









University of Computer Studies (Mandalay)

Department of Advanced Science and Technology

Ministry of Science and Technology

The Republic of the Union of Myanmar

JOURNAL OF INFORMATION TECHNOLOGY, RESEARCH AND INNOVATION

June, 2024

JiTRi 2024

JOURNAL OF INFORMATION TECHNOLOGY, RESEARCH AND INNOVATION

June, 2024

Organized by

University of Computer Studies

(Mandalay)

Volume-4, Issue-1

JOURNAL OF INFORMATION TECHNOLOGY, RESEARCH AND INNOVATION

JITRI, 2024

Vol-4, Issue-1

UNIVERSITY OF COMPUTER STUDIES (MANDALAY)

Journal of Information Technology, Research and Innovation (JITRI), 2024 Committee

University of Computer Studies (Mandalay)

Editor in Chief *Prof. Dr. San San Tint, Pro Rector*

Technical Program Committee & Editorial Board

- Prof. Dr. Thin Thin Naing
- Prof. Dr. Aye Aye Chaw
- Prof. Dr. Thandar Aung
- Prof. Dr. Mya Thida Kyaw
- Prof. Dr. Yu Ya Win
- Prof. Dr. Saw Thandar Myint
- Prof. Dr. May Mya Aung
- Prof. Dr. Nwe Nwe Htoon
- Assoc. Prof. Dr. Thein Htay Zaw
- Assoc. Prof. Dr. Yin Min Tun
- Assoc. Prof. Daw San San Lwin
- Dr. May Phyo Thu

Acknowledgements for Reviewers and Organizing members of JITRI

The Journal of Information Technology, Research and Innovation (JITRI) 2024 technical program committee would like to express the deepest appreciation to the external reviewers and organizing members for collaborating to publish this journal successfully. JITRI (2024) is published as the sixth time in order to undertake research in their respective fields of studies. Finally, we would like to thank all the authors who had submitted their research papers by making their great efforts in the Journal of Information Technology, Research and Innovation (JITRI), 2024, Vol- 4, Issue-1.

Table of Contents

Sr. No.	Title	Page
1.	Four Times Image Upscaling from Low-Resolution Input Using Custom Fast Super Resolution Convolutional Neural Network *Min Thway Han, Hlaing Moe Than, Sai Zaw Latt, and Kyaw Kyaw Lin*	1-8
2.	Online Food Order Clustering using DP-Means and K-means Algorithms on Apache Spark *Phyu Phyu Myint, and Thin Thin Yi*	9-13
3.	Cultural Heritage in Myanmar using Knowledge Management System *Hmway Hmway Tar, and Zay Ya Zaw*	14-17
4.	Attention-Based Neural Machine Translation Between Myanmar and English *Phyu Moe Nyunt, and Su Su Win*	18-25
5.	Human Faces Expression Recognition using CNN Model *Phyu Moe Nyunt, and Su Su Win*	26-32
6.	Scrutiny of Optical Extinction Bands of COLLOIDAL Colloidal Ag NPs and Agcore-Cushell Nanoparticles in Different Stabilizing Agent for Stability Evaluation *Thaung Hlaing Win*	33-39
7.	Particles Size and Structural Properties of Colloidal Silver Nanoparticles by Green Synthesis Method *Yin Yin Htwe, Htet Htet, and Thaung Hlaing Win*	40-47
8.	Finetuning Whisper for Domain Specific Multilingual ASR Application *Sai Myat Htoo, and Ei Ei Moe*	48-58
9.	Comparative Analysis of Hash Algorithms Performance in Decentralized Messaging Platform *Ko Ko Min, Hein Tun, Aung Ko Latt, and Nay Lynn*	59-63
10.	Comparative Study of Strong and Soft Reducing Agent Concentrations on Size Evolution of Silver Nanorods *Thiriwin, Myint Myint Kyi, and Thaung Hlaing Win*	64-70

Sr. No.	Title	Page
11.	Effect of Natural Dyes Concentration on Size and Surface Plasmon Properties of Colloidal Silver Nanoparticles *Htet Htet, Yin Yin Htwe, and Htun Win*	71-78
12.	IOT-Based Real-Time Vehicle Tracking and Battery Monitoring System for Electric Vehicles *Nann Sandar Aye, Pa Pa Winn San, and Aye Min Soe*	79-92
13.	Comparative Analysis of U-Net and ResNet-Enhanced U-Net Architectures Evaluating Binary Cross-Entropy and Focal Tversky Loss Functions in Satellite Image Segmentation *Myo Myat Thu, Phone Naing, and Kyaw Kyaw Lin*	93-100
14.	Studying Importance of Encryption and Testing Some Popular Tool *Zin Mar Oo, Theint Theint Htwe, and Myintzu Phyo Aung*	101-106
15.	Recommendation System Based on Apache Spark using Big Data *Naw May Myat Noe, and Myintzu Phyo Aung*	107-115
16.	Rice Leaf Disease Classification with SVM Classifier *Than Than Htwe*	116-121
17.	WEATHER MNITORING SYSTEM BASED ON MACHINE LEARNING WITH RASPBERRY PI *Ngu War Tun, and Atar Mon*	122-128

FOUR TIMES IMAGE UPSCALING FROM LOW-RESOLUTION INPUT USING CUSTOM FAST SUPER RESOLUTION CONVOLUTIONAL NEURAL NETWORK

Min Thway Han[1], Hlaing Moe Than [2], Sai Zaw Latt [3], and Kyaw Kyaw Lin [4]

[1,2,3,4]Department of Computer Technology, High Education Centre (Pyin Oo Lwin)
[1]minthwayhan53@gmail.com,[2]hlaingmoe2008@gmail.com,[3]saizawlatt46@gmail.com,
[4]kklin1500@gmail.com

ABSTRACT

In this paper, we propose the Custom Fast Super Resolution Convolutional Neural Network (CFSRCNN) model. Other FSRCNN models can upscale images four times from tiny to large, but our proposed model can not only process upscaling images four times, but it can also process these images without losing their quality, and the model's processing time is very quick. In this paper, the researcher utilized the proposed architecture and our own dataset for four times the original size upscaling of tiny images. The experimental results show that CFSRCNN is an effective and efficient solution for upscaling small images to four times their original size, making it a valuable tool for a wide range of image enhancement applications.

KEYWORDS

Convolutional Neural Network, Deep learning, FSRCNN, Image restoration, Super-resolution

1. INTRODUCTION

The FSRCNN model's CNN convolutional neural network architecture is designed to efficiently upscale images with minimal computational effort. It consists of layers of convolution and deconvolution that alternate and learn to capture and reconstruct fine image details during the upscaling process [1]. The model's lightweight design enables real-time performance on various platforms, making it suitable for real-world applications, including image restoration, video upscaling, and other related tasks.

In this research, the FSRCNN model is trained on a diverse dataset of low-resolution images and corresponding high-resolution actual data. The training process leverages loss functions, such as mean squared error, to minimize the discrepancy between the upscaled output and the ground truth. During inference, the trained FSRCNN model effectively upscales small input images by four times, significantly improving their visual quality and revealing finer details not present in the original low-resolution versions.

Experimental results demonstrate its superiority over traditional interpolation-based methods and competitive performance compared to more complex super-resolution models, all while requiring significantly reduced computational resources. The findings highlight FSRCNN's effectiveness as an efficient solution for upscaling small images, rendering it a valuable tool for enhancing image quality across various applications.

The aim of using Custom Super-Resolution Convolutional Neural Network architecture is to describe a method for improving image quality by increasing resolution using a proposed CFSRCNN, particularly enhancing low-resolution inputs fourfold.

In this paper, the next section 2 describes about the related works of the proposed system and then section 3 presents about research methodology of the proposed system. In section 4, the own dataset is presented and

model training and model testing are discussed at section 5 and 6. Finally, the system is concluded in section 7.

2. Related Works

2.1. Image Super-Resolution Using Deep Learning

The FSRCNN (Fast Super Resolution Convolutional Neural Network) is a deep learning model designed for image super-resolution tasks, particularly upscaling low-resolution images. Several other models like EDSR (Enhanced Deep Super-Resolution), ESPCN (Efficient Sub-Pixel Convolutional Neural Network), and LapSRN (Laplacian Pyramid Super-Resolution Network) also target similar objectives [2]. The mentioned models, EDSR, ESPCN, LapSRN and FSRCNN all contribute to the field of image super-resolution. EDSR stands out for its high accuracy, yet it suffers from slow processing speed and large file sizes. ESPCN, while tiny and fast, lags behind in visual performance compared to newer models. FSRCNN excels due to its speed, compact size, and reasonable accuracy. Notably, the FSRCNN-small variant sacrifices some accuracy for even faster performance. LapSRN offers multi-scale super-resolution with a single pass, accommodating various scaling factors efficiently. However, it's slower than ESPCN and FSRCNN, and its accuracy falls short of EDSR's standards.

Among these models, FSRCNN emerges as a compelling choice for upscaling images. It strikes a balance between speed, accuracy, and size, making it suitable for real-time applications and situations with resource constraints. While EDSR, LapSRN, and ESPCN possess their respective strengths, FSRCNN's combined advantages make it a valuable option for image upscaling tasks that require both efficiency and reasonable visual quality [3].

2.2. FSRCNN for Image Super-Resolution

The objective of image super-resolution is to reconstruct high-resolution images from low-resolution inputs [4]. Dong et al. first introduced the SRCNN in 2014, marking the use of convolutional neural networks in the field of super-resolution of images, which resulted in significant improvements in image enhancement [1]. One notable limitation of SRCNN is its reliance on learning mappings from bicubic interpolated low-resolution images rather than their original low-resolution counterparts [8]. Nevertheless, SRCNN's use of a 5x5 convolution kernel for learning nonlinear mappings imposes restrictions on the network's learning capacity within this context [6]. To address these limitations, a range of approaches were proposed by Dong et al. and Kim et al., such as the Extended Super Resolution Convolutional Neural Network (SRCNN-ex) and the FSRCNN [1].

In light of these developments, the utilization of FSRCNN emerges as particularly compelling for image upscaling. By addressing the limitations of SRCNN, FSRCNN offers a refined solution. It not only reduces computational complexity by utilizing a compact model, but it also leverages a more efficient sub-pixel convolutional structure. This results in a network that strikes a balance between computational efficiency and accurate image restoration. FSRCNN's evolution from the pioneering SRCNN exemplifies the continuous refinement and optimization of deep learning techniques for the task of image super-resolution. Figure 1 shows the differences in network architecture between FSRCNN and SRCNN [1].

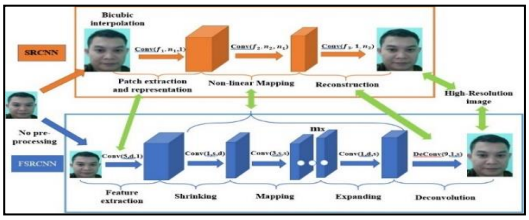


Figure 1. SRCNN and FSRCNN's Network Architectures

3. Methodology

3.1. Fast Super-Resolution Convolutional Neural Networks (FSRCNN)

Dong et al. (2016) introduced the FSRCNN, which aims to speed the SRCNN that was

previously suggested [1]. In contrast to SRCNN, FSRCNN can learn the mapping from the original LR picture to the HR image without the need for a deconvolution layer by adding one at the network's the end [7]. The aforementioned techniques have helped to minimize the overall network's calculating costs to some extent. Additionally, in this improved non-linear mapping appearance, FSRCNN adds a decreasing layer at starting point and an increasing layer at the conclusion, respectively (He et al., 2016) [1]. A smaller filter (size=3*3) is used in the nonlinear layer by FSRCNN [1]. The entire network is built as a small hourglass-shaped CNN structure, which consists of five steps: feature extraction, shrinking, non-linear mapping, expanding, and deconvolution [1]. Convolution layers make up the first four portions, while deconvolution layers make up the final component. For ease of comprehension, the term "convolution layer" is used. as Conv (f_i; n_i; c_i) and a deconvolution layer as DeConv (f_i; n_i; c_i) wheref_i; n_i; c_i stand for the filter size, the number of filters, and the channel count, respectively [1]. We are unable to look at every variable in the network because there are tens of them ($\{f_i; n_i; c_i\}\ \sigma_{i=1}$) [1]. As a result, a fair value is given to the insensitive variables in advance while leaving the sensitive variables unset. When a little change in a variable may have a large impact on how well something performs, that variable can be referred as sensitive. These sensitive variables always signify some significant SR-influencing elements, as will be seen from the explanations that follow.

Feature Extraction: Feature extraction is the initial step in the FSRCNN process, where the low-resolution input image is analyzed to capture important patterns and features. FSRCNN adopts a convolutional layer with smaller filters to perform feature extraction [1]. These filters scan the input image to identify low-level features such as edges, textures, and simple shapes. This step's objective is to create a concise illustration of the essential information from the input image that can be further processed for super-resolution.

Shrinking: In the shrinking step, a filter of size 1*1 is used by the shrinking layer to change the LR feature dimension from d to s, where d is the size of the features in the LR image after feature extraction and s is the number of filters ($s \leq d$) [1]. This is achieved by convolving the feature maps (generated in the previous step) with a set of smaller filters. The application of these smaller filters reduces the feature maps' dimensions while retaining important information. The shrinking process helps reduce the network's computational complexity while capturing significant features efficiently.

Mapping: The mapping step is where the magic of super-resolution happens in FSRCNN [1]. After the shrinking stage, the network utilizes multiple convolutional layers to learn non-linear mappings from the low-resolution feature space to the high-resolution feature space. These layers are responsible for capturing complex relation-ships between low-resolution and high-resolution features. In essence, this mapping process learns how to reconstruct higher-resolution information from the extracted lower-resolution features.

Expanding: After mapping, the expanding step takes place. Here, deconvolutional layers (also known as transposed convolutional layers) are used to upsample the features back to a higher resolution. Deconvolutional layers are applicable to enlarge the feature maps while filling in the missing details based on the learned mappings from the previous step [1]. The result is that the expanding step helps reconstruct a HR representation of the image.

Deconvolution (Up-sampling): Deconvolu-tional layers, used in the expanding step, are the reverse of standard convolutional layers [1]. While convolutional layers take an input and apply filters to downsample or extract features, deconvolutional layers take a lower-resolution input and use filters to upsample and reconstruct higher-resolution feature maps. They achieve this by applying the transpose of the convolution operation.

3.2. Model Architecture

Our proposed CFSRCNN model architecture consists of four convolutional layers which shown in Table 1. It takes an input image of shape (None, None, 3), where the first two dimensions can be of any size, and the last dimension represents the three-color channels (RGB). And we created a 2D convolutional layer with 56 filters, a 5x5 kernel size, and the same padding. The activation function used is ReLU. This layer is utilized as the input layer, effectively connecting the input to this convolutional layer. And we created another 2D convolutional layer with 16 filters,1x1 kernel size, and the same padding. The activation function used is ReLU. The output from the preceding layer is contained in this layer.

This adds another convolutional layer to the model. And we create a 2D transposed convolutional layer (also known as deconvolution) with 16 filters, a 9x9 kernel size, 4 strides, and the same padding. The activation function used is ReLU. In this layer, we add the transposed convolutional layer to the model. We create another 2D transposed convolutional layer with 3 filters, a 5x5 kernel size, and the same padding. Since no activation function is specified, it defaults to linear activation (no activation). In this layer, we add the last transposed convolutional layer to the model. The last Lambda layer that applies the last output transposed convolutional layer must ensure the output values are within the valid range for image pixel values (0 to 255).

Table 1. Proposed CFSRCNN Model Architecture

Layer (type)	Output Shape	Parameters
Input_1 (input layer)	(None, None, None, 3)	0
conv2d (Conv2D)	(None, None, None, 56)	4256
conv2d_1 (Conv2D)	(None, None, None, 16)	912
conv2d_ transpose (Conv2DTranspose)	(None, None, None, 16)	20752
conv2d_ transpose_1 (Conv2Dtranspose)	(None, None, None, 3)	1203
lambda (Lambda)	(None, None, None, 3)	0
Total parameters: 27,123		
Trainable parameters: 27,123		
Non-trainable parameters: 0		

4. DATASET

In this experiment, we used the own dataset that included 30 different kinds of face shapes of 42 men, the collection has 1260 images in total. Figure 2 shown 30 different images of one person from a dataset of 42 people. Figure 3 shows the 42 individual image samples from the dataset.

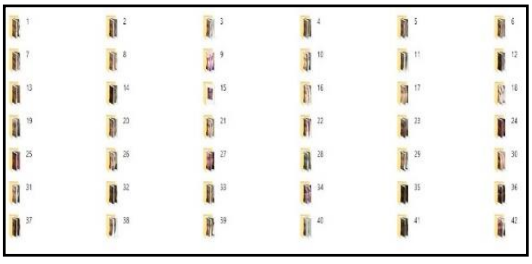


Figure 2. The Dataset of 42 People

Figure 3. Describing 30 images of a person with different facial shapes from a dataset of 42 people

5. MODEL TRAINING

For this model training, the own architecture was used by us. Figure 4 shown the model training diagram step-by-step and the explanations that follow.

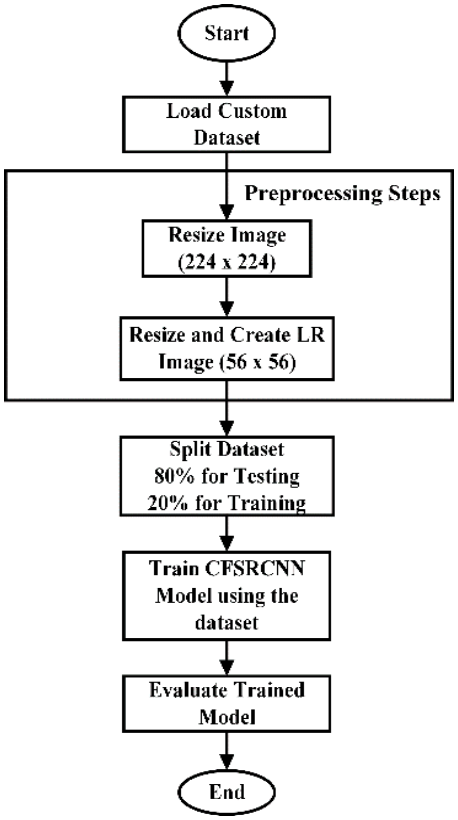


Figure 4. Model Training Diagram

5.1. Training Data

For this training, the researcher used a dataset of low-resolution images and their corresponding high-resolution images. During the pre-processing stage, the various input image sizes are downsized to 224x224 images, and then these resized images are again resized to create 56x56 low-resolution images. The researcher used 20% of the images for testing, and 80% of the images for training.

5.2 Model Training

The model was trained using the Adam optimizer and Mean Squared Error (MSE) loss. The training is performed for 1500 epochs with a batch size of 32.

