to count the number of matches [47]. The output of BLEU score range is always between 0 and 1, value nearly to 1 display that the produced captions are more equivalent to the ground-truth cations and 0 is no match at all. Equation 2.1 and Equation 2.2 are used to evaluate the BLEU scores.

$$\text{BLEU} = \min\left(1,\frac{hypothesis\ length}{reference\ length}\right)\left(\prod_{i=1}^{4} precision_i\right)^{1/4} \qquad (2.1)$$

Where, hypothesis length = generated caption length
reference length = ground truth caption length

$$\text{Precision} = \frac{Max\ number\ of\ times\ N\ gram\ occurs\ in\ reference}{Total\ no\ of\ N\ grams\ in\ hypothesis} \tag{2.2}$$

2.3.2 Example Calculation of BLEU bi-gram

Reference R: လူ လေး ယောက် လမ်း ပေါ် မှာ စက် ဘီး စီး နေ တယ် (11 words)

Predicted P: လူ နှစ် ယောက် က စက် ဘီး စီး နေ တယ် (9 words)

Bi-gram for Reference: (လူ, လေး), (လေး, ယောက်), (ယောက်, လမ်း), (လမ်း, ပေါ်), (ပေါ်, မှာ), (မှာ, စက်), (စက်, ဘီး), (ဘီး, စီး), (စီး, နေ), (နေ, တယ်)

Bi-gram for Predicted: (လူ, နှစ်), (နှစ်, ယောက်), (ယောက်, က), (က, စက်), (စက်, ဘီး), (ဘီး, စီး), (စီး, နေ), (နေ, တယ်) ➔ 8

$$\text{Precision} = \frac{0}{8} + \frac{0}{8} + \frac{0}{8} + \frac{0}{8} + \frac{1}{8} + \frac{1}{8} + \frac{1}{8} + \frac{1}{8} = \frac{4}{8} = 0.5$$

$$\text{BLEU} = \min\left(1, \mathbf{\frac{9}{11}}\right)\left(\frac{1}{2} * \frac{1}{2} * \frac{1}{2} * \frac{1}{2}\right)^{1/4} = 0.82*0.5 = 0.41$$

2.3.3 Recall-Oriented Understudy of Gisting Evaluation (ROUGE)

ROUGE is an abbreviation of Recall-Oriented Understudy of Gisting Evaluation (ROUGE) [17] that is a set of metrics used for the computing of automatic text summarization, machine translation and image captioning. The metrics fundamentally compare automatically machine generated summary with reference summary or multiple reference summaries. There are the five-evaluation metrics such as ROUGE-N, ROUGE-L, ROUGE-W, ROUGE-S and ROUGE-SU.

ROUGE-N: evaluates the overlapping of n-grams in a candidate caption and a set of reference captions. The n-grams value can be differed from 1 to n but if the value of n increase, the evaluation cost will also increase immediately. Commonly utilized n-gram metrics are uni-gram and bi-gram.

ROUGE-1: calculates the overlapping of unigram between reference captions and Machine generated captions.

ROUGE-2: Evaluates the overlapping of bi-grams in reference captions and machine generated captions.

ROUGE–L: compute the longest common subsequence between reference captions and candidate caption. Each sentence in a summary is considered as a sequence of words. Two summaries which have longer common sequence of words are more similar to each other. One advantage of ROUGE-L is that it does not contain successive matches of words that is only consider sequence of words. Next advantage is that it does not need to predefine sequence of n-gram it automatically finds the longest sequence of n-grams.

ROUGE-W: consider consecutive sequence from machine generated captions and assigned more weight to sentence which in actual is more analogous to reference sentence.

ROUGE-S: evaluates the skip bigram co-occurrences in reference captions and candidate caption. Ordering of bigrams is crucial but skip bigram is fundamentally any pair of words in sentence order. Any arbitrary gaps are allowed.

ROUGE-SU4: The disadvantage of ROUGE-S is that it considers only bigrams. If a sentence does not occur any overlapping bigrams, it will not assign any weight value to these sentences. ROUGE-S is upgraded as the ROUGE-SU that is used to overcome the problem of ROUGE-S. It is also considered unigram with bigrams. In this work, ROUGE-L and ROUGE-SU4 are used for comparison with other models. To measure the accuracy of the machine generated captions, it is necessary to evaluate the Precision, Recall and F-measure for any of this metric.

In ROUGE Recall refers that how much words of candidate summary are extracted from reference summary.

$$\mathrm{R} = \frac{number\ of\ overlaping\ words}{Total\ words\ in\ reference\ summary} \tag{2.3}$$

In ROUGE Precision refers that how much candidate summary words are relevant.

$$\mathrm{P} = \frac{number\ of\ overlaping\ words}{Total\ words\ in\ candidate\ summary} \tag{2.4}$$

F measure provides the complete information that recall and precision provides separately.

β=0.9,

$$\text{F-measure}=\frac{(1+\beta^2)RP}{R+\beta^2+P} \qquad (2.5)$$

2.3.4 Example Calculation of ROUGE-L

Reference R: လူ လေး ယောက် လမ်း ပေါ် မှာ စက် ဘီး စီး နေ တယ် (11 words)

Predicted P: လူ နှစ် ယောက် က စက် ဘီး စီး နေ တယ် (9 words)

ROUGE-L: လူ ယောက် စက် ဘီး စီး နေ တယ် (7 words)

$$\text{Recall} = \frac{\mathbf{7}}{\mathbf{11}} = 0.636$$

$$\text{Precision} = \frac{\mathbf{7}}{\mathbf{9}} = 0.777$$

B = 0.9,

F-measure = (1.81 * 0.64 * 0.78) / (0.64 + 0.78) = 0.903 / 1.42 = 0. 6363

2.3.5 METEOR Metric

Metric for Evaluation of Translation with Explicit Ordering (METEOR) [50] evaluated the average scores of precisions and recall values according to the unigram. METEOR is combination of both precision and recall metric that is the main difference with BLEU. METEOR can overcome the restriction of rigorous matching by using the word and similar meaning based on unigram although BLEU and ROUGE have the difficulties to overcome that restriction.

Precision,

$$\mathrm{P}=\frac{\mathrm{m}}{\mathrm{w_t}} \qquad (2.6)$$

Recall,

$$\mathrm{R}=\frac{m}{w_r} \qquad (2.7)$$

$$F_{mean} = \frac{PR}{\alpha P + (1-\alpha\ )R} \tag{2.8}$$

where,

m: Number of unigrams in the candidate translation also found in reference

w_t : Number of unigrams in candidate translation

w_r : Number of unigrams in reference translation

2.3.6 Example Calculation of METEOR

Reference R: လူ လေး ယောက် လမ်း ပေါ် မှာ စက် ဘီး စီး နေ တယ် (11 words)

Predicted P: လူ နှစ် ယောက် က စက် ဘီး စီး နေ တယ် (9 words)

Precision, $P = \frac{\boldsymbol{m}}{\boldsymbol{w_t}} = \frac{\mathbf{7}}{\mathbf{9}} = 0.777$

Recall, $R = \frac{m}{w_r} = \frac{\mathbf{7}}{\mathbf{11}} = 0.636$

$\alpha = 0.9$,

$F_{mean} = \frac{\boldsymbol{PR}}{\alpha P + (\mathbf{1}-\alpha\ )\boldsymbol{R}} = 0.6496$

2.4 Summary

In this chapter, overview of image captioning system and other different languages such as Arabic, Chinese, Hindi, Indonesian and English are reviewed. In accordance with the review of previous image captioning system on different languages, there is no research on image captioning system for Myanmar Language. Moreover, three evaluation methods such as BLEU, ROUGE and METEOR metrics are described in this chapter.

CHAPTER 3
COMPUTATIONAL MODEL

In this chapter, the computational model of deep learning has mainly discussed. Deep learning is a machine learning approach that is used to teach computer to do what appear natural to humans: determine by example to carry out classification tasks directly from images, text, and sound. Deep learning models can obtain state-of-the-art accuracy, sometimes superior human-level performance. A large set of labeled data are used to train the models and neural network architectures that include a lot of layers. In this work, the comparison on various neural network models is done for both encoder and decoder such as various pre-trained feature extraction models of Convolutional Neural Network (CNN) as encoder, Gated Recurrent Unit (GRU), Bidirectional Gated Recurrent Unit (Bi-GRU), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory, and Word embedding model as decoder.

3.1 Artificial Neural Networks (ANN)

Artificial Neural Networks (ANNs) are computational networks which are massively stimulated by the idea of biological nervous systems (such as the human brain) function. ANNs essentially consists of a high number of interconnected processing elements, which intertwine in a distributed fashion altogether to find out from the input in order to optimize its final output.

Figure 3.1 presents the fundamental architecture of an ANN. All of the weight-adjusted input values to a computational processing element are aggregated using a multidimensional vector of scalar function. After the input value is evaluated which will distribute it to the hidden layers. The process is repeated by the hidden layers to make decisions from the previous layer and weigh up how a stochastic change within itself detriments or improves the final output, and this is referred to as the process of learning. Having several hidden layers one by one and propagate information sequentially creating a deep structure is commonly called deep learning.

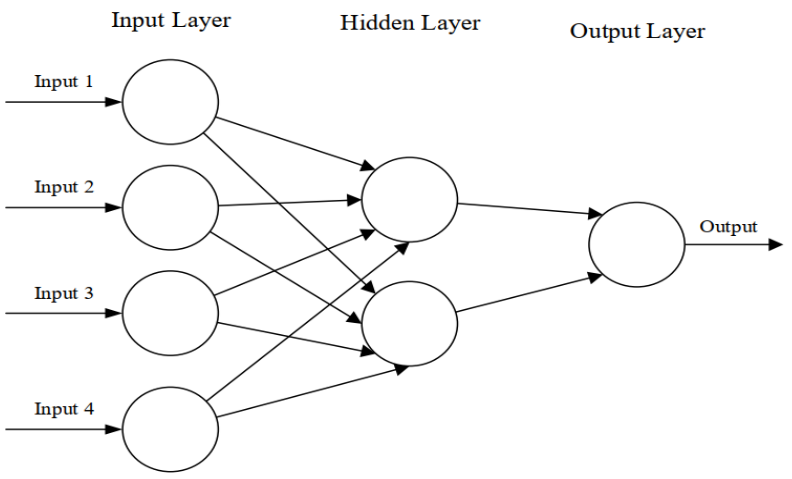


Figure 3.1 A Simple Three-Layered Feedforward Neural Network (FNN)

There are two key learning approaches in machine learning: supervised and unsupervised learning. **Supervised learning** requires the pre-labelled inputs, which is utilized to classify new and to predict outcomes for unseen datasets. **Unsupervised learning** does not require any labels data. Training is done on raw and unlabeled data that is used to recognize patterns and to cluster similar features into a definite number of groups. The biggest difference between supervised and unsupervised learning is that labeled data is used to assist prediction outcomes or to classify data in one approach, whereas unsupervised learning is especially applied to comprehend relationships within datasets.

3.2 Convolutional Neural Network (CNN)

Convolutional neural networks are distinguished from other neural networks by their superior performance with image, speech, or audio signal inputs. Figure 3.2 shows the block diagram of Convolutional Neural Network.

Especially, there are four main operations in the Convolutional Neural Network as shown in Figure 3.2 below:

1. Convolutional layer
2. Non-Linearity (ReLU)
3. Pooling layer or Sub Sampling
4. Fully-connected (FC) layer

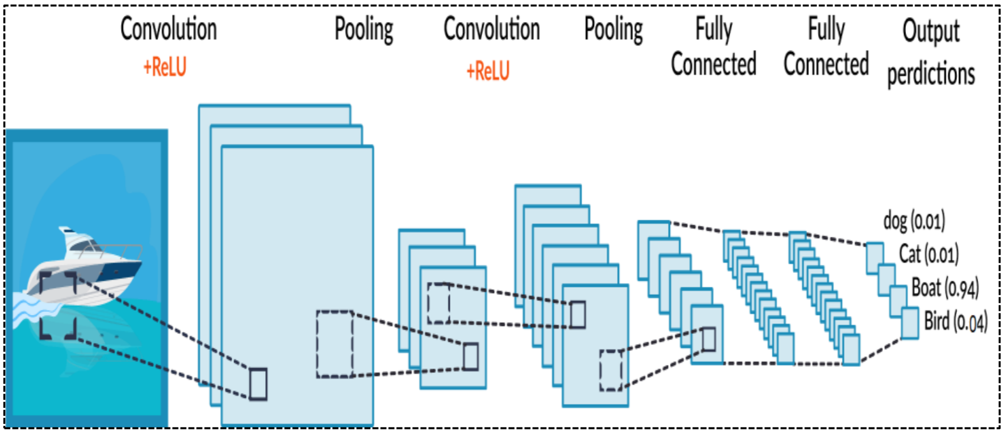


Figure 3.2 Block Diagram of Convolutional Neural Network

CNN grows in its complication for each layer, recognizing larger portions of the image. Former layers recognize on simple features, such as colors and edges of images that are improved by the way of the layers of CNN. It begins to identify greater elements or shapes of the object as far as it lastly focused the expected objects.

3.2.1 Convolutional Layer

The core building block of a CNN is the convolutional layer where the greater part of computation takes place. Whenever CNN is applied for image captioning system, some components such as original input image, a filter, also known as a feature detector or a kernel that are the part of original input images are needed for moving across the receptive fields of the image to check if the feature is present. Every image can be considered as a matrix of pixel values. Let’s assume that the input may be a color image (3D image) which is the range of 0-to-255-pixel values. There are three dimensions in the correspondence RGB images such as height, width, and depth.

The size of the filter can vary generally a 3x3 matrix weights that express the portion of the images to decide the size of receptive field. Let consider the 5x5 input image and 3x3 filter whose pixel values are as follows:

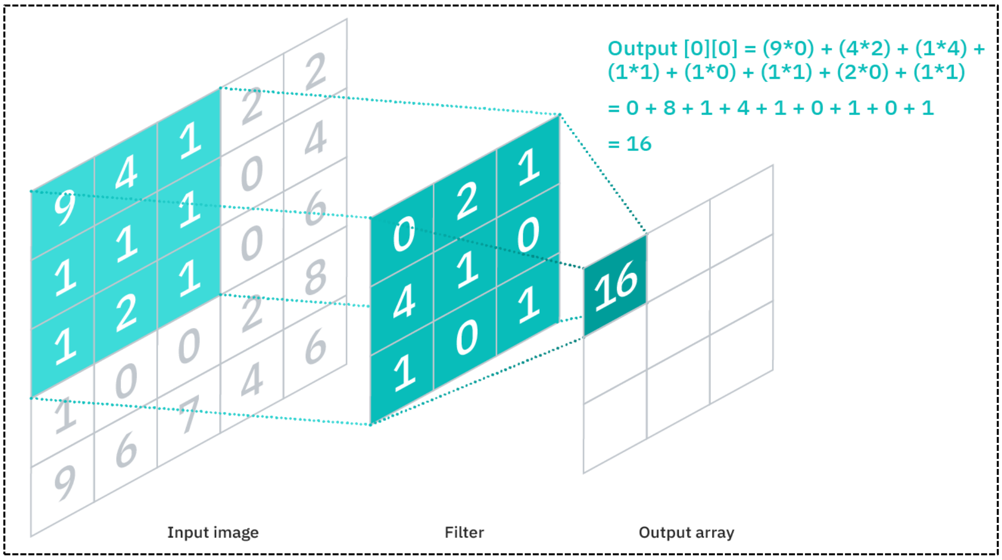

Output [0][0] = (9*0) + (4*2) + (1*4) + (1*1) + (1*0) + (1*1) + (2*0) + (1*1)
= 0 + 8 + 1 + 4 + 1 + 0 + 1 + 0 + 1
= 16
16
Input image
Filter
Output array

Figure 3.3 The Convolution Step

As can be seen in Figure 3.3 above, the feature detector matrix is slide over our original input image matrix by 1 pixel that are also called stride and for every position. It computes element wise multiplication between input image matrix and filter matrix and then the multiplication outputs are added to achieve the final integer values that are feed forward into the output array. Subsequently, the filter is moved according to a stride value and the process is repeating just before the kernel has swept across the whole image. The latest output from the series of dot products from the input and the filter is known as a feature map, activation map, or a convolved feature.

There are three hyperparameters to control the size of the output volume: the depth, stride, and zero-padding.

- First, the **depth** of the output volume is a hyperparameter: it correlates to the size of filters that are used for each learning to look for something different in the input. For example, the raw image is taken as input by the first Convolutional Layer, then different neurons along the depth dimension may activate in presence of various oriented edges, or blobs of color. A set of neurons will be mention that are all looking at the similar area of the input as a **depth column.**
- Second, the **stride** value has to define to slide the filter. If the stride is 1 then we shift the filters one pixel at a time. If the stride is 2 (or if the stride value is 3 or more, it is uncommonly and rare in practice), the

filters will jump 2 pixels at a time as we slide them around. After this operation will be produced the smaller output volumes.

- The last one is **Zero-padding** that is utilized meanwhile the input images are not met with the filters. This sets all elements that the outside of the input matrix is filled with zero, generating a greater or equally sized output.

3.2.2 Non-Linearity (ReLU)

After the convolutional layer operation is done, Rectified Linear Unit (ReLU) is used to replace all negative pixel values in the feature map by zero. In this stage, we can also be used other non-linear functions such as Tanh or Sigmoid instead of ReLU. According to the experimental results, ReLU has been found to perform better in most situations. After applying the ReLu function, the rectified feature map (only non-negative values) is obtained.

As we specified previously, the initial convolution layer is followed by another convolution layer. Meanwhile this occurs, the architecture of the CNN turns into hierarchical as the next layers can observe the pixels inside of the receptive fields of previous layers.

3.2.3 Pooling Layer

Pooling layers are also called down sampling which is used to decrease the dimensionality of each feature map and the size of parameters in the original input images. The operation of pooling layers is very similar with the convolutional layer, but the largest distinction is that this filter does not have any weights. Alternatively, the feature detector uses an aggregation function to the values within the receptive field, commonly the output array. It has three main kinds of pooling operations and they are as follows:

- **Max pooling:** It takes the largest element from the rectified feature map within that window and assign to the output array. This operation is commonly used compared to average pooling.
- **Sum pooling:** It evaluates the addition of all elements from the rectified feature map within that window and assign to the output array.

- **Average pooling:** Instead of taking the largest elements or sum of all elements, the average of all elements could be taken from the rectified feature map withing that window and then assign to the output array.

Although a lot of information is vanished in the pooling layer, but retains the most important information. It also has a number of advantages to the CNN. They assist to decrease complication, promote capability, and restrict the risk of overfitting. Figure 3.4 shows the operation of max pooling step.

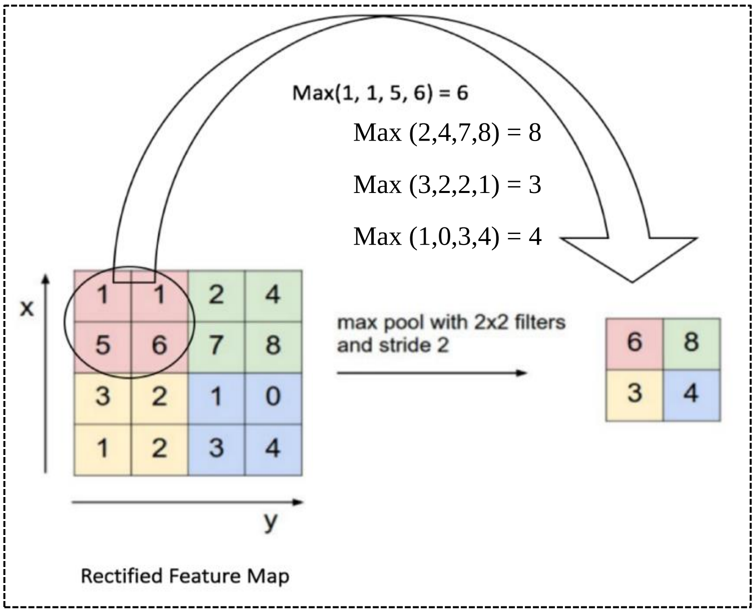


Figure 3.4 Max Pooling

3.2.4 Fully-Connected Layer

In the fully-connected layer, the pixel values of the input image are not directly communicated to the output layer in fractionally connected layers. Nevertheless, in the fully-connected layer, each node in the output layer connects directly to a node in the former layer.

This layer carries out the task of classification based on the features extraction through the former layers and their different filters. While convolutional and pooling layers turn to apply ReLu functions, FC layers normally used a softmax activation function to classify the given input images appropriately, generating a probability value from 0 to 1. This layer classified the given input image with their probability values using softmax activation function. Figure 3.5 displays the fully connected layer- each node is connected to every other node in the adjacent layer [32][70]. According to the four possible outputs as shown in Figure 3.5, a given input image may be boat because it has the highest probability value 0.94 among them. These are the overview operations of CNN.

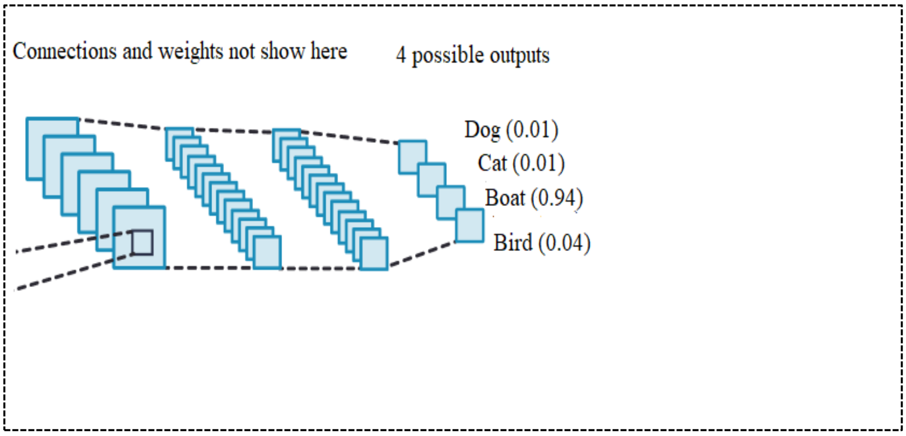


Figure 3.5 Fully Connected Layer

3.3 Types of Convolutional Neural Networks

A number of various CNN architectures have developed with the initiation of new datasets, such as MNIST and CIFAR-10, and competitions, ImageNet Large Scale Visual Recognition Challenge (ILSVRC). Keras applications module is applied to produce pre-trained model for deep neural networks that are utilized for prediction, feature extraction and fine tuning.

3.3.1 Pre-trained Feature Extraction Models

Pre-trained model contains two modules: model Architecture and model Weights. Model weights are large file so we have to download and extract the feature from ImageNet database. In this thesis, six popular pre-trained feature extraction models are used as follows:

- VGG16
- VGG19
- InceptionV3
- InceptionResNetV2

- NASNetLarge
- EfficientNetB7

VGG16: Visual Geometry Group (VGG) OxfordNet 16-layer of CNN is a pre-trained model on ImageNet dataset that is utilized for image classification. The Output of VGG16 is the probability of each class to classify for the image classification system. The last layer of the VGG16 is removed because we need to utilize the output of second last layer as feature parameters for individual image. It has 4096 parameters to extract the feature vectors for each image and default input image size of VGG16 is 224x224 that are next processed by a Dense layer to generate a 256 elements representation of an image [34][53].

VGG19: The functionality of Visual Geometry Group (VGG) OxfordNet 19-layer is very similar with VGG16. VGG16 and VGG19 networks have the total number of weight layers 16 and 19 respectively. VGG19 has 3 more convolutional layers than VGG16 [34].

InceptionV3: is applied in image feature extraction module. During the training module, the model has two inputs: firstly, an image feeds into the pre-trained InceptionV3 model by removing output layer and added a fully connected layer that is called dense map which is utilized to connect the extracted features of images with the first state in Language Model. The second input is a caption that has been preprocessed which is become a sequence index of tokens. During the testing module, the input is an image as well as index of token and will generate the index of next tokens one by one. The index tokens have previously generated that will be passed again until the model generates maximum sequence length or end of the token. Inception-V3 has different feature extraction capabilities from another CNN architectures. It uses a lot of convolution filters and combines the convolution results in inception module. The features vector of input images is defined to be 2048 elements and processed by a Dense layer to produce a 256 elements representation of the photo [13][14].