Adam Optimizer: The optimization algorithm Adam (Adaptive Moment Estimation) commonly utilized during training to update the neural network weights [9]. It combines the ideas of both the RMSprop and AdaGrad algorithms and introduces momentum to speed up convergence. In deep learning applications, the Adam optimizer is frequently utilized because it works well with complex, large-scale models. The Adam optimizer's update rule is as follows:[9]

$$m_t = \beta_1 . m_{t-1} + (1 - \beta_1) . g_t$$

$$v_t = \beta_2 . v_{t-1} + (1 - \beta_2) . g_t^2$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t}$$

$$\theta_t = \theta_{t-1} - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} . \hat{m}_t$$

Where θ_t is the vector of weights at iteration t, g_t is the loss's gradient with respect to θ at iteration t, η is the learning rate, which controls the step size during weight updates, ,β_1 and β_2 are hyperparameters that determine the first and second moments' exponential decay rates, respectively, m_t and v_t are the gradients' first and second moments, respectively, $\widehat{m_t}$ and $\widehat{v_t}$ are bias-corrected estimates of m_t and v_t , ϵ is a small constant (usually a very small value) to avoid division by zero. The Adam optimizer is an algorithm for optimizing adaptive learning rates that effectively combines momentum and adaptive scaling of learning rates, being therefore suitable for effective and efficient neural network training. [9]

Mean Squared Error (MSE) Loss: For regression tasks, Mean Squared Error is a widely used loss function. The difference between actual (ground truth) and anticipated values is calculated using the average squared difference. With regard to deep learning, MSE is often used as a measure of how well a neural network is performing on a given task, and the goal during training is to minimize this loss. The MSE loss is calculated as follows:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

Where n is data points number (samples), y_i is the true value (ground truth) of the i-th sample, $\hat{y}_i$ is the predicted value for the i-th sample.

Each data point's loss is calculated individually and then averaged over all the data points to obtain the final MSE value. The main advantage of MSE is that it

penalizes large errors heavily, making it particularly sensitive to outliers.

Checkpoints: Model checkpoints are saved during training, and the best model (based on validation loss) is saved with a unique filename containing the epoch number and validation loss. After training 1500 epochs using CFSRCNN, the best result is achieved at epoch 1388, which is shown in table 2.

Table 2. The Best Result of Trained Model

Dataset	Epoch	Validation Loss	Validation Accuracy
Own Dataset	1388 / 1500	19.9093	0.9747

Evaluation: After training, the model is evaluated on the test set using the evaluation method. The evaluation result includes validation loss and validation accuracy.

6. Model Testing

For this model testing, the researcher used best-trained model. Figure 5 shown the model testing diagram step-by-step and the explanations that follow.

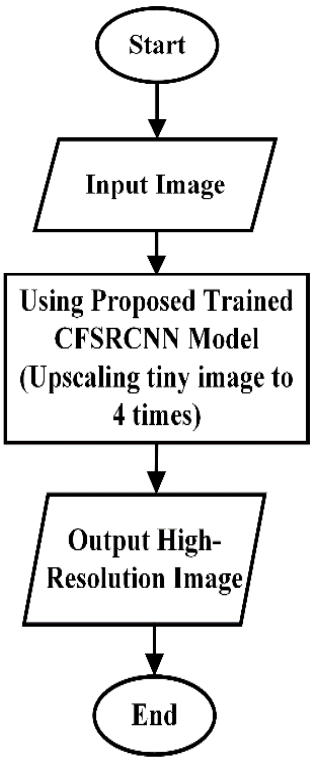


Figure 5. Model Testing Diagram

Input Image Pre-processing: The testing image is first pre-processed by converting it from BGR to RGB format.

High-Resolution Image Prediction: The FSRCNN model is used to generate a high-resolution version of the low-resolution image. And the high-resolution (HR) output image is obtained up to four times the input image.

7. Experimental Result

To carry out this experiment, the researchers use a laptop with a Ryzen 9 (5900HX) processor, 32 GB of RAM, and a 3050Ti GPU (4GB) with the tensorflow_gpu (2.10.0) framework. In this study, the researchers use first an 800x800 image, and then resize this image to 200x200 for the input image.

The researcher changed this resized 200x200 input image to an 800x800 upscaling output image by using the proposed CFSRCNN architecture. Researchers compared the final upscaled super resolution image to the original image in my experimental results using the PSNR, SSIM, and MSE methodologies.

The PSNR formula is used to compute a ratio of the highest signal strength to the amount of power of the distorted noise that impairs the accuracy of its representation. Decibel form is used to calculate this ratio between the two images. The degradation of a picture is regarded within the SSIM (Structural Similarity Index Method) framework as a shift in the perceptual comprehension of its structural elements. By evaluating the similarities of two images the source image and the restored version SSIM measures the quality that is perceived of pictures and videos.

Among image quality measurement metrics, Mean Squared Error (MSE) stands as the most prevalent. Operating as a full-reference standard, it signifies higher precision when its values approach zero. As a comprehensive benchmark, lower MSE values correspond to heightened quality [5][10]. Figure 6 shows (a) the input 200x200 image and (b) the output 800x800 image. Table 3 shown the comparison results and processing time of PSNR, SSIM and MSE.

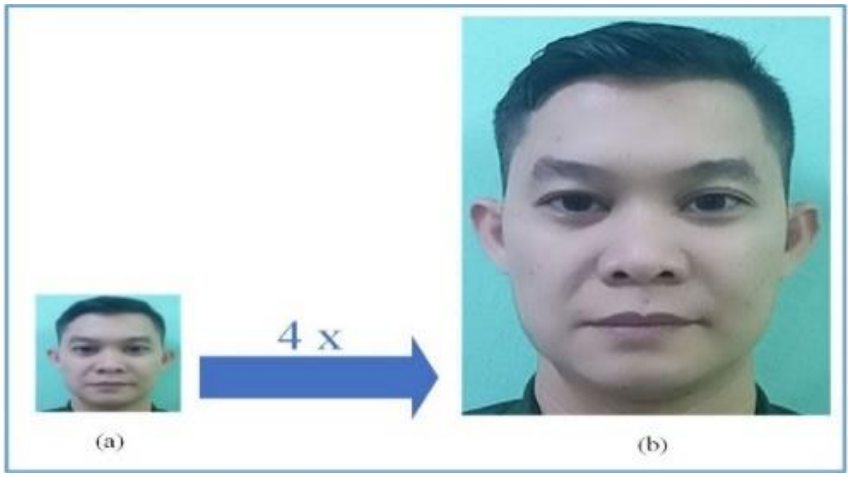


Figure 6. Upscaling 4 Times from 200x200 to 800x800 Image

Table 3. The comparison results and procession time of PSNR, SSIM and MSE

Input Image	Super Resolution	Ground Truth	Scale	Comparison SR and GT			Processing Time
				MSE	SSIM	PSNR	
50x50	200x200	200x200	4x	55.28	0.99	30.70 dB	0.1s
200x200	800x800	800x800	4x	4.32	1.00	41.78dB	0.4s

8. CONCLUSIONS

In summary, the researcher's own Custom Fast-Super-Resolution Convolutional Neural Network (CFSRCNN) architecture has proven to be a highly efficient and effective solution for 4x image upscaling from low-resolution inputs. The network's fast processing speed and computational efficiency make it appropriate for real-time and resource-constrained applications. So, this architecture can be used in surveillance areas, home security, and restricted areas for the detection of expected people and thieves by using CCTV. The success of this FSRCNN model represents a significant development in image super-resolution, enhancing both computer vision and image processing technology.

ACKNOWLEDGEMENTS

I extend my heartfelt gratitude to my supervisors, Dr. Hlaing Moe Than, Dr. Sai Zaw Latt, and Dr. Kyaw Kyaw Lin. Their invaluable guidance and unwavering support throughout my research journey have been transformative. Their expert advice and encouragement shaped this paper's success. Their dedication and mentorship fuelled my motivation. I'm privileged to work with such exceptional individuals who played a pivotal role in my academic growth. My sincere thanks for their boundless energy and genuine interest in my success.

REFERENCES

[1] Chao Dong, Chen Change Loy, and Xiaoou Tang, "Accelerating the Super-Resolution Convolutional Neural Network", arXiv: 1608. 00367v1 ,1 Aug 2016.

[2] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken " Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network ", arXiv: 1609. 05158v2, 16 Sep 2016.

[3] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah and Kyoung Mu Lee, "Enhanced Deep Residual Networks for Single Image Super-Resolution", arXiv: 1707.02921v1 , 10 Jul 2017.

[4] W. -S. Lai, J. -B. Huang, N. Ahuja and M. -H. Yang, "Deep Laplacian Pyramid Networks for Fast and Accurate Super-Resolution," 2017 IEEE

Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, pp. 5835-5843, doi: 10.1109/ CVPR. 2017. 618.

[5] Umme Sara, Morium Akter, Mohammad Shorif Uddin, “Image Quality Assessment through FSIM, SSIM, MSE and PSNR—A Comparative Study”, 10.4236/ jcc.2019. 73002, March 4, 2019.

[6] S. Schulter, C. Leistner and H. Bischof, "Fast and accurate image upscaling with super-resolution forests", 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 2015, pp. 3791-3799, doi: 10.1109/ CVPR.2015.7299003.

[7] C. K S, R. R. G, S. T N and C. Gururaj, "Image Super-Resolution Using Convolutional Neural Network," 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon), Mysuru, India, 2022, pp. 1-7, doi: 10.1109/ MysuruCon 55714.2022. 9972459.

[8] Dong, C., Loy, C.C., He, K., Tang, X. (2014). Learning a Deep Convolutional Network for Image Super-Resolution .2014. Lecture Notes in Computer Science, vol 8692. Springer, Cham. https:// doi.org/10. 1007 / 978-3-319-10593-2_13.

[9] Diederik P. Kingma, Jimmy Ba, "Adam: A Method for Stochastic Optimization", arXiv:1412. 6980v9 ,30 Jan 2017.

[10] J. Erfurt, C. R. Helmrich, S. Bosse, H. Schwarz, D. Marpe and T. Wiegand, "A Study of the Perceptually Weighted Peak Signal-To-Noise Ratio (WPSNR) for Image Compression," 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 2019, pp. 2339-2343, doi: 10.1109/ ICIP.2019.8803307.

Authors

Biography

Min Thway Han received M.Sc (Radio Engineering) degree from Moscow Power Engineering Institute (MPEI), Russian Federation in 2016. Now he is a Ph.D. candidate of Computer Technology Department, High Education Center (HEC).

Dr. Hlaing Moe Than received the M.I.C.Sc. degree from the Moscow Institute of Electronic Technology (MIET), Russian Federation in 2008. He continued his studies and got his Ph.D. (Computer Science) in 2014. He is now serving as an assistant lecturer at the Department of Computer Technology at the Higher Education Center (HEC), Pyin Oo Lwin.

Dr. Sai Zaw Latt earned his master degree from Moscow Power Engineering Institute (MPEI), (Russian Federation) in 2009. He obtained his Ph.D. (Computer Technology) from HEC in 2015. Now he is a lecturer of Department of Computer Technology in HEC.

Dr. Kyaw Kyaw Lin received the M.I.C.Sc. and Ph.D. (IT) from the Moscow Institute of Electronic Technology (MIET), Russian Federation in 2008 and 2017. He has served as an assistant lecturer at the Department of Computer Technology at the Higher Education Center (HEC), Pyin Oo Lwin.

ONLINE FOOD ORDER CLUSTERING USING DP-MEANS AND K-MEANS ALGORITHMS ON APACHE SPARK

Phyu Phyu Myint[1], and Thin Thin Yi[2]

[1,2] Faculty of Computer Science, University of Computer Studies (Kyaing Tong)
[1]phyumyint.ppm@gmail.com,[2]thinthinyee.pku@gmail.com

ABSTRACT

With the burgeoning popularity of online food ordering platforms (such as Food Panda, Grab, etc.), understanding customer behaviour and preferences becomes paramount for service providers to enhance user experience and optimize business operations. This study proposes a scalable approach for clustering online food orders using K-Means and DP-Means algorithms implemented on the Apache Spark framework. The suggested solution effectively processes massive volumes of order data in real-time by utilizing Spark's distributed computing capabilities, providing fast insights for corporate decision-making. The comparative analysis between K-Means and DP-Means provides insights into the trade-offs between conventional centroid-based clustering and the non-parametric nature of threshold (lambda)-based clustering. The outcomes of the experiments demonstrate the scalability and effectiveness of the methods in identifying important, meaningful food order clusters, which led to customized recommendations and targeted marketing strategies in the virtual food industry.

KEYWORDS

K-Means, DP-Means, Clustering, Apache Spark

1. INTRODUCTION

The development of online meal ordering platforms has brought about a revolution in the food service sector, offering consumers an unprecedented level of convenience and decision-making, while also posing novel difficulties and prospects for businesses. Identifying the underlying trends and preferences in customer habits is crucial for these platforms because it may guide the development of individualized marketing campaigns, operational optimizations, and tailored suggestions. In this situation, clustering analysis proves to be a useful tool since it makes it possible to identify different consumer base groups or segments based on ordering patterns.

In addition to their effectiveness and simplicity, traditional clustering methods like K-Means have found widespread application in this domain. But, especially in high-dimensional or noisy datasets, they depend on an expectation of pre-defined cluster centroids, which might not necessarily coincide with the underlying data distribution. An alternate method is provided by threshold-based clustering algorithms, such DP-Means, which define clusters based on local density estimation. This method provides greater flexibility and robustness against outliers and unevenly formed clusters.

The current research proposes a unique method for clustering online food orders utilizing the Apache Spark framework and a combination of K-Means and DP-Means algorithms. The distributed computing architecture offered by Apache Spark makes it possible to handle vast amounts of data effectively in parallel, which makes it an excellent choice for managing the enormous amounts of order data that online food platforms create in real time. The suggested solution seeks to draw significant conclusions from the data by utilizing Spark's scalability and fault tolerance, enabling better decision-making for food service providers.

This study has two main goals: first, it will investigate whether distinct online food order clusters can be determined using traditional centroid-based grouping (K-Means) and

threshold-based clustering (DP-Means). Secondly, it will assess the performance and scalability of these algorithms when they are implemented on the Apache Spark platform. The suggested system aims to identify the advantages and disadvantages of each strategy and offer insights into their real-world applications for the online food business through empirical research and comparative assessment. The suggested system's ultimate objective is to equip food service providers with useful information gleaned from their order data, allowing them to improve customer happiness, streamline operations, and spur business expansion in the cutthroat internet market.

This paper is organized mainly in the sections that follow: The relevant research on a thorough examination of clustering algorithms on diners' meal orders is covered in Section 2. An overview of the dataset and its preparation is given in Section 3. The experimental work's methodology is covered in Section 4, which provides a detailed description of the suggested system, procedure, and algorithm. The paper's conclusion, included in Section 5, offers insights into the system's future direction.

2. RELATED WORK

The growing significance of decisions based on data in the food service business has led researchers and industry practitioners to focus heavily on clustering analysis in the online food ordering space. Numerous clustering techniques have been used in this field, such as the well-known centroid-based K-Means and density or threshold-based DP-Means approaches. However, more research and assessment of these techniques is still necessary in the context of massive amounts order data and distributed computing situations.

Previous research efforts have shown that K-Means clustering can be used to segment customers according to their ordering behaviour and preferences. The author in [1] used K-Means clustering to identify different customer segments in the online food delivery industry, which allowed for personalized recommend-ations and targeted marketing campaigns. The study [2] used K-Means clustering to analyse food preferences and dietary habits among customers of an online food delivery platform, which revealed insightful information for menu optimization and retention techniques.

The capacity of threshold-based algorithms for clustering, such as DP-Means, to automatically identify the number of clusters and manage unevenly shaped data distributions, on the other hand, has drawn attention. Although DP-Means has not been utilized much in the online food ordering arena, its flexibility and robustness make it a viable option for investigative data analysis and pattern recognition in this field.

Furthermore, clustering algorithms are now more scalable and parallelizable because to the advent of distributed computing frameworks like Apache Spark, which makes it possible to analyze large datasets in real time with effectiveness. Apache Spark has been utilized in several studies to cluster workloads in various sectors, demonstrating its ability to manage large-scale big data analytics. For instance, Zhang et al. (2018) showed how the platform can manage massive volumes of transaction data and generate insights that can be used to improve business operations by segmenting clients in the e-commerce industry using Spark-based clustering techniques.

The literature currently lacks a comparative examination of the K-Means and DP-Means clustering algorithms, despite these advancements. This is especially true when comparing the techniques' application to distributed computing platforms like Apache Spark and data from online food orders. This study [3] intended to bridge this gap by conducting a thorough evaluation of these tactics and assessing their benefits, drawbacks, and practical applications for food service enterprises operating in online markets. Through empirical evaluation and comparative analysis, the proposed system aims to offer new insights and methods to the growing body of research in this field, ultimately promoting economic success in the online food market and enhancing user experience.

3. DATASET

This study uses a 329-user dataset with 12 distinct behaviors that was downloaded from Kaggle. The dataset can be tabulated (in Excel or CSV, for example), where columns represent the various qualities that are covered

below, and rows represent alternative ordering. Larger datasets may also be stored in relational databases or distributed data storage systems.

When working with such a dataset, pre-processing techniques such as data cleaning, feature engineering, ETL process, and normalization may be necessary to ensure data quality and prepare the data for clustering analysis using K-Means and DP-Means algorithms. In order to improve the quality and performance of the algorithms, the proposed system first performs ETL pre-processing and data cleaning.

The first step in the ETL process is extracting data and data might come from: Order data: Information about what customers ordered, Customer data: Customer profiles, including location addresses. Data cleaning for handling missing values: If some data points are missing, remove rows with missing data. Removing duplicates: Make sure there are no duplicate records, such as repeated customer orders or duplicate entries in the database.

Table 1. Dataset Attributes

Attribute	Data Type
Age	Integer
Gender	Categorical
Marital Status	Categorical
Occupation	Categorical
Annual Income	Integer
Education	Categorical
Family Size	Integer
Latitude	Float
Longitude	Float
Spending Score	Integer

4. PROPOSE SYSTEM AND METHODOLOGY

The recommended procedure for clustering online food orders on Apache Spark using K-Means and DP-Means is described below:

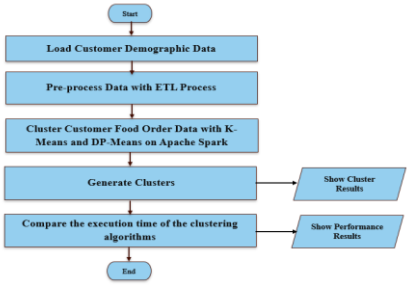


Figure 1. Flowchart of the Proposed System

4.1. Data Collection and Pre-processing

The proposed method obtains the online meal order dataset from Kaggle, which is a relevant source. After that, preparation is applied to this dataset in order to eliminate any missing or inaccurate values, standardize data formats, and address outliers as necessary.

4.2. ETL Process and Analysis

The ETL process is a versatile method commonly employed in data management and analytics and also ensures data integrity in the warehouse through standardization and the removal of redundant entries [4]. This process automates selecting, collecting, and conditioning data from a data warehouse, ensuring the output data is formatted optimally for further processing or business purposes [4]. Different tests served specific functions, revealing a research gap in advanced analysis techniques for distributed data sources [5, 6]. The proposed system uses ETL process and shows the visualization of the data using plots, histograms, and other graphical representations to identify potential clusters or trends.

Scale numerical features to a range (0, 1) so that no single feature dominates the clustering algorithm. For example, if the total order value is in thousands while the number of orders is a single-digit number, normalizing these features ensures they have equal weight. If the data contains categorical variables (e.g., gender, marital status, occupation, education), the system converts these into numerical values.

```
#CSV file
df = pd.read_csv("../input/mall-customerscsv/Mall_Customers.csv")
display(df.head())
print(df.shape)
```

	CustomerID	Gender	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

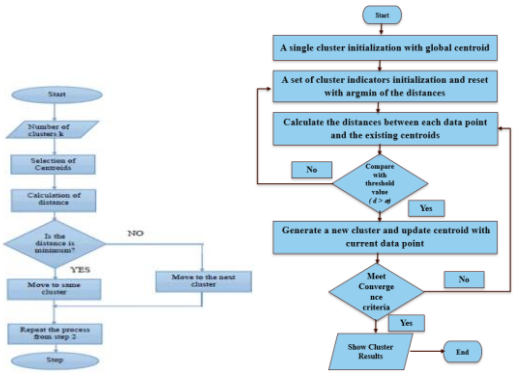
```
(200, 5)
```

Figure 2. Reading Customer Data with Selected Attributes

4.3. Clustering Algorithms

The recommended approach chooses the most appropriate clustering methods based on the type of data, the requirement for scalability, and the desired cluster properties. This system

selects K-Means and DP-Means as the primary clustering methods to compare due to their distinctions in threshold-based and centroid-based clustering.



(a) (b)

Figure 3. Flowchart of (a) K-Means and (b) DP-Means Algorithms

4.4. Implementation on Apache Spark

The proposed system used distributed computing capabilities of Apache Spark by setting up an environment and processing large amounts of data. Then, using Spark's MLlib library or other similar libraries, this system distributes the dataset among Spark's cluster nodes and uses the K-Means and DP-Means clustering algorithms in order to parallelize the calculation and maximize performance.

4.5. Model Training and Evaluation

The K-Means and DP-Means clustering models are then trained using the preprocessed dataset, with method settings (such the number of clusters and convergence criteria) adjusted as needed. This system uses appropriate metrics such as silhouette score, Davies-Bouldin index, or within-cluster sum of squares (WSS) to assess the quality of the generated clusters and evaluates the trained models by comparing the performance of K-Means and DP-Means in terms of clustering accuracy, scalability, and robustness to various data distributions. Figure 5 compares the execution times (ms) of two clustering techniques.

The performance of clusters in K-Means and DP-Means clustering can be evaluated using several metrics. These metrics generally assess how well the clustering algorithm has grouped similar data points together and how well-separated the clusters are. Inertia (Within-Cluster Sum of Squares - WCSS) measures how well the clusters have been formed based on the sum of squared distances of samples to their closest cluster centre. Lower inertia implies better clustering, but you need to balance it with the number of clusters.

Table 2. Cluster Validation Scores

Cluster Validation Score in K-Means	Cluster Validation Score in DP-Means
Silhouette Score:1.0 **Calinski-Harabasz Index:**1.0 **Davies-Bouldin Index:**0.0	**Silhouette Score:** 0.9963321725628299 **Calinski-Harabasz Index:** 1004642.5300620429 **Davies-Bouldin Index:** 0.006481393982631485

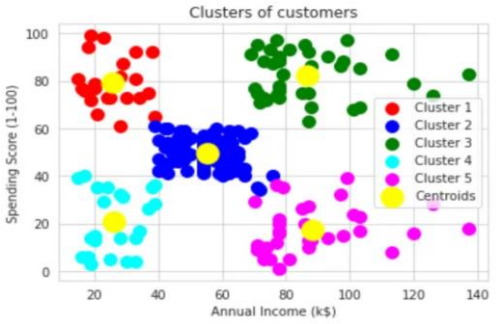


Figure 4. Customer Clusters Based on Annual Income and Spending Score

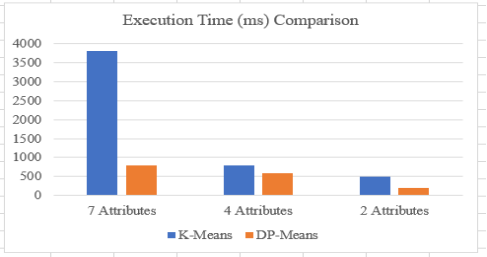


Figure 5. Execution Time Comparison of Two Clustering Algorithms

5. CONCLUSION

This research proposed a novel approach to cluster online food orders using K-Means and DP-Means algorithms from the Apache Spark framework. Leveraging the scalability and parallelism of Spark's distributed computing environment, we conducted a thorough analysis of several clustering algorithms, evaluating their effectiveness, scalability, and practicality for the online food industry. The results of the experiment demonstrated that both the K-Means and DP-Means algorithms could effectively identify significant clusters from the online meal order dataset.

REFERENCES

[1] E. Omol, "Application Of K-Means Clustering For Customer Segmentation In Grocery Stores In Kenya", International Journal Of Science Technology & Management, January 2024, 5(1):192-200, DOI:10.46729/ijstm.v5i1.1024

[2] J. S. Pasaribu, "Appilication of K-Means algorithm to predict consumer interest according to the season on place reservation and food online software", Journal of Physics: Conference Series 1477 (2020) 032004 IOP Publishing doi:10.1088/1742-6596/1477/3/032004

[3] R. Galici, L. Ordile, M. Marchesi, A. Pinna, "Applying the etl process to blockchain data. prospect and findings", Information, vol. 11, no. 4, pp. 204, MDPI

[4] K. Geetha, Param, "Data Analysis and ETL Tools in Business Intelligence", International Research Journal of Computer Science (IRJCS), vol. 7, pp. 127-131.

[5] A. A. Yulianto, "Extract transforms load (ETL) process in distributed database academic data warehouse. APTIKOM", Journal on Computer Science and Information Technologies, vol. 4, no. 2, pp. 61-68, 2019

[6] G. G. W. Mhon, "ETL Preprocessing with Multiple Data Sources for Academic Data Analysis", 2020 IEEE Conference on Computer Applications (ICCA), February 27-28, 2020.

AUTHOR BIOGRAPHY

Phyu Phyu Myint (Associate Professor) is currently working in the Faculty of Computer Science at the University of Computer Studies (Kyaing Tong). She has been received Master of Information Science (M.I.Sc) degree from University of Computer Studies, Mandalay in 2007. she has more than 19 years of experiences in teaching.

COAUTHOR BIOGRAPHY

Thin Thin Yi is an associate professor in the Faculty of Information Sciences at the University of Computer Studies (Pyay). She received her master degree (M.I.Sc) from University of Computer Studies (Mandalay) in 2007. have more than nineteen years of experiences in teaching.

CULTURAL HERITAGE IN MYANMAR USING KNOWLEDGE MANAGEMENT SYSTEM

Hmway Hmway Tar[1], and Zay Ya Zaw[2]

[1,2] University of Information Technology, Mottama Holding
[1]hmwaytar@uit.edu.mm, [2]zayyazaw9@gmail.com

ABSTRACT

Myanmar is a rich cultural heritage spanning thousands of years. Because of this rich and diverse cultural heritage, it can boost the country's economic assets. The proposed system applies the strength of an ontology of semantic-based level for knowledge management. The system constructs a robust and effective ontology that accurately represents the knowledge about the Cultural Heritage in Myanmar, Bagan pagoda. The implementation can greatly enhance cultural knowledge. By providing a structured framework that enables more effective management of cultural heritage data for future generations. An ontology can represent various pagodas in Bagan. This enables a rich understanding of the heritage and history of Bagan. For tourists, a semantic-based ontology can enhance the search experience by providing contextually relevant information based on visitor queries about specific pagodas or themes like "Buddhist art" or "historical events". The outcomes indicate not only accessibility but also flexibility. The experimental results of the ontology for pagodas in Bagan, Myanmar, is functional and user-friendly and a valuable resource for knowledge discovery and community engagement.

KEYWORDS

Semantic, Ontology, Knowledge management

1. INTRODUCTION

There are some key elements of Myanmar's cultural heritage such as Buddhist influence, Traditional art, and crafts, Traditional Festivals, Ethics diversity, Traditional dress, Traditional dance and music and Traditional cuisine. Among them the proposed system mainly impacts on Buddhist influence sector. The system is applied in the *Bagan Archaeological Zone* where Bagan is an ancient cited which was located in the Mandalay Region, Myanmar. It has over *10,000 Buddhist temples*, pagodas, and monasteries in Southeast Asia. That reveals that Myanmar is a rich cultural heritage.

The objective of the proposed system is to manage digitization of cultural heritage applying ontology. The use of technology to counteract the issue of manual documents. The system uses of digital technologies to digitize and preserve cultural documents, and other forms of knowledge. This allows for easier access and dissemination of cultural heritage information to a wider audience. In an increasingly complex digital world, the need for digitization is required –than traditional one. Domain ontology emerges as a powerful tool to address this challenge by providing a formal representation of knowledge specific to a particular domain. It organizes concepts and their relationships within that domain, facilitating consistent understanding, data sharing, and interoperability among various systems

A knowledge base is a centralized repository of information, data, and resources that is accessible to users for reference, learning, and problem-solving. It is a valuable tool for organizations and businesses to store and share information with employees, customers, and etc.

2. METHODOLOGY

The domain ontology serves as a foundational building block for knowledge management and digital applications. Ontological Engineering is still a relatively immature discipline; each work group employs its own methodology. [1]. In the work made by the PROTÉGÉ group, a method ontology is

intended to be part of the ontology library besides the domain ontology, and the link with the application ontology is handled by explicit mapping rules acting as “mediators”; in the KADS group this link is handled by an indexing mechanism. [2]. This paper presents the study on phase three which involved the development and evaluation of the pagoda metadata schema, resulting in a metadata schema which comprised of 14 main data elements and 68 data elements in total. The schema will serve as a vital tool for preserving and documenting Myanmar's pagoda heritage, and foster collaboration among libraries, museums, and cultural institutions, both nationally and internationally [4]. It aims to analyze information related to the development of pagoda metadata standards for libraries in Myanmar. It is part of a broader effort to develop metadata standards for pagoda information in the future. The author point is to know the general notion of cultural heritage and to be aware the importance to protect the cultural properties and lastly to analyse the legal protection and preservation of cultural heritage in Myanmar [5].

The paper [6] proposed the concept weight for text clustering with domain ontology. These use mathematical concept for the system strength. This paper [7] investigates ontologies can also be applied to the clustering process. The experimental results used dissertations papers from Google Search Engine and the proposed method demonstrated its effectiveness and practical value. A domain ontology captures the essential vocabulary and rules that govern a specific area of interest—be it healthcare, finance, education, cultural heritage, or any other field. Domain ontologies enable more effective data management and knowledge representation by defining precise concepts, properties, and relationships. They form the backbone of semantic web technologies, artificial intelligence applications, and advanced information retrieval systems, thereby enhancing the ability to process and analyse information. By following these steps in knowledge management for cultural heritage, organizations and communities can effectively preserve, protect, and promote their cultural heritage for future generations. In the content of the proposed knowledge base, the system only deals with tangible cultural heritage.

3. PROPOSED SYSTEM

The proposed system uses the following steps for the ontology construction creation process. This system use OWL ontology that is a foundational framework that can be built upon as needed to encompass more detailed aspects of the Bagan pagodas and related entities, providing a useful knowledge representation for research, tourism, and cultural heritage initiatives.

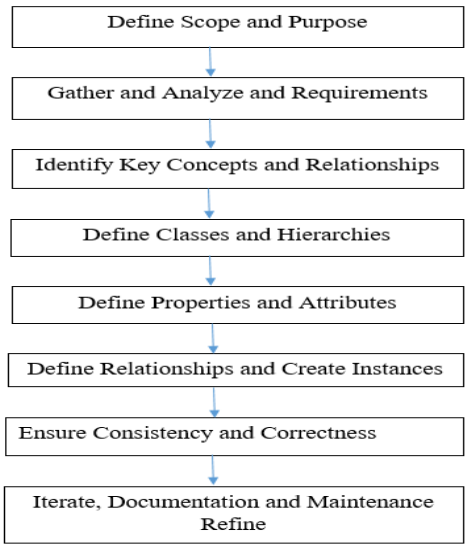


Figure1: Proposed Ontology Construction Process

The scope the system is only Bagan in Myanmar. The data is collected from Google. The following is the flow diagram representing the key stages. The data is collected as text after that removing the duplicate data and normalization is applied. For example, ensuring that the same cultural site is referred to consistently (e.g., “Bupaya Pagoda” vs. “Bupaya”). The Proposed Ontology Construction Process is for pagoda the system used Anada Temple, Dhammayangyi Temple and Thatbyunyu pagoda. For properties the system use Name: Anada, Location: Bagan Archaeological Zone, Built Year : 1105 AD and etc. The Class of ontology in the domain is Pagoda. Subclass of Pagoda class is Ananda, Dhammyangyi, Thatbinnyu.

For example

Classes:

- *Pagoda*
- *Architect*
- *Cultural Practice*

- ***Relationships:***
  - *builtBy: (Pagoda → Architect)*
  - *associatedWith: (Pagoda → Cultural Practice)*

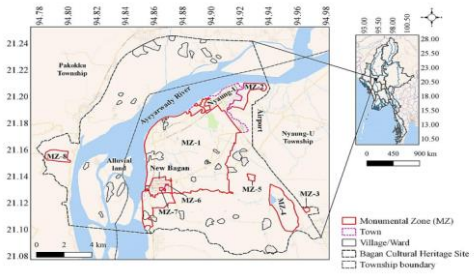


Figure2: General composition of Bagan Cultural Heritage Site and its location in Myanmar.

3.1. Ontology deployment for Proposed System

The proposed system used Protégé editors for the construction process. The proposed system used the construction specification method. These specifications ensure that the project meets the desired quality and performance criteria. Ontology construction is the process of creating a formal representation of knowledge within a particular domain.

The proposed system use the tool because it is user friendly and it support many various ontology languages, including OWL (Web Ontology Language) and RDF .The system use Protégé tool [6] using formal ontology languages such as OWL (Web Ontology Language). This tool is Protégé is a free, open-source ontology editor and framework for building intelligent systems. It is widely used for creating, visualizing, and managing ontologies in various domains, including biomedical, social sciences, and cloud computing. The OWL formally defines and represents the ontology. These allow for machine-reliable formats that can be used in various applications.

Figure 3: Bagan (UNESCO World Heritage Site in Myanmar)

3.2. Ontology Structure for Pagodas in Bagan, Myanmar

Firstly, the proposed ontology is named as *BaganPagodaOntology*. The top level class can be *Pagoda, HistoricalPeriod, Location.* For pagoda class there will be a description that represents various pagodas found in Bagan. The Object Properties define the relationships between classes, such as where a pagoda is located. An *Instance* of a pagoda (Ananda Temple) is provided with relevant properties and values. The following Figure shows the class of proposed ontology.

```
@prefix : <http://example.org/bagan-pagoda-ontology#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://example.org/bagan-pagoda-ontology>
    a owl:Ontology .

### Classes
:Pagoda a owl:Class .
```

Figure 4: Turtle Format OWL class in Ontology

```
:AnandaTemple a :Pagoda ;
    :hasName "Ananda Temple" ;
    :hasDateEstablished "1105-01-01"^^xsd:date ;
    :hasHeight "51"^^xsd:decimal ;
    :hasDescription "A prominent temple known for its impressive architecture and seated
    :isLocatedAt [ a :Location ; :hasLatitude 21.1623 ; :hasLongitude 94.8583 ] ;
    :hasArchitecturalStyle :BurmanStyle ;
    :isSignificantFor :Buddhism ;
    :belongsToHistoricalPeriod :PaganDynasty ;
    :hasCulturalFeature :Festivals .
```

Figure 5: Instance of the proposed system in Ontology

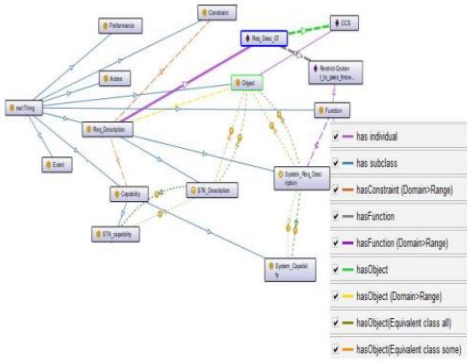


Figure 6: Proposed Ontology between domain concepts

4. CONCLUSION

In conclusion, the proposed system uses semantic platforms that are scalable and adaptable to the local context. To the future, the Bagan pagoda ontology can serve as a living document that evolves with ongoing research, discoveries, and digital initiatives. The development and evaluation of advanced ontology-based techniques for cultural heritage in Myanmar represent interesting and essential future research directions. The system could expand to include related cultural heritage elements such as local arts, crafts, to provide more holistic understanding of Bagan Region. By integrating technological solutions with community participation, Myanmar can enhance its ability to safeguard its rich cultural legacy while providing opportunities for education, tourism, and sustainable development. Another direction is to link this work to ontology mapping methods.

REFERENCES

[1] Fernández López, M., "Overview Of Methodologies For Building Ontologies", https://ceur-ws.org/Vol-18/4-fernandez.pdf

[2] Nicola Guarino, "Understanding Building, AND using Ontologies", International Journal of Human Computer Studies, 46 (2-3), pp 293-310-1997

[3] https://protege.stanford.edu/

[4] Tin Tin Pipe1 and Kulthida Tuamsuk2, "Development of Metadata Schema for Myanmar Pagodas' Information Management", International Journal of Religion, 2024.

[5] Htet Htet Zaw, "Legal Protection and Preservation of Cultural Heritage in Myanmar"

[6] Hmway Hmway Tar, T.T.Soe Nyunt, " Ontology based concept weighting for text documents", 2011, World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering

[7] Hmway Hmway Tar, Thi Thi Soe Nyunt "Enhancing Traditional Text Documents Clustering based on Ontology" International Journal of Computer Applications (0975 – 8887)

Author Biography

Hmway Hmway Tar is currently working as a professor, Faculty of Computer Science, at the University of Information and Technology, Yangon, Myanmar. Her research area is Semantic Web, AI, Machine learning, Data Sciences and other IT related fields. She got many papers including IEEE science 2011.