Google's InceptionResNetV2: GoogleAI's ingenious InceptionResNetV2 model is the main part of image understanding module. The CNN is one of the most powerful feature extraction models which is comprised of various innovative approaches and helps us to generate captions. The number of weight layers in the network are 449 layers deep and it is widely accepted by now. The deeper networks yield the better understanding the contents of images. The initial layers of network consist of 3 standard convolutional layers followed by a max-pooling layer which is again followed by 2 convolutional layers and a max pooling layer. The inception convolution is the next stage in the network and simultaneously convoluting an input using various sizes of filters for each convolution and then stacking the result together and feeding it forwards to the rest of the network. The inceptions and residual part of the network perform where the network uses dropout layers to prevent overfitting. What's more, the second last layer is a fully connected layer which influence all neurons on the basis of learnings. Finally, softmax layer is used to distribute the probability scores to the final 1000 neurons [13][14].

NASNetLarge: NASNetLarge is a pre-trained feature extraction model of Convolutional Neural Network (CNN) that achieved 82.5% top-1 classification accuracy on ImageNet dataset. This model is trained using ImageNet database which contains over the million images. It can be categorized the images into 1000 object classification. Accordingly, the network has learned the details of features representations for a large range of images. It has 4032 parameters to extract the feature vectors for each image and default input image size of NASNetLarge is 331x331 that are next processed by a Dense layer to generate a 256 elements representation of images. The output of second fully connected layer before the softmax layer is eliminated because we don't need to classify the image. This pre-trained model is applied to retrieve all of the feature vectors of images within the dataset and the transfer-values is save with the pickle file extension so that we can be reloaded faster for next evaluation [11][9].

EfficientNetB7: EfficientNetB7 is the combination of Mobile Nets and ResNet that are scaled up to improve the effectiveness of the model. To go even further, a new baseline network is designed by using neural architecture search and scale it up to achieve a family of models, called EfficientNets, which obtained much better accuracy and efficiency than previous ConvNets. In particular, EfficientNetB7 obtained the state-of-the-art 84.3% top-1 accuracy on ImageNet. Higher resolutions, such as 600x600, are also widely used in object detection ConvNets.

It has 2560 parameters to extract the feature vectors for each image and default input image size of EfficientNetB7 is 600x600 that are next processed by a Dense layer to generate a 256 elements representation of the images. The last layer of

EfficientNetB7 model is removed to avoid classification of the image. The output of second fully connected layer is taken as the initial state of Bi-LSTM in the decoder after it is downsized by the dense map layer [41].

In the image features extraction part, we compared these popular convolution networks architectures- VGG16, VGG19, InceptionV3, InceptionResNetV2, NASNetLarge and EfficientNetB7 as encoders for the same image captioning model in order to find out which method is the best at feature extraction to apply for caption generation. According to our experimental results, we found that EfficientNetB7 is significantly better performance than other feature extraction models without changing the decoder model, therefore, EfficientNetB7 is used as the encoder of the proposed model.

3.4 Recurrent Neural Network (RNN)

Recurrent neural network is the state-of-the-art algorithm for sequential data and that are used by Apple’s Siri and Google’s voice search. It is the first algorithm that remembers important things about the input they received, which allows them to be very precise in predicting what’s coming next, because of their internal memory. The reason for that, RNN makes it perfectly suited for machine learning problems that involve time series, speech, text, financial data, audio and video much more. However, RNN has the limitations like exploding gradients and vanishing gradients. Exploding gradients occurs when the algorithm assigns a stupidly high importance to the weights and vanishing gradients occur when the values of a gradient are too small and the model stops learning or takes way too long as a result. These major problems can be solved the concept of Long Short-Term Memory.

3.5 Long Short-Term Memory (LSTM)

Long Short-Term Memory network is considered as an extension of Recurrent Neural Network that is designed to handle temporal sequences and long-term dependencies that is more accurately than conventional RNNs. A typical LSTM network is comprised of different memory blocks called cells. The memory block is responsible to remember things and manipulation is done through three major mechanisms, called input gate, output gate and forget gate. The responsibility of input gate is to add the information to the cell states that is important and is not redundant. The output flow of cell activations into the rest of the network is controlled by the output gate. A forget gate is responsible to remove information from the cell state that is no longer required for the LSTM to understand things. Less importance information is removed via multiplication of a filter that is required to optimize the performance of the LSTM network [61]. LSTM network takes the inputs from various sources: current input x_t, the previous hidden state of all LSTM units h_{t-1} as well as previous memory cell state c_{t-1} at given time step t. At time step t, the updating of those gates for given inputs x_t, h_{t-1} and c_{t-1} as follows:

Input Gates: $i_t = \sigma(W_{xi}\, x_t + \mathrm{Whi\ ht\text{-}1} + \mathrm{bi})$ (3.1)

Forget Gates: $f_t = \sigma(W_{xf}\, x_t + W_{hf}\, h_{t-1} + b_f)$ (3.2)

Output Gates: $o_t = \sigma(W_{xo}\, x_t + W_{ho}\, h_{t-1} + b_o)$ (3.3)

$g_t = \emptyset(W_{xc}\, x_t + W_{hc}\, h_{t-1} + b_c)$ (3.4)

Cell States: $c_t = f_t\, \Theta\, c_{t-1} + i_t\, \mathit{\Theta}\, g_t$ (3.5)

Cell Output: $h_t = o_t\, \mathit{\Theta}\, \emptyset(c_t)$ (3.6)

Where W is the weight matrices learned from the network and b is bias vectors. σ is the sigmoid activation function, $\emptyset$ describes hyperbolic tangent and $\mathit{\Theta}$ means the products with gate values. The architecture of Long Short-Term Memory network is shown in Figure 3.6.

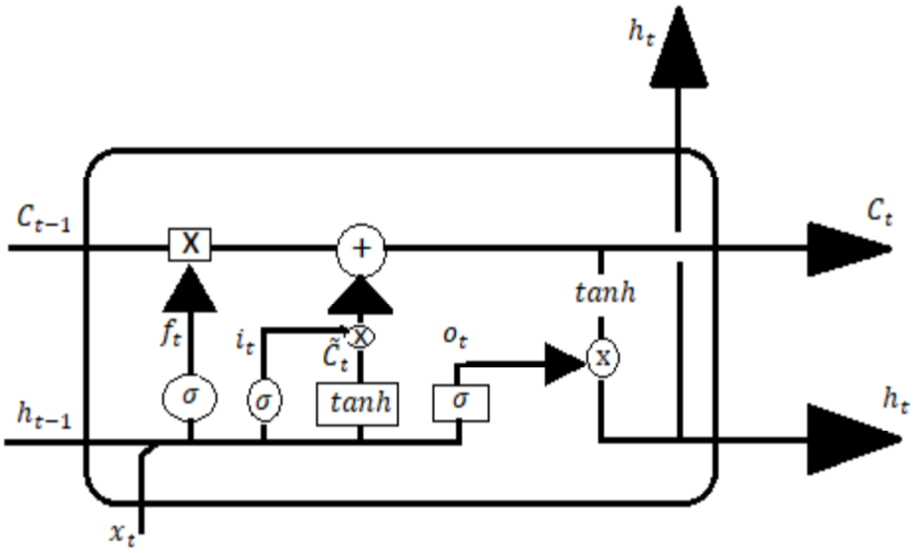


Figure 3.6 Architecture of Long Short-Term Memory

3.6 Bidirectional Long Short-Term Memory (Bi-LSTM)

Bi-LSTM network is an enhancing of Long Short-Term Memory (LSTM) neural network that is involved an input layer, two hidden layers and an output layer.

Input layer: Meanwhile the training module, the pre-tokenized words in our image captions corpus and their corresponding image features vectors from the former feature extraction model are taken by the input layer or word embedding layer as input. Individual word in the image descriptions sentences is transformed into one-hot encoded format in word embedding layer. Afterwards, the words and syllable vectors are the input parameters for the Bi-LSTM language model.

Hidden layer: The hidden layer contains two isolated forward and backward LSTM networks, associating to the similar output layer. Meanwhile the training module, both of the forward hidden sentences $\overrightarrow{h}_t = (\overrightarrow{h}_1, \overrightarrow{h}_2, ...\overrightarrow{h}_k)$ and backward hidden sentences $\overleftarrow{h}_t = (\overleftarrow{h}_1, \overleftarrow{h}_2, ...\overleftarrow{h}_k)$ are used for the similar sentences of word vectors coming from the previous input layer to assign the parameters of the system to correctly generate captions. The forward and backward hidden layer are evaluated as the following equations:

$$\overrightarrow{h}_t = \sigma\,(W_{\overrightarrow{h}}\;[\,\overrightarrow{h}_{t-1}, w_t\,] + b_{\overrightarrow{h}}\,) \tag{3.7}$$

$$\overleftarrow{h}_t = \sigma\,(W_{\overleftarrow{h}}[\,\overleftarrow{h}_{t-1}, w_t\,] + b_{\overleftarrow{h}}\,) \tag{3.8}$$

The integration of forward and backward layer created the final encoded hidden vector,

$$h_t = [\overrightarrow{h}_t, \overleftarrow{h}_t] \tag{3.9}$$

$$h_t = W_{\overrightarrow{h}}\overrightarrow{h}_t + W_{\overleftarrow{h}}\overleftarrow{h}_t + b_{\hat{y}} \tag{3.10}$$

Output layer: In this output layer or dense layer, softmax activation function is utilized to selects the suitable words according to the sequences of data from both hidden layers, which is powerful as concerns with categorizations and probability dispersion difficulty. The output of this function is in the form of one-hot encoded word which is then transformed backward to word form in a high-level representation for image descriptions [40]. Figure 3.7 presents the architecture of Bidirectional Long Short-Term Memory.

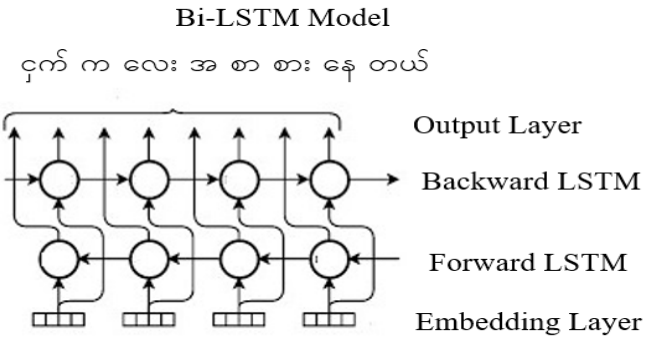


Figure 3.7 Architecture of Bidirectional Long Short-Term Memory

3.7 Gated Recurrent Unit (GRU)

Gated Recurrent Unit (GRU) has been proposed by Kyunghyun Cho et al. [5] to apply favorably for machine translation as well as sequence generation. GRU is an advanced version of the Recurrent Neural Network that is utilized to solve the “vanishing” or “exploding” gradient difficulty of typical RNN. GRU can also be considered as a variation on LSTM both of them are designed similarly and, in some cases, produce equally excellent results. GRU has two gates, a reset gate and update gate. The update gate helps the model to determine what information to throw away and what new information to add which is similar to forget and input gate of LSTM. Essentially, the reset gate is used to decide how much of the past information to forget.

In Figure 3.8, z_t is the update gate, r_t is the reset gate, $\tilde{h}_t$ is the candidate hidden state of the currently hidden node, h_t is the current hidden state, x_t is the input of the current neural network, and h_{t-1} is the hidden state at the previous moment. The detailed evaluation formulas are shown in the following equations:

$$z_t = \sigma(w_{zx}x_t + u_{zh}h_{t-1}) \qquad (3.11)$$

$$r_t = \sigma(w_{rx}x_t + u_{rh}h_{t-1}) \qquad (3.12)$$

$$\tilde{h}_t = \tan(w_{hx}x_t + r_t\,\Theta\,u_{hh}h_{t-1}) \qquad (3.13)$$

$$h_t = (1\text{-}z_t)\,\Theta\,\tilde{h} + z_t\,\Theta\,h_{t-1} \qquad (3.14)$$

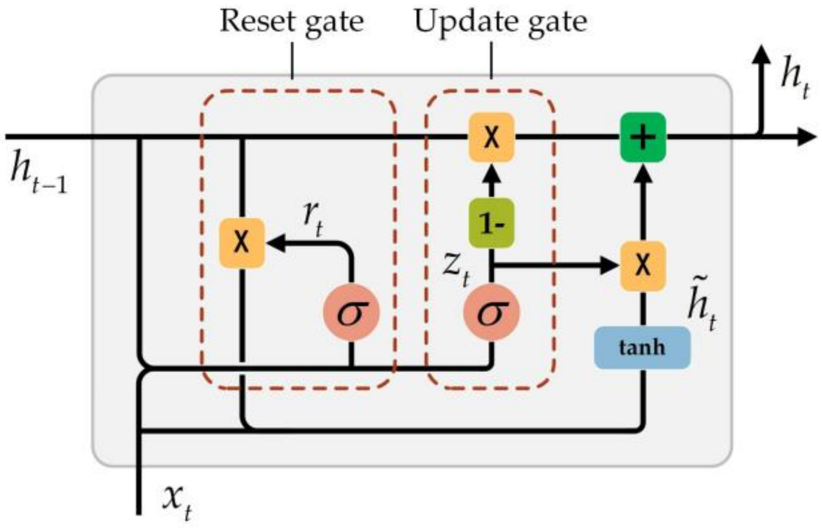


Figure 3.8 Gated Recurrent Unit (GRU) Neural Network Structure

Where σ is the *sigmoid* activation function, which ranges from 0 to 1, Θ is the Hadamard product of the matrix, w and u are the weight matrices that need to be learned, and z_t and r_t range from 0 to 1.

3.8 Bidirectional Gated Recurrent Unit (Bi-GRU)

A Bi-GRU neural network [45] is enhancing of a Gated Recurrent Unit (GRU) neural network that is contained with two diffident GRU structure -forward and backward. This two-layer structure gives the output layer with the complete contextual information of the input information at every moment. The fundamental idea of the Bi-GRU neural network is that the input sequence is fed through a forward neural network and a backward neural network, and then, the outputs of the two layers are connected in the same output layer. Figure 3.9 describes the two-layer Bi-GRU neural network utilized in this article in a time series expansion form.

In Figure 3.9, the individual layer of Bi-GRU neural network, the forward layer evaluates the output of the hidden layer at each time from forward to backward, and the backward layer evaluates the output of the hidden layer at each time from backward to forward. The output layer superimposes and normalizes the output results of the forward layer and backward layer at each moment:

$$\overrightarrow{h_t^1} = \mathrm{f}\,(w_{x\overrightarrow{h^1}} x_t + w_{\overrightarrow{h^1}\overrightarrow{h^1}} \overrightarrow{h_{t-1}^1} + b_{\overrightarrow{h^1}}) \tag{3.15}$$

$$\overleftarrow{h_t^1} = \text{f}\ (w_{x\overleftarrow{h^1}}\ x_t + w_{\overleftarrow{h^1}\overleftarrow{h^1}} \overleftarrow{h_{t+1}^1} + b_{\overleftarrow{h^1}}\ ) \qquad (3.16)$$

Where $\vec{h}^1{}_t$ € R^H are the output vectors of the forward hidden layer of the Bi-GRU neural network at time t, H is the number of units in the GRU cell, $\overleftarrow{h_t^1}$ € R^H is the output vectors of the backward hidden layer of the Bi-GRU neural networks at time t.

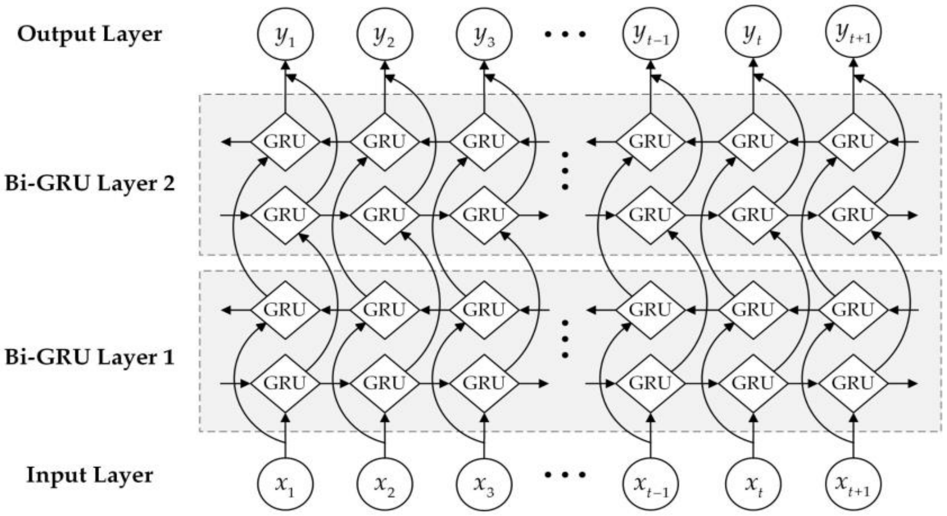


Figure 3.9 Bi-GRU Neural Network Structure

3.9 Word Embedding

The words require to be built meaningfully for comprehending of machine learning or deep learning algorithms. Hence, they must be represented statistical. Algorithms such as One Hot Encoding, Word2Vec, FastText, GloVe embedding methods are capable words to be represented statistical form as word embedding techniques utilized to handle such difficulty.

Word embedding can supply dense expression of words and identify their meaning [56]. Word Embedding is a language modeling technique that is used to map words to vectors of statistic values. Words or phrases can be stated in vector space with various dimensions (e.g., 50, 100, 200, 300). Word embedding can handle huge dimension difficulty and it describes dense vector expression and identify morphological association in the middle of words. If words have same meaning, word vectors will be near with each other [38]. The main objective of word embedding is to overcome infrequent and huge dimensional problems in Natural Language Processing (NLP).

3.9.1 One Hot Encoding

One hot encoding is one of the simplest approaches that is applied to express words numerically. In this method, a vector is constructed in the size of total values of individual words. The number of vectors is set such that the number of individual word association to its index is 1 and otherwise 0. This technique considers the fixed length with existence in image captions corpus in which all words are categorized in order and individual word has owned number.

Let the size of corpus length is equivalence with *n*. Hence, the word with the number *m* from the corpus is matched with a vector size *n* including zeros but it has precisely one member similar to one as a substitute *m*. This way of transforming words into a vector is fine for its clarity and understandable division of vector descriptions for various words. However, commonly corpus has a moderately big length, vector descriptions for words of such a corpus achieved utilizing the represented approach are too enormous in size during being slightly inadequate. Concurrent, enormous dimensionality, paucity and inadequate capture of semantic association between words makes this way not well appropriate for apply in the difficulty of image captioning.

3.9.2 Word2Vec

Another commonly applied word embedding methods is Word2vec. When the vector preparation process is done by deciding the target word occurs with more often, the entire corpus is scanned. In such manner, the semantic adjacency of the words is also mentioned with each other.

Unlike One Hot Encoding methods, Word2Vec method is performed by using unsupervised learning. Artificial neural networks are used to train the unlabelled for building the Word2Vec model that produces word vectors. Dissimilar with other techniques, the vector size is not plenty as well as the number of unique words in the corpus. The size of the vector can be changed based on the size of corpus and kind of the projects. This is especially good for huge amount of text data. For example, if we assume that huge corpus contains 200 000 unique words, by the time vector construction is done with one hot encoding, vector size 200 000 is built for individual word, with the value of only one element of 1 and otherwise 0. Nonetheless, by selecting the vector size 200 (it can be greater or smaller according to the user's selection) on the Word2Vec side, irrelevant big amount of vector size operations is avoided [72].

3.9.3 FastText

The process flow of FastText method is very analogous to Word2Vec, nevertheless the largest dissimilarity is that it also utilized N-grams characters as the minimum unit during training iterations [4]. For instance, let's say that the word vector "Orange" contains in the training dataset but we have to take the vector of the word "Oange" after the training is completed. When Word2Vec model is utilized for this condition, an error will be occurred due to the word "Oange" does not contain in the corpus, and any vectors will not be given. Nonetheless, FastText model can be utilized for this condition, both of the vector will be given for the word of "Orange" and similarity of its words. As discussed above, not only the word itself but also N-gram alternative are contained in training (Example 3-gram representation for the word "Orange" -> "Ora", "ran", "ang", "nge"). The greatest benefit of applying FastText is that it obtains the vectors for rare words or even words not found during training. It is one of the reasons to apply other alternatives in problem where words mistakes may occur [20][44][72].

3.9.4 GloVe (Global Vectors for Word Representation)

GloVe stands for Global Vectors for word representation. It is an unsupervised algorithm for generating word vectors by aggregating global word co-occurrence matrices from a given corpus. The major objective of GloVe word embedding is to obtain the association in the midst of words from Global word-to-word co-occurrence statistics. Co-occurrence matrix formulate how many of frequently words co-occur with one another in the training corpus [30]. Therefore, GloVe embedding method is more suitable for image captioning system. The ratio of probabilities between two pairs of words are computed by using the following equation 3.17.

$$F(w_i, w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}} \qquad (3.17)$$

3.10 Summary

This chapter has discussed various neural network models for both encoder and decoder. In image understanding part, various pre-trained feature extraction models of Convolutional Neural Network (CNN) such as VGG16, VGG19, InceptionV3, InceptionResNetV2, NASNetLarge and EfficientNetB7 are described to know which feature extraction model is the best at image captioning in Myanmar Language as encoder. In caption generation part, four different language generation models such as Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Bidirectional Gated Recurrent Unit (Bi-GRU), Bidirectional Long Short-Term Memory (Bi-LSTM) and word embedding vectors are reported in which language modelling is the best at image captioning system. The best result is obtained from Bi-LSTM with GloVe embedding vectors on both tasks word and syllable vectors as decoder of the image captioning system. Therefore, the model with the combination of EfficientNetB7 has been proposed as an encoder and Bi-LSTM with GloVe embedding vectors as a decoder of the system. It can be proved that the proposed model significantly better performance than other different neural networks models according to the various experimental results in next chapters.

CHAPTER 4
DATASET PREPARATION

This chapter has mainly discussed the creation of image captions corpus in Myanmar language, image preprocessing, text preprocessing and building of word and syllable GloVe embedding vectors.

4.1 Myanmar Image Captions Corpus Preparation

Due to the limited time, Flickr8k dataset [25] has been chosen and 2k images are selected from Flickr30k dataset [46], a total 10k images for our investigation that are generally utilized dataset for captions annotation in English language. This dataset contains complicated everyday activities with common objects in normally appearing contexts and can be downloaded freely. Therefore, it covered large accessible classification of images. There is no Myanmar image caption dataset applicable in the literature, the captions of this dataset have been manually annotated. It contains five annotated captions per image to generate image captioning dataset for Myanmar language. Myanmar image captions corpus is created in two distinct forms: 1) Automatic translation from English captions to Myanmar captions and 2) Direct image annotation with Myanmar language [42].

4.1.1 Translation English Captions to Myanmar Captions without Images

Before translation of English captions dataset is done, it is necessarily to clean the text in this dataset in sequence to decrease the amount of the vocabulary of words. Therefore, all words in the English captions dataset are converted to small letter, eliminate all punctuation, eliminate all words that are one character or less in length (e.g., ‘a’) and remove all words with number in them. Once cleaned, the size of the vocabulary can be summarized. If the size of vocabulary is small, the training time for smaller model will train faster. After the text preprocessing steps are done, all of the English image descriptions of the Flickr10k (Flickr8k+2k images from Flickr30) dataset are translated to Myanmar image captions without matching images by applying English to Myanmar Machine Translation. Attention based Neural Machine Translation model is used to transform from English to Myanmar language [66]. The training process is done on UCSY Corpus which contains 220k English Myanmar Parallel sentences. Since the domain of training data is common and most of the sentences contain about news and conversations, the translated sentences are not satisfying to apply directly in our image captioning system. Even though the translated captions sentences do not correct accurately, the translated Myanmar image captions sentences assist to decrease the manual annotation time.

4.1.2 Direct Building Myanmar Captions from Images

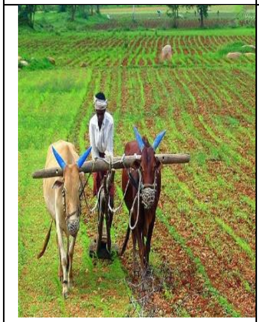
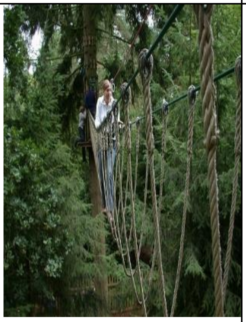
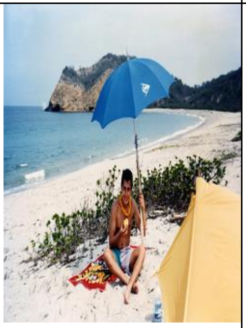
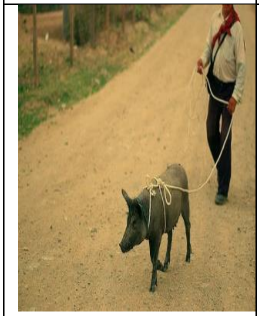
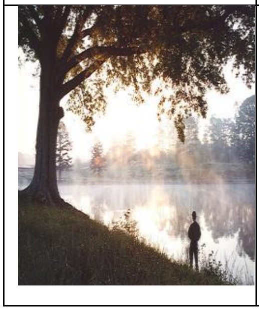
In this step, the translated Myanmar image captions are manually checked and corrected by looking one by one to match the descriptions with their correspondent images. The researcher has created own natural language definition as in rooted impression for the image without using English image captions. The total Myanmar captions for 10k images are 50460 sentences with a vocabulary size of 3,350. The maximum captions length is 24 in words level and 32 in syllable level.

Validation set contained 650 images to monitor the accuracy of trained model. The model accuracy improved and stabilized at the end of 15 epoch and then saved that model to obtain the best-learned model on the training dataset. Test set contained 650 images to measure the achievement of the learned model and its generation on a test set. The rest of the images 8792 are used as training. Table 4.1 shows the example images and five annotated captions for each image with ID number.

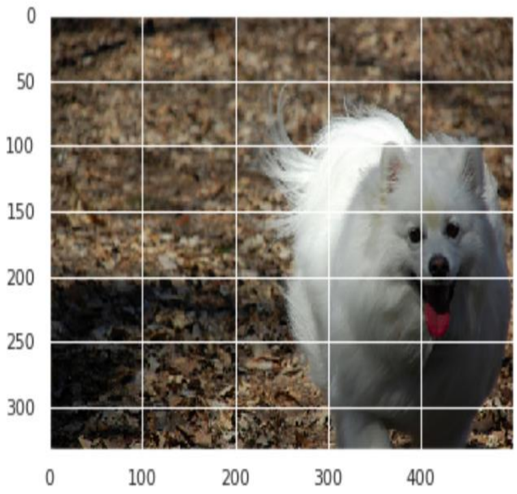
4.2 Image Preprocessing

Data pre-processing plays the vital role in every deep learning algorithm. In Myanmar IC system, two different types of data pre-processing are required such as image pre-processing and text pre-processing. In image pre-processing step, the input images need to resize based on the specified format, i.e., 331x331 for NASNetLarge, 224x224 for VGG16 and VGG19, 299x299 for InceptionV3 and InceptionResNetV2, and 600x600 for EfficientNetB7 to get the better quality and to avoid any numerical inconsistency during training and testing phases. The image pre-processing module can be offered by TensorFlow that can use freely for them to be read into memory, decoded as jpg, jpeg and resized utilizing pre-trained model. After the image pre-processing is done, the pre-processed image is provided as input to the image features extraction

Table 4.1 Example Images and Structure of Creation Corpus

Images	ID No	Annotated Captions with Myanmar Language
	136310496	လူ တစ်ယောက် က နွား နှစ်ကောင် နဲ့ ယာ ထွန် နေတယ် လူ တစ်ယောက် က ယာခင်း ထဲမှာ နွား နှစ်ကောင် နဲ့ ယာ ထွန် နေတယ် လူ တစ်ယောက် က လယ်ကွင်း ထဲမှာ အလုပ်လုပ် နေတယ် လူ တစ်ယောက် က လယ်ကွင်း ထဲမှာ နွား နှစ်ကောင် နှင့် ယာ ထွန် နေတယ် အမျိုးသား က ယာခင်း ထဲမှာ နွား နှစ်ကောင် နဲ့ ယာ ထွန် နေတယ်
	136644885	အမျိုးသမီး က ကြိုးတံတား ပေါ်မှာ လမ်းလျှောက် နေတယ် ကြိုးတံတား ပေါ်မှာ အမျိုးသမီး တစ်ယောက် လမ်းလျှောက် နေတယ် လူ တစ်ယောက် က ကြိုးတံတား ပေါ်မှာ လမ်းလျှောက် နေတယ် လူ များ က ကြိုးတံတား ပေါ်မှာ လမ်းလျှောက် နေ ကြ တယ် အမျိုးသမီး တစ်ယောက် က ကြိုးတံတား ပေါ်မှာ လမ်းလျှောက် နေတယ်
	136645716	လူ တစ်ယောက် က ပင်လယ်ကမ်းခြေ မှာ ထီး ဆောင်း ပြီး ထိုင် နေတယ် လူ တစ်ယောက် က ထီး ဆောင်း ပြီး ထိုင် နေတယ် အမျိုးသား က ကမ်းခြေ မှာ ထီး ဆောင်း ပြီး ထိုင် နေတယ် အမျိုးသား တစ်ယောက် က ကမ်းခြေ မှာ ထီး ဆောင်း ပြီး ထိုင် နေတယ် ထီး ဆောင်း ထား သော အမျိုးသား က ကမ်းခြေ မှာ ထိုင် နေတယ်
	13327175	လူ တစ်ယောက် က ဝက် ကို ကြိုး နဲ့ ဆွဲ ထား တယ် လူ တစ်ယောက် က ဝက် ကို ကြိုး ဆွဲပြီး လမ်းလျှောက် နေတယ် လူ တစ်ယောက် က ဝက် ကို ကြိုး ချည် ထား တယ် အမျိုးသား က ဝက် ကို ကြိုး နဲ့ ဆွဲပြီး လမ်းလျှောက် နေတယ် အမျိုးသား တစ်ယောက် က ဝက် ကို ကြိုး နဲ့ ဆွဲပြီး လမ်းလျှောက် နေတယ်
	136644886	လူ တစ်ယောက် သစ်ပင် အောက် မှာ ငါးမျှား နေတယ် ရေကန် ဘေး သစ်ပင် အောက်မှာ လူ တစ်ယောက် ငါးမျှား နေတယ် အမျိုးသား က သစ်ပင် အောက်မှာ ငါးမျှား နေတယ် သစ်ပင် အောက်မှာ လူ တစ်ယောက် ငါးမျှား နေတယ် အမျိုးသား တစ်ယောက် သစ်ပင် အောက်မှာ ငါးမျှား နေတယ်

Shape of the image: (333, 500, 3)



Shape After resize: (224, 224, 3)

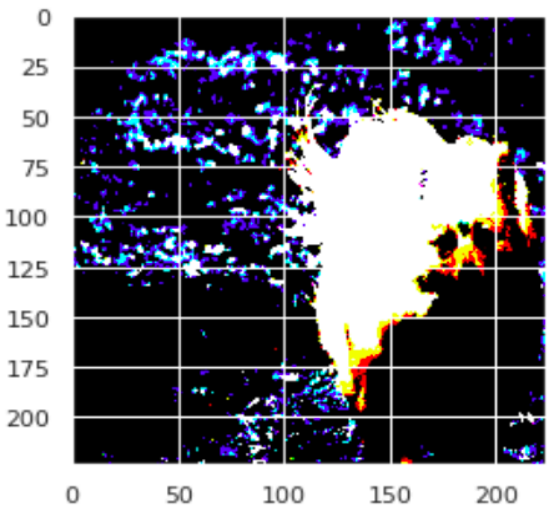


Figure 4.1 Original Image

Figure 4.2 After Resize Image with (224, 224, 3)

model of CNN and then the extracted features are passed as input to the Bi-LSTM unit. For example, Figure 4.1 is the original image which has the shape (333, 500, 3) and this image is resized with the dimension of 224x224 as show in Figure 4.2.

4.3 Text Preprocessing

Basically, Myanmar language is a syllabic language. Myanmar script has 33 consonants, 8 vowels, 2 diacritics, 11 medials, a vowel killer or ASAT, 10 digits and 2 punctuation marks. Each consonant has default vowel sound and itself works as a syllable. Words in Myanmar language are composition of one or more syllables and a syllable may also contain one or more characters. A word “Myanmar” “မြန်မာ” is composed of two syllables ‘မြန်’ and ‘မာ’. A character can stand as a syllable itself or a syllable in Myanmar language may be made up of one or several characters. For example, in the first syllable, “မြန်”, the sub syllabic elements are Consonant(မ) + Medial (ြ) + Consonant (န)and Vowel Killer (်) constitute to form the syllable ‘မြန်’. In this work, two distinct kinds of segmentation such as word and syllable segmentation are used in text preprocessing step and then built the own GloVe embedding vectors for both segmentations to compared with one-hot encoding vectors.

4.3.1 Word Segmentation

The sentences manually created in Myanmar image descriptions corpus are not tokenized accurately to get the quality improvement for Myanmar image captioning. Word segmentation is the essential preprocessing step to generate image captions with Myanmar language due to Myanmar text particularly does not contain white space between words although space sometimes exists between phrases. Therefore, Myanmar word segmentation process is done by using UCSYNLP word segmenter [62] in this work. After segmenting all of Myanmar image captions sentences in the corpus, the '|' symbols from the segmented sentences are eliminated and recovered with space.

136310496	လူ တစ်ယောက် က ယာခင်း ထဲမှာ နွား နှစ်ကောင် နဲ့ ထယ် ထိုး နေတယ်
136310496	လူ တစ်ယောက် က နွား နှစ်ကောင် နဲ့ ယာ ထွန် နေတယ်
136310496	လူ တစ်ယောက် က လယ်ကွင်း ထဲမှာ နွား နှစ်ကောင် နှင့် အလုပ်လုပ် နေတယ်
136310496	လူ တစ်ယောက် က လယ်ကွင်း ထဲမှာ နွား နှစ်ကောင် နှင့် ယာ ထွန် နေတယ်
136310496	အမျိုးသား က ယာခင်း ထဲမှာ နွား နှစ်ကောင် နဲ့ ယာ ထွန် နေတယ်
13327175	လူ တစ်ယောက် က ဝက် ကို ကြိုး နဲ့ ဆွဲ ထား တယ်
13327175	လူ တစ်ယောက် က ဝက် ကို ကြိုး နဲ့ ဆွဲပြီး လမ်းလျှောက် နေတယ်
13327175	လူ တစ်ယောက် က ဝက် ကို ကြိုး ချည် ထား တယ်
13327175	အမျိုးသား က ဝက် ကို ကြိုး နဲ့ ဆွဲပြီး လမ်းလျှောက် နေတယ်
13327175	အမျိုးသား က ဝက် တစ်ကောင် ကို ကြိုး နဲ့ ဆွဲပြီး လမ်းလျှောက် နေတယ်
136644885	အမျိုးသမီး က ကြိုးတံတား ပေါ်မှာ လမ်းလျှောက် နေတယ်
136644885	ကြိုးတံတား ပေါ်မှာ အမျိုးသမီး တစ်ယောက် လမ်းလျှောက် နေတယ်
136644885	လူ တစ်ယောက် က ကြိုးတံတား ပေါ်မှာ လမ်းလျှောက် နေ တယ်
136644885	လူ များ က ကြိုးတံတား ပေါ်မှာ လမ်းလျှောက် နေ ကြ တယ်
136644885	အမျိုးသမီး တစ်ယောက် က ကြိုးတံတား ပေါ်မှာ လမ်းလျှောက် နေတယ်

Figure 4.3 Structure of Myanmar Image Captions Corpus for Word Segmentation

The sample structure of Myanmar image captions corpus for word segmentation is shown in Figure 4.3. The structure of word and syllable segmentation process for a

Myanmar image caption sentence in the corpus is illustrated as shown below and the meaning is “Bird is eating the food”:

Before Segmentation: ငှက်ကလေးအစာစားနေတယ်

After Word Segmentation: ငှက်ကလေး အစာ စား နေတယ်

After Syllable Segmentation: ငှက် က လေး အ စာ စား နေ တယ်

4.3.2 Syllable Segmentation

In Natural Language Processing, syllable segmentation [71] is very important preprocessing step. In Myanmar, words can be made up of one or more syllables, and syllable consists of one or more character.

136310496	လူ တစ် ယောက် က ယာ ခင်း ထဲ မှာ နွား နှစ် ကောင် နဲ့ ထယ် ထိုး နေ တယ်
136310496	လူ တစ် ယောက် က နွား နှစ် ကောင် နဲ့ ယာ ထွန် နေ တယ်
136310496	လူ တစ် ယောက် က လယ် ကွင်း ထဲ မှာ နွား နှစ် ကောင် နှင့် အ လုပ် လုပ် နေ တယ်
136310496	လူ တစ် ယောက် က လယ် ကွင်း ထဲ မှာ နွား နှစ် ကောင် နှင့် ယာ ထွန် နေ တယ်
136310496	အ မျိုး သား က ယာ ခင်း ထဲ မှာ နွား နှစ် ကောင် နဲ့ ယာ ထွန် နေ တယ်
13327175	လူ တစ် ယောက် က ဝက် ကို ကြိုး နဲ့ ဆွဲ ထား တယ်
13327175	လူ တစ် ယောက် က ဝက် ကို ကြိုး နဲ့ ဆွဲ ပြီး လမ်း လျှောက် နေ တယ်
13327175	လူ တစ် ယောက် က ဝက် ကို ကြိုး ချည် ထား တယ်
13327175	အ မျိုး သား က ဝက် ကို ကြိုး နဲ့ ဆွဲ ပြီး လမ်း လျှောက် နေ တယ်
13327175	အ မျိုး သား က ဝက် တစ် ကောင် ကို ကြိုး နဲ့ ဆွဲ ပြီး လမ်း လျှောက် နေ တယ်
136644885	အ မျိုး သ မီး က ကြိုး တံ တား ပေါ် မှာ လမ်း လျှောက် နေ တယ်
136644885	ကြိုး တံ တား ပေါ် မှာ အ မျိုး သ မီး တစ် ယောက် လမ်း လျှောက် နေ တယ်
136644885	လူ တစ် ယောက် က ကြိုး တံ တား ပေါ် မှာ လမ်း လျှောက် နေ တယ်
136644885	လူ များ က ကြိုး တံ တား ပေါ် မှာ လမ်း လျှောက် နေ ကြ တယ်
136644885	အ မျိုး သ မီး တစ် ယောက် က ကြိုး တံ တား ပေါ် မှာ လမ်း လျှောက် နေ တယ်

Figure 4.4 Structure of Myanmar Image Captions Corpus for Syllable Segmentation

Regular Expression (RE) based Myanmar syllable segmentation algorithm "sylbreak" is applied to token from the Myanmar image descriptions sentences into syllable level. When all of Myanmar image descriptions have been segmented into syllable sentences, the "|" symbol from the segmented sentences is eliminated and recovered with white space that are leading the trim process.

In this work, two distinct kinds of corpus such as word segmentation corpus and syllable segmentation corpus are created for training to compare which segmentation ways is more effective on Myanmar image captioning system. After that, our own GloVe vectors for both word and syllable segmented corpus are built using GloVe v.1.2. Figure 4.4 shows the sample structure of Myanmar image captions corpus for syllable segmentation.

4.4 Word Embedding

Word embedding is basically a form of word representation that transforms human understanding language to form vectors of each word. In word embedding, each word in our corpus is described as real-valued vectors in a specified vector space. Individual word is assigned to one vector and the vector values are accomplished in a way that resembles a neural network. Word2Vec and GloVe are the most commonly use techniques to learn word vectors by utilizing shallow neural network. In this work, GloVe is used in word and syllable embedding phase based on Bi-LSTM neural network after pre-processing step.

GloVe: GloVe (Global Vectors for Word Representation) is an approach to achieve vector representations utilizing unsupervised learning methods as stated by matrix factorization techniques on the word-context matrix [30]. A huge matrix of co-occurrence information is constructed and count the number of each “word” (rows), and how to see frequently this word in some “context” (columns) in a huge corpus. Using the GloVe library, each word and syllable in the monolingual Myanmar corpus is transformed to a 300-dimensional vector.

4.4.1 Construction of Word and Syllable GloVe Embedding Vectors

Recently, word embedding model has been used in text to speech [6], text summarization [67] with their own corpus for Myanmar language. In [48-21], only two set of pre-trained word vectors models can be accessed publicly for Myanmar language. The pre-trained word embedding models cannot be utilized directly due to the words are not corresponded with our image captioning system. Therefore, our own word and syllable vectors are built with standard Unicode encoding for more coverage and much better performance of Myanmar image captioning system. While building the syllable and word GloVe vectors for our own image captions corpus, there are some issues in counting vocabulary and find out unknown terms for each word in the image captions corpus because of the inadequate amount of data in building GloVe embedding vectors. Hence, the text data is gathered to create a vast monolingual Myanmar corpus for the purpose of creation better efficiency word embedding model with broad coverage. Myanmar news corpus [67] (around 10k sentences) is utilized by gathering several Myanmar News websites which involves different types of news such as World news, business, health, politics, Entertainment, education and sport.

In monolingual Myanmar corpus, the sentences are combined from our Myanmar image captions corpus (50460 sentences) and the sentences from Myanmar News corpus. Finally, it consists of the total 60460 sentences. After gathering the text data, the further step is constructing GloVe embedding model for both word and syllable Myanmar image captions corpus that are mapped with vectors of real values using the GloVe v.1.2.

4.4.2 GloVe Embedding Vector Setting

After collecting the text data, GloVe embedding vectors are created for both tasks using the GloVe v.1.2. Individual word and syllable are presented as real-valued vectors with various dimensionality (50, 100, 200, 300). The training iteration was repeated 15 times with negative sampling. The length of training context is set to 8 for all models. The window size is set to 15 and the minimum vocabulary count is denoted as 5. Embedding layer contains the number of vocabularies, dimension of each word vector and maximum length of input vector. After doing experiment with various dimensions (50, 100, 200, 300), 300-dimension is selected in our experiment for better performance.

Some example vectors values of 300 dimensions are shown for syllable vector “မျိုး” and word vector “အမျိုးသမီး” in the following:

- ❖ Syllable Vector မျိုး ➔ 1.217643 0.760280 -0.657219 -1.344656 1.557007 0.722238 0.253888 1.079134 -0.059733 0.468703 -0.379546 -1.748968……

- ❖ Word Vector အမျိုးသမီး ➔ -0.128325 -0.828009 1.758781 -0.788344 0.298282 0.200081 -0.018604 1.738469 -0.316292 -1.658677 0.996643 0.449300 -0.600717 -0.554563….

4.5 Summary

In this chapter, the way of how to build the Myanmar image captions corpus and image preprocessing step have been presented. The nature of Myanmar language is also presented in brief. Text preprocessing, building of word and syllable vectors have been presented in this chapter. Word and syllable embedding features from our trained GloVe vectors will be used in Myanmar image captioning system to enhance the language modelling in next chapter.

CHAPTER 5

VGG16 AND LONG SHORT-TERM MEMORY FOR MYANMAR IMAGE DESCRIPTION

Image Captioning is a challenging task in the area of artificial intelligence problem where a textual description with different languages is produced for a given input photo. It needs both methods from computer vision to comprehend the content of the images and a language model from the domain of natural language processing to predict the interpreting feature vectors of the image into words in the correct order. Currently, deep learning techniques obtained the state-of-the-art results in the domain of image captioning difficulty. What is most impressive about these methods is a specific end-to-end model can be specified to generate a description, given a photo, instead of needing complicated data modification or a pipeline of especially considered models.

This chapter describes the comparison of two different feature extraction models: VGG16 and VGG19 of CNN as encoder and Long Short-Term Memory as decoder. These combination models' results are presented in this chapter.

5.1 VGG16 and LSTM Based Image Captioning System

The Architecture of VGG16 and LSTM based image captioning system is shown in Figure 5.1. This architecture especially consists of two different modules. The first one is image understanding module using the pre-trained feature extraction model of CNN and the second one is text understanding module using Long Short-Term Memory. In image understanding module, Convolutional Neural Network is widely utilized because image classification problems can be solved successfully with high accuracy. Two different types of models such as VGG16 and VGG19 for feature extraction of images dataset are compared and tested. The two different features extractions models have the different capabilities, and the input image size of both models are 224× 224× 3 and the convolutional feature size of VGG is 4096. The last layer of the pre-trained feature extraction models is removed because the feature vectors are needed instead of classification the images and then the output is applied from second last layer as feature parameters for each image. Individual image has 4096 parameters that are extracted to process by a Dense layer to generate a 256-element depiction of an image. Figure 5.2 describes the model summary of VGG16 with LSTM.

In text understanding module, LSTM can keep information in memory for a long time and extract series of information through time. The text understanding part predicts the appropriate words or phrases according to the word embedding vector of former module. The language generation model is trained to produce individual word in the image captions after it has found both image feature vectors and all of the prior words in our corpus. For all given sentences in Myanmar image captions corpus, two extra symbols like “startseq” and “endseq” are added to know exactly for begin word and end word of image captions. Whenever endseq is observed, generating caption is stopped that is denoted end of the sentence. In this model, the input sequences length is defined 21 words which are passed into an Embedding layer and then utilize a mask to omit filled values and followed by an LSTM layer with 256 memory units. Both of the models produced a 256-element vector and regularization of 50% dropout is utilized to decrease over fitting during the training.