ATTENTION-BASED NEURAL MACHINE TRANSLATION BETWEEN MYANMAR AND ENGLISH

Phyu Moe Nyunt[1], and Su Su Win[2]

[1]Department of Computer Studies, Yadanabon University
[2]Faculty of Computer Science, University of Computer Studies (Mandalay)
[1]moenyuntphyu@gmail.com, [2]susuwin.loikaw@gmail.com

ABSTRACT

In order to translate natural languages using computers, machine translation, is a significant area of study within natural language processing. End-to-end Neural Machine Translation (NMT) has become the new standard approach in real-world machine translation systems after seeing tremendous success in recent years.[1] This study investigates the use of NMT using an attention model designed especially for translating between English and Myanmar.[2] In this paper, an encoder-decoder architecture utilizing Long Short-Term Memory (LSTM) is presented. The encoder analyzes an input text from Myanmar and produces a number of hidden representations. The translated English output is then produced by the decoder using these representations and paying attention to pertinent portions of the original phrase. The quality of the translation is greatly enhanced by attention, which enables the decoder to concentrate more on pertinent words in the input, particularly in lengthy and syntactically complicated sentences.[7]

KEYWORDS

Neural Machine Translation, Attention Mechanism, Myanmar Language, English Language, Deep Learning, LSTM

1. INTRODUCTION

The crucial task of machine translation (MT) seeks to use computers to translate sentences in natural language. Early machine translation techniques mostly rely on linguistic expertise and manually created translation rules. Given the intrinsic complexity of natural languages, it is challenging to cover every linguistic quirk with manual translation standards. By providing notable advancements over conventional statistical machine translation systems, NMT has completely transformed the language translation industry. Large-scale parallel corpora have made data-driven methods—which extract linguistic information from data—more and more popular. Statistical Machine Translation (SMT) acquires latent structures, including word alignments or phrases, directly from parallel corpora, in contrast to rule-based machine translation. Long-distance word dependencies cannot be modeled by SMT, and the translation quality is far from ideal. Neural Machine Translation (NMT) has become a new paradigm and has fast supplanted SMT as the accepted method of machine translation (MT) with the development of deep learning.[5]

Text can be translated from one source language into another target language using machine translation (MT), often known as automated translation. The state-of-the-art technique in machine translation (MT) is now neural machine translation (NMT). Unlike traditional rule-based or statistical methods, it has made tremendous progress because of its ability to learn to translate based on massive amounts of bilingual data. By allowing the model to focus on the portions of the source sentence that are most pertinent to each word in the target sentence, the addition of attention mechanisms has significantly improved translation accuracy and fluency. The Attention mechanism, which enables the model to concentrate on pertinent portions of the input sentence when producing each word of the output sentence, is one of the main innovations of NMT. An encoder-decoder architecture with attention mechanisms, trained using maximum

likelihood estimation (MLE), is a feature of the NMT system.[7]

The foundation of NMT is the idea of deep learning, which involves processing incoming data through a multi-layered neural network. In order to train the network to predict the likelihood of a series of words in the target language given a sequence of words in the source language, translation necessitates exposing it to a sizable corpus of bilingual text. An encoder-decoder framework is commonly used in the architecture of NMT systems. After processing the input sentence, the encoder compresses the data into a context vector that symbolizes the sentence's semantic meaning. The translated sentence in the target language is then produced by the decoder using this vector.[8] The system learns to map input sequences to output sequences directly without the need for intermediary stages or manually created rules because the entire process is end-to-end.

In this system, LSTM model is served as an encoder-decoder framework and the Attention mechanism is designed to predict the alignment of a target word with respect to source words. The system focuses on developing an attention-based NMT model for translating between Myanmar and English, languages with distinct syntactic and lexical characteristics. The attention-based NMT model for Myanmar-English translation demonstrated high accuracy and fluency.

2. Related Work

Compared to conventional statistical machine translation (SMT) techniques, neural machine translation has drastically improved translation quality, revolutionizing the area of machine translation. The sequence-to-sequence (Seq2Seq) learning model was presented by Sutskever et al. (2014). It converts input sequences into output sequences using an encoder-decoder architecture. This paradigm served as a foundation for other NMT innovations.

The attention method was proposed by Bahdanau et al. (2014) and enables the model to generate each word in the output sequence by focusing on distinct parts of the input sequence. This method improves translation quality, particularly for longer sentences, by reducing the bottleneck issue associated with encoding lengthy sentences into fixed-size vectors. The success of the attention mechanism prompted the creation of a number of attention-based models, such as Vaswani et al. (2017)'s Transformer model, which exclusively uses attention processes and has raised the bar in NMT.

While high-resource languages have been the main focus of NMT research, low-resource languages, such as Myanmar, are beginning to attract interest. In their discussion of the difficulties and approaches to NMT in low-resource situations, Koehn and Knowles (2017) emphasized the value of transfer learning, data augmentation, and the use of multilingual corpora.

There is a dearth of research devoted to translating words from Myanmar into English. In their 2019 study, Aung et al. investigated a phrase-based SMT system for translating words from Myanmar into English. They found some progress, but acknowledged that neural techniques could yield better results. Using a rudimentary Seq2Seq model with LSTMs, Khin et al. (2020) constructed a preliminary NMT system for translating words from Myanmar to English. The findings were encouraging, but they also made clear the need for more sophisticated methods, such as attention mechanisms.

Bojar et al.'s (2020) recent work examined multilingual NMT models that incorporate Myanmar and use shared representations to enhance low-resource language translation quality. By pretraining models on high-resource languages and fine-tuning on low-resource language pairings, Conneau et al. (2020) investigated cross-lingual transfer learning and showed notable gains in low-resource language translation. The BLEU score, a commonly used statistic for assessing the caliber of machine-generated

translations, was first presented by Papineni et al. in 2002.

3. METHODOLOGY

Neural Machine Translation (NMT) represents a significant advancement in the field of machine translation, leveraging the power of deep learning to translate text from one language to another. Unlike traditional statistical machine translation (SMT) methods, which rely on phrase-based translation models and extensive feature engineering, NMT employs neural networks to model the entire translation process end-to-end.[1]

One of the foundational architectures in NMT is the use of Long Short-Term Memory (LSTM) networks. LSTMs are a type of recurrent neural network (RNN) designed to handle the vanishing gradient problem, making them particularly effective for tasks involving long-range dependencies, such as language translation.

LSTMs were introduced by Hochreiter and Schmidhuber in 1997 to address the limitations of traditional RNNs. They incorporate memory cells that can maintain information over long periods, enabling them to capture dependencies in sequential data effectively. Each LSTM cell contains three gates: the input gate, the forget gate, and the output gate, which regulate the flow of information and the updates to the cell state.[4]

3.1 NMT Architecture with LSTM

The core architecture of NMT using LSTMs typically follows the sequence-to-sequence (Seq2Seq) model, comprising two main components: an encoder and a decoder.

1. **Encoder**: The encoder reads the input sentence (source language) and converts it into a fixed-size context vector. It consists of an embedding layer followed by one or more LSTM layers. The final hidden states of the LSTM are used to summarize the input sentence.
2. **Decoder**: The decoder generates the output sentence (target language) using the context vector from the encoder. It also consists of an embedding layer and one or more LSTM layers. The initial states of the decoder LSTM are set to the final states of the encoder LSTM. At each time step, the decoder predicts the next word in the target sequence.
3. **Attention Mechanism**: To improve the translation quality, an attention mechanism can be integrated into the Seq2Seq model. The attention mechanism allows the decoder to focus on different parts of the input sentence at each time step, rather than relying on a single context vector. This is particularly useful for handling long sentences and capturing complex dependencies.

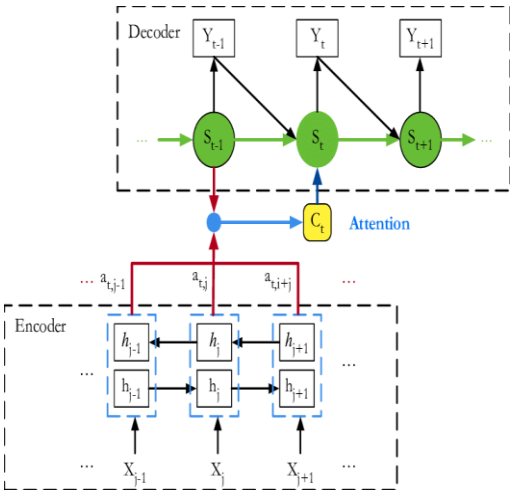


Figure 1. Attention based NMT

3.2 Dataset

The dataset that ALT corpus, which consists of twenty thousand Myanmar-English parallel phrases from Wikinews, is a component of the Asian Language Treebank (ALT) project (Riza et al., 2016). The NLP Lab, University of Computer Studies, Yangon (UCSY), Myanmar, is building the UCSY corpus (Yi Mon Shwe Sin and Khin Mar Soe, 2018). The corpus comprises two hundred thousand parallel sentences in Myanmar and English that were gathered from various sources, such as textbooks and news articles. Because the ALT corpus is so tiny, a bigger out-of-domain corpus—also referred to as the UCSY corpus—for the same language pair is offered. The training

data consists of about 220,000 lines of sentences and phrases from the UCSY corpus and a subset of the ALT corpus. The ALT corpus is the source of the development and test data. As a result, a combination of domain data gathered from various sources serves as the training set for translation jobs between English and Myanmar and Myanmar and English.[3] Data statistics utilized in the trials are displayed in Tabel 1.

Table 1: Statistics of Dataset

Data Type	# of Sentence	# of Myanmar Syllables	# of English Words
TRAIN (UCSY)	238,014	6,285,996	3,357,260
TRAIN (ALT)	18,088	1,038,640	413,000
DEV	1,000	57,709	27,318
TEST	1,018	58,895	27,929

4. EXPERIMENTAL SETUP

Attention-based neural machine translation (NMT) using Long Short-Term Memory (LSTM) networks is a popular architecture in the field of deep learning for translating text from Myanamar language to English. Below is a step-by-step process for the implementation of the system:

4.1 Data Preparation

- Collect Data**:** Obtain a parallel corpus of source and target language sentence pairs.
- Preprocess Data**:** Tokenize, clean, and normalize the text. Convert words to lowercase, remove punctuation, and handle special characters.
- Create Vocabulary**:** Build a vocabulary for both source and target languages. Each word is assigned a unique integer index.
- Convert Text to Sequences**:** Map each word in the sentences to its corresponding index from the vocabulary.

4.2 NMT with Attention Model

The system is based on NMT with attention model, which links blocks of Long Short-Term Memory (LSTM) in an RNN. It trained the word-based NMT using OpenNMT, an opensource project. Create embedding layers for both source and target languages. These layers convert word indices into dense vectors of fixed size. Training times vary depending on parameter settings. Table 2 lists the network hyper-parameter settings for NMT models.

Tabel 2: Hyper-parameter of NMT Models

Hyper-parameter	NMT models
src vocab size	25,087
tgt vocab size	50,004
Embedding Dim	256
LSTM Units	512
Encoder layer	2
Decoder layer	2
Learning rate	0.01
Dropout rate	0.3
Mini-batch size	64
Optimizer	adam
Loss Function	sparse categorical crossentropy

Use an LSTM to process the source sentence. The encoder LSTM processes the input sequence and outputs a sequence of hidden states. Input the source sentence to the embedding layer to obtain embedded representations. Pass the embeddings through the LSTM to get hidden states for each time step.

Add an attention layer that helps the decoder focus on relevant parts of the source sentence. The attention mechanism computes a context vector as a weighted sum of the encoder's hidden states. Compute the attention weights for each hidden state of the encoder. Multiply the attention weights by the encoder's hidden states to get the context vector.

The visualization showing the attention weights for each word in the input sentence

as the model generates each word in the output sentence. Visualization of the attention mechanism can improve the performance of the system. For example, the input sentence “မင်္ဂလာပါ။ သင့်ရဲ့နေရာမှာ ဒီနေရာကို ပြောင်းလဲလို့ရလား။”, the visualization is:

- When generating the word "Hello," the model pays attention to "မင်္ဂလာပါ" in the input sentence.
- For the phrase "Can you change," the model focuses on the corresponding part "သင့်ရဲ့နေရာမှာ ဒီနေရာကို ပြောင်းလဲ" in the input sentence.
- The word "place" in the output is aligned with "နေရာ" in the input.
- The phrase "to your place" corresponds to "သင့်ရဲ့နေရာမှာ" in the input sentence.

Use another LSTM to generate the target sentence. The decoder LSTM takes the context vector and the previous target word (during training) or the predicted word (during inference) as inputs to predict the next word in the sequence. Input the previous target word to the embedding layer to obtain its embedded representation. Combine the context vector and the embedded previous target word and pass them through the decoder LSTM to generate the next hidden state. Use the hidden state to predict the next word in the target sequence.

Define the model that takes source and target sentences as input and outputs the predicted target sentences. The model was compiled using the Adam optimizer with a learning rate of 0.001. Sparse categorical cross-entropy was used as the loss function, appropriate for multi-class classification problems like NMT. The model was trained for 50 epochs with a batch size of 64. The training set consisted of 80% of the collected data, with 10% used for validation and 10% for testing.

During inference, use the encoder to process the input sentence and obtain the encoder's hidden states. Use the initial decoder states (from the encoder) and start with the start token. For each time step, use the decoder to generate the next word, update the decoder states, and use the predicted word as the input for the next time step. After the training processes, combine the translated words to form the final translated sentence and remove the start and end tokens from the translated sentence.

The encoder-decoder architecture with attention is a key component of neural machine translation (NMT) models, especially for translating between Myanmar and English. Here's an explanation of how it works, using an example translation. For example: Myanmar sentence: "ကျောင်းသားများ စာသင်ခန်းထဲသို့ဝင်ကြသည်။" English translation: "The students enter the classroom."

The encoder processes the source sentence, and the decoder generates the translated target sentence. The attention mechanism allows the decoder to focus on different parts of the source sentence when translating.

1. Encoder Stage:

The encoder processes the input Myanmar sentence and creates a representation of it. This representation is then passed to the decoder.
Step 1: The input sentence is tokenized.
Tokenized sentence:

["ကျောင်းသားများ", "စာသင်ခန်း", "ထဲ", "သို့", "ဝင်", "ကြ", "သည်"]

These tokens represent words like "students," "classroom," and other meaningful parts of the sentence.

Step 2: Each token is converted into an embedding vector. These embeddings are fed into the encoder, which is typically an LSTM (long short-term memory) or GRU (gated recurrent unit).

Step 3: The encoder processes each word step-by-step, updating its internal hidden state to capture the meaning of the sentence. It outputs a sequence of hidden states:
Hidden states: [h1, h2, h3, h4, h5, h6, h7]

h1: "ကျောင်းသားများ" (students)

h2: "စာသင်ခန်း" (classroom)

h3: "ထဲ" (into)

h4: "သို့" (to)

h5: "ဝင်" (enter)

h6: "ကြ" (plural marker)

h7: "သည်" (auxiliary verb)

Each hidden state captures not only the meaning of the current word but also the context of the previous words in the sentence.

2. Decoder Stage with Attention

The decoder generates the English translation, one word at a time, using the encoded representation and the attention mechanism.

Step 1: The decoder begins with a start token `<start>` and uses the final hidden state of the encoder as an initial context. For each word in the target sentence, the decoder uses the attention mechanism to focus on different parts of the Myanmar sentence. At each decoding step, the attention mechanism looks at all the hidden states from the encoder. It computes an alignment score to determine which parts of the Myanmar sentence are most relevant to generating the next word in English. For example, when translating "students," the attention mechanism will focus on h1 (the hidden state corresponding to "ကျောင်းသားများ").

Step 2: The decoder predicts the next word in the English sentence based on the attention-weighted hidden states and the previous decoder state.

- First word: It predicts "The students" based on h1 (because "ကျောင်းသားများ" means "students").
- Second word: Next, the decoder attends to h2 (which corresponds to "စာသင်ခန်း" or "classroom") and predicts "enter."
- Third word: The decoder attends to h3 and h4 to predict the word "the."
- Fourth word: Finally, it focuses on `h5` ("ဝင်" or "enter") to predict "classroom."

Step 3: The decoder continues this process until it generates the entire English sentence: "The students enter the classroom."

4.3 Experiment Result

The performance of the attention-based neural machine translation (NMT) model for translating between Myanmar and English was evaluated by BLEU scores. It measures the similarity between the machine-generated translation and reference translations. The Steps in BLEU Score Calculation:

1. N-Gram Precision: BLEU uses n-grams (contiguous sequences of words) to compare the candidate translation to the reference translations. The precision is calculated for different n-gram sizes (unigram, bigram, trigram, etc.), up to a specified maximum n-gram length. For example, for a unigram (n=1), it checks how many individual words in the candidate match any word in the references.
2. Clipping: BLEU applies a "clipping" mechanism to handle cases where the candidate translation overuses words that appear in the reference translation. The candidate's count of each n-gram is clipped by the maximum number of times that n-gram appears in the reference.
3. Brevity Penalty: A brevity penalty (BP) is applied to discourage overly short translations. If the candidate translation is shorter than the reference, a penalty reduces the final score. If the candidate length is equal to or longer than the reference length, the penalty is 1. The Formula of Brevity Penalty is as follow:

$$BP = \begin{cases} 1, if\ c > r \\ e^{\left(1-\frac{r}{c}\right)}, if\ c \leq r \end{cases}$$

where:
- c is the length of the candidate translation.
- r is the effective reference length (typically the length of the closest reference to the candidate).

4. Final BLEU Score: The final score is a geometric mean of the precision scores for all n-grams, multiplied by the brevity penalty. Mathematically, it's expressed as:

$$BLEU = BP.\exp\left(\sum_{n=1}^{N} w_n \log p_n\right)$$

where:
- BP is the brevity penalty.
- w_n is the weight for each n-gram precision (commonly equal for all n-grams).
- p_n is the precision for n-grams of size n.
- N is the maximum n-gram size.

The BLEU scores for the model on the test set are presented in Table 3. In this system, 80% of the collected parallel Myanmar-English sentences is trained, 10% of the data used as validation set and 10% of the data used for final evaluation.

Tabel 3: Performance of the System

Language Pair	**BLEU Scores**
Myanmar to English	35.7
English to Myanmar	32.5

The performance evaluation of an attention-based Neural Machine Translation (NMT) model between Myanmar and English. The system used metrics such as BLEU scores, training loss, and validation loss over time. The Training Loss is a loss function value during training. The Validation Loss is a loss function value on the validation set, used to check how well the model generalizes.

The qualitative analysis revealed that the model performs well in generating fluent and contextually appropriate translations. However, there are some limitations these are: Handling of Rare Words, the model sometimes struggles with rare or out-of-vocabulary words, leading to inaccuracies and Long Sentences, while the attention mechanism helps with long sentences, the model occasionally misses nuances in very lengthy or complex sentences.