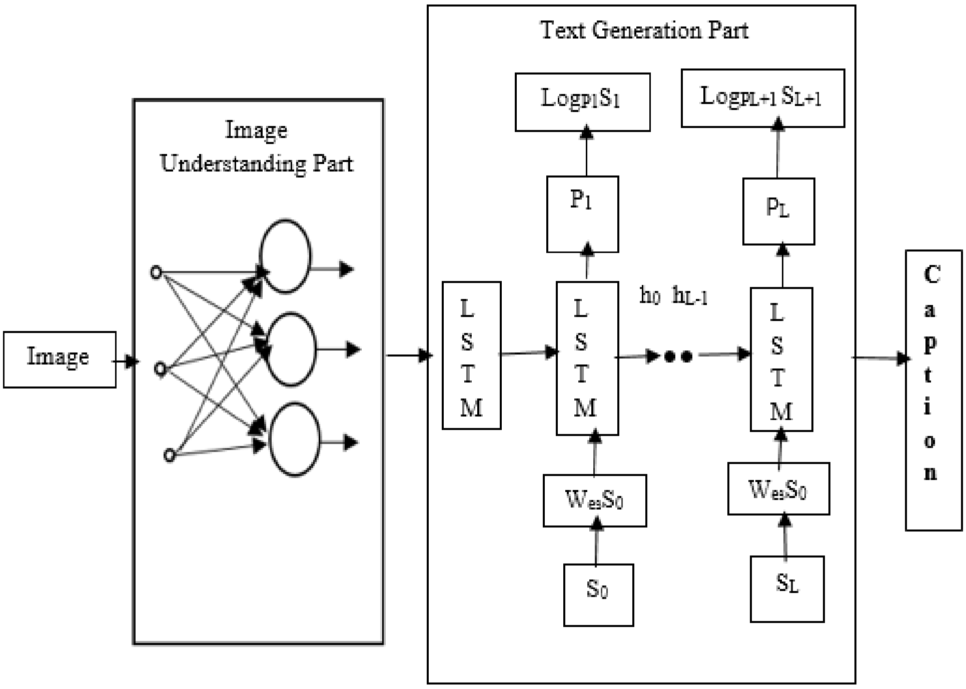


Figure 5.1 Architecture of VGG16 and LSTM Based Image Captioning

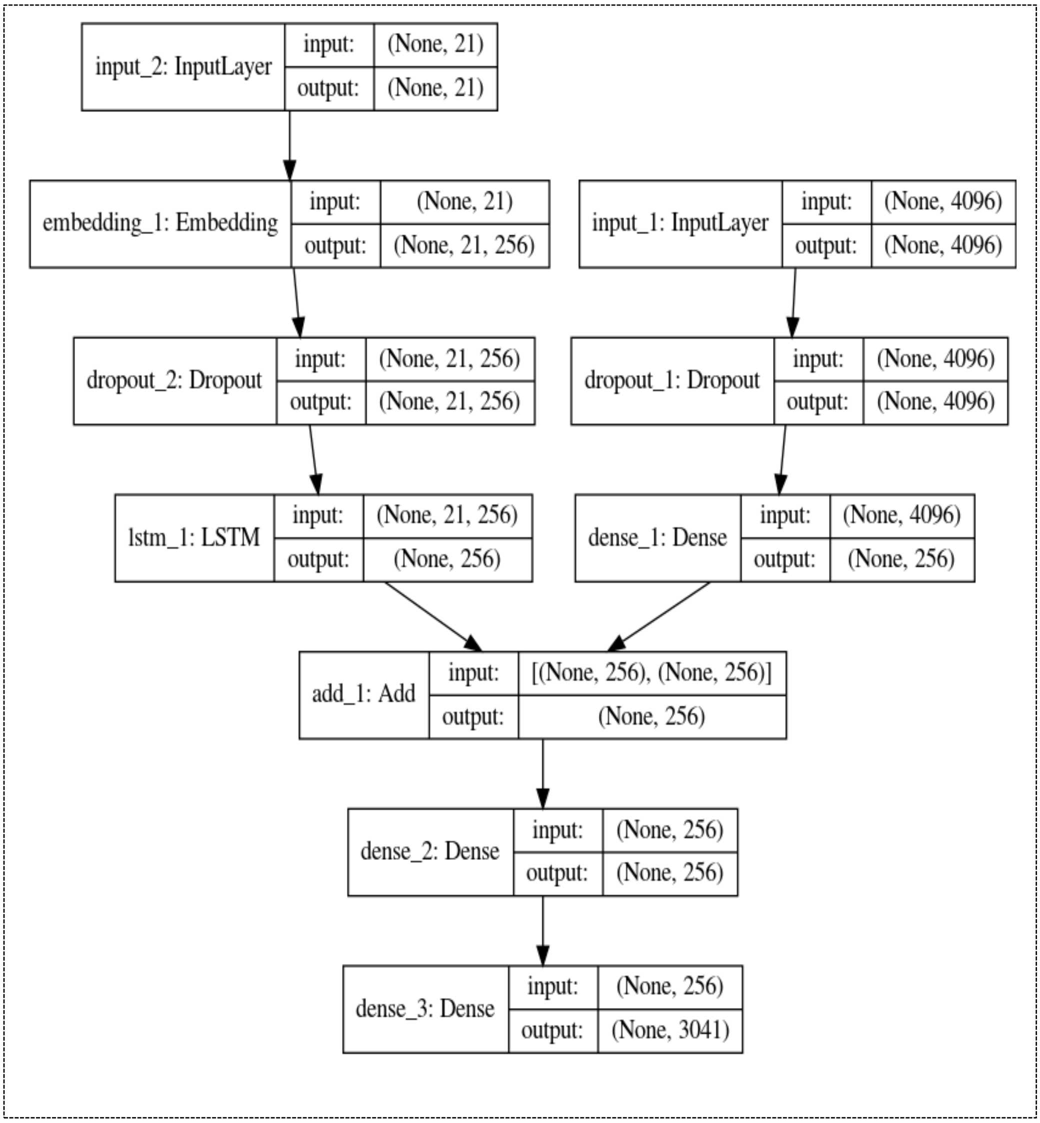


Figure 5.2 Model Summary of VGG16 with LSTM

In decoding part, the model mixed the vectors from both previous feature vectors and one-hot vectors by utilization an extension action and then passed to a Dense 256 neuron layer to produce the softmax classification for the next word in the captions over the whole corpus. Loss value is calculated for both models by using the following equation:

$$L\ (I, S) = - \sum_{t=1}^{N} \log p_t(S_t) \qquad (5.1)$$

Where I is given input image and S is machine predicted caption, N is the length of predicted description. p_t and S_t are probability and generate word at time t respectively. While the training iteration is doing, we have attempted to decrease the loss value.

This paper has done experiment only 15000 Myanmar image captions sentences for 3k images that are taken from the Flickr8k dataset which contains five annotated captions for each image. The structure of building Myanmar image captions corpus is described in Chapter 4. Although the size of the corpus is small, the acceptable performance can be obtained.

5.2 Experimental Setups

The two different models for creating and training deep neural networks are conducted on K80 GPU machine. Keras API library is used with TensorFlow backend. When the given amount of training data is trained, both of the models are set at the 10 epochs. Both of the models are stable after the 5th iterations; therefore, the best learned model and loss values are saved for each training times. In 10 folds cross validation setting, the minimal loss value is 2.097 on trained data and the minimal validation loss is 2.513 on development dataset by applying VGG16 with LSTM. Next, the minimal loss is 2.114 for the trained dataset and the validation loss is 2.513 for the development dataset by applying VGG19 with LSTM. As can be seen that the minimal lost value for validation of both models is the absolutely identical. Figure 5.3 and 5.4 display the alternative loss values of training and validation by utilizing two distinct models.

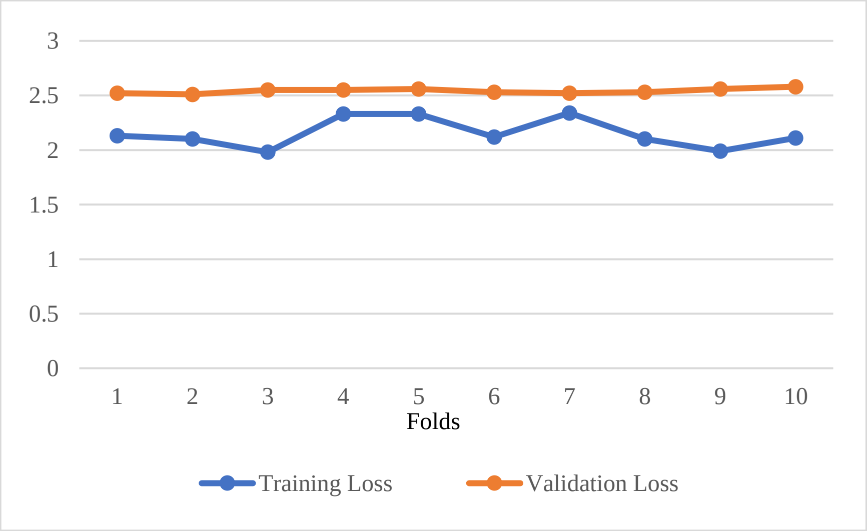


Figure 5.3 Alternative Loss Values of Training and Validation in Each Fold Utilizing VGG16 with LSTM

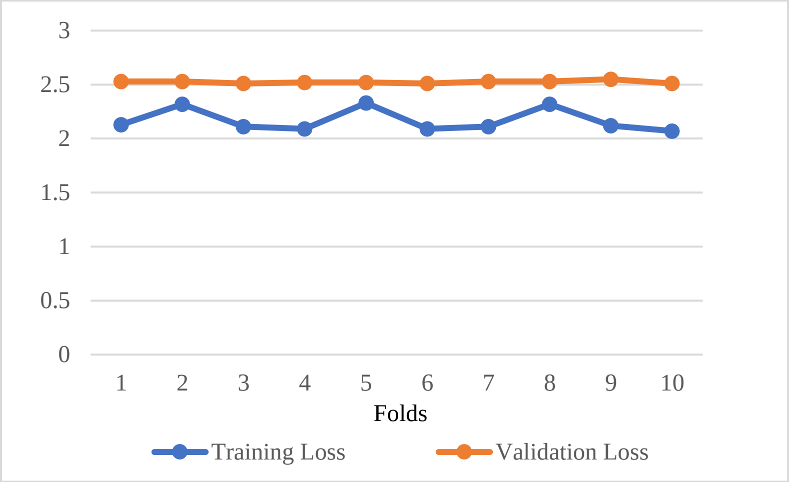


Figure 5.4 Alternative Loss Values of Training and Validation in Each Fold Utilizing VGG19 with LSTM

5.3 10-Fold Cross Validation

In this chapter, the performance of predictive models is evaluated using BLEU and 10-fold cross validation setting that are randomly partitioned the original dataset into 10 equal subsets. 9 sets are used for training dataset to train the model and the rest 1 set is used to evaluate for testing dataset. The cross-validation process is iterated 10 times (the fold) and individual subsets is utilized accurately once for the validation data. The average evaluation results for all the iterations are calculated to generate a single evaluation result. In Table 5.1 and Table 5.2, the BLEU scores of individual iterations with distinct testing datasets are shown. The average BLEU scores for the comparison of VGG16 with LSTM model and VGG19 with LSTM model are shown in Figure 5.5.

As can be seen in Table 5.1, we achieved the average BLEU-1 score of 64.14%, 48.58% of BLEU-2, 39.86% of BLEU-3 and 24.38% of BLEU-4 score using VGG16 with LSTM model. In Table 5.2, we obtained the average BLEU-1 score of 63.51%, 48.12% of BLEU-2, 39.55% of BLEU-3 and 24.18% of BLEU-4 score using VGG19 with LSTM model. According to these experimental results, VGG16 and VGG19 achieved the relatively similar results and do not give any qualitative difference.

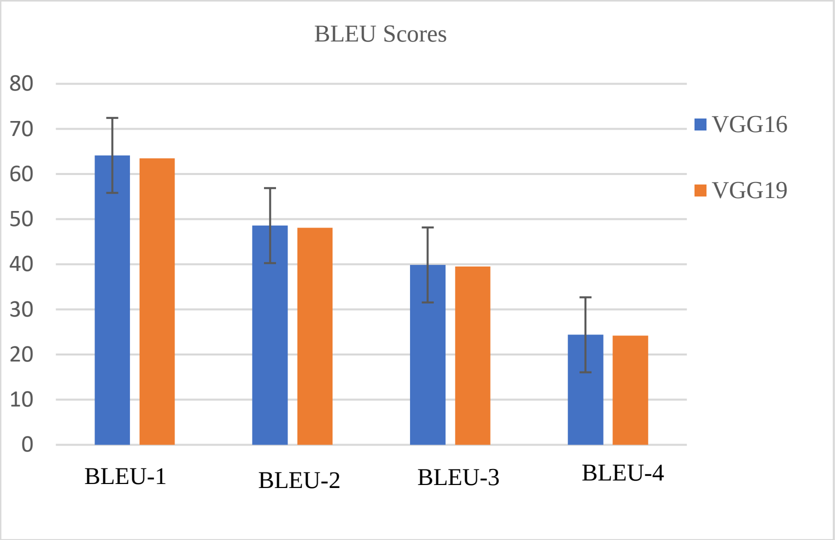


Figure 5.5 Comparison of VGG16 and VGG19

Table 5.1 10-Fold Cross Validation for VGG16 with LSTM [P1]

Training Times	**BLEU-1**	**BLEU-2**	**BLEU-3**	**BLEU-4**
Fold 1	61.5	47.6	40.3	25.9
Fold 2	62.4	46.4	37.1	22.2
Fold 3	64.8	49.3	40.5	23.8
Fold 4	65.6	49.8	41.2	26.0
Fold 5	66.2	50.8	41.6	25.2
Fold 6	64.3	48.7	40.5	25.6
Fold 7	62.5	46.4	36.9	21.4
Fold 8	63.1	46.8	37.9	22.8
Fold 9	64.9	50.2	42.4	26.8
Fold 10	66.1	49.8	40.2	24.1
Total	641.4	485.8	398.6	243.8
Average	**64.14**	**48.58**	**39.86**	**24.38**

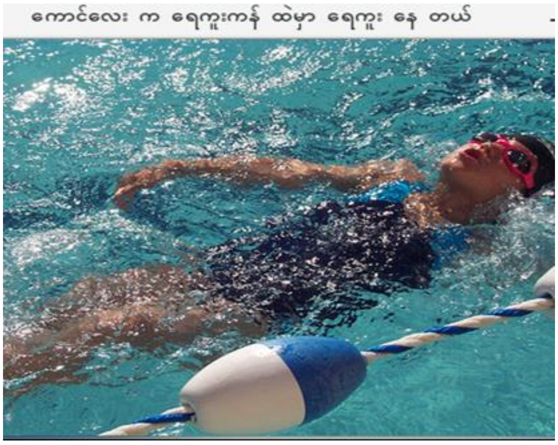
Table 5.2 10-Fold Cross Validation for VGG19 with LSTM [P1]

Training Times	BLEU-1	BLEU-2	BLEU-3	BLEU-4
Fold 1	65.1	50.6	42.7	28.0
Fold 2	60.6	44.7	36.4	20.9
Fold 3	58.8	44.1	36.3	21.2
Fold 4	65.6	49.3	39.2	22.7
Fold 5	67.0	51.1	41.5	24.9
Fold 6	65.9	50.0	41.0	25.4
Fold 7	54.6	40.7	34.2	21.3
Fold 8	65.5	49.2	40.1	24.6
Fold 9	65.4	51.4	43.6	28.2
Fold 10	66.6	50.1	40.5	24.6
Total	635.1	481.2	395.5	241.8
Average	**63.51**	**48.12**	**39.55**	**24.18**

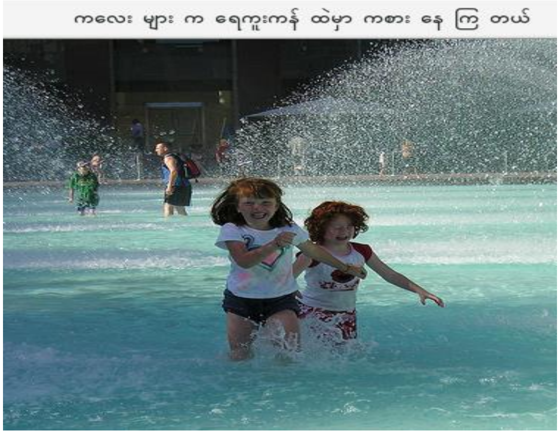
5.4 Experiment Results and Analysis

The generated captions for both VGG16 and VGG19 pre-trained feature extraction models are nearly similar and is not provided any qualitative difference. Accordingly, the generated results from VGG16 with LSTM is mainly focused in this chapter. In Figure 5.6 (a), the model is able to generate the major features and actions of the images accurately such as "ကောင်လေး" ("the boy") "ရေကူးကန် ထဲမှာ" ("in the swimming pool") "ရေကူး နေတယ်" ("is swimming") and the relationship between images and captions are also described accurately. In Figure 5.6 (b), the model can capture the count of the objects, place, and activities like "ကလေး များ" ("Children"), "ရေကူးကန်" ("in the lake") and "ကစား နေ ကြ တယ်" ("are playing"). In Figure 5.6 (c), the model can predict the count of the objects like "ခွေး နှစ် ကောင်" ("two dogs") and also identify the place correctly "မြက်ခင်းစိမ်း ထဲမှာ" ("in the green grass"), "ကစား နေ ကြ တယ်" ("are playing"). If we look at Figure 5.6 (a), Figure 5.6 (b) and Figure 5.6

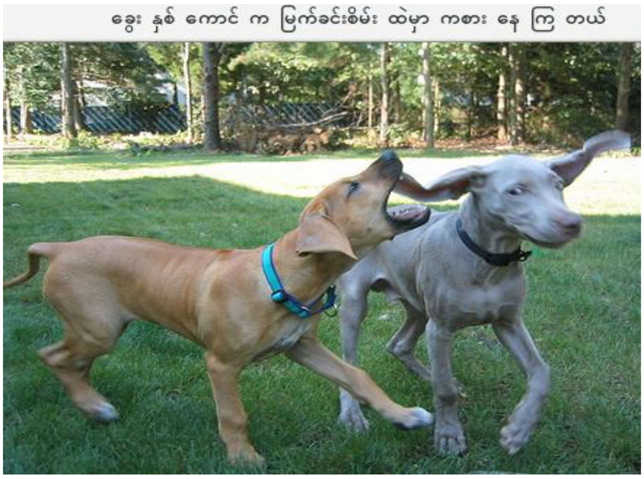
(c), the model can capture the significant features, number of objects, activities of the images and also predicts grammatical correct sentences. In spite of that, we can be seen in Figure 5.6 (d) for open test image, the model can identify the main feature which is “လူ တစ်ယောက်” (“a person”) and “ထိုင် နေတယ်” (“sitting”), nonetheless, it is not able to capture the object completely and misidentify the object like နံရံ (wall) instead of ခုံတန်းရှည် (bench). In conclusion, it is the restriction of this model and we will be focusing the necessary for future work regarding the model. In Chapter 7, the proposed models can resolve this problem, and can generate precisely the association between images and its captions even for open test images. All of the Figures 5.6 (a), 5.6 (b), 5.6 (c) and 5.6 (d) are generated with captions automatically in Myanmar Language without any human intervention.



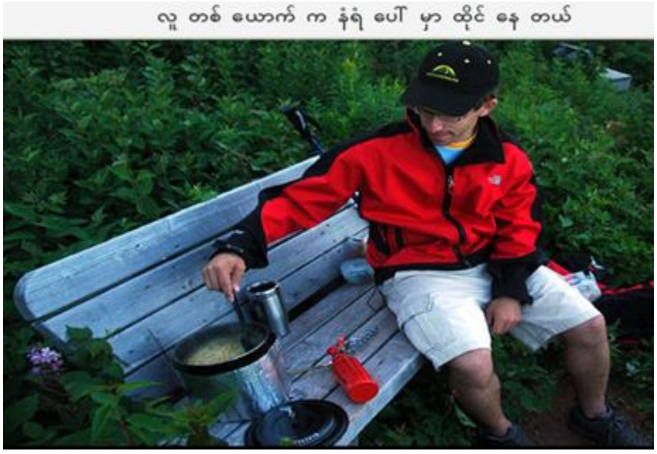
(a) In English: The boy is swimming in the swimming pool



(b) In English: Children are playing in the swimming pool



(c) In English: Two dogs are playing in the green grass



(d) In English: A person is sitting on the wall

Figure 5.6 Generated Captions Using VGG16 with LSTM

5.5 Summary

In this chapter, Visual Geometry Group (VGG) OxfordNet 16 layers and 19 layers of Convolutional Neural Network are compared as encoder to know which is the best at feature extraction of images, and single hidden layer Long Short-Term Memory model is applied as decoder for Myanmar automatic image caption generation. The system performance is measured on the image captions corpus (around 15k Myanmar sentences for 3k images) using 10-fold cross validation and Bilingual Evaluation Understudy Score (BLEU). According to the experiment results, it is found that the performance of VGG16 and VGG19 are not different significantly. Moreover, the combination of pre-trained VGG16 and Long Short-Term Memory for Myanmar image captioning system can give acceptable performance on our tiny corpus. It is believed that the system performance will be given more acceptable if the size of corpus is enlarged as we proved in next chapters. After doing various experiments on different size of the corpus, it is found that the size of training data is very important in machine learning algorithms.

CHAPTER 6

INCEPTIONRESNETV2 AND RECURRENT NEURAL NETWORK FOR MYANMAR IMAGE DESCRIPTION

Image captioning is one of the most challenging tasks in Artificial Intelligence which combines computer vision as well as natural language processing (NLP). Computer vision plays an important role to extract key information in images and natural language processing generates the corresponding descriptions. The aim of image captioning is to generate a suitable sentence of the content of given image. Nowadays, social media networks such as Facebook and Twitter can directly generate captions from images. The exact information can be gained from these photos where are the places: (e.g., beach, cafe, and road), what are the people wearing and importantly what are they doing there [68]. It is very useful and has a great impact on visually impaired person who can only feel the world by touch. The automatic generation of descriptions from the images with proper sentences are very difficult and challenging task for machine.

The task of automatic caption generation for a given image is significantly harder than object recognition and image classification. The caption generation of an image involves not only the objects in the image, but also relation between these objects with their attributes and activities shown in images [12]. In this chapter, InceptionV3 and InceptionResNetV2 are used the feature extraction models as encoder and Gated Recurrent Unit and Long Short-Term Memory are used the language models as decoder. The experiment results are reported by comparing the VGG16 with LSTM model described in Chapter 5.

6.1 InceptionResNetV2 and RNN Based Myanmar Image Description

The basic framework of Myanmar image captioning system is depicted in Figure 6.1. Firstly, a given image is preprocessed with the dimensions of 299 × 299 and is provided as input to the pre-trained image features extraction model of CNN and then the extracted features are passed as input to the LSTM unit. Secondly, Myanmar captions are segmented word by word from the sentences and then feed forward to LSTM which finally generates caption with Myanmar Language. Word segmentation process is presented in Chapter 4. Long Short-Term Memory (LSTM) acts as the decoder and extracted features are fed to it and then used the common representation of all gathered information based on it provided sentence.

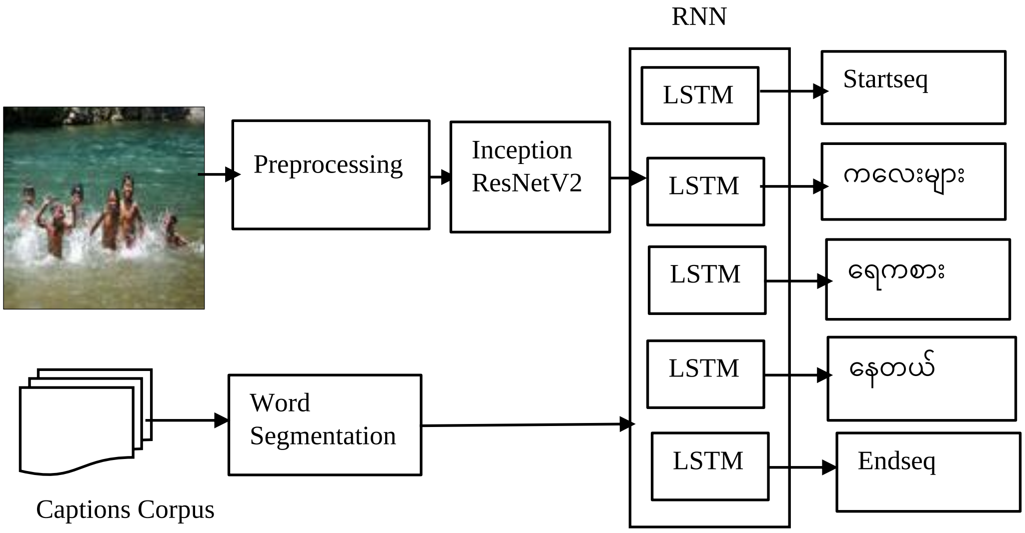


Figure 6.1 The Process Flow of Myanmar Image Captioning Using InceptionResNetV2 and RNN

6.2 Results and Discussion

The comparison of four different encoder and decoder architectures for Myanmar automatic image captioning are discussed in this chapter. The following Table 6.1 is a listing of the models that we experimented on Myanmar image captions corpus which contains 40460 sentences for 8k images that are taken from Flickr8 dataset. In this chapter, only word segmentation is considered in text preprocessing step.

The four models are data-driven and it is trained end to end. Manually annotated Myanmar image captions corpus is used for training purpose. The models are trained

Table 6.1 List of Models

IV3-GRU	InceptionV3 is utilized as an encoder and GRU is utilized as decoder
IV3-LSTM	InceptionV3 is utilized as an encoder and LSTM is utilized as decoder
IRNV2-GRU	InceptionResNetV2 is utilized as an encoder and GRU is utilized as decoder
IRNV2-LSTM	InceptionResNetV2 is utilized as an encoder and LSTM is utilized as decoder

on Tesla K80 GPU and implemented with Python by using Keras library, which is run on Tensorflow as backend. We have evaluated BLEU scores of our models with 10-fold cross validation. We attained the best result from a combination of InceptionResNetV2 as an encoder and LSTM as a decoder.