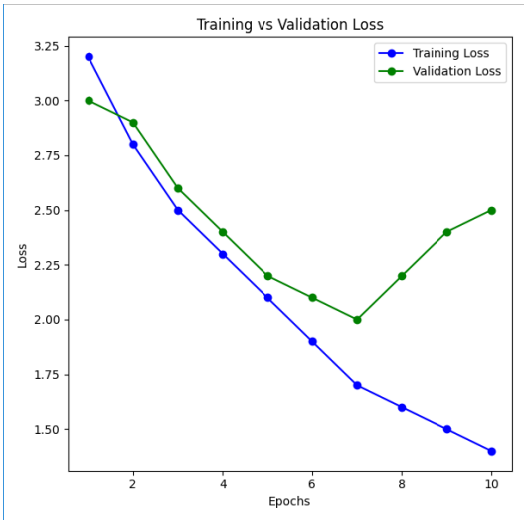


Figure 2. Training Vs Validation Loss

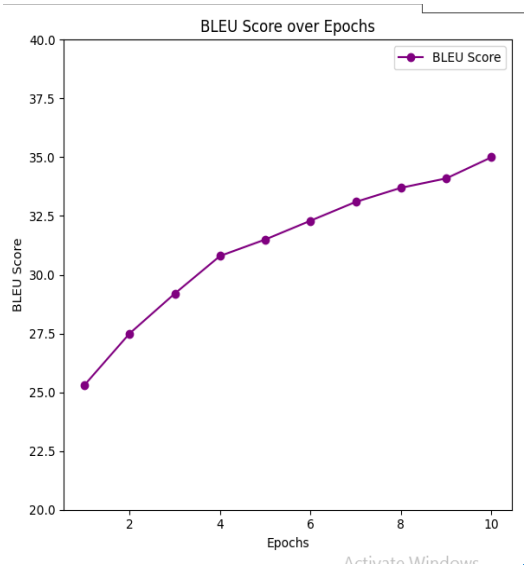


Figure 3. BLEU Score over Epochs

5. CONCLUSIONS

NMT using LSTM has proven to be a powerful approach for language translation, offering significant improvements over traditional methods. The integration of attention mechanisms further enhances the model's ability to handle long sentences and

complex linguistic structures. This paper presents an attention-based NMT model for Myanmar-English translation, leveraging the strengths of LSTMs and attention mechanisms to achieve high-quality translations.

The system presented an effective attention-based NMT system for Myanmar-English translation, achieving competitive performance and demonstrating the potential of attention mechanisms in handling diverse language pairs. It demonstrated high accuracy and fluency. The attention mechanism's ability to focus on relevant parts of the input sequence proved crucial in handling the linguistic differences between Myanmar and English.

The experimental results demonstrate that the attention-based NMT model performs effectively in translating between Myanmar and English, achieving reasonable BLEU scores and producing fluent translations. The attention mechanism significantly contributes to handling long sentences and capturing dependencies, leading to improved translation quality. Further improvements could be achieved by expanding the training dataset and fine-tuning the model architecture.

6. REFERENCES

[1] Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to Sequence Learning with Neural Networks. Advances in Neural Information Processing Systems, 27.

[2] Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural Machine Translation by Jointly Learning to Align and Translate. arXiv preprint arXiv:1409.0473.

[3] Ye Kyaw Thu, Win Pa Pa, Masao Utiyama, A. Finch and E. Sumita, “Introducing the Asian Language Treebank (ALT)”, 10th edition of the Language Resources and Evaluation Conference, Portorož, Slo-venia, May 23-28, 2016.

[4] YonghuiWu, Mike Schuster, Zhifeng Chen, et al. 2016. Google’s neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144.

[5] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is All You Need. Advances in Neural Information Processing Systems, 30.

[6] Yi Mon Shwe Sin and Khin Mar Soe, “Large Scale Myanmar to English Neural Machine Translation System”. Proceeding of the IEEE 7th Global Conference on Consumer Electronic (GCCE 2018).

[7] Yi Mon Shwe Sin and Khin Mar Soe. 2019.Attention-based syllable level neural machine translation system for myanmar to english language pair. International Journal on Natural Language Computing, 8(2):01–11.

[8] Chen, Peng-Jen et al. “Facebook AI’s WAT19 Myanmar-English Translation Task Submission.” Conference on Empirical Methods in Natural Language Processing (2019).

[9] Ding, C., Aye, H. T. Z, Pa, W. P., Nwet, K. T., Soe, K. M., Utiyama, M., and Sumita, E. (2019). "Towards Burmese (Myanmar) Morphological Analysis: Syllablebased Tokenization and Part-of-Speech Tagging", ACM TALLIP, Vol. 19, Issue 1, Article No. 5.

[10] Xuewen Shi, Heyan Huang, Ping Jian, and Yi-Kun Tang. Improving neural machine translation with sentence alignment learning. Neurocomputing, 420:15–26, 2021. ISSN 0925-2312. doi: https://doi.org/10.1016/j.neucom. 2020.05.104.

HUMAN FACES EXPRESSION RECOGNITION USING CNN MODEL

Phyu Moe Nyunt[1], and Su Su Win[2]

[1]Department of Computer Studies, Yadanabon University
[2]Faculty of Computer Science, University of Computer Studies (Mandalay)
[1]moenyuntphyu@gmail.com,[2]susuwin.loikaw@gmail.com

ABSTRACT

In order for, human to human interaction to be more smoothly communicated, emotions are important. Enhancing human-computer interaction (HCI) is another benefit of employing emotion recognition systems. Recent years have seen a rise in the need for improved human-computer interfaces, leading to the development of facial expression recognition (FER) systems. Computer-based technologies known as facial expression recognition systems can recognize faces, interpret facial expressions, and identify emotional states [3]. In this system, human faces expression is recognized by using Deep Learning models called Convolutional Neural Networks (CNNs). The system emphasizes the use of CNNs for feature extraction due to their ability to capture spatial hierarchies in facial images. It also examined the role of data augmentation and transfer learning in enhancing model performance, particularly in scenarios with limited training data [5]. After extensive testing and analysis on standard FER datasets AffectNet, RAF-DB result of the system demonstrate significant improvements in recognition accuracy and robustness.

KEYWORDS

Human-Computer Interaction, Deep Learning, Facial Expression Recognition, Convolutional Neural Networks

1. INTRODUCTION

In the field of human-computer interaction, facial expression recognition (FER) is essential because it allows computers to recognize and react to human emotions. Accurate facial expression recognition is necessary for many applications, such as vehicle safety systems, security surveillance, healthcare diagnostics, and virtual reality environment user experience enhancement. [2]

FER systems historically depended on manually created features and traditional machine learning techniques, which frequently had trouble generalizing well across a variety of datasets and real-world situations. These techniques produced less than ideal results since they were sensitive to changes in individual facial structures, occlusions, and illumination. Convolutional neural networks (CNNs), in particular, have transformed the field of deep learning by providing strong tools for robust categorization and automatic feature extraction. [8]

CNNs' capacity to extract hierarchical features from unprocessed pixel data has allowed them to achieve impressive results in image identification challenges. Multiple convolutional filter layers enable CNNs to capture the complex patterns and spatial hierarchies present in facial expressions. CNNs are especially well-suited for facial expression recognition (FER), where it is necessary to precisely identify and interpret small changes in facial muscles.[10]

The use and assessment of a CNN-based model for recognizing human facial expressions are the main topics of this research. The goal of the system is to investigate how well various CNN designs, data augmentation methods, and training approaches work together to produce high recognition accuracy. Utilizing popular FER datasets as AffectNet and RAF-DB, Finally, the system compares the performance of the model to current state-of-the-art techniques.

The system begins by providing an overview of related work in the field of FER, highlighting the transition from traditional approaches to deep learning-based methods. Next, we delve into the specifics of CNN architecture, including convolutional layers, pooling layers, and activation functions, explaining their roles in feature extraction and classification. The subsequent sections detail our experimental setup, including data preprocessing, model training, and evaluation metrics.

Through rigorous experimentation and analysis, we demonstrate the superiority of CNN-based models in recognizing facial expressions under various conditions. The results underscore the importance of deep learning in advancing the accuracy and robustness of FER systems. By harnessing the power of CNNs, this study aims to contribute to the development of more intuitive and responsive human-computer interaction systems, ultimately bridging the gap between human emotions and machine understanding.

2. RELATED WORK

For many years, researchers have been actively working in the field of facial expression recognition (FER), and the development of deep learning and machine learning approaches has led to notable progress in this field. Traditional machine learning techniques and handmade features were the mainstays of early FER approaches. Nevertheless, these techniques frequently suffered from changes in position, lighting, and occlusions, which reduced their accuracy and robustness. The advent of deep learning, particularly Convolutional Neural Networks (CNNs), has significantly transformed the landscape of FER. Numerous studies have shown that CNNs outperform conventional techniques in FER tasks.

Because deep learning models, in particular CNNs, automatically learn hierarchical features from raw data, they have shown improved performance in a variety of computer vision applications. The AlexNet architecture was first presented in a seminal paper by Krizhevsky et al. [5], and it demonstrated remarkable performance on the FER-2013 dataset. CNNs have been proven to be an effective technique for FER by this success.

Several CNN architectures were investigated in later studies to improve FER performance. In FER tests, the VGGNet architecture [6], renowned for its intricate yet straightforward sequential convolutional layer design, demonstrated increased accuracy. The vanishing gradient problem was addressed by He et al. with the introduction of ResNet [7], which used residual learning to enable the training of far deeper networks. This invention greatly increased the FER models' robustness and accuracy.

Szegedy et al. proposed inception networks [8], which significantly improved recognition performance by capturing a variety of facial traits by utilizing multi-scale processing capabilities. These sophisticated CNN architectures served as a strong basis for the creation of more complex FER models.

Hybrid models that combine CNNs with Recurrent Neural Networks (RNNs) have been developed to capture temporal and spatial information. These models employ RNNs, such as Long Short-Term Memory (LSTM) networks, for temporal sequence modeling and CNNs for the extraction of spatial features. The efficacy of this method in video-based FER, where temporal context is essential for precise expression identification, was shown in a study by Fan et al. [9].

Data augmentation approaches have been widely employed to expand the diversity and amount of training data due to the limited availability of labeled FER datasets. To provide training examples that are more diverse, techniques like rotation, scaling, and flipping have been used [10]. Enhancing FER performance has also been made possible via transfer learning, which is honing pre-trained CNNs using sizable datasets like ImageNet. The efficiency of transfer learning in raising the accuracy of

FER models trained on fewer datasets was shown in a study by Ding et al. [11].

3. SYSTEM ARCHITECTURE

The system architecture for human facial expression recognition using CNNs typically involves several components, each designed to handle specific tasks from data preprocessing to final result. Here's an overview of the system architecture:

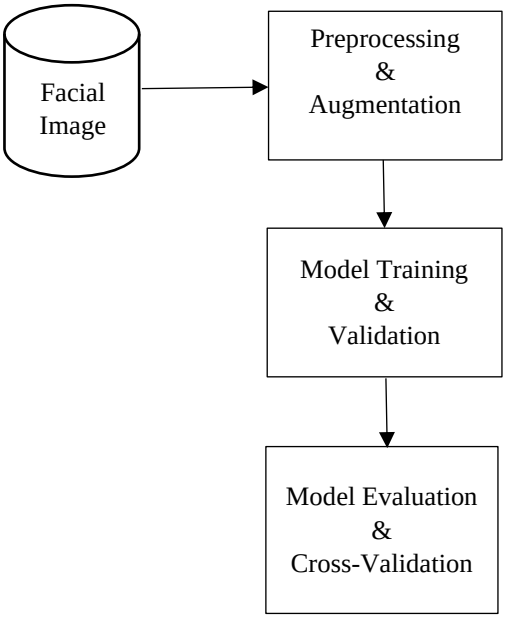


Figure 1. Overview of the System Architecture

3.1 Dataset

456,349 photos of various facial expressions are included in the AffectNet database. These photographs were gathered from Yahoo, Bing, and Google and are available in six different languages. Eleven emotions were assigned labels to the images: none, uncertain, nonface, disgust, rage, surprise, fear, happiness, sadness, and nonface. Among these feelings, "nonface" denotes that an image has watermarks, animated graphics, exaggerated facial expressions, or drawings; "uncertain" denotes that an image cannot be placed in any of the other categories.

Annotators were employed by Mollahosseini et al. (2015) to manually classify the emotions defined in AffectNet. Furthermore, there is a significant imbalance in AffectNet's image count for each emotion category. For instance, there are about thirty times as many photos of people smiling as there are of people disgusted. The number of images for each category is shown in Table1. In this paper, the system uses seven categories, surprise, fear, disgust, anger, sadness, happiness and neutrality, in AffectNet.

The Beijing University of Posts and Telecommunications' Pattern Recognition and Intelligent System Laboratory (PRIS Lab) provides the RAF-DB. More than 300,000 face photos from the internet make up the database, which is divided into seven sections: surprise, fear, disgust, rage, sadness, happiness, and neutrality.

There are 37 automated landmark locations and 5 precise landmark positions in each image. A vast range of data including ages, races, head gestures, light exposure levels, and blocking may also be found in the RAF-DB. Five times as many images are in the training set as in the test set. Table 1 shows the number of images used in the system for each emotion from each database.

Table 1. Number of images in each database

Category	Number of images in AffectNet	Number of images in RAF-DB
Neutrality	80,276	3,204
Happiness	146,198	5,957
Sadness	29,487	2,460
Surprise	16,288	1,619
Fear	8,191	355
Disgust	5,264	877
Anger	28,130	867
Contempt	5,135	NA
None	35,322	NA
Uncertain	13,163	NA
Nonface	88,895	NA
Total	456,349	15,339

3.2 Preprocessing of FER

Preprocessing is a crucial step in facial expression recognition (FER) using Convolutional Neural Networks (CNNs). Proper preprocessing ensures that the input data is in the optimal format for training and evaluating the CNN model. The following preprocessing steps are involved in preparing facial images for FER.

1. Face Detection and Alignment
2. Image Resizing and Grey Scale Conversion
3. Data Augmentation
4. Normalization

Firstly, Face Detection process is worked to ensure the model focuses on the facial region. Haar Cascades is the face detection algorithms that is a method to detect an object to identify in an image. After detecting faces, alignment is performed to normalize the facial orientation. This step involves aligning the faces based on key landmarks (e.g., eyes, nose, and mouth) to ensure consistency in pose and scale. In this paper, the Dlib library's facial landmark detector techniques can be used for this purpose.

Then, all detected and aligned face images are resized to a fixed dimension to ensure uniform input size for the CNN model, typically 48x48 or 224x224 pixels. Resizing ensures that the model processes images consistently, facilitating efficient training. Converting images to grayscale can simplify the model and reduce computational complexity. Grayscale images reduce the input dimensionality while retaining essential facial features.

After that, data augmentation techniques are applied to enhance the diversity and size of the training dataset. Augmentation helps the model generalize better by simulating variations that may occur in real-world scenarios. In this system, rotation, scaling, translation, brightness and contrast adjustment and noise addition techniques are used.

Finally, normalization is performed to scale pixel values to a specific range, typically [0, 1] or [-1, 1]. This step ensures that the input data has a consistent distribution, which aids in faster convergence during training. Normalization can be performed using Min-Max Normalization which work by scaling pixel values to the [0, 1] range.

3.3 CNNs Models Training and Validation

Convolutional Neural Networks (CNNs) are a specialized type of artificial neural network designed for processing structured grid data, such as images. CNNs are particularly effective for tasks like image classification, object detection, and facial expression recognition due to their ability to automatically and adaptively learn spatial hierarchies of features. The keys component of CNNs model are: Convolutional Layers, Activation Functions, Pooling Layers, Fully Connected (Dense) Layers, Dropout Layers.

In a Convolutional Neural Network (CNN) model for Facial Expression Recognition (FER), the process involves a series of operations that transform an input image through multiple layers, including convolutional layers, pooling layers, and fully connected layers. Each layer applies a specific mathematical operation to extract features and ultimately classify the facial expression.

The convolution operation is the core operation in a CNN. It involves sliding a filter (kernel) over the input image and computing the dot product between the filter and the receptive field in the input. The activation function Rectified Linear Unit (ReLU) is used in the system.

$$ReLU(x)=max(0,x) \quad (1)$$

After applying the ReLU activation function to the convolution output O:

$$A_{i,j}=ReLU(O_{i,j}) \quad (2)$$

The pooling layer reduces the spatial dimensions of the feature maps while retaining the most important information. Max pooling selects the maximum value from each patch of the feature map. In the fully connected layer, the input feature map is flattened into a vector, and then a standard neural network layer is applied. The output

layer uses the softmax activation function to convert the raw scores into probabilities for each class (facial expression).

$$softmax(z_i) = \frac{e^{z_j}}{\sum_j e^{z_j}} \quad (3)$$

In the system, create a sequential model with layers as described earlier. And then, train the model using the training set and validate it using the validation set. Validation is a crucial step in training Convolutional Neural Networks (CNNs) for facial expression recognition to ensure that the model generalizes well to unseen data. The system used train validation test split are 70% used to train the model, 15% used to tune hyperparameters and make decisions about the model architecture and 15% used to evaluate the final performance of the model.

4. PERFORMANCE EVALUATION

In this paper, the CNNs mdoel is used to classify facial expression from the AffectNet and RAF-DB image dataset that is obtained from Kaggle.com. It is a collection of approximately 456,349 and 15,339 images data, partitioned evenly across 10 different categories in AffectNet dataset and 8 different categories in RAF-DB dataset. However, this paper will focus on 7 categories of the image dataset consisting of the categories surprise, fear, disgust, anger, sadness, happiness and neutrality.

The performance of the system is evaluated by comparing the result of the image classification with the annotations found in the original dataset. The system performance is evaluated by using various evaluation metrics, (i) Precision, (ii) Recall, (iii) F1-score. Here, the confusion matrix is used to compute Precision, Recall and F1-score of the system. In a confusion matrix, four metrics can be defined: true positive (TP), true negatives (TN), false positive (FP), false negative (FN).

$$\text{Precision (P)} = \frac{TP}{TP+FP} \quad (4)$$

$$\text{Recall (R)} = \frac{TP}{TP+FN} \quad (5)$$

$$\text{F1-Score (F1)} = \frac{2 * Precision * Recall}{(Precision + Recall)} \quad (6)$$

The following result show that the CNNs model can predict the facial expressions correctly with the accuracy of 85.4%. The system successfully developed and evaluated a CNN model for facial expression recognition using the AffectNet and RAF-DB dataset. The CNN architecture, trained with rigorous data augmentation and optimized hyperparameters, achieved significant performance metrics. This evaluation underscores the potential of CNNs in facial expression recognition, paving the way for enhanced emotion detection technologies applicable to diverse fields such as human-computer interaction and mental health assessment.

Tabel 2. Confusion metrics of AffectNet Dataset for Facial Expression Recognition

		Predict							
		Neutral	Happy	Sad	Surprise	Fear	Disgust	Anger
Actual	Neutral	560	30	25	10	5	10	20
	Happy	20	670	15	40	5	0	10
	Sad	30	10	580	5	20	15	20
	Surprise	5	50	5	620	10	0	5
	Fear	10	5	20	5	550	20	30
	Disgust	15	0	15	5	20	580	20
	Anger	25	10	25	5	30	10	570

Tabel 3. Confusion metrics of RAF-DB Dataset for Facial Expression Recognition

Actual	Predict							
	Neutral	Happy	Sad	Surpris	Fear	Disgust	Anger
Neutra	500	20	15	5	10	8	12
Happy	25	610	10	25	3	5	12
Sad	20	5	530	8	18	10	25
Surpris	5	35	8	580	10	5	5
Fear	12	5	18	10	520	15	25
Disgust	8	5	12	5	20	540	10
Anger	15	10	25	5	30	10	530

Tabel 4. Performance Evaluation of AffectNet Dataset for Facial Expression Recognition

Expression	Precision	Recall	F1 Score	Accuracy
Neutral	0.85	0.83	0.84	0.86
Happy	0.90	0.92	0.91	0.93
Sad	0.82	0.80	0.81	0.82
Surprise	0.88	0.86	0.87	0.85
Fear	0.75	0.78	0.76	0.76
Disgust	0.73	0.71	0.72	0.73
Anger	0.80	0.77	0.78	0.79

Tabel 5. Performance Evaluation of RAF-DB Dataset for Facial Expression Recognition

Expression	Precision	Recall	F1 Score	Accuracy
Neutral	0.88	0.85	0.86	0.85
Happy	0.93	0.94	0.93	0.92
Sad	0.82	0.80	0.81	0.82
Surprise	0.90	0.87	0.88	0.89
Fear	0.77	0.79	0.78	0.78
Disgust	0.74	0.73	0.73	0.74
Anger	0.81	0.79	0.80	0.80

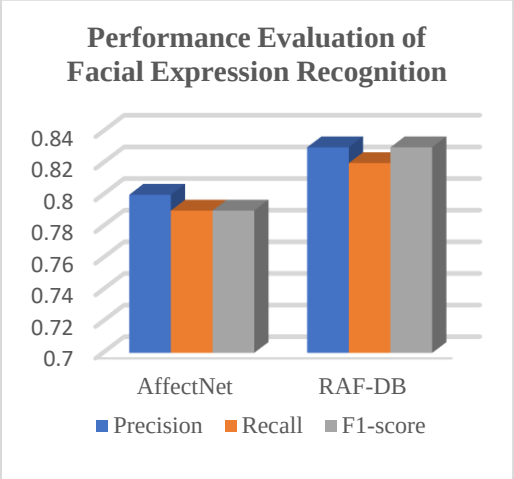


Figure 2. Overall Performance Evaluation of the System

5. CONCLUSIONS

The learning process of the neural network model to recognize face emotions is examined in this paper. CNN was used to extract the features seen on emotion images, and these emotional aspects were then visualized to identify the key facial landmarks that hold important information. The system demonstrated that CNNs exhibit high accuracy and robustness in identifying faces across diverse conditions and datasets. CNNs model achieved an accuracy of 81.4% on the AffectNet and 83.6% on the RAF-DB, outperforming several existing methods in

the field. The effectiveness of CNNs in feature extraction and classification was evident, underscoring their suitability for real-world applications such as security systems and personalized user interfaces.

While our study showcased promising results, several challenges remain, including variations in illumination, occlusion, and pose. Future research should focus on integrating multi-modal approaches and exploring advanced CNN architectures to enhance performance further. By addressing these challenges, we can pave the way for more reliable and efficient human face recognition systems."

6. REFERENCES

[1] *Ahonen, T., Hadid, A., & Pietikäinen, M. (2006). Face Description with Local Binary Patterns: Application to Face Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(12), 2037-2041.*

[2] *Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 886-893.*

[3] *Lowe, D. G. (1999). Object Recognition from Local Scale-Invariant Features. Proceedings of the International Conference on Computer Vision (ICCV), 1150-1157.*

[4] *Shan, C., Gong, S., & McOwan, P. W. (2009). Facial Expression Recognition Based on Local Binary Patterns: A Comprehensive Study. Image and Vision Computing, 27(6), 803-816.*

[5] *Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. Advances in Neural Information Processing Systems, 25, 1097-1105.*

[6] *Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. arXiv preprint arXiv:1409.1556.*

[7] *He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778.*

[8] *Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going Deeper with Convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1-9.*

[9] *Fan, Y., Lam, J. C. K., & Li, V. O. K. (2016). Video-based Emotion Recognition Using CNN-RNN and C3D Hybrid Networks. Proceedings of the International Conference on Multimedia and Expo (ICME), 1-6.*

[10] *Shorten, C., & Khoshgoftaar, T. M. (2019). A Survey on Image Data Augmentation for Deep Learning. Journal of Big Data, 6(1), 60.*

[11] *Ding, L., & Tao, J. (2018). Facial Expression Recognition Using Deep Learning and Transfer Learning. Proceedings of the International Conference on Affective Computing and Intelligent Interaction (ACII), 1-6.*

[12] *Wang, Z., Yin, L., Wei, X., & Sun, Y. (2020). Attention Mechanism Enhanced Facial Expression Recognition With Deep Networks. IEEE Transactions on Multimedia, 22(11), 3033-3046.*

[13] *Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., ... & Bengio, Y. (2014). Generative Adversarial Nets. Advances in Neural Information Processing Systems, 27, 2672-2680.*

AUTHORS BIOGRAPHY

Phyu Moe Nyunt, Associate Professor at Department of Computer Studies, Yadanabon University (Mandalay). She received M.SC (Computer Science) degree in 2004 at Yangon University.

Su Su Win is a Lecturer of the Faculty of Computer Science at University of Computer Studies (Mandalay). She received M.C.Sc (Master of Computer Science) degree in 2010 at University of Computer Studies (Loikaw). She has published papers in University Journals. Her research interests include Natural Language Processing, Speech Recognition and Image Processing.

SCRUTINY OF OPTICAL EXTINCTION BANDS OF COLLOIDAL AG NPS AND AG_{CORE}-CU_{SHELL} NANOPARTICLES IN DIFFERENT STABILIZING AGENT FOR STABILITY EVALUATION

Thaung Hlaing Win[1]

[1] Department of Physics, Pakokku University
[1]thaung.naing001@gmail.com

ABSTRACT

The silver nanoparticles (AgNPs) were developed in distilled water by chemical reduction method using three different stabilizers (polyvinyl pyrrolidone (PVP), polyethylene glycol (PEG) and ascorbic acid (AA)). We carried out a careful examination of the time evolution of surface plasmon resonance (SPR) bands (specifically, peak positions and intensities) of colloidal AgNPs so as to evaluate their stability. In addition, the changing pattern of SPR peak positions and intensities during the stability time period was also investigated. Effects of stabilizer materials, stabilizer concentration, Cu capping on the stability of AgNPs colloids have been highlighted. The maximum stability of AgNPs is 45 day with stabilizer PVP, PEG and AA. The stability time of AgNPs in EG is further lengthened to 30 days in the presence of Cu capping ($Ag_{core}Cu_{shell}$ NPs). Thus a proper selection of the stabilizing/capping agent and the reaction medium is critical in determining the stability of AgNPs colloids. The benefits of stabilization of AgNPs for real world applications are immense and this study would help in examining the stability of other novel plasmonic metal nanostructures.

KEYWORDS

$Ag_{core}Cu_{shell}$, nanoparticles, different stabilizer, PVP, PEG, AA.

1. INTRODUCTION

Metal nanoparticles attract attention of the researchers due to their plasmonic and physicochemical properties. The former one deals with free surface electrons confined to metal nanoparticles which possess surface plasmon resonances (SPR) in the visible region giving rise to its intense colors. The applications of SPR are impressively numerous. To take photovoltaics as an example, harnessing the SPR excitation of metal nanoparticles would enable the photovoltaic devices to be physically thin but optically thick leading to the cost-effective as well as efficient photovoltaic devices. The gold (Au) and silver (Ag) nanoparticles are well known for their SPR in photovoltaics [1]. However their high cost is detrimental to the price sensitive applications such as large area photovoltaics. The cost-effective copper nanoparticles (CuNPs) are becoming an attractive alternative to costly Au and Ag since CuNPs exhibit good electrical conductivity and SPR absorption peak in visible region, and of comparable intensity which are important for charge transport and light trapping respectively in photovoltaics. Several prior reports demonstrated that like Au and Ag NPs, integration of CuNPs enhanced the efficiency of photovoltaic devices by taking advantage of SPR [2].

In addition to photovoltaic application, the CuNPs bear a wide range of other potential applications such as antibacterial agents, drug delivery, biosensors, catalysts, conductive ink and cooling fluid. The CuNPs have been synthesized by a variety

of physical and chemical methods, most recently by biological and green methods. All methods have a certain advantages along with limitations. One of the most common methods for the synthesis of metal nanoparticles is chemical reduction method where metal ions in a medium are reduced to elemental metal using various reducing agents and surfactants [3].

To the best of our knowledge, the stability examination of colloidal CuNPs via careful and thorough monitoring of their SPR band positions and intensities against time has not widely been reported. With this methodology, we can study the changing trend of SPR properties (peak positions and intensities) during the stability period beyond which their SPR properties fall into rapid decline. In the present work, the SPR bands of colloidal AgNPs and Cu-capped CuNPs in different reacting media (distilled water and ethylene glycol) were scrutinized so as to evaluate their stability [4].

2. EXPERIMENT

In the synthesis of colloidal AgNPs, two solutions of the ascorbic acid (0.02 M) and silver nitrate $AgNO_3$ (0.01 M) were separately prepared in distilled water and mixed under strong magnetic stirring. Various molar amounts of stabilizing agents (either polyethylene glycol (PEG 6000) or polyvinyl pyrrolidone (PVP 30000)) were added to the mixed solution. The solution pH was adjusted up to 12 by adding NaOH. After stirring at room temperature for about 1 hour, a solution of $NaBH_4$ (0.1 M) in distilled water was added to the mixture under continuous strong stirring for about 10 min. The initial blue color of the reaction mixture turned yellow and eventually dark yellow indicating the formation of AgNPs. The same NP synthesis was also carried out in different medium [ethylene glycol (EG) [4]]. In the preparation of $Cu_{core}Ag_{shell}$ NPs, various concentrations of silver salt ($AgNO_3$ to Cu^{2+} ratios of 0.1:1, 0.2:1 and 0.3:1) were added to the AgNPs solution. Core-shell Ag-CuNPs were formed by transmetalation where the reduction of Cu ions by Ag takes place directly on the surface of AgNPs. The colloidal NPs were characterized by UV-vis spectrophotometer (Thermo Scientific GENESIS 10S) in the wavelength range of 300-900 nm. The NPs were transferred to Si substrates by dip-coating for size estimation. The flow charts and preparation of steps of Core-shell Ag-CuNPs are shown in Figure 1 (a-b).

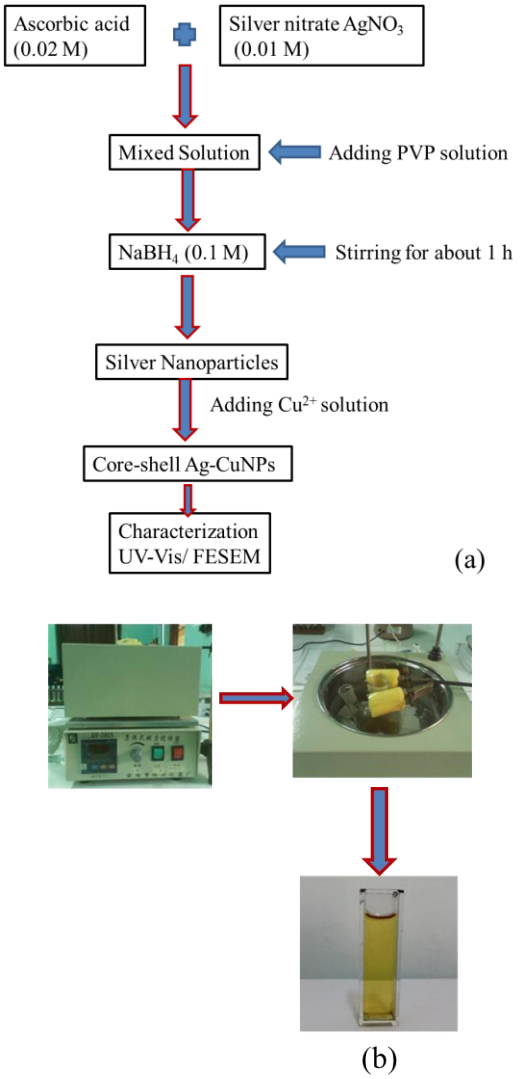


Figure 1 (a-b). The flow charts and preparation of steps of Core-shell Ag-CuNPs

2.1 Characterization Methods-UV-Vis Spectrophotometry

UV-Vis Spectrophotometry is a technique to measure either absorbance or transmittance of the sample in ultraviolet and visible region. The optical properties of

silver nanoparticles were measured by UV-Vis spectrophotometer: (GENESYS 10S UV-Vis (Materials Research Lab, Department of Physics, and University of Mandalay) shown in Figure 2. The extinction spectra of NPs were recorded in the wavelengths ranging from 300-900 nm. A glass substrate was used as baseline scanning before the measurement. The extinction of NRs was obtained from the transmission (T %) of NPs by using the equation [Extinction (%) = 1- T %].

Figure 2. UV-Vis Spectrophotometry

2.2. Field Emission Scanning Electron Microscope (FESEM)

The surface morphological features of metal nanoparticles were assembled on Si substrate by field-emission scanning electron microscopy: Agilent 8500 FE-SEM (Defence Services Science and Technology Research Centre (DSSTRC), Pyin Oo Lwin), as shown in Figure 3.

Figure 3. Field Emission Scanning Electron Microscopy (Agilent 8500FE-SEM)

3. RESULTS AND DISCUSSION

3.1. Effect of Stabilizing Agents Concentration

In the first part of my present work, the effect of stabilizing agents PVP, PEG, and AA on optical extinction of silver nanoparticles. Figure 4 shows the maximum optical extinction spectra for colloidal AgNPs with different stabilizing agents PVP, PEG, AA:$AgNO_3$ (3:1). According to the spectroscopic results, The optical maximum extinction peak position at 419 nm in PVP, 425 nm in PEG and 405 nm in AA with extinction values are 0.54, 0.40 and 0.26 for different stabilizing agents PVP, PEG and AA. The highest extinction peak is observed at stabilizing agent PVP, it attributing to the smaller size of nanoparticles in PVP.

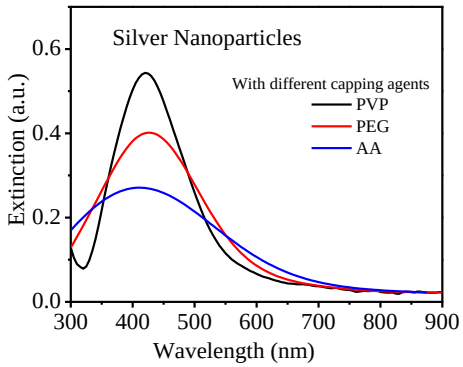


Figure 4. The maximum optical extinction spectra for colloidal AgNPs with different capping agents PVP, PEG, AA :$AgNO_3$ (3:1)

3.2 Time Evolution on Stability of Colloidal Silver Nanoparticles

In the second part of this research work, the stability of colloidal AgNPs with stabilizer (PVP) concentrations in distilled water by monitoring the time evolution of the SPR bands of AgNPs colloids. Figure 5 (a) shows the time evolution of the SPR bands of AgNPs colloids (PVP to $AgNO_3$ ratio of 3:1) during the record time 0 day up to 45 day and (b) Plots of SPR peak intensity against maximum wavelength for colloidal AgNPs. Time recording has started immediately upon adding PVP into the mixture solution. At the reaction time of 0 day (as-obtained) up to 45 day, the extinction peak of AgNPs colloids were red-shifted from 419 nm to 470 nm and peak intensity decreased indicating the bigger nanoparticles formation of colloidal AgNPs. Upon ageing up to 45 days, the SPR peak of NPs retain at this position. Thus, the stability time of colloidal AgNPs (PVP to $AgNO_3$ ratio of 3:1) is 45 days.

Following this, we examined the stability of AgNPs colloids by using different stabilizer (PEG) in the same reaction medium (distilled water). The same procedure is applied for stability. Figure 6 (a) shows the time evolution of optical extinction spectra (SPR bands) and (b) plots of SPR peak intensity against maximum wavelength for AgNPs colloids with PEG: $AgNO_3$ ratios of 3:1. Applying the same stability extraction protocol as in the PEG case, at the reaction time of 0 day (as-obtained) up to 45 day, the extinction peak of AgNPs colloids were red-shifted from 425 nm to 467 nm and peak intensity decreased indicating the bigger nanoparticles formation of colloidal AgNPs. It is observed that the stability period of AgNPs colloids with 45 days for PEG stabilizer.

Figure 7 shows time evolution of optical extinction spectra for colloidal AgNPs with AA: $AgNO_3$ (3:1) and (b) the plots of SPR peak intensities against maximum wavelength for colloidal AgNPs. It is clearly seen in the plots that the SPR peak position is red- shifted (dispersion of NPs) and peak intensity decreased. The extinction peak position is red-shifted from 405 nm to 467 nm at the record time of 0 day (as-obtained) up to 45 day. These record times are indeed the stability time for colloidal AgNPs since they degrade beyond this period. It is noted from this study that the AA: $AgNO_3$ provides the maximum stability period of 45 day. They can stable up to 45 days for all stabilizing agent (PVP), (PEG) and (AA). The 45-days-stability of colloidal AgNPs is still sufficient for their potential applications. Figure 8 shows the plots of SPR peak position and intensity against record time 0 day up to 45 day for colloidal AgNPs with stabilizing agents PVP, PEG, AA (3:1). According to the above results, the highest optical extinction peak was observed at stabilizing agents PVP.