Although LSTM and GRU [1]provide similar results for some applications, LSTM more accurate on dataset using longer sequences. We should use LSTM if sequence is large and accuracy is very critical whereas GRU should be used for less memory consumption and faster operation because it uses less training parameters and less memory than LSTM [22].

6.3 10-Fold Cross Validation

In this work, the performance of predictive models is evaluated using 10-fold cross validation setting that are randomly partitioned the original dataset into 10 equal sets. 9 sets are used for training dataset to train the model and 1 set is used for testing dataset to evaluate it. Sparse softmax cross entropy is applied to measure the loss value of the trained model which evaluate the probability error to reduce for image captions generation tasks. The adaptive moment estimation (Adam) optimizer is utilized for better performance results instead of RMSprop optimizer. The models differ in the method used to extract features from the images and to generate captions with Myanmar Language. Figure 6.2, Figure 6.3, Figure 6.4 and Figure 6.5 show the loss from the variation of training error on the training dataset, and also the validation error on the development dataset in 10-fold using different models. The smallest loss values in 10-fold cross validation setting using different models are given in Table 6.2.

Table 6.2 The Smallest Loss Value in 10-Fold Using Different Models

Models	Folds	Training-Loss	Validation-Loss
IV3-GRU	Fold 8	2.18	1.96
IV3-LSTM	Fold 7	2.09	1.97
IRNV2-GRU	Fold 6	2.01	1.69
IRNV2-LSTM	Fold 6	1.93	1.62

[1] https://www.quora.com/Whats-the-difference-between-LSTM-and-GRU

It is visible from the Table 6.2 that IV3-GRU model achieved the smallest loss value 1.96 in fold 8 and 1.97 in fold 7 using IV3-LSTM model. These two models are not significantly different in validation error. IRNV2-LSTM performs much better in validation accuracy with the smallest loss values is 1.62 in fold 6 compared to other three models. On the other hand, InceptionResNetV2 obtains the smaller loss value than InceptionV3 for both language models (shown as IRNV2-GRU and IRNV2-LSTM). According to these experiments results, we can be said that InceptionResNetV2 feature extraction model is more powerful than InceptionV3.

Table 6.3 Evaluation Result of 10-Fold Cross Validation with IV3-GRU

Fold-N	**BLEU-1**	**BLEU-2**	**BLEU-3**	**BLEU-4**
Fold 1	62.71	46.76	37.8	24.36
Fold 2	60.97	44.93	35.75	23.11
Fold 3	58.6	42.39	33.65	20.75
Fold 4	63.46	47.24	36.97	22.35
Fold 5	61.63	44.68	34.69	20.9
Fold 6	62.12	45.3	35.63	22.27
Fold 7	59.03	43.58	34.97	22.39
Fold 8	60.85	44.5	35.63	22.42
Fold 9	61.44	44.91	35.6	22.38
Fold 10	60.45	43.04	33.52	20.43
Total	611.26	447.33	354.21	221.36
Average	**61.13**	**44.73**	**35.42**	**22.13**

Evaluation results of 10-fold cross validation for each different models are presented in Table 6.3, Table 6.4, Table 6.5 and Table 6.6. In each model, it is necessary to do 10 training times with different testing dataset and take the average score to estimate the single prediction as shown in bold in each table. Therefore, 40 training times have been done for four different models in this chapter. The 10-fold cross validation should be used in small dataset because it takes for a long time.

Table 6.4 Evaluation Result of 10-Fold Cross Validation with IV3-LSTM

Fold-N	**BLEU-1**	**BLEU-2**	**BLEU-3**	**BLEU-4**
Fold 1	61.99	45.65	36.92	23.69
Fold 2	61.16	45.32	36.49	23.46
Fold 3	63.21	47.35	38.29	24.84
Fold 4	68.01	51.5	40.28	24.64
Fold 5	63.22	46.47	36.46	22.09
Fold 6	62.29	45.74	35.18	21.43
Fold 7	61.86	46.58	37.82	25.17
Fold 8	61.83	46.3	37.91	24.86
Fold 9	60.34	44.19	35.27	22.21
Fold 10	61.25	43.77	33.12	19.3
Total	625.16	462.87	367.74	231.69
Average	**62.51**	**46.29**	**36.77**	**23.17**

Table 6.5 Evaluation Result of 10-Fold Cross Validation with IRNV2-GRU

Fold-N	**BLEU-1**	**BLEU-2**	**BLEU-3**	**BLEU-4**
Fold 1	63.12	47.2	38.61	25.47
Fold 2	62.51	46.64	37.7	24.88
Fold 3	60.95	44.89	35.2	21.78
Fold 4	64.97	48.79	37.93	22.85
Fold 5	65.85	49.48	38.72	23.71
Fold 6	64.96	49.22	39.72	26.02
Fold 7	64.69	48.65	38.72	25.11
Fold 8	63.23	47.17	37.21	23.45
Fold 9	63.3	47.17	37.21	23.45
Fold 10	63.81	46.86	36.78	23.61
Total	637.39	476.07	377.8	240.33
Average	**63.74**	**47.61**	**37.78**	**24.03**

Table 6.6 Evaluation Result of 10-Fold Cross Validation with IRNV2-LSTM

Fold-N	**BLEU-1**	**BLEU-2**	**BLEU-3**	**BLEU-4**
Fold 1	63.12	47.2	38.61	25.47
Fold 2	64.77	49.52	40.81	27.38
Fold 3	64.55	49.62	40.18	27.53
Fold 4	68.07	51.5	40.93	26.14
Fold 5	64.9	48.27	38.11	24.05
Fold 6	63.57	47.04	37.57	24.09
Fold 7	61.61	45.3	36.5	23.64
Fold 8	63.45	46.43	36.01	22.01
Fold 9	68.1	51.5	40.93	26.13
Fold 10	64.27	47.4	37.81	24.28
Total	646.41	483.78	387.46	250.72
Average	**64.64**	**48.38**	**38.75**	**25.07**

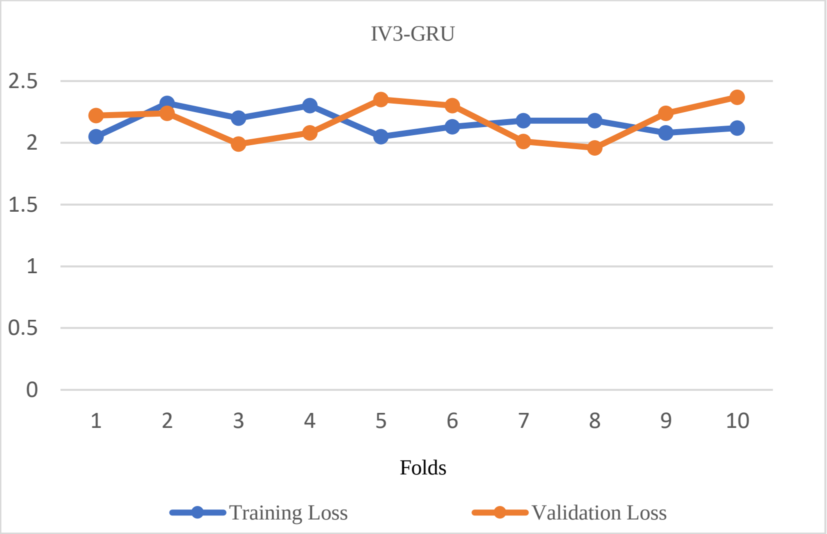


Figure 6.2 Alternative Loss Values of Training and Validation in Each Fold Using IV3-GRU

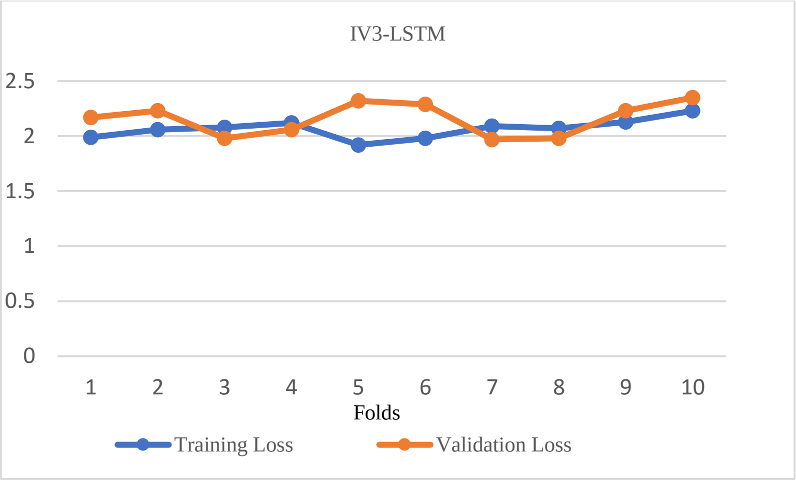


Figure 6.3 Alternative Loss Values of Training and Validation in Each Fold Using IV3-LSTM

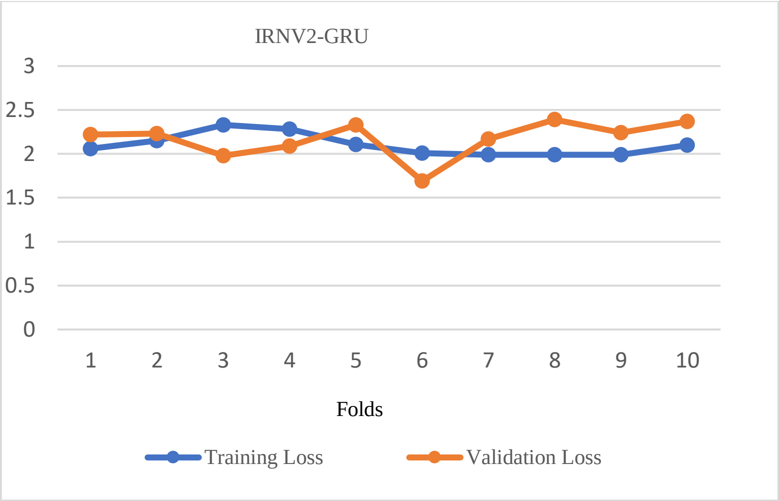


Figure 6.4 Alternative Loss Values of Training and Validation in Each Fold using IRNV2-GRU

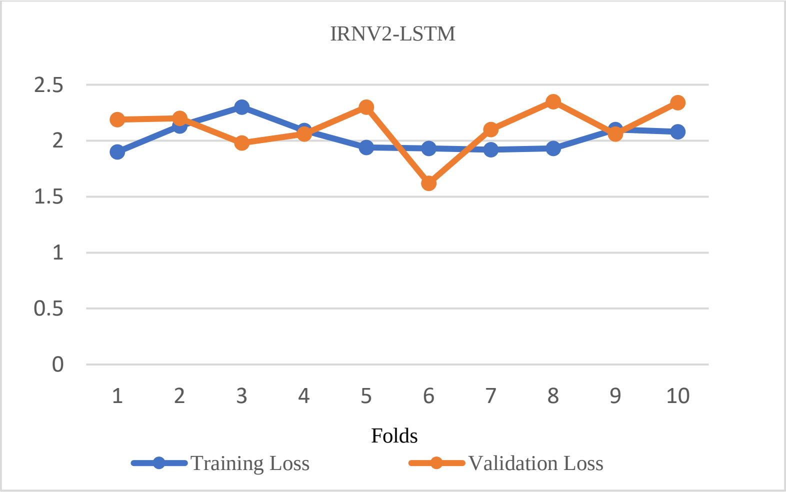


Figure 6.5 Alternative Loss Values of Training and Validation in Each Fold Using IRNV2-LSTM

Table 6.7 Average BLEU Scores of Different Models

Models	**BLEU-1(%)**	**BLEU-2(%)**	**BLEU-3(%)**	**BLEU-4(%)**
Baseline [P1]	64.14	48.58	39.86	24.38
IV3-GRU	61.13	44.73	35.42	22.13
IV3-LSTM	62.51	46.29	36.77	23.17
IRNV2-GRU	63.74	47.61	37.78	24.03
IRNV2-LSTM	**64.64**	**48.38**	**38.75**	**25.07**

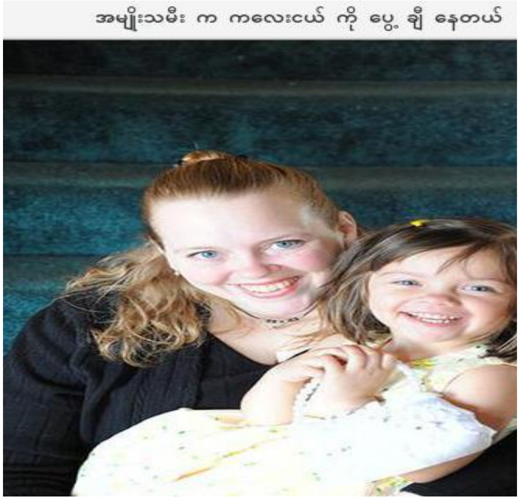
Table 6.7 shows the average BLEU scores of four different models and baseline model. InceptionResNetV2 with LSTM achieved the BLEU-1 score of 64.64%, BLEU-2 score of 48.38%, BLEU-3 score of 38.75% and BLEU-4 score of 25.07, which are slightly superior than other three different models. Nonetheless, this model gaps only 0.69-point of BLEU-4 score compared with baseline model as shown in Table 6.7. The BLEU scores results are taken as percentage (e.g., 0.2507 * 100 = 25.07%).

6.4 Experiment Results

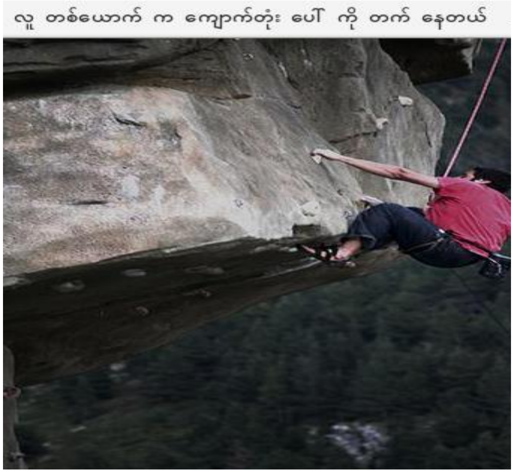
This section focusses on generated results from InceptionResNetV2 with LSTM. This model precisely generates the main features and relationship between these features within images. In Figure 6.6, results of generated Myanmar descriptions are given. In Figure 6.6(a), the generated sentence is "အမျိုးသမီး က ကလေးငယ် ကို ပွေ့ချီ ထားတယ်" ("Woman is holding the toddler") the model can accurately identify the gender and age like "အမျိုးသမီး" ("Woman"), "ကလေးငယ်" ("toddler") and also actions like "ပွေ့ချီ နေတယ်" ("is holding"). In Figure 6.6 (b), the generated sentence is "လူ တစ်ယောက် က ကျောက်တုံး ပေါ် ကို တက် နေတယ်" ("A man is climbing on the rock") and in Figure 6.6 (c), the generated caption is "လူ တစ်ယောက် က တော ထဲမှာ စက်ဘီး စီး နေတယ်" ("A man is riding the bike through the forest"). In Figure 6.6 (d), the generated description is "ခွေး က တန်း ကို ခုန် ကျော် နေတယ်" ("The dog is jumping over hurdle").

If we observe at Figure 6.6 (a), (b), (c), (d), the model is able to speculate the most of objects that appear in the pictures and also generates grammatically correct sentences.

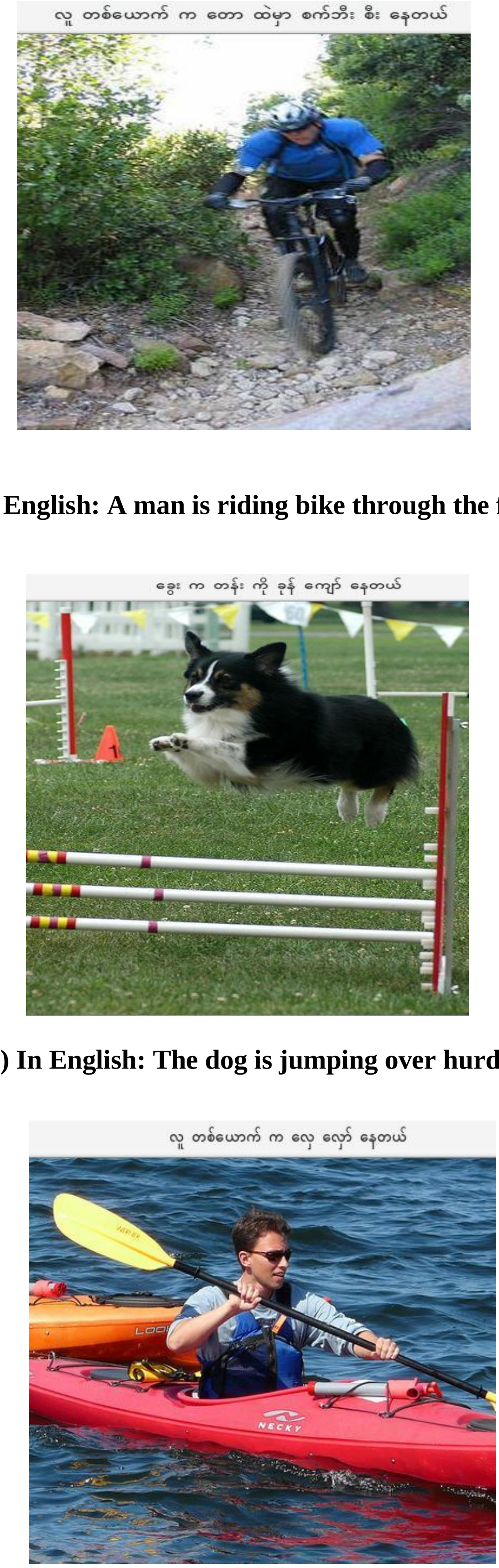
As can be seen these generated descriptions are much more similar with the content of input image nevertheless there is a weakness to progress in the identification generation of ***color*** descriptions. For example, in Figure 6.6 (e), the model generates a caption like "လူ တစ်ယောက် က လှေ လှော် နေတယ်" ("A man is paddling the boat"), but there is actually "အနီရောင် လှေ" ("***red boat***"). It fails to depict the minor features like "color". Nevertheless, our proposed model can identify the color of the objects as we confirmed that in next Chapter 7. All of the Figures 6.6 (a), 6.6 (b), 6.6 (c), 6.6 (d) and 6.6 (e) are automatically produced descriptions with Myanmar Language by using InceptionResNetV2 with LSTM without any human intervention.



(a) In English: Woman is holding the toddler



(b) In English: A man is climbing on the rock


လူ တစ်ယောက် က တော ထဲမှာ စက်ဘီး စီး နေတယ်
ခွေး က တန်း ကို ခုန် ကျော် နေတယ်
လူ တစ်ယောက် က လှေ လှော် နေတယ်

(c) In English: A man is riding bike through the forest

(d) In English: The dog is jumping over hurdle

(e) In English: A man is paddling the boat

Figure 6.6 Some Example of Generated Captions by IRNV2-LSTM

6.5 Summary

In this chapter, the attempt of various combinations of feature extractions models are done for encoder and language generation models for decoder such as InceptionV3 with GRU, InceptionV3 with LSTM, InceptionResNetV2 with GRU and InceptionResNetV2 with LSTM. According to the experimental results, the best result is obtained from a combination of InceptionResNetV2 as an encoder and LSTM as a decoder. It is efficient and robust system than other three different models, and can be produced the captions more specific and related to the content of that image. However, InceptionResNetV2 with LSTM is not significantly different compared with baseline model on Myanmar image captions corpus (around 40460 sentences for 8k images).

CHAPTER 7

EFFICIENTNETB7 AND BIDIRECTIONAL LSTM FOR MYANMAR IMAGE CAPTIONING SYSTM

An image contains a lot of information. The extraction of information from an image is one of the challenging tasks in the field of Computer Vision and Natural Language Processing, two of the main domains in Artificial Intelligence. However, most research in this area generated image captions in English while there are a lot of different languages exist in the world. With their distinctive languages, there is a necessity of particular research to generate captions in those isolated language. As far as being aware and up to our knowledge, there is no image captioning system for Myanmar language. Therefore, neural network-based Myanmar image captioning system is proposed by comparing different encoding and decoding techniques.

In this chapter, Myanmar image caption generation system is examined by dividing into two parts: 1) image features extraction acts as encoder and 2) generating a caption with Myanmar language as decoder. In the image features extraction part, three popular pre-trained Convolution Neural Networks - VGG16 [52], NASNetLarge [8] and EfficientNetB7 are compared as encoders for the same image captioning model in order to find out which method is the best at feature extraction to apply for caption generation. According to the experimental results, we found that EfficientNetB7 is significantly better performance than for both VGG16 and NASNetLarge models without changing the decoder model, therefore, EfficientNetB7 is used as the encoder of the proposed model. In caption generation part, four different language generation models such as Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), Bidirectional Gated Recurrent Unit (Bi-GRU) and Bidirectional Long Short-Term Memory (Bi-LSTM) with and without GloVe embedding vector are investigated to apply which language modelling is the best at image captioning system. The best result is obtained from a combination of EfficientNetB7 as an encoder and Bi-LSTM with GloVe vectors as a decoder. Moreover, the effectiveness of applying GloVe vectors features is investigated on EfficientNetB7 with Bi-LSTM based Myanmar image captioning system in this chapter. To the best of our knowledge, this is the first work to apply Bi-LSTM with GloVe vectors features in Myanmar image captioning system.

7.1 EfficientNetB7 and Bidirectional LSTM for Myanmar Image Captioning

The system flow diagram of proposed Myanmar image captioning system is depicted in Figure 7.2. Data pre-processing plays the vital role in every deep learning algorithm. In training module of Myanmar image captions generation system, two different types of data pre-processing are needed such as image pre-processing and text pre-processing because of the better quality of the system. In image pre-processing step, the input images need to resize for the specified format, i.e. (331,331) for NASNetLarge, (224,224) for VGG16 and (600,600) for EfficientNetB7 to get the better quality and to avoid any numerical inconsistency during training and testing phases. TensorFlow offers the image pre-processing libraries files that can be accessed easily for them to be read into memory, decoded as jpg, jpeg and resized utilizing various pre-trained feature extractions models. After the image pre-processing is done, the pre-processed images are provided as input to the image features extraction model of CNN and then the extracted features are feed forward as input to the Bidirectional Long Short-Term Memory language model.

In text pre-processing step, two distinct types of segmentations such as word segmentation [69] and syllable[2] segmentation were presented in detail in Chapter 4 which are used in training to compare which segmentation level affects in Myanmar image captioning system. The structure of word segmentation for a Myanmar image description sentence in the proposed corpus is depicted as follows in Figure 7.1:

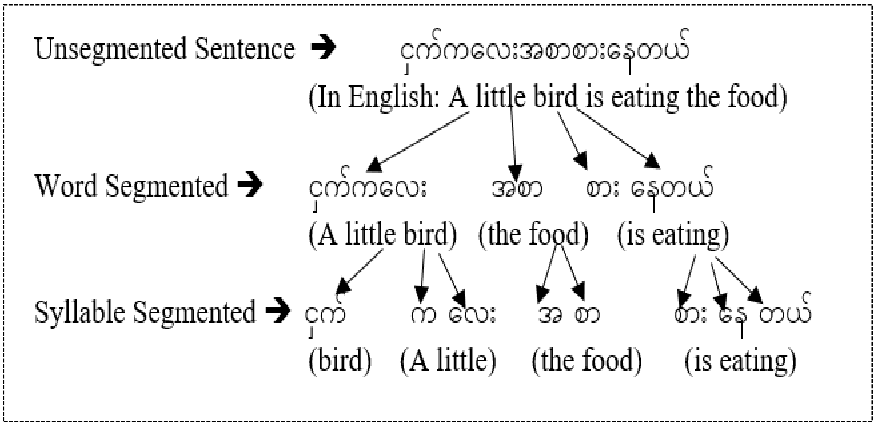

Unsegmented Sentence ➔ ငှက်ကလေးအစာစားနေတယ်
(In English: A little bird is eating the food)
Word Segmented ➔ ငှက်ကလေး အစာ စား နေတယ်
(A little bird) (the food) (is eating)
Syllable Segmented ➔ ငှက် က လေး အ စာ စား နေ တယ်
(bird) (A little) (the food) (is eating)

Figure 7.1 The Structure of Myanmar Word Segmentation

[2] https://github.com/ye-kyaw-thu/sylbreak

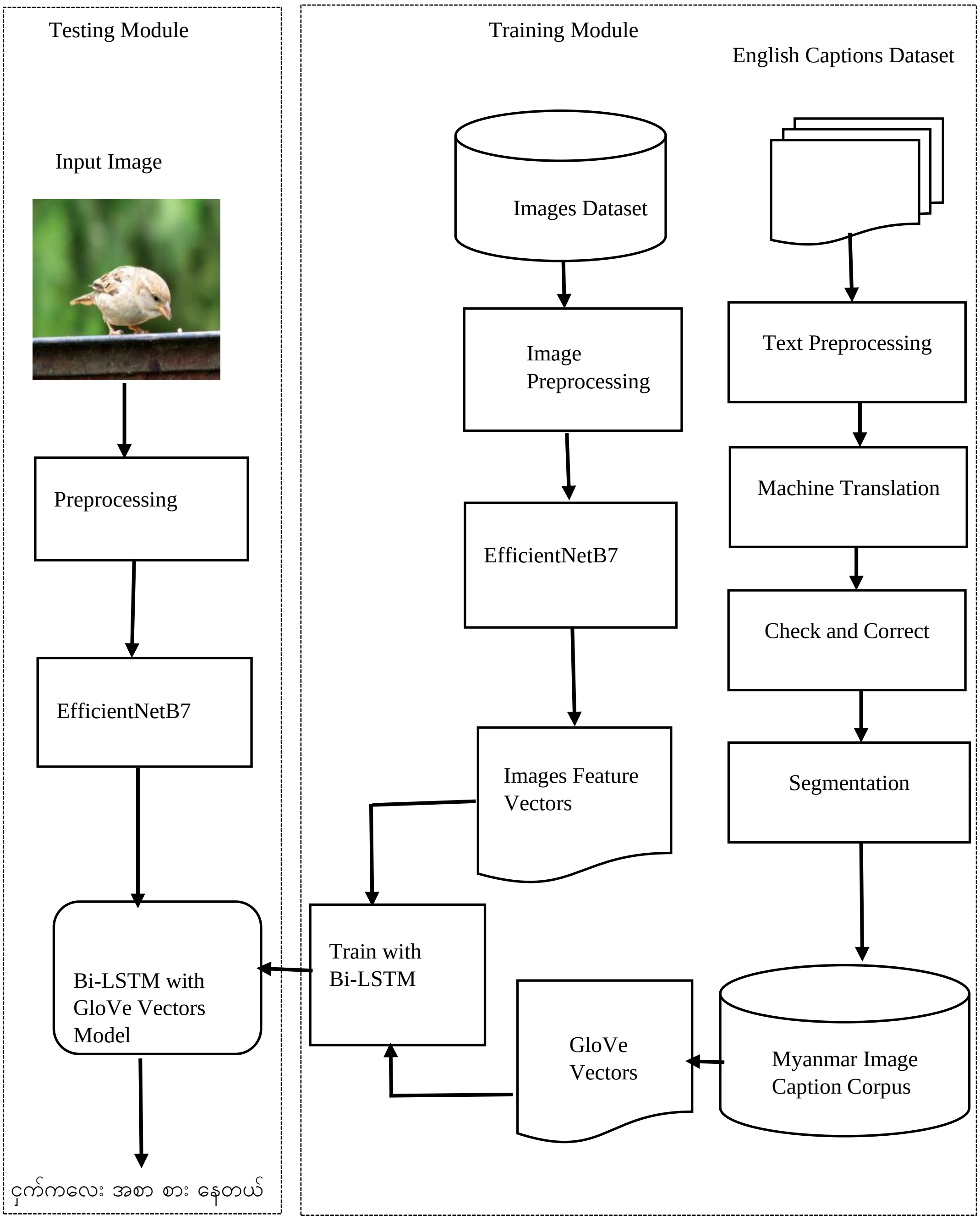


Figure 7.2 The Proposed Architecture of EfficientNetB7 and Bi-LSTM Based Myanmar IC System

Text pre-processing is very important role in language modelling, and syllable segmentation is significantly better than word segmentation for Myanmar image captioning system. After pre-processed the text data, we got the clean Myanmar image captions corpus which contains 50460 sentences for 10k images. And then, GloVe vectors on both word segmentation corpus and syllable segmentation corpus are built using GloVe v.1.2. Building of GloVe embedding vectors are described in Section 4.4.1. Furthermore, the efficiency of using GloVe vectors as the additional input features are examined on Bi-LSTM based Myanmar image captioning system to achieve the best learned model. After the training module is completed, it is necessary to give an input image for testing our proposed system.

7.2 Experiments

In this chapter, two different experiments are done to measure the performance of Myanmar image captioning on syllable and word segmentation. To find the most appropriate encoder and decoder network architecture for Myanmar image captioning system. Both of the experiment 1 and 2 used the same hyperparameters setting list as shown in Table 7.1.

- **Experiment 1 (Exp1)**

  Experiment 1 is done on Myanmar image captions corpus which contains 40460 sentences for 8k images. All of the images are taken from Flickr8k dataset. The system performance is evaluated using BLEU scores on both word and syllable segmented corpus. In this experiment, we did not consider about GloVe embedding vectors.

- **Experiment 2 (Exp2)**

  Experiment 2 is done on Myanmar image captions corpus which contains 50460 sentences for 10k images. All of the images are taken from Flickr8k dataset and 2k images are selected from Flickr30k dataset. The system performance is measured using BLEU, ROUGE-L, ROUGE-SU4 and METEOR metrics on both word and syllable segmented corpus. Furthermore, we added word and syllable GloVe vectors into the Bi-LSTM model.

7.2.1 Experimental Setups

All of the investigations were conducted on NVIDIA GeForce MX250, RAM 16GB, Ubuntu Linux machine. Keras library is used to implement with Python programming language. Sparse softmax cross entropy is used to evaluate the loss values which are measured the possibility error in distinct classification functions. The

Adaptive moment estimation (Adam) optimizer is used for more excellent performance instead of RMSprop optimizer. A dropout of 50 % was set, which is the efficient regularization technique to mitigate the excessive during the training time. The following equation is used to compute the loss values for all models.

$$\text{L (I, S)} = - \sum_{t=1}^{N} \log p_t(S_t) \qquad (7.1)$$

Where I is given input image and S is predicted caption, N is the length of output description. p_t and S_t are probability and predicting word at time t respectively. While the training iterations are doing, we have attempted to decrease the loss values.

Table 7.1 Hyperparameters Setting List of Our Models

Parameters	Best Values	Values
Embedding size	300	50, 100, 200, 300
Hidden layer size	256	128, 256, 512, 1024
Max-sequence length of word level	24	
Max-sequence length of syllable level	32	
Dense layer size	256	128, 256, 512, 1024
Batch size	32	32, 64
Number of epochs	15	10, 15, 20, 25
Beam search(k)	3	2, 3, 5, 7, 9
Random seeds	1035	

Table 7.1 shows the best hyperparameters setting list for Myanmar image captioning system according to the various parameters values we have tested to achieve the best accuracy. We set the different embedding size such as 50, 100, 200 and 300 among them 300 dimensions is selected for better experiment. Both of the Hidden layer and Dense layer size, 128, 256, 512 and 1024 were used to train in our experiments. Although the layer size that 512 and 1024 took longer training time than 256, they achieved comparable accuracy. Therefore, layer size 256 is selected in the further experiments. Maximum sequence length is the number of words in the longest caption of Myanmar image captions corpus. The maximum sequence length is 24 for word level and 32 for syllable level. We initially set batch size 64, however, due to the limitations of computational resources, we selected batch size 32 for our investigations. Number

of epochs mean number of iterations to train the model. We tried various epochs such as 10, 15, 20, 25 among them epoch 15 gave the highest performance to learn the best model. Beam search is the number of search times in generated captions with Myanmar language. It is tested various k values with 2, 3, 5, 7, and 9 are tested and it is found that k value 3 could reach the much better performance for our experiments. Random seeds are also chosen 1035. According to these various experiments, Table 7.1 displays the best hyperparameters setting list of our automatic Myanmar image captioning models.

In Table 7.2, Table 7.3 and Table 7.4, the term superscript "E" refers "EfficientNetB7", "G" is denoted as "VGG16", "A" is "AlexNet", "R" means "ResNet101", "N" is "NASNetLarge" Feature Extraction Models. "+M" is denoted as the Multi-task Learning, "+W" is using Word GloVe vectors, "+S" is using Syllable GloVe Vectors. "-" indicates in unused. The superscripts are also applicable in other sections in this chapter.

7.2.2 Experiment 1 (Exp1)

In this experiment 1, two different types of pre-trained feature extraction models VGG16 and NASNetLarge are used as encoder. Three different language models such as Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM) and Bidirectional

Table 7.2 Performance Comparison of Various Models on Two Different Segmented Corpus [P3]

	Word Segmented Corpus (%)				**Syllable Segmented Corpus (%)**			
Models	**B-1**	**B-2**	**B-3**	**B-4**	**B-1**	**B-2**	**B-3**	**B-4**
LSTM^{G} (baseline)	64.14	48.58	39.86	24.38	-	-	-	-
Bi-LSTM$^{G, A}$ [15]	65.5	46.8	32	21.5	-	-	-	-
GRU^{N}	66.45	50.36	40.65	26.4	69.35	57.34	50.97	38.2
LSTM^{N}	66.76	50.06	40.31	26.5	69.73	58.17	51.24	39.04
Bi-LSTM^{G}	65.37	49	39.13	25	69.81	57.69	50.87	38.06
Bi-LSTM^{N}	**67.24**	**51.29**	**41.75**	**27.55**	**70.74**	**58.74**	**52.44**	**40.05**

Long Short-Term Memory (Bi-LSTM) are used as decoder. VGG16 with LSTM based image captioning implemented in Chapter 5 is utilized as the baseline system in this Exp1.

Table 7.2 shows the performance comparison of various encoder decoder pair. The combination of NASNetLarge and Bi-LSTM model achieved the highest BLEU-4 score of 27.55% on word segmented corpus and 40.05% of BLEU-4 score on syllable segmented corpus compared with other different models such as NASNetLarge with GRU, NASNetLarge with LSTM, VGG16 with Bi-LSTM, baseline model LSTM^G as well as state-of-the-art model [15]. According to these experimental results, it is found that the syllable segmentation results obtained the better results than the word segmentation results. As can be seen in Table 7.2, NASNetLarge is better than VGG16 as encoder (shown as Bi-LSTM^N), it is gaps 2.55% of BLEU-4 score compared with 25% of BLEU-4 score (shown as Bi-LSTM^G) on word segmentation and 1.99% of BLEU-4 on syllable segmentation. It is observed that the selection of not only encoder but also decoder is very important for image captioning system as it also proved that in Exp2. Myanmar text preprocessing plays the vital role in language modelling of automatic Myanmar image caption generation.

7.2.3 Experiment 2 (Exp2)

In this experiment 2, three visual models are compared for encoding images: VGG16, NASNetLarge and EfficientNetB7, to examine the effects of applied encoding techniques. In language modelling, four different types of models such as such as Gated Recurrent Unit (GRU), Bidirectional Gated Recurrent Unit (Bi-GRU), Long Short-Term Memory (LSTM) and Bidirectional Long Short-Term Memory (Bi-LSTM) with and without GloVe embedding models are compared to investigate the effects of applied decoding techniques. The experimental results are presented in Table 7.3 and Table 7.4. VGG16 with LSTM based image captioning implemented in Chapter 5 and Bi-LSTM^N (Exp1) are utilized as baseline models in this experiment 2 (Exp2).

7.2.3.1 Effectiveness of Feature Extraction Models

Table 7.3 describes that the evaluation results for word segmented corpus of Myanmar image captioning models. The highest scores are displayed in highlighted. It is clear to see that without using EfficientNetB7 feature extraction model and keep other configurations unchanged (shown as Bi-LSTM^G and Bi-LSTM^N). There are 28.03% of BLEU-4, 42.96% of ROUGE-L, 49.45% of ROUGE-SU4 and 18.74% of METEOR scores using VGG16 with Bi-LSTM model (shown as Bi-LSTM^G) and 28.82% of BLEU-4, 43.79% of ROUGE-L, 51.79% of ROUGE-SU4 and 19.69% of METEOR scores using NASNetLarge with Bi-LSTM (shown as Bi-LSTM^N). The performance of both models significantly drops on all evaluation metrics for word segmentation compared with using EfficientNehtB7 feature extraction models.

Table 7.3 Performance Comparison of Various Models on Word Segmented Corpus [P4]

Models	Word Segmented Corpus (%)						
	B-1	B-2	B-3	B-4	ROUGE-L	ROUGE-SU4	METEOR
LSTM^G (baseline)	64.14	48.58	39.86	24.38	-	-	-
Exp1 (baseline)	67.24	51.29	41.75	27.55	-	-	-
Bi-LSTM$^{G, A}$ [15]	65.5	46.8	32	21.5	-	-	19.4
Bi-LSTM$^{G, +M}$ [16]	66.7	48.3	33.7	23	-	-	19.1
LSTM$^{R,+W}$ [60]	69	51.6	37.6	26.9	50.3	-	22.4
GRU^E	68.1	53.62	45.6	32.22	47.55	52.39	20.72
LSTM^E	69.65	54.98	47.38	33.57	47.17	51.45	21.06
Bi-GRU^E	68.84	54.28	46.25	32.99	47.52	54.05	21.02
Bi-LSTM^G	67.07	51.37	41.82	28.03	42.96	49.45	18.74
Bi-LSTM^N	67.63	51.82	42.4	28.82	43.79	51.79	19.69
Bi-LSTM^E	70.12	55.07	47.85	34.91	46.18	52.79	21.25
Bi-LSTM$^{E,+W}$	**71.42**	**56.73**	**48.45**	**35.09**	**49.52**	**54.34**	**21.3**

Table 7.4 shows that the evaluation results for syllable segmented corpus of Myanmar image captioning models. The highest scores are displayed in bold. It can be obviously observed that without using EfficientNetB7 feature extraction model and other configurations are kept no change. VGG16 with Bi-LSTM model (shown as Bi-LSTM^G) achieved the highest BLEU-4 score of 39.47%, ROUGE-L score of 58.14%, ROUGE-SU4 of 63.11% and METEOR score of 24.16%. NASNetLarge with Bi-LSTM (shown as Bi-LSTM^N) obtained the highest BLEU-4 score of 42.11%, ROUGE-

L score of 59.41%, ROUGE-SU4 score of 64.34% and METEOR score of 25.17% respectively. Both of the model performance obviously decreases on all evaluation metrics for syllable segmentation compared with using EfficientNetB7 feature extraction models.

The experiments result also stated how utilizing encoder effects on image captioning system performance. According to the results presented in Table 7.3 and Table 7.4, it can be seen that selection of encoder plays a special valuable part in image captioning system and can be quite upgraded model performance without modification a decoder architecture. To this end, EfficientNetB7 is applied as encoder for our proposed feature extraction model, because it has higher resolutions, such as 600x600, are also widely used in object detection ConvNets. It is believed that feature extraction on insufficient data is more challenging and assistance to evaluate the benefits brought by encoder. Replacing VGG16 and NASNetLarge with EfficientNetB7 brings significantly better performance on all evaluation metrics.

7.2.3.2 Effectiveness of Word Embedding

Furthermore, the efficiency of the system performance is improved by utilizing EfficientNetB7, next effective language understanding model is GloVe vectors for Myanmar image generation system. The dataset is gathered for construction of own GloVe vectors for both segmented corpus in order to progress the quality of the system.

Next, the Bi-LSTM$^{E,+S}$ and Bi-LSTM$^{E,+W}$ models utilizing GloVe embedding vectors are trained and measuring has been performed on each validation set to examine the achievement and generality of the system. The model with GloVe embedding vectors requires more training time than one-hot encoding vector and it will take to 5400 seconds per epoch. The comparison with different models in terms of BLEU, ROUGE-L, ROUGE-SU4 and METEOR scores results are presented in Table 7.3 and Table 7.4.

The best performing baseline model Bi-LSTM^{N} (Exp1) without using GloVe vectors is chosen to compare with the evaluation results. Table 7.3 presents the experimental results for word segmentation corpus. EfficientNetB7 and Bi-LSTM with word GloVe vectors (shown as Bi-LSTM$^{E,+W}$) significantly improve the BLEU score

Table 7.4 Performance Comparison of Various Models on Syllable Segmented Corpus [P4]

Models	Syllable Segmented Corpus (%)						
	B-1	B-2	B-3	B-4	ROUGE-L	ROUGE-SU4	METEOR
LSTM^{G} (baseline)	64.14	48.58	39.86	24.38	-	-	-
Exp1 (baseline)	70.74	58.74	52.44	40.05	-	-	-
Bi-LSTM$^{G, A}$ [15]	65.5	46.8	32	21.5	-	-	19.4
Bi-LSTM$^{G, +M}$ [16]	66.7	48.3	33.7	23	-	-	19.1
LSTM$^{R,+W}$ [60]	69	51.6	37.6	26.9	50.3	-	22.4
GRU^{E}	72.02	61.65	56.13	44.46	62.46	65.28	26.76
LSTM^{E}	73.06	62.02	57.02	44.82	64.16	65.69	26.67
Bi-GRU^{E}	72.46	61.74	55.92	43.97	61.17	65.84	26.88
Bi-LSTM^{G}	69.76	58.08	51.86	39.47	58.14	63.11	24.16
Bi-LSTM^{N}	72.19	60.71	54.44	42.11	59.41	64.34	25.17
Bi-LSTM^{E}	73.66	63.02	57.2	45.22	65.14	67.13	26.9
Bi-LSTM$^{E,+S}$	**73.9**	**63.45**	**57.8**	**46.2**	**65.62**	**68.43**	**27.07**

4.18%, 5.44%, 6.7% and 7.54% for BLEU-1, BLEU-2, BLEU-3, and BLEU-4 respectively using word GloVe vectors compared with baseline model Exp1. It can be seen that the proposed model (shown as Bi-LSTM$^{E,+W}$) achieved much better performance compare to without using GloVe vectors (shown as Bi-LSTM^{E}) as well as other neural network models on word segmented corpus.

Table 7.4 shows the experimental results for syllable segmented corpus. EfficientNetB7 and Bi-LSTM with syllable GloVe vectors (shown as Bi-LSTM$^{E,+S}$) significantly improve the 3.16%, 4.71%, 5.36%, 6.15% for BLEU-1, BLEU-2, BLEU-3, and BLEU-4 respectively compared with baseline model Bi-LSTM^{N} (Exp1). The Bi-LSTM$^{E,+S}$ model also obtained much better evaluation results than other neural network models like GRU^{E}, LSTM^{E}, Bi-GRU^{E}, Bi-LSTM^{G}, Bi-LSTM^{N}, Bi-LSTM^{E} without using GloVe embedding models as well as state-of-the-art models. In addition, Bi-LSTM$^{E,+S}$ utilizing syllable GloVe embedding model attained higher evaluations results than Bi-LSTM$^{E,+W}$ using word GloVe embedding model because a word is made up of one or more syllables in Myanmar language(i.e., a word ‘Woman’ in Myanmar ‘အမျိုးသမီး’ consists of four syllables like ‘အ’, ‘မျိုး’ ‘သ’ and ‘မီး’). Evaluation scores are computed by measuring how many words or syllables are similar between the machine generated descriptions and reference descriptions, the reason for that, the output results scores of syllable segmentation are much higher than the output results scores of word segmentation.

Regarding ROUGE-L, ROUGE-SU4 and METEOR performance, the effectiveness of the GloVe embeddings vectors is examined for both segmented corpus while the overall model is training on a specific dataset. As the GloVe embeddings accomplished the finest during all of our experiments, it reaches 49.52%, 54.34% and 21.3% for ROUGE-L, ROUGE-SU4 and METEOR scores respectively on word segmented corpus as present in Table 7.3. As we can be seen in Table 7.4, our proposed model (shown as Bi-LSTM$^{E,+S}$) attained the highest scores 65.62%, 68.43% and 27.07% for ROUGE-L, ROUGE-SU4 and METEOR scores respectively whereas model without GloVe embedding vectors (shown as Bi-LSTM^{E}) obtains 65.14% on ROUGE-L, 67.13% on ROUGE-SU4 and 26.9 on METEOR score. ROUGE-L score is sightly inferior on word segmented corpus compare to 50.3% in [60] and METEOR scores also exceed the rest of the models for both tasks.

According to the evaluation results, it can be concluded that the effectiveness of GloVe vectors can be observed clearly in Myanmar image captioning system for both tasks although the size of GloVe vectors is not large. Nonetheless, it is believed that the sufficient amount of GloVe vectors into the system can achieve more improvements, note that the proposed model achieved the best performance on all evaluation metrics.

7.2.3.3 Comparison with State-of-the-Art Methods

In this section, the proposed EfficientNetB7 and Bi-LSTM with GloVe vectors (shown as Bi-LSTM$^{E,+W}$ and Bi-LSTM$^{E,+S}$) models are compared with state-of-the art methods. The comparison results are presented in Table 7.3 and Table 7.4. Our approach obtained the highest scores on all evaluation metrics for both segmented corpora. EfficientNetB7 and Bi-LSTM using syllable GloVe vectors mostly achieved better performance compared to EfficientNetB7 and Bi-LSTM using word GloVe vectors as well as other various neural networks models. It should be aware that a recent interesting work [15] is significantly inferior to 5.92%, 9.93%, 16.45%, and 13.59% for

BLEU-1, BLEU-2, BLEU-3, BLEU-4 and METEOR respectively compared to Bi-LSTM$^{E,+W}$ on word segmented corpus, and 8.4% of BLEU-1, 16.65% of BLEU-2, 25.8% of BLEU-3, 24.7% of BLEU-4 and 7.67% of METEOR score using Bi-LSTM$^{E,+S}$ on syllable segmented corpus.

Furthermore, the former model [16] is quite reduce to 12.09% of BLEU-4 and 2.2% of METEOR score using Bi-LSTM$^{E,+W}$ with word level, 23.2% of BLEU-4 score and 7.97% of METEOR score utilizing Bi-LSTM$^{E,+S}$ with syllable level on updated corpus. In addition, our best results obtained 35.09% of BLEU-4, 49.52% of ROUGE-L and 21.3% of METERO score (compare to 26.9% of BLEU-4, 50.3% of ROUGE-L and 22.4% of METEOR score in [60]) on word segmented corpus and 46.2% of BLEU-4, 65.62% of ROUGE-L and 27.07% of METERO score (compare to 26.9% of BLEU-4, 50.3% of ROUGE-L and 22.4% of METEOR score in [60]) on syllable segmented corpus.

What is more, in former state-of-the-art interesting work [15-16], the authors observed that the small dataset Flickr8K which has difficulty to train the deep models because of inadequate data. Nevertheless, the proposed EfficientNetB7 and Bi-LSTM with GloVe embedding model substantially outperforms on all evaluation metrics for both word and syllable segmented corpus although the size of the corpus is small (around 50460 sentences for 10K images), compare with other various neural networks models namely GRU^{E}, Bi-GRU^{E}, LSTM^{E}, Bi-LSTM^{G}, Bi-LSTM^{N}, Bi-LSTM^{E}, the baseline models as well as the state-of-the-art models [15-16, 60].