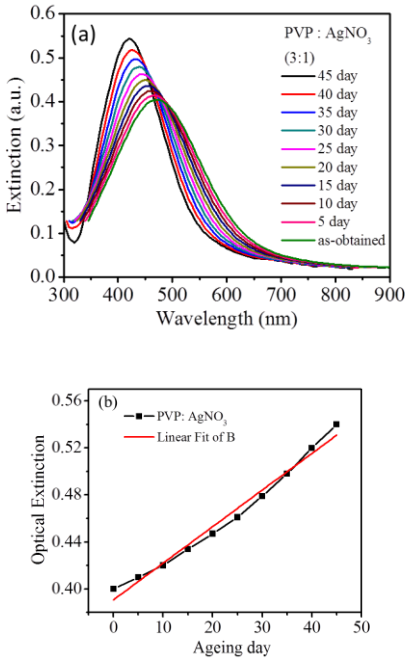


Figure 5. (a) Time evolution of extinction spectra for colloidal AgNPs with PVP: $AgNO_3$ (3:1) and (b) Plots of SPR peak intensity against record time for colloidal AgNPs

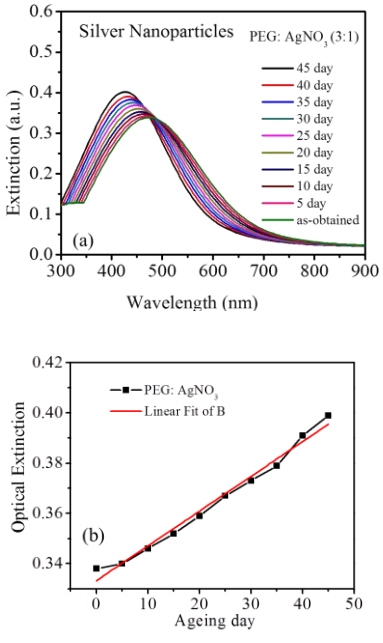


Figure 6. (a) Time evolution of extinction for colloidal AgNPs with PEG: $AgNO_3$ (3:1) and (b) Plots of SPR peak intensity against record time for colloidal AgNPs in distilled water

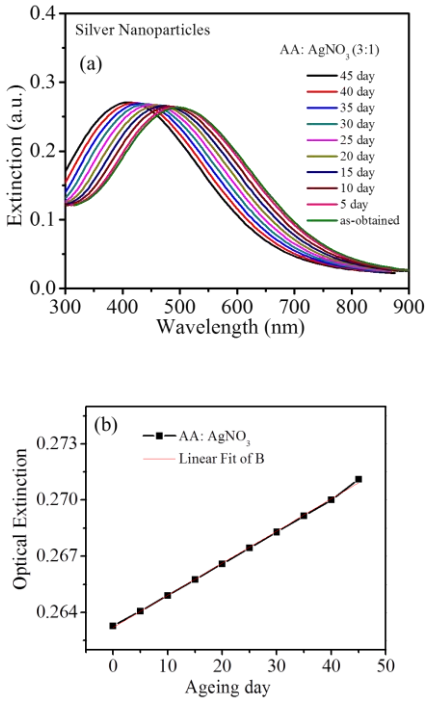


Figure 7. (a) Time evolution of optical extinction spectra for colloidal AgNPs with AA: $AgNO_3$ (3:1) and (b) Plots of SPR peak intensity against record time for colloidal AgNPs

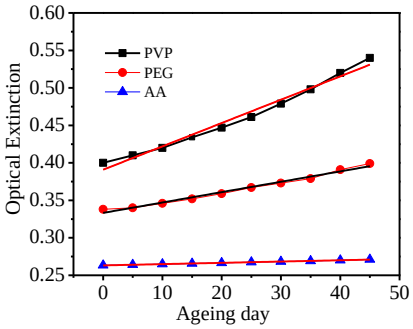


Figure 8. Time evolution of optical extinction spectra for colloidal AgNPs with different stabilizing agents PVP, PEG, AA: $AgNO_3$ (3:1)

3.3 Effect of Capping Agent (Cu^{2+}) Concentration on Extinction of $Ag_{core}Cu_{shell}$ NPs

The next step of my present work, the effect of capping agents concentration on optical extinction of $Ag_{core}Cu_{shell}$ NPs. Figure 9 shows the maximum optical extinction spectra $Ag_{core}Cu_{shell}$ NPs in EG using different molar ratios of $AgNO_3$: Cu^{2+} (0.1:1, 0.2:1, 0.3:1). Upon increasing Cu^{2+} concentration, the optical extinction peak was blue shifted and peak intensity decreased. Shifted of extinction peak cab be attributed to the smaller nanoparticles size of Ag_{core}-Cu_{shell} nanoparticles. In synthesized nanoparticles, there are two SPR peaks corresponding to SPR of Ag at 471 nm and SPR of Cu at 592 nm with (0.1:1), 449 nm and 563 nm with (0.2:1) and 438 nm to 552 nm with (0.3:1) suggesting the formation of Ag-Cu NPs with. Upon increasing capping agent concentration (0.1 up to 0.3), the optical extinction peak blue-shifted and peak intensity decreased. It attributed to the smaller nanoparticles size of A-Cu NPs.

The colloidal Ag-Cu NPs were transferred onto the silicon substrates and their sizes were estimated using field emission scanning electron microscopy (FESEM). Figure 10 (a-c) shows the FESEM image of $Ag_{core}Cu_{shell}$ NPs in EG using different molar ratios of $AgNO_3$: Cu^{2+} (0.1:1, 0.2:1, 0.3:1). The average NPs size is 90 nm for the molar ratio of 0.1:1, 80 nm for the molar ratio of 0.2:1 and 70 nm for molar ratio 0.3:1. It was found that Ag NPs size decreased with increasing capping agent concentrations which agreed well with our previous speculation (decreased NP size attributed to the blue shift of extinction peak).

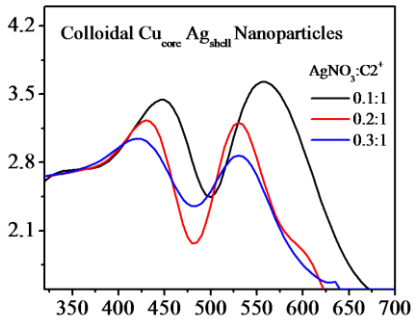


Figure 9. The maximum optical extinction spectra $Ag_{core}Cu_{shell}$ NPs in EG using different molar ratios of $AgNO_3$: Cu^{2+} (0.1:1, 0.2:1, 0.3:1)

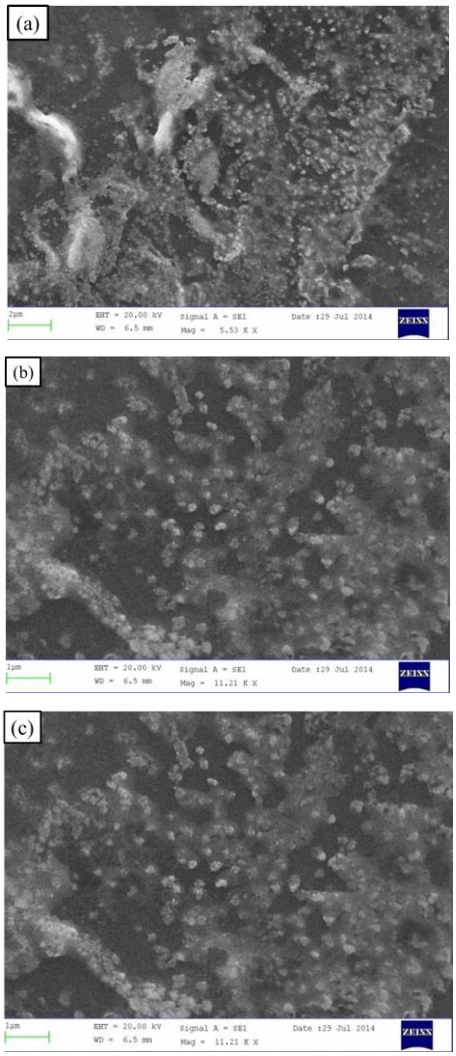
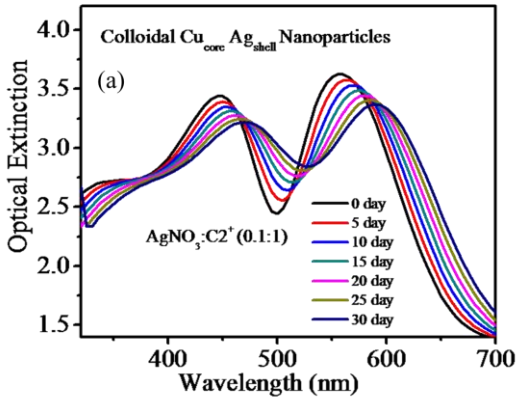


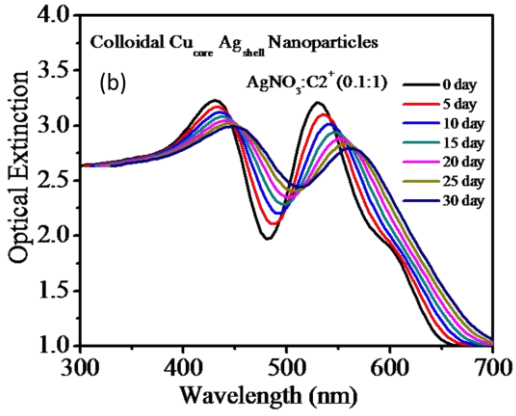
Figure 10. (a-c) The FESEM image of $Ag_{core}Cu_{shell}$ NPs in EG using different molar ratios of $AgNO_3$: Cu^{2+} (0.1:1, 0.2:1, 0.3:1)

3.4 Time Evolution on Stability of Ag_{core}-Cu_{shell} Nanoparticles

In the next attempt, colloidal AgNPs were capped with Cu shell using different amounts of Cu^{2+} and the stability of Ag-CuNPs was tested in EG medium. Figure 11 (a) shows the optical extinction spectra of colloidal Ag_{core} Cu_{shell} NPs with ratios of $AgNO_3$:Cu^{2+} (0.1:1) recorded for the time period of 30 days. Upon ageing 0 day (as-obtained) up to 30 day, the extinction peak intensity decreased and peak position red-shifted. This ageing study indicates that the maximum extinction peak observed at 0 days (as-obtained).

Figure 11 (b-c) shows the time evolution of optical extinction spectra $Ag_{core}Cu_{shell}$ NPs in EG using different molar ratios of $AgNO_3$: Cu^{2+} (0.2:1, 0.3:1). It is noted that the changing trend is similar for the Ag-Cu NPs with $AgNO_3$:Cu^{2+} (0.2:1 and 0.3:1) where the SPR peak positions (Ag_{core} and Cu_{shell}) was red shifted and peak intensities are decreased upon ageing 30 days. With $AgNO_3$:Cu^{2+} (0.2:1), the SPR peak are shifted to the longer wavelength regions (first peak 429 nm to 449 nm) and (second peak 528 nm to 563 nm). With stabilizing agent concentrations (0.3:1), the SPR peak is shifted from first peak 425 nm to 438 nm to and (second peak 530 nm to 552 nm) respectively. Thus the maximum extinction period for Ag_{core} and Cu_{shell} NPs is 0 day (as-obtained) for stabilizing agent concentration (0.1:1, 0.2:1 and 0.3:1). The optimum condition of Ag_{core}-Cu_{shell} NPs in the present work is as-obtained for all ratios in EG medium.





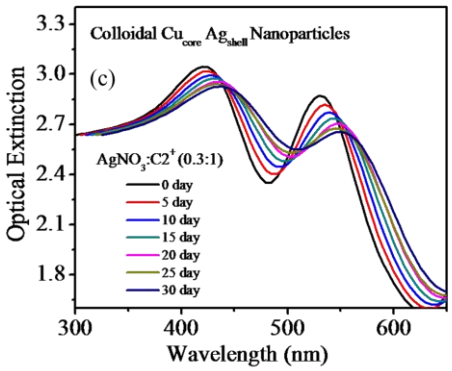


Figure 11. (a-c) Time evolution of optical extinction spectra Ag_{core}-Cu_{shell} NPs in EG using different molar ratios of $AgNO_3$: Cu^{2+} (0.1:1, 0.2:1, 0.3:1)

4. CONCLUSION

The application potential of nanoparticles in catalysis ranges from fuel cell to catalytic converters and photocatalytic devices. Catalysis is also important for the production of chemicals and it depends on the size and shape of metal nanostructures. Colloidal silver nanoparticles (AgNPs) were prepared in reaction media (distilled water) with stabilizers (PVP, PEG and AA) by chemical reduction method and the metal (Cu) capping on AgNPs was achieved by transmetalation method. The synthesis methods used are simple, rapid and economical. A careful and thorough examination on time evolution of surface plasmon resonance peak positions and intensities of colloidal Ag and Ag-Cu NPs in reaction media (EG) with different concentration (0.1:1,0.2:1, 0.3:1) was carried out to evaluate their stability period. The observed stability period of colloidal AgNPs is 45 day with stabilizer PVP, PEG and AA. The Cu capping decreases the stability of Ag-Cu NPs to as-obtained 0 days. It is found that the SPR peak positions are red shifted while the peak intensities decreasing in DW (reaction continuing). In its real applications, the instability of NPs would not guarantee to explore an exact processing-structure-property performance relationship for a given system which is indispensable for future process development and system design. The size estimation of $Ag_{core}Cu_{shell}$ NPs with FESEM image is observed at 90 nm for stabilizing agent (0.1:1), 80 nm for (0.2:1) and 70 nm for (0.3:1). Upon increasing precursor ($AgNO_3$) concentrations, the size of nanoparticles was decreased, it attributed to the good solubility of nanoparticles solution at precursor concentration (0.3:1).

ACKNOWLEDGEMENTS

We would like to thank Dr Tin Tun Aung, Acting Rector, Pakokku University. We are also thankful to Dr Win Win Ei (Pro-rector) and Dr Khin Khin Htoot, (Pro-rector), Pakokku University. We would like to express my gratitude to Dr Htun Win (Professor and Head), Dr Win Mar (Professor) and Dr Win Myint Kyu (Professor), Department of Physics, Pakokku University, for their kind permission to give us the opportunity to do research.

REFERENCES

[1] H.X. Zhang, U. Siegert, R. Liu, W.B. Cai, Facile fabrication of ultrafine copper nanoparticles

in organic solvent, Nanoscale Res. Lett. 4 (2009) 705

[2] T.M. Dang, T.T. Le, E.F. Blanc, M.C. Dang, Synthesis and optical properties of copper nanoparticles prepared by a chemical reduction method.

[3] G.M. Blanc, M.C. Dang, Synthesis and optical properties of silver nanoparticles prepared by a polyol method.

[4] P.M. Tanc, M.C. Dang, Synthesis and optical properties of silver-Copper nanoparticles prepared by a polyol method.

Authors

Biography

1. **First Author:**

Dr Thaung Hlaing Win

Lecturer

Department of Physics

Pakokku University.

PARTICLES SIZE AND STRUCTURAL PROEPERTIES OF COLLOIDAL SILVER NANOPARTICLES BY GREEN SYNTHESIS METHOD

Yin Yin Htwe[1], Htet Htet[2], and Thaung Hlaing Win[3]

[1,2,3] Department of Physics, Pakokku University

[1]yinyinpku777888@gmail.com,[2]htethtetphys0011.11@gmail.com,[3]thaung.naing001@gmail.com

ABSTRACT

Synthesis of silver nanoparticles (AgNPs) using plant resources has so many advantages over conventional methods like chemical synthesis, due to its environmental friendly activity and low cost for efficient production. The present study is a novel work where the synthesis of silver nanoparticles using natural dyes extract as a potential reducing agents for dry and fresh Caribbean Trumpet (Tubeuiaaurea) flower (1% - 5%) with the help of 1mM of silver nitrate solution. Thus the synthesized nanoparticles were characterized using UV-Vis spectrophotometer, Field Emission Scanning Electron Microscope (FESEM) and X-Ray Diffraction (XRD). The optical maximum absorption peak of the synthesized nanoparticles was observed at 420 nm, using UV-Vis spectrophotometer. In FESEM images, the average nanoparticles size was observed at ~ 52 nm and 69 nm for dry and fresh Caribbean Trumpet flowers.

KEYWORDS:

Absorbance, reducing agent, crystalline structure, green synthesis, silver nanoparticles

1. INTRODUCTION

Nanotechnology offers fields with effective applications, ranging from traditional chemical techniques to medicinal and environmental technologies. AgNPs have emerged with leading contributions in diverse applications, such as drug delivery, nanomedicine, chemical sensing, data storage, textiles, the food industry, photocatalytic organic dye–degradation activity and antimicrobial agents [1]. Despite the contradictions reported on the toxicity of AgNPs, its role as a disinfectant and antimicrobial agent has been given considerable appreciation. The available documented data and the interest of the community in this field prompted us to work on plant-mediated green synthesis and biological activities of AgNPs. Nanotechnology explores a variety of promising approaches in the area of material sciences on a molecular level and silver nanoparticles (AgNPs) are of leading interest in the present scenario. This review is a comprehensive contribution in the field of green synthesis, characterization, and biological activities of AgNPs using different biological sources [2].

Different methods like electrochemical reduction, photochemical reduction, heat evaporation and biological are used for the synthesis of silver nanoparticles. As reported in literature, these methods are very expensive and also involve in the use of hazardous chemicals which may cause the biological and environmental risks. Currently, silver nanoparticles synthesis from plants extract is receiving a greater attention because of environment friendly behavior. The plants extract works as reducing agent as well as capping agent in synthesis of nanoparticles. Different plants extract have been successfully used for the synthesis of metal nanoparticles [3].

Different green synthesis methods are used for the synthesis of silver nanoparticles. Green synthesis defined as the practice of biologically friendly elements like plants, bacteria and fungi for nanoparticles synthesis.

These striking green approaches are free of the short falls as per conventional synthetic approaches [4]. One of them is the bio-organism in which plant extract is being used. It is environment friendly, having non-toxicity effects as compared to chemical reduction methods. The plants extract works as reducing agent as well as capping agents making steady and exact shape of the AgNPs. Bio molecules in the plant extract such as amino acids, vitamins, enzymes, and proteins work to reduce the concentration of silver ions Ag^{+}. In this present work, different flower extracts (1% up to 5%) are used for this purpose to reduce the concentration of silver ions. The effect of dyes concentration on optical and size of AgNPs were investigated by UV-Vis Spectrophotometry and Field Emission Scanning Electron Microscope (FESEM). And then, the crystal nature and surface-morphology feature of AgNPs was calculated by Debye Scherrer formula. Finally, the obtained results of NPs size were compared by the size of FESEM image.

2. MATERIALS AND METHOD

2.1 Preparation of Flower Extract

Fresh Caribbean Trumpet flowers were collected from Pakokku University. The flowers were washed thoroughly using tap water four to five times, incised into small pieces, drying in the sun light for about one day and were grind into fine powder. For extract preparation, 1 g of fine powder of flowers was added into 100 ml of ethanol solvent. Then it was stirring for about 2 minutes. The obtain extract was filtered using filter paper and the extract obtained was filtered through Whatman No. 1 filter paper and stored at 4·C for further use.

2.2 Precursor (silver nitrate solution) Preparation

Silver nitrate solution was used as precursor for the synthesis of silver nanoparticles from the flowers extract. 1mM solution of silver nitrate was prepared using ethanol solvent.

2.3 Synthesis of Colloidal Silver Nanoparticles (AgNPs)

The green synthesis method was followed for the synthesis silver nanoparticles. For the preparation of silver nanoparticles (AgNPs) colloids, silver nitrate ($AgNO_3$) was used as a metallic precursor and the extract of flower with different concentrations was used reducing agent as well capping agents. The extract exhibited strong potential in rapid reduction of silver ions for synthesis of silver nanoparticles. For the reduction of Ag^{+} ions, 1g of Caribbean Trumpet (S_1) was added drop wise into 100 ml of 1 mM aqueous solution of silver nitrate and stirring for about 2 minutes [5]. The change in color was observed from light yellow to dark yellow which indicated the formation of silver nanoparticles. Similarly, 2g, 3g, 4g and 5g of flowers extract was suspended for the preparation of S_2, S_3, S_4 and S_5 respectively. The experimental setups for the synthesis of colloidal AgNPs of the present work are shown in Figure 1.