7.3 Subjective Evaluation

The efficiency of EfficientNetB7 and Bi-LSTM based image captioning system is subjectively measured by perceptual tests. The 18 images are selected with various categories from the dataset and open test Burmese images that are taken from Google. There are four generated captions for each input image (18 * 4 = 72 generated captions) by using EfficientNetB7 and Bi-LSTM with and without GloVe vector for two different segmented corpora. The 10 non-expert native persons of age range from 30 to 40 years were participated to give the marks in each generated caption result which are matched with input image and its generated captions by using four different models. It is the subject to rate the generated caption on a scale from 1 to 5 where 1 is bad and 5 is excellent. The scores of Bi-LSTM$^{E,+W}$, Bi-LSTM$^{E,+S}$ and Bi-LSTM^{E} on two distinct segmented corpora are shown in Figure 7.3. It can be observed that the proposed model

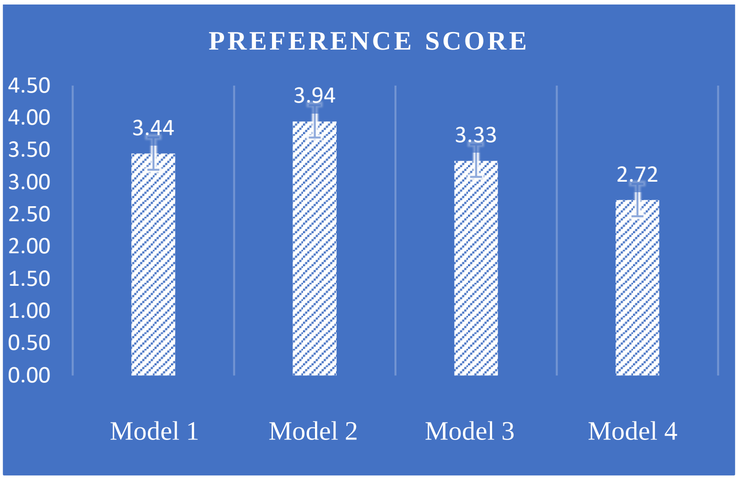


Figure 7.3 Preference Score of EfficientNetB7 and Bi-LSTM With and Without GloVe Vectors Models on Two Segmented Corpus

obtained the maximum preference score 3.94 using LSTM$^{E,+W}$ word vector model (Model 2), 3.44 preference score using Bi-LSTM$^{E,+S}$ syllable vector model (Model 1), 3.33 preference score using syllable model without syllable GloVe vectors (Model 3) and 2.72 preference score using word model without word GloVe vector (Model 4). The proposed model obtained the highest score in terms of both objective and subjective evaluation. It is found that word vectors model is more preferable than syllable vectors model by human evaluators in subjective evaluation results. In the contradiction, with no GloVe vectors, syllable models gained higher performance scores than word models in both objective and subjective evaluation. Therefore, the evaluation results shown that text preprocessing and word representation are effective for Myanmar image captioning.

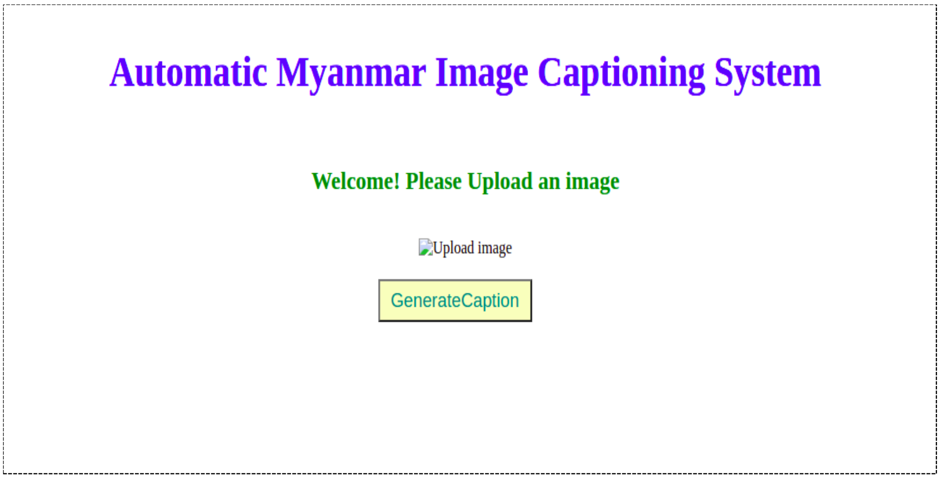
7.4 System Demonstration

The features and the flow of the system with various testing results can see clearly by using the program demonstration. It gives visual support to improve the quality of the system presentation. The flow of program demonstration is described in detail as the follows:

7.4.1 Main View of System Demonstration

Figure 7.4 shows the first page of EfficientNetB7 and Bi-LSTM-based Myanmar image captioning by system demonstration. The UI design of the system is very simple and user-friendly design. First, the user needs to upload an image and generate caption button is provided to generate caption with Myanmar language. Four different Myanmar captions will be produced for a given input image by using EfficientNetB7 and Bi-LSTM with and without GloVe vectors features on two different segmented corpora. Four generated captions for each image are the following:

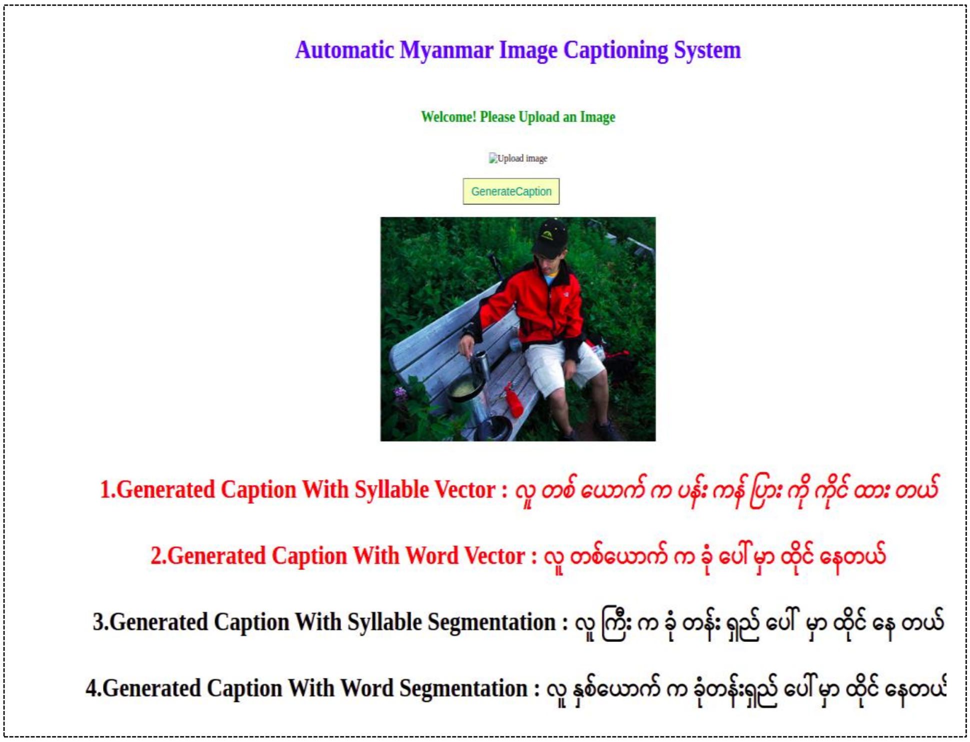
- 1 is the generated caption result with syllable vectors that are described in red font
- 2 is the generated caption result with word vectors that are described in red font
- 3 is the generated caption result with syllable model without using syllable GloVe vectors
- 4 is the generated caption result with word model without using word GloVe vectors



Automatic Myanmar Image Captioning System

Welcome! Please Upload an image

Figure 7.4 Main View by Proposed Models



Automatic Myanmar Image Captioning System

Welcome! Please Upload an Image

1.Generated Caption With Syllable Vector : လူ တစ် ယောက် က ပန်း ကန် ပြား ကို ကိုင် ထား တယ်

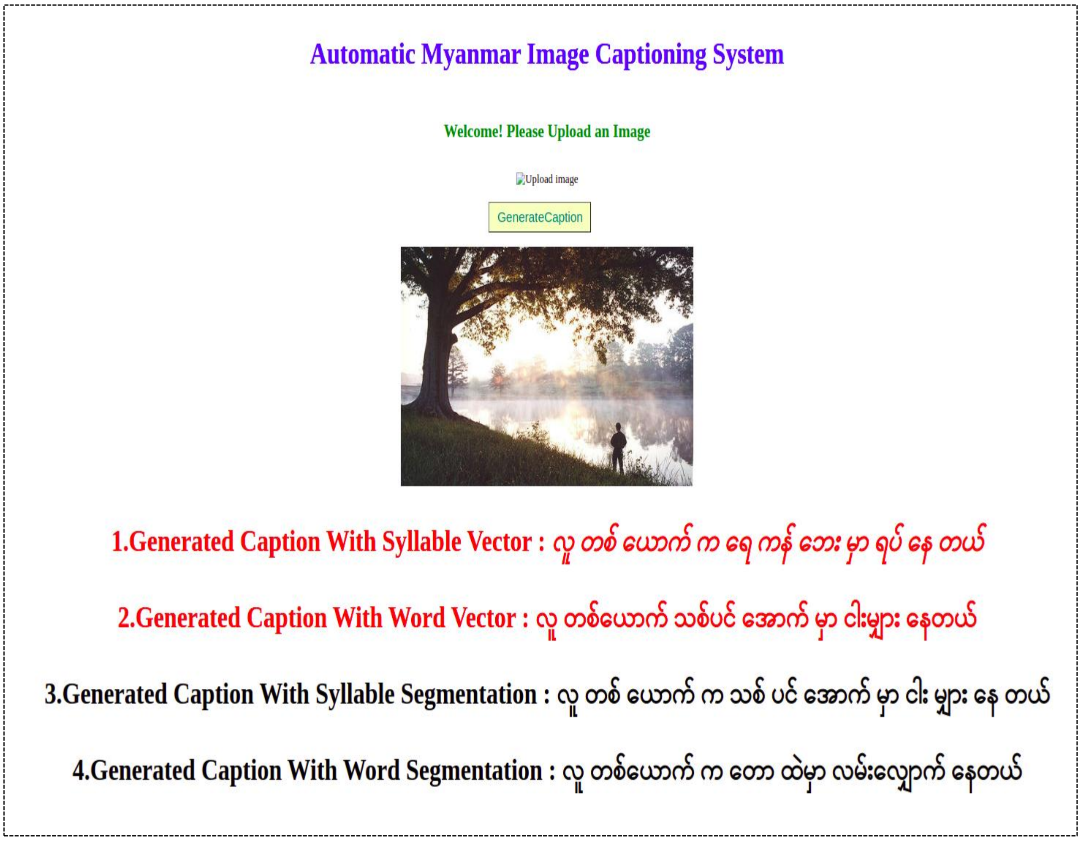
2.Generated Caption With Word Vector : လူ တစ်ယောက် က ခုံ ပေါ် မှာ ထိုင် နေတယ်

3.Generated Caption With Syllable Segmentation : လူ ကြီး က ခုံ တန်း ရှည် ပေါ် မှာ ထိုင် နေ တယ်

4.Generated Caption With Word Segmentation : လူ နှစ်ယောက် က ခုံတန်းရှည် ပေါ် မှာ ထိုင် နေတယ်

Figure 7.5 Generated Caption View by Proposed Models

The Figure 7.5 was described by using VGG16 with LSTM that are implemented in Chapter 5 but the generated result misses to present the minor features and misidentify the object like “နံရံ” (“wall”) instead of “ခုံတန်းရှည်” (“bench”). However, as can be seen that the proposed models can be resolved this issue. Both of the models with GloVe vectors can capture the objects in details and also cover the different semantic information. For example, generated caption with syllable vectors captures ‘ပန်း ကန် ပြား ကိုင် ထား တယ် (“holding the plate”) while the generated caption with word vector describes ‘ခုံ ပေါ် မှာ ထိုင် နေ တယ်’ (“sitting on the seat”). The generated caption with syllable segmentation without GloVe vectors is more specific and can identify this image accurately like “ခုံ တန်း ရှည်” (“bench”) although the generated caption with word vector misidentifies the count of person “နှစ်ယောက်” (“two”) instead of (“one”). Moreover, all of the generated captions 1, 2, 3 and 4 can predict accurately, and also identify the action and information of the major objects.



Automatic Myanmar Image Captioning System

Welcome! Please Upload an Image

1.Generated Caption With Syllable Vector : လူ တစ် ယောက် က ရေ ကန် ဘေး မှာ ရပ် နေ တယ်

2.Generated Caption With Word Vector : လူ တစ်ယောက် သစ်ပင် အောက် မှာ ငါးမျှား နေတယ်

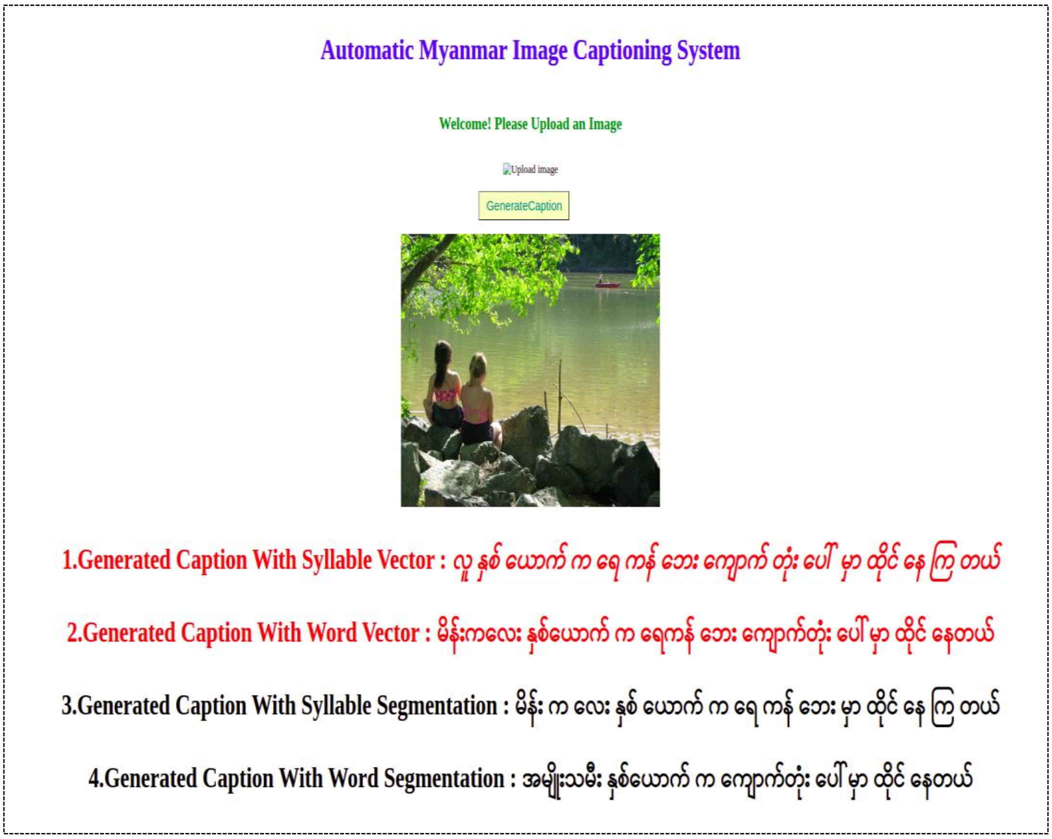
3.Generated Caption With Syllable Segmentation : လူ တစ် ယောက် က သစ် ပင် အောက် မှာ ငါး မျှား နေ တယ်

4.Generated Caption With Word Segmentation : လူ တစ်ယောက် က တော ထဲမှာ လမ်းလျှောက် နေတယ်

Figure 7.6 Generated Caption View by Proposed Models

In this Figure 7.6, both of the generated captions 1 and 2 are much more similar to one of the ground-truth captions and can capture the objects “ငါးမျှား” (“fishing”),

“သစ်ပင် အောက်” (“under the tree”), “ရေ ကန် ဘေး” (“beside the lake”) although the contents of the image hard to capture precisely. In generated captions 3 and 4, it can be found that the syllable segmentation result is better than word segmentation without GloVe vectors features.



Automatic Myanmar Image Captioning System

Welcome! Please Upload an Image

1.Generated Caption With Syllable Vector : လူ နှစ် ယောက် က ရေ ကန် ဘေး ကျောက် တုံး ပေါ် မှာ ထိုင် နေ ကြ တယ်

2.Generated Caption With Word Vector : မိန်းကလေး နှစ်ယောက် က ရေကန် ဘေး ကျောက်တုံး ပေါ် မှာ ထိုင် နေတယ်

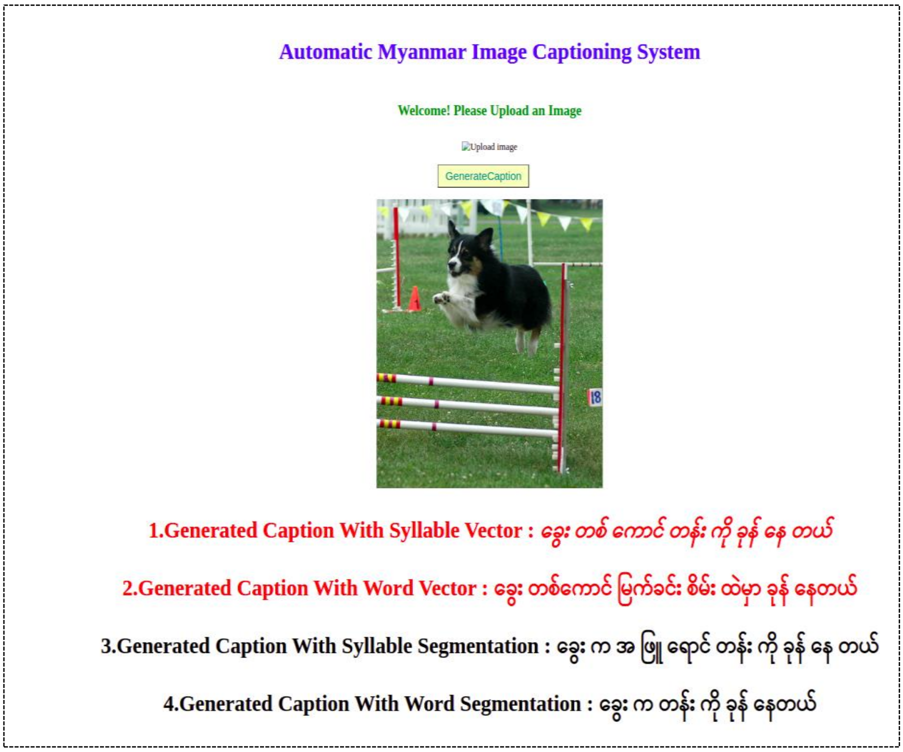
3.Generated Caption With Syllable Segmentation : မိန်း က လေး နှစ် ယောက် က ရေ ကန် ဘေး မှာ ထိုင် နေ ကြ တယ်

4.Generated Caption With Word Segmentation : အမျိုးသမီး နှစ်ယောက် က ကျောက်တုံး ပေါ် မှာ ထိုင် နေတယ်

Figure 7.7 Generated Caption View by Proposed Models

In this Figure 7.7, the proposed models effectively predict the activities and information of the main objects in this image. For example, Generated caption 1 with syllable vector model captures the objects like “လူ နှစ် ယောက်” (“two persons”) “ရေ ကန် ဘေး” (“beside the lake”) “ကျောက် တုံး” (“stone”) the action like “ထိုင် နေ” (“sitting”) but the generated caption 2 with word vector model can capture the object more detail like “မိန်းကလေး နှစ်ယောက်” (“two girls”) instead of “လူ နှစ် ယောက်” (“two persons”). The generated captions 3 are not completely identified the objects like “ကျောက် တုံး” (“stone”) and “ရေကန် ဘေး” (“beside the lake”) in generated captions 4.

Nonetheless, all of the generated captions in this figure are quite accurately and correspondence with each other.



Automatic Myanmar Image Captioning System

Welcome! Please Upload an Image

1.Generated Caption With Syllable Vector : ခွေး တစ် ကောင် တန်း ကို ခုန် နေ တယ်

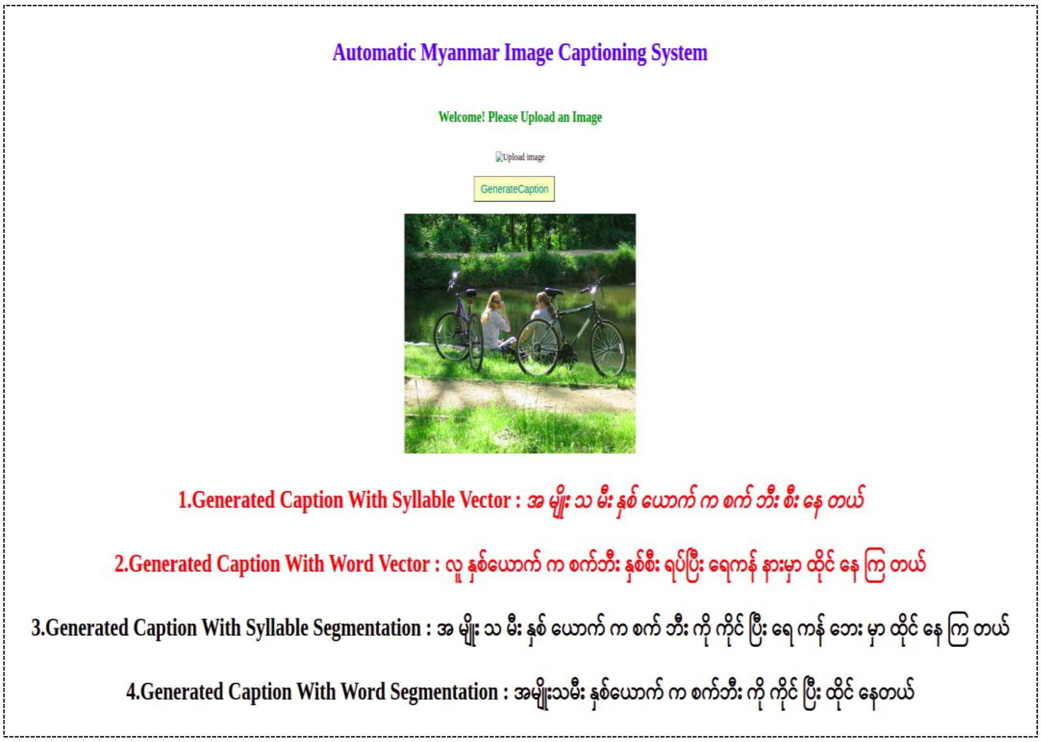
2.Generated Caption With Word Vector : ခွေး တစ်ကောင် မြက်ခင်း စိမ်း ထဲမှာ ခုန် နေတယ်

3.Generated Caption With Syllable Segmentation : ခွေး က အ ဖြူ ရောင် တန်း ကို ခုန် နေ တယ်

4.Generated Caption With Word Segmentation : ခွေး က တန်း ကို ခုန် နေတယ်

Figure 7.8 Generated Caption View by the Proposed Models

In Figure 7.8, generated caption 1 and 2 with GloVe vectors models can capture the objects in details, and also predict the word accurately. As can be seen, the proposed model can identify the object "ခွေး" ("dog") and count of the object like "တစ် ကောင်" ("one"), action like "ခုန် နေ တယ်" ("is jumping"). The generated caption 2 with word vector model identify the place of object "မြက်ခင်း စိမ်း ထဲမှာ" ("in the green grass") while the generated caption 1 predicts the word "တန်း" ("hurdle"). The generated captions 3 can capture the color of object like 'အ ဖြူ ရောင် တန်း' ('white hurdle") this result is the same with ground-truth caption. All of the generated captions 1, 2, 3 and 4 can produce the reasonable captions. The syllable model can identify the most of the objects in details rather than word model without using word embedding model. In the contradiction, word vectors model is more specific in generating captions than syllable vectors model.



Automatic Myanmar Image Captioning System

Welcome! Please Upload an Image

1.Generated Caption With Syllable Vector : အ မျိုး သ မီး နှစ် ယောက် က စက် ဘီး စီး နေ တယ်

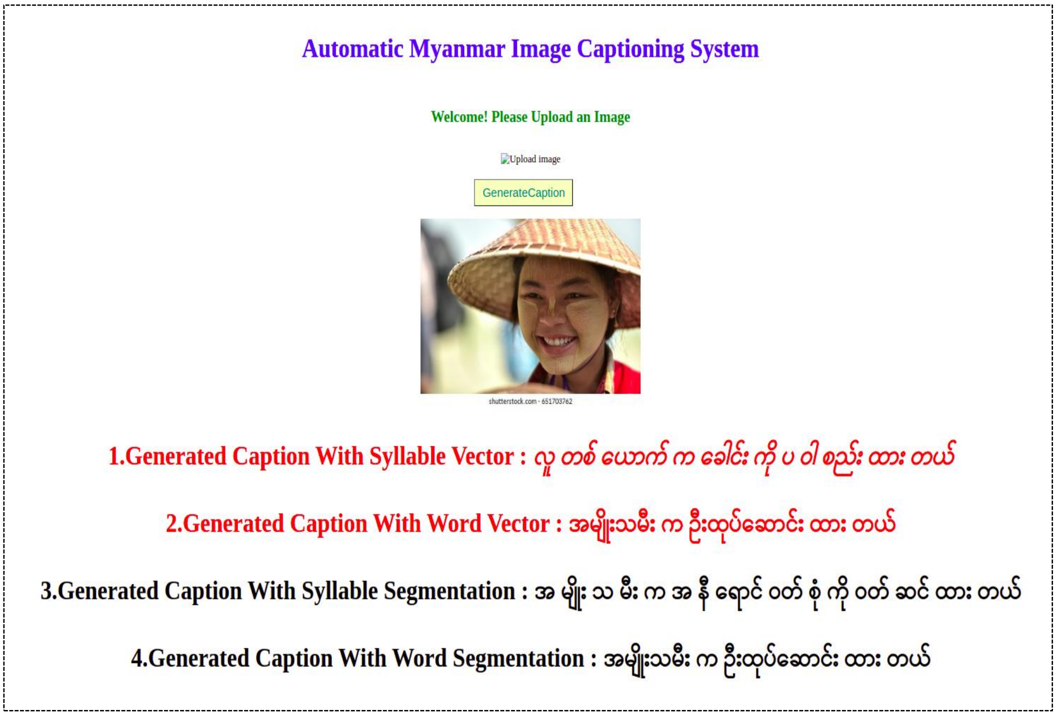
2.Generated Caption With Word Vector : လူ နှစ်ယောက် က စက်ဘီး နှစ်စီး ရပ်ပြီး ရေကန် နားမှာ ထိုင် နေ ကြ တယ်

3.Generated Caption With Syllable Segmentation : အ မျိုး သ မီး နှစ် ယောက် က စက် ဘီး ကို ကိုင် ပြီး ရေ ကန် ဘေး မှာ ထိုင် နေ ကြ တယ်

4.Generated Caption With Word Segmentation : အမျိုးသမီး နှစ်ယောက် က စက်ဘီး ကို ကိုင် ပြီး ထိုင် နေတယ်

Figure 7.9 Generated Caption View by the Proposed Models

In Figure 7.9, all of the generated captions for both with and without GloVe vectors models identified the activities, count, gender and information of the objects in details. The models also generated the grammatically correct captions and relationship between images and captions.



Automatic Myanmar Image Captioning System

Welcome! Please Upload an Image

1.Generated Caption With Syllable Vector : လူ တစ် ယောက် က ခေါင်း ကို ပ ဝါ စည်း ထား တယ်

2.Generated Caption With Word Vector : အမျိုးသမီး က ဦးထုပ်ဆောင်း ထား တယ်

3.Generated Caption With Syllable Segmentation : အ မျိုး သ မီး က အ နီ ရောင် ဝတ် စုံ ကို ဝတ် ဆင် ထား တယ်

4.Generated Caption With Word Segmentation : အမျိုးသမီး က ဦးထုပ်ဆောင်း ထား တယ်

Figure 7.10 Generated Caption View for Open Test Image by the Proposed Models

In Figure 7.10, it is the Burmese photo that was taken by Google search engine. The proposed EfficientNetB7 and Bi-LSTM with GloVe vector for both tasks word and syllable vectors accurately predict the major features, activities and relationship of the image and also generated grammatically correct sentence even with the open test image. As can be seen in generated captions 3 and 4, EfficientNetB7 and Bi-LSTM without GloVe vectors for both tasks word and syllable segmentations are also predicted correctly in this open test image.

To conclude the experiment results, EfficientNetB7 with Bi-LSTM using GloVe vectors features for both tasks word and syllable vectors can give highly performance results than EfficientNetB7 with Bi-LSTM without using GloVe vectors as well as the other different models for Myanmar IC system even with the open test images. It illustrates that the proposed EfficientNetB7 and Bi-LSTM with GloVe vectors model has a powerful ability to learn visual-language correlation and predicts grammatically correct captions in Myanmar language for both tasks word and syllable segmentations. In addition, increasing the size of word vectors and variety of the training data may assist in decreasing the type of errors. All of the Figure 7.5, 7.6, 7.7, 7.8, 7.9 and 7.10 are automatically produced descriptions in Myanmar language without any human interruption.

7.5 Summary

In this chapter, the proposed architecture of EfficientNetB7 with Bi-LSTM using GloVe vectors features was presented in detail. The effect of text preprocessing and word vectors features was explored. It can be seen that word embedding features for both tasks word and syllable vectors can give highly performance results than EfficientNetB7 with Bi-LSTM using one-hot encoding model as well as the other different models for Myanmar IC system even with the open test images. Therefore, it can be concluded that the proposed model has a powerful ability to learn visual-language correlation and predicts grammatically correct captions with Myanmar language for both tasks word and syllable segmentations.

CHAPTER 8
CONCLUSION AND FUTURE WORKS

In the conclusion of the research work, the advantages and the limitation of the proposed model are presented.

This paper reports the research results of EfficientNetB7 and Bi-LSTM with GloVe vectors features for enhancing Myanmar image captioning system. All five contributions in features extraction module and language modelling module meet the objectives of this dissertation presented in Chapter 1.

The main contribution of the research work is the very first evaluation of deep neural network architecture on Myanmar image captioning. As part of this research, Myanmar image captions corpus (around 50460 sentences for 10k images) is manually created based on the Flickr8k and 2k images are selected from Flickr30k dataset in Chapter 4. This Myanmar image captions corpus was prepared for both word and syllable segmentation. The correct prediction captions of Myanmar text with syllable information have been obtained by this syllable image captions corpus.

The effect of word information is examined by using NASNetLarge with Bi-LSTM on Myanmar image captioning in Exp1 of Chapter 7 and it can be proved that word information can progress the quality of description on Myanmar image captioning system even though word segmentation process is still indefinite. As stated in this fundamental result, several contextual linguistic information in addition to word information are taken into consideration for next investigations of Myanmar image captioning. VGG16 with LSTM and NASNetLarge with Bi-LSTM model are used as the baseline systems.

Moreover, the effectiveness of word representation on EfficientNetB7 with Bi-LSTM based Myanmar image captioning system for both word and syllable segmentation tasks are proposed. Moreover, words and syllable GloVe vectors were also constructed for Myanmar image captioning by utilizing the gathered monolingual Myanmar corpus for much better performance. The comparisons are done on various neural network models, namely EfficientNetB7 with GRU, EfficientNetB7 with Bi-GRU, EfficientNetB7 with LSTM, VGG16 with Bi-LSTM, NASNetLarge with Bi-LSTM, EfficientNetB7 and Bi-LSTM without word vectors, EfficientNetB7 and Bi-LSTM with GloVe vectors, baseline models and state-of-the-art models in Chapter 7.

Although the size of GloVe vectors is small, word and syllable vectors features can give the effectiveness of Myanmar image captioning system performance. However, this exploration of using word and syllable vectors features for image captioning is the initial attempt to use Bi-LSTM network in Myanmar language. The main objective of the research is to enhance the Bi-LSTM network on Myanmar image captioning system. In comparing the results of different encoder and decoder techniques are reported and it is approved by the much more experimental results that are clearly presented in this research.

According to subjective and objective evaluation results, EfficientNetB7 with Bi-LSTM using GloVe embedding model achieved significantly better performance than EfficientNetB7 with Bi-LSTM without using GloVe embedding model as well as other neural network models. Furthermore, model-based word vectors are more preferable over model-based syllable vectors by human evaluators in the subjective evaluation results.

8.1 Advantages and the Limitation of the System

The more accurate image captions have been achieved by applying word representation on EfficientNetB7 with Bi-LSTM based image captioning for Myanmar language as it has been approved in former chapters. This system can be used in several application areas such as for assisting visually impaired people who can only feel the world by touch, intelligent human computer interactions, developing image search engines, teaching concept for children, social media platforms like Facebook and Twitter can directly generate captions from images. The exact information can be gained from these photos where are the places (e.g., beach, cafe, and road), what are the people wearing and importantly what are they doing there.

The effectiveness of word representation for implementing language modelling of Myanmar image captioning system has been predicted by the text analysis part and it can be used in next research of Myanmar IC system.

The sufficient amount text data for Myanmar image captions corpus was built and can be used not only in Myanmar image captioning system but also in another NLP research.

As the limitation, Out-of-Vocabulary problem can be occurred for rare words in Myanmar Language due to the size of the Myanmar image captions corpus is small in this system.

8.2 Future Works

In this work, image captions corpus has been developed that is the extension of Flickr8k and 2k images are selected from Flickr30k dataset. In the future, image captions corpus for the Myanmar culture images will be gathered to conduct the investigations and to generate the better efficiency in Myanmar image captioning task. Moreover, the size of the GloVe vectors will be extended for word and syllable vectors features that can give more effectiveness of Myanmar image captioning performance.

In the future, more comparative analysis will be examined on more different word embedding models with distinct parameters setting. Moreover, transformer and attention mechanism will be investigated for the language modelling, and other new feature extraction models such as EfficientNetV2L, ConvNeXtBase, ConvNeXtLarge and ConvNeXtXLarge models will be explored. Furthermore, as the extension of this research popular encoding methods such as 2D-CNN features, 3D-CNN features and semantic features, and LSTM or GRU will also be used to generate tokens circularly as decoding methods for Myanmar video captioning.

AUTHOR'S PUBLICATIONS

[P1] San Pa Pa Aung, Win Pa Pa and Tin Lay Nwe, “Automatic Myanmar Image Captioning using CNN and LSTM-Based Language Model”, Proceeding of the 1st Joint SLTU and CCURL Workshop **(SLTUCCURL 2020)**, pp. 139–143, Marseille, France, 2020.

[P2] San Pa Pa Aung, Win Pa Pa and Tin Lay Nwe, “InceptionResNetV2 and Recurrent Neural Network for Myanmar Image Description”, Proceedings of the 23rd Conference of the Oriental COCOSDA **(Oriental COCOSDA 2020)**, pp. 187-192, Yangon, Myanmar, November 5-7, 2020. (**Poster**)

[P3] San Pa Pa Aung, Win Pa Pa and Tin Lay Nwe, “Improving Myanmar Image Caption Generation Using NASNetLarge and Bi-directional LSTM”, Proceeding of the IEEE 19th International Conference on Computer Applications **(IEEE-ICCA 2021)**, pp. 164-169, Yangon, Myanmar.

[P4] San Pa Pa Aung and Win Pa Pa, “EfficientNetB7 and Bi-LSTM with GloVe vectors Based Myanmar Image Captioning”, International Journal of Intelligent Engineering and Systems **(IJIES),** Japan, 2022. ISSN: 2185-3118 **(Scimago index)**

BIBLIOGRAPHY

[1] A. A. Nugraha, A. Arifianto and Suyanto, “Generating Image Description on Indonesian Language using Convolutional Neural Network and Gated Recurrent Unit”, 7th International Conference on Information and Communication Technology (ICoICT), 2019.

[2] A. Chetan, and J. Vaishli, “Image Caption Generation using Deep Learning Technique”, Fourth International Conference on Computing Communication Control and Automation (ICCUBEA).

[3] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth, “Every Picture Tells a Story: Generating Sentences from Images”, European conference on computer vision, Springer, pp. 15–29, 2010.

[4] A. Graves and J. Schmidhuber. “Framewise phoneme classification with bidirectional LSTM and other neural network architectures”. In: Neural Networks 18.5-6 (2005), pp. 602–610.

[5] A. Huda, A1-muzaini, N. Tasniem and B. Hafida, “Automatic Arabic Image Captioning using RNNLSTM-Based Language Model and CNN”, International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 9, No.6, 2018.

[6] A. M. Hlaing and W. P. Pa, “Word Representations for Neural Network Based Myanmar Text-to-Speech System”, International Journal of Intelligent Engineering and System, Vol.13, No.2, pp. 239-249, 2020.

[7] A. Puscasiu, A. Fanca, D. I. Gota and H. Valean, "Automated image captioning", 2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR), 2020.

[8] A. Rathi, “Deep learning apporach for image captioning in Hindi language”, IEEE,2020.

[9] B.Heinzerling and M.Strube, BPEmb: “Tokenization-free Pre-trained Subword Embeddings in 275 Languages”, 2018.

[10] B. Yajurv, B. Aman, R. Deepanshu, and M. Himanshu, “Image Captioning using Google’s Inceptionresnetv2 and Recurrent Neural Network”, IEEE, 2019.

[11] B. Zoph, V. Vasudevan, J. Shlens and Q. V. Le, “Learning Transferable Architectures for Scalable Image Recognition”, arXiv:1707.07012v4 [cs.CV] 11 Apr 2018.

[12] C. Amritkar and V. Jabade, “Image Caption Generation using Deep Learning Technique”, Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), 2018.

[13] C. Szegedy, S. Ioffe, V. Vanhoucke and A. Alemi, “Inception-v4, InceptionResNet and the Impact of Residual Connections on Learning”, arXiv: 1602.07261v2 [cs.CV], 2016.

[14] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna “Rethinking the Inception Architecture for Computer Vision”, arXiv:1512.00567v3 [cs.CV], 2015.

[15] C. Wang, H. Yang, C. Bartz and C. Meinel, “Image Captioning with Deep Bidirectional LSTMs”, ACM, DOI: http://dx.doi.org/10.1145/ 2964284.2964299, Amsterdam, Netherlands, 2016.

[16] C. Wang, H. Yang, and C. Meinel, “Image Captioning with Deep Bidirectional LSTMs and Multi-Task Learning”, ACM Transaction on Multimedia Computing Communications, and Application, Vol. 14, No. 2s, Article 40. ,2018.

[17] C. Y. Lin, “ROUGE: A Package for Automatic Evaluation of Summaries”, *In: Proc. of the Workshop on Text Summarization Branches Out (WAS 2004)*, Barcelona, Spain, 2004.

[18] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate”, arXiv preprint arXiv:1409.0473, 2014.

[19] D. Jacob, G. Saurabh, and G. Ross, “Exploring Nearest Neighbor Approaches for Image Captioning”, arXiv: 1505.04467, 2015.

[20] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, T. Mikolov, “Learning Word Vectors for 157 Languages”.

[21] E. Grave, P. Bojanowski, P. Gupta, A. Joulin, and T. Mikolov, “Learning word vectors for 157 languages”, axXiv preprint arXiv:1802.06893, 2018.

[22] E. Mulyanto, E. I. Setiawan, E. M. Yuniarno and M. H. Purnomo, “Automatic Indonesian Image Caption Generation using CNN-LSTM

Model and FEEH-ID Dataset”, 978-1-5386-8344-6/19/$31.00 ©2019 IEEE.

[23] H. Parikh, H. Sawant, B. Parmar, R. Shah, S. Chapaneri and D. Jayaswal, “Encoder-Decoder Architecture for Image Caption Generation”, 3rd International Conference on Communication System, Computing and IT Applications (CSCITA), 2020.

[24] H. Peng and N. Li, “Generating Chinese Captions for Flickr30K Images,” 2016.

[25] H. Micah, Y. Peter, and H. Julia, “Framing image description as a ranking task: Data, models and evaluation metrics”, Journal of Artificial Intelligence Research, Vol. 47, pp. 853-899, May, 2013.

[26] H. Xinwei, Y. Yang and S. Baoguang, “VD-SAN: Visual-Densely Semantic Attention Network for Image Caption Generation”, Neurocomputing, 2018.

[27] I. Afyouni, I. Azhara, A. Elnagara, “AraCap: A hybrid deep learning architecture for Arabic Image Captioning”, 5th International Conference on AI in Computational Linguistics, 2021.

[28] I. Sutskever, O. Vinyals and Q. V. Le, “Sequence to Sequence Learning with Neural Networks”, Advances in neural information processing systems, pp. 3104–3112, 2014.

[29] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. “Empirical evaluation of gated recurrent neural networks on sequence modeling”. In: 2014.

[30] J. Pennington, R. Socher, C. Manning, “GloVe: Global Vectors for Word Representation”, In: Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543, 2014.

[31] K. Cho, B. V. Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk and Y. Bengio, “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation”, arXiv preprint arXiv:1406.1078, 2014.

[32] K. O’Shea1 and R. Nash, “An Introduction to Convolutional Neural Networks”.

[33] K. Papineni, S. Roukos, T. Ward, and W. J. Zhu, “BLEU: a method for automatic evaluation of machine translation,” In: *Proc. of the 40th Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, pp. 311–318, 2002.

[34] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition”, arXiv:1409.1556v6 [cs.CV] 10 Apr 2015.

[35] L. Shuang, B. Liang and Yanming, “Reference Based on Adaptive Attention Mechanism for Image Captioning", IEEE Fourth International Conference on Multimedia Big Data (BigMM), 2018.

[36] M. A. Jishan, “ImagetoText: Image Caption Generation Using Hybrid Recurrent Neural Network”, 2019.

[37] M. Faruk, H. A. Faraby, M. M. Azad, M. R. Fedous and M. K. Morol, “Image to Bengali Caption Generation Using Deep CNN and Bidirectional Gated Recurrent Unit”, 23rd International Conference on Computer and Information Technology (ICCIT), arXiv:2012.12139v1 [cs.CV], 2020

[38] M. Naili, A. H. H. B. Ghezala, “Comparative study of word embedding methods in topic segmentation”, International Conference on Knowledge Based and Intelligent Information and Engineering System, KES2017, 6-8 September 2017, Marseille, France.

[39] M. R. S. Mahadi, A. Arifianto and K. N. Ramadhani, “Adaptive Attention Generation for Indonesian Image Captioning”, 8th International Conference on Information and Communication Technology (ICoICT), 2020.

[40] M. Schuster and K. K. Paliwal, “Bidirectional recurrent neural networks,” IEEE Transactions on Signal Processing, vol. 45, no. 11, pp. 2673–2681, 1997.

[41] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks”, arXiv:1905.11946v5 [cs.LG] 11 Sep 2020.

[42] N. Khumaisu, E. Johanes, S. Sakriani, A. Mirna and N. Satoshi “Corpus Construction and Semantic Analysis of Indonesian Image Description”, The 6th Intl. Workshop on Spoken Language Technologies for Under-Resourced Languages, Gurugram, India, 2018.

[43] O. Vinyals, A. Toshev, S. Bengio and D. Erhan, “Show and Tell: A Neural Image Caption Generator”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3156-3164, 2015.

[44] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, “Enriching Word Vectors with Subword Information”.

[45] P. Li, A. Luo, J. Liu, Y. Wang, J. Zhu, Y. Deng and J. Zhang, "Bidirectional Gated Recurrent Unit Neural Network for Chinese Address Element Segmentation", International Journal of Geo-Information, pp.1-19, 2020.

[46] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions", Transactions of the Association for Computational Linguistics (TACL), pp. 67–78, 2014.

[47] Q. Shiru, X. Yuling and D. Songtao, "Visual Attention Based on Long-Short Term Memory Model for Image Caption Generation", 29th Chinese Control and Decision Conference (CCDC), 2017.

[48] R. Al-Rfou, B. Perozzi, and S. Skiena, "Polyglot: Distributed word representations for multilingual NLP", arXiv preprint arXiv:1307.1662, 2013.

[49] R. Dhir, S. K. Mishra, S. Saha and P. Bhattacharyya, "A Deep Attention based Framework for Image Caption Generation in Hindi Language", Computacion y Sistemas, Vol. 23, No.3, pp. 693-701, 2019.

[50] S. Bannerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments", In: *Proc. of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pp. 65–72, 2005.

[51] S. Hochreiter and J. Schmidhuber. "Long short-term memory". In: Neural Computation 9.8 (1997), pp. 1735–1780.

[52] S. K. Mishra, R. Dhir, S. Saha, and P. Bhattacharyya, "A Hindi Image Caption Generation Framework Using Deep Learning", ACM Trans. Asian Low-Resour. Lang Inf. Process., Vol.20, No. 2, 2021.

[53] S. Lakshminarasimhan, S. Dinesh and A. Amutha, "Image Captioning- A Deep Learning Approach", Internaltional Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, 2018.

[54] S. Parth, B. Vishvajit, and P. Supriya, "Image Captioning using Deep Neural Architectures", International Conference on Innovations in information Embedded and Communication Systems (ICIIECS), 2017.

[55] S. P. S. Gurjar1, S. Gupta1 and R. Srivastava, "Automatic Image Annotation Model using LSTM Approach", Signal & Image Processing: An

International Journal (SIPIJ) Vol.8, No.4, August 2017.

[56] S. R. Sreela and M. I. Sumam, “AIDGenS: An Automatic Image Description System using Residual Neural Network”, International Conference on Data Science and Engineering (ICDSE).

[57] T. Mikolov, S. Kombrink, L. Burget, J. Cernocky, and S. Khudanpur. “Extensions of recurrent neural network language model”. In: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE. 2011, pp. 5528–5531.

[58] T. Mikolov, W. Yih, G. Zweig, “Linguistic Regularities in Continuous Space Word Representations”, Proceedings of NAACL HLT, 2013.

[59] V. Atliha and D. Sesok, “Comparison of VGG and ResNet used as Encoders for Image Captioning", 2020 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream), 2020.

[60] V. Atliha and D. Sesok, “Pretrained Word Embeddings for Image Captioning”, IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream), 2021.

[61] V. Jindal, “Generating Image Captions in Arabic using Root-Word Based Recurrent Neural Networks and Deep Neural Networks”, Proceedings of NAACL-HLT 2018: Student Research Workshop, pages 144–151, 2018.

[62] W. P. Pa and N. L. Thein, "Myanmar Word Segmentation using Hybrid Approach", In: Proc. of 6th International Conf. on Computer Applications, Yangon, pp-166-170, 2008.

[63] X. Li, W. Lan, J. Dong, and H. Liu, “Adding Chinese Captions to Images”, 2016.

[64] Y. Bhatia, A. Bajpayee, D. Raghuvanshi and H. Mittal, “Image Captioning using Google’s InceptionResnetV2 and Recurrent Neural Network”, IEEE, 2019.

[65] Y. Fan, J. Xu, Y.Sun, B. He. “Long-Term Recurrent Merge Network Model for Image Captioning”, IEEE 30th International Conference on Tools with Artificial Intelligence, 2018.

[66] Y. M. S. Sin, W. P. Pa and K. M. Soe, “UCSYNLP-Lab Machine Translation Systems for WAT 2019”, Proceedings of the 6th Workshop on Asian Translation, pp.195-199, 2019.

[67] Y. M. Thu and W. P. Pa, “Myanmar News Headline Generation with Sequence-to-Sequence model”, In: Proc. of the 23rd Conference of the Oriental COCOSDA, Yangon, Myanmar, pp. 117-122, 2020.

[68] Z. Hossain, F. Sohel, M. F. Shiratuddin and H. Laga “A Comprehensive Survey of Deep Learning for Image Captioning”, ACM Computing Surveys, 2018.

[69] Z. Luo, H. Kang, P. Yao and W. Wan, “Chinese Image Caption Based on Deep Learning”, 2016.

[70] https://www.ibm.com/cloud/learn/convolutional-neural-networks.

[71] https://github.com/ye-kyaw-thu/sylbreak.

[72] https://towardsdatascience.com/word-embedding-techniques-word2vec-and-tf-idf-explained.