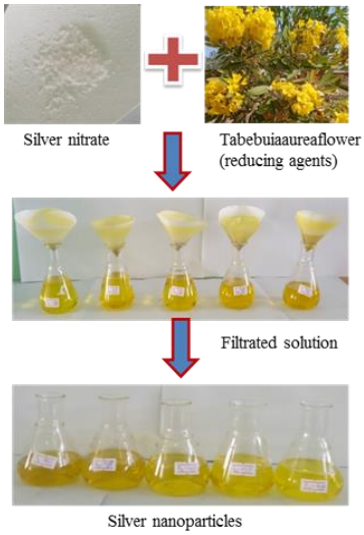


Figure 1. Sample preparation for the green synthesis colloidal AgNPs

3. CHARACTERIZATION OF METAL NANOPARTICELS

The characterization methods used are UV-Vis spectrophotometry, field emission scanning electron microscopy (FESEM) and X-rays Diffraction (XRD).

3.1 UV-Vis Spectrophotometry

UV-Vis Spectrophotometry is a technique to measure either absorbance of the sample in ultraviolet and visible region. The optical absorption and transmission spectra of novel metal nanoparticles were recorded in the wavelengths ranging from 300-1100 nm. The optical properties of colloidal nanoparticles

were measured by UV-Vis spectrophotometer, as shown in Figure 2.

3.2. Field Emission Scanning Electron Microscope (FESEM)

The surface morphological features of metal nanoparticles were assembled on Si substrate by field-emission scanning electron microscopy: Agilent 8500 FE-SEM (Defence Services Science and Technology Research Centre (DSSTRC), Pyin Oo Lwin), as shown in Figure 3.

3.3. X-Ray Diffraction (XRD)

The structural and surface morphological properties of metal nanoparticles were characterized by X- ray diffraction,(RIGAKU Multiflex)using CuKα radiation (λ = 1.5418 Å) as shown in Figure 4. The 2θ scan range was from 30˚-70˚ with a step of 0.03˚. Operating conditions are 50 kV and 40 mA.

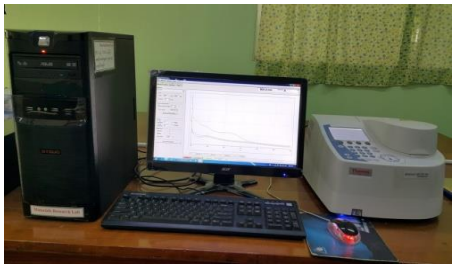


Figure 2. UV-Vis Spectrophotometry (GENESYS 10S)

Figure 3. Field Emission Scanning Electron Microscopy (Agilent 8500FE-SEM)

Figure 4. RIGAKU Multiflex X-Ray Diffractometer

4. RESULTS AND DISCUSSION

4.1. Effect of Reducing Agents Concentration on Absorption of AgNPs

The optical absorption properties of colloidal AgNPs were explored varying with the reducing agent (dry flower extract) concentrations (1%, 2%, 3%, 4%, 5%). Figure 5 shows the optical absorption spectra of colloidal AgNPs. As shown in Figure 6, the optical absorption peak positions of AgNPs colloids with different reducing agents concentration are observed at 420 nm for 1%, 429 nm for 2%, 442 nm for 3% and 450 nm for 4% and 462 nm for 5%, respectively. It is the formation of colloidal AgNPs for all molar concentrations [6]. Upon increasing reducing agents concentration, the optical absorption peak position of colloidal AgNPs was red-shifted from 420 nm to 462 nm and then the absorption peak intensity decreased. From the above observation (red-shifted optical absorption peak positions and decreased peak intensity), it is suggested that the larger nanoparticles size is formed.

The optical absorption properties of colloidal AgNPs were explored varying with the reducing agent (fresh flower extract) concentrations (1%, 2%, 3%, 4%, 5%). Figure 6 shows the optical absorption spectra of colloidal AgNPs. As shown in Figure 6, the optical absorption peak positions of AgNPs colloids with different reducing agents concentrations are observed at 423 nm for 1%, 437 nm for 2%, 450 nm for 3% and 453 nm for 4% and 478 nm for 5%. Upon increasing reducing agents concentrations, the optical absorption peak positions of colloidal AgNPs red-shifted from 423 nm to 478 nm and then the absorption peak intensity decreased as well. From the variation of optical absorption peak positions and peak intensity, the size of nanoparticles can be altered. From the above observation (red-shifted optical absorption peak positions and decreased peak intensity), it is suggested that the larger nanoparticles size is formed. This speculation is supported for the size estimation of nanoparticles [7]. At the larger reducing agents concentrations (natural dyes), the optical absorption property of colloidal AgNPs is not strong enough for the light harvesting process.

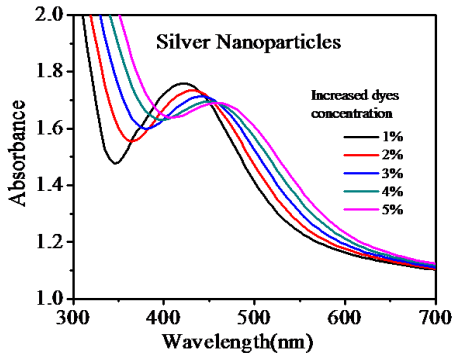


Figure 5. Optical absorption spectra of colloidal AgNPs with different dry Caribbean Trumpet flower concentrations

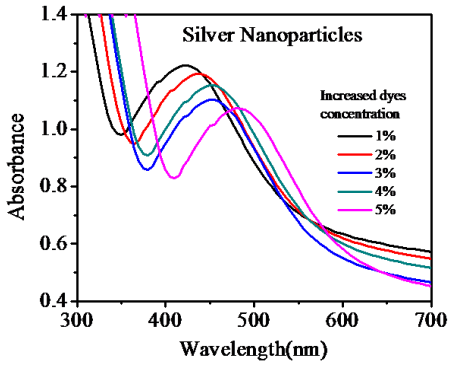
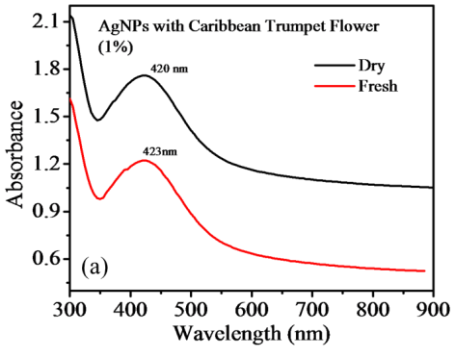
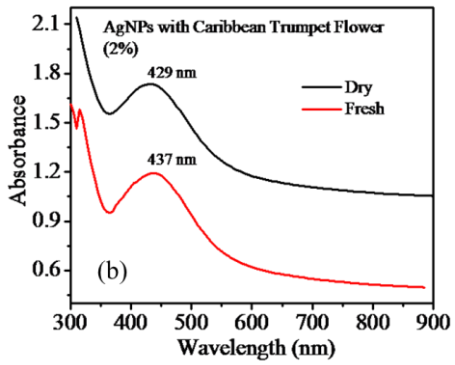


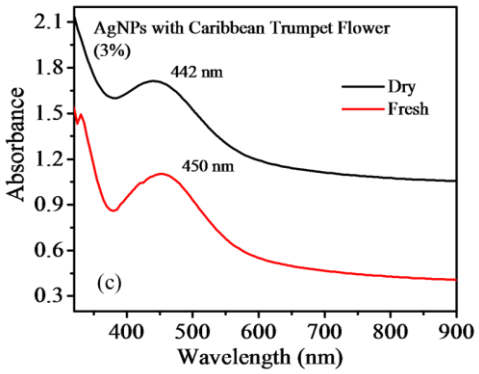
Figure 6. Optical absorption spectra of colloidal AgNPs with different fresh Caribbean Trumpet flower concentrations

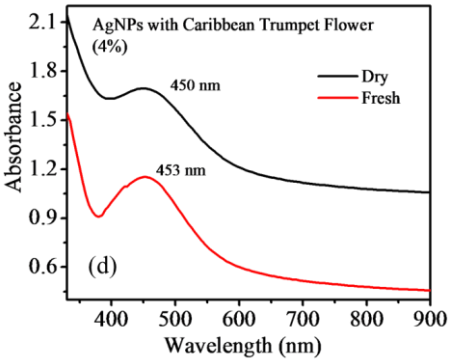
4.2 Effect of Reducing Agent (Dry and Fresh) on Absorption of AgNPs

The reducing agent (natural dyes) plays an important role in the formation of Ag NPs and can adjust the surface plasmon resonance (SPR) band positions, thus controlling the aspect ratio of nanoparticles. We studied the effect of reducing agent (dry and fresh) on optical absorption of colloidal Ag NPs. Figure 7 (a-e) shows the absorption spectra of colloidal Ag NPs with different reducing agent concentrations for dry and fresh Caribbean Trumpet flower. With fresh flower extract, absorption peak maximum λ_{max} changes from 420 nm into 423 nm, 429 nm into 437 nm, 442 nm into 450 nm, 450 nm into 453 nm for Ag NPs with reducing agents concentration of 1%, 2%, 3%, 4% and 1% respectively. It is obvious that the absorption peak maximum (λ_{max})) is red-shifted. It suggests that the formation of larger nanoparticles with increasing reducing agents concentration.









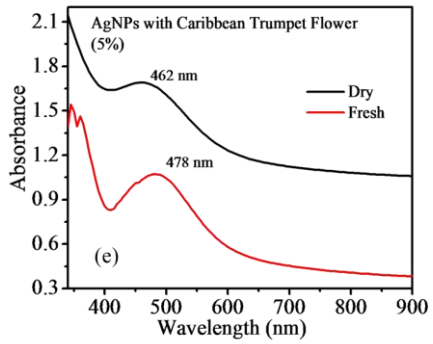


Figure 7. (a-e) Optical absorption spectra of colloidal AgNPs with dry and fresh dyes concentrations (1%,2%,3%,4%, 5%)

4.3 Effect of Reducing Agent Concentration (1% and 5 %) on Absorption of AgNPs

Figure 8 (a) shows the optical absorption spectra of colloidal AgNPs with dry Caribbean Trumpet flower concentrations (1% and 5%). Upon increasing reducing agents concentration, the optical absorption peak position was red-shifted and the absorption peak intensity decreased. Red–shifted is attributed to the larger nanoparticles formation of colloidal silver nanoparticles. Figure 8 (b) shows the optical absorption spectra of colloidal AgNPs with fresh Caribbean Trumpet flower concentrations (1% and 5%). The similar results were observed with fresh flower extract of reducing agents concentration (1 % and 5 %).

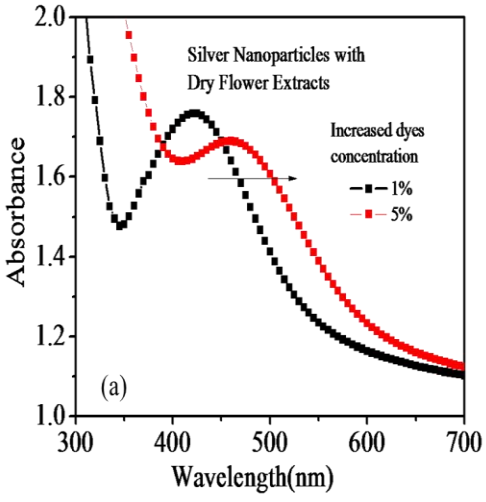


Figure 8. (a)Optical absorption spectra of colloidal AgNPs with dry Caribbean Trumpet flower concentrations (1%, 5%)

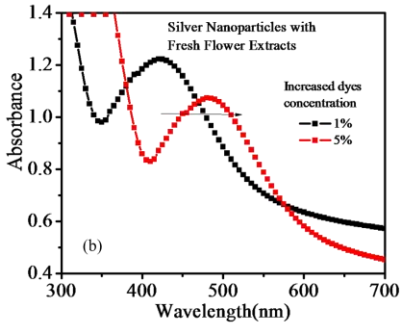


Figure 8. (b) Optical absorption spectra of colloidal AgNPs with fresh Caribbean Trumpet flower concentrations (1%, 5%)

4.4 Effect of Dyes Concentration on Size Estimation of AgNPs

The surface features of AgNPs were measured by field emission scanning electron microscopy (FESEM). Figure 9 (a-b) shows the FESEM micrograph of colloidal AgNPs with dry Caribbean Trumpet flower concentrations (1% and 5%). The average particle sizes of colloidal AgNPs with dry natural dyes extracts are ~ 54 nm for 1% (highest absorption peak) and 68 nm for concentrations 5 % (lowest absorption peak). The average size of AgNPs increased with increasing reducing agent of natural dyes. Like the red-shift of optical absorption result, the largest average size of colloidal AgNPs was observed at the higher reducing agent concentrations.

Figure 10 (a-b) shows the FESEM micrograph of colloidal AgNPs with fresh Caribbean Trumpet flower concentrations (1% and 5%). The average particle sizes of colloidal AgNPs with fresh natural dyes extracts are ~ 57 nm for 1% (highest absorption peak) and 69 nm for concentrations 5 % (lowest absorption peak). The average size of AgNPs increased with increasing reducing agent of natural dyes.

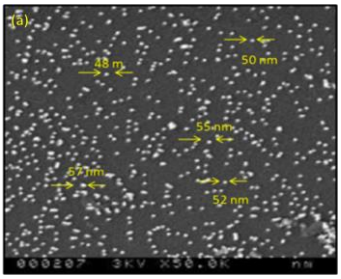


Figure 9. (a) FESEM images of colloidal AgNPs with dry Caribbean Trumpet flower concentration (1%)

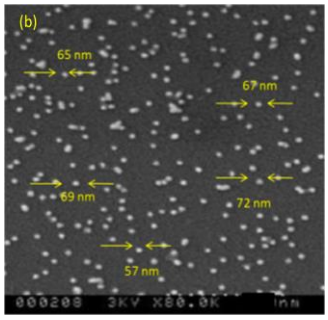


Figure 9. (b) FESEM images of colloidal AgNPs with dry Caribbean Trumpet flower concentration (5%)

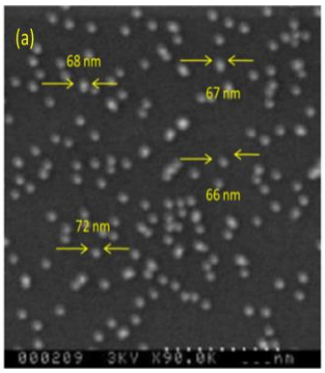


Figure 10. (a) FESEM images of colloidal AgNPs with fresh Caribbean Trumpet flower concentration (1 %)

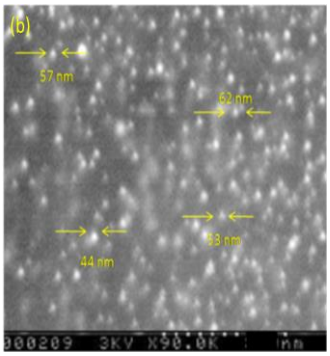


Figure 10. (b) FESEM images of colloidal AgNPs with fresh Caribbean Trumpet flower concentration (5%)

4.5 Ageing Effect on Absorption properties of colloidal AgNPs

The stability of Ag NPs with dry Caribbean Trumpet flower (optimum concentration 1%) was examined upon ageing the colloids for 0 day up to 15 day. Figure 11 (a) shows the absorption spectra of colloidal AgNPs solutions (0 day (as prepared) and aged for 5, 10, 15 days). Upon ageing for 0 day up to 5 day, the absorption peak positions of Ag NPs remained unchanged at 420 nm but the absorption peak intensity significantly increased. Upon ageing for 10 day and 15 days, the absorption peak intensity significantly decreased. Thus, the stability of colloidal silver nanoparticles with natural dyes 1% is within 5 day.

Figure 11 (b) shows the absorption spectra of Ag NPs colloids with fresh Caribbean Trumpet flower (optimum concentration 1%) as-prepared and aged for 5, 10,15 and 20 days. The absorption peak of Ag NPs is increased after 5 and 10 days. This ageing study indicates that the stability of Ag NPs colloids starts deteriorated beyond 10 days-ageing.

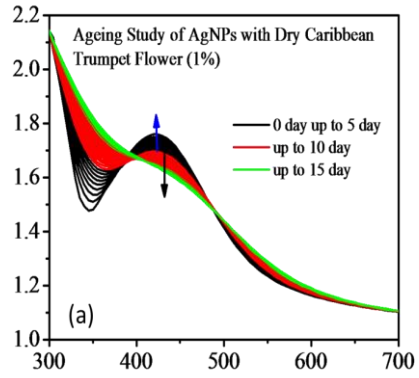


Figure 11. (a) Stability of colloidal AgNPs with dry dyes concentrations (1%)

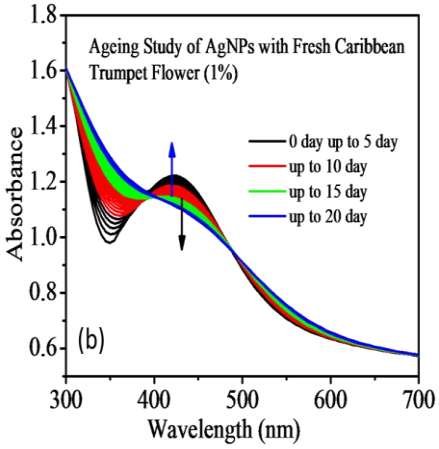


Figure 11. (b) Stability of colloidal AgNPs with fresh dyes concentrations (1%)

4.6 Structural and Size Estimation of AgNPs

The XRD analysis confirmed the crystalline nature of the nanoparticles as shown in Figure 12. The diffraction peaks at 38.2° 44.64°, 64.48°, and 78.4° related to 111, 200, 220 and

222 facts of the face centred cubic (FCC) crystal lattice, corresponded to silver nanoparticles. The extra peak obtained at 2θ nearly equal to 82 may be due to the bio-organic phase crystallization over silver nanoparticles surface. The crystallinity of the synthesized silver nanoparticles using flower extract 1% of dry and fresh was examined through X-ray diffraction (XRD) as shown in Figure 9. The size of the nanoparticles was calculated based on the Debye Scherrer equation:

$$D = k\lambda/\beta\cos\theta.$$

In the above equation, represents particle diameter size, k: a constant with a value of 0.9, λ: X-ray source wavelength (0.1541nm), β and θ represent the FWHM (full width at half maximum), and diffraction angle. So, in the present study, the average crystalline nanoparticles size was observed to be approximately 52 nm for all lattice planes.

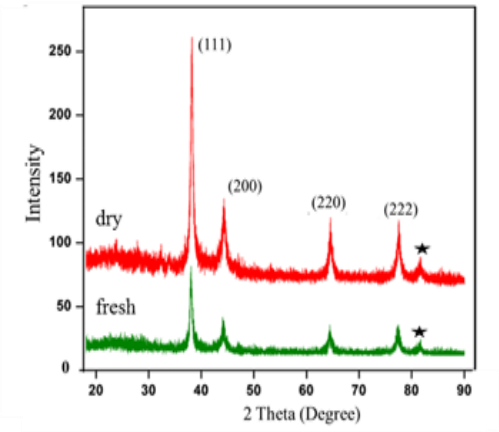


Figure 12. XRD patterns of Ag NPs for dry and fresh Caribbean Trumpet flower extract

5. Conclusion

Nanoparticles are used increasingly in catalysis to boost chemical reactions. Silver nanoparticles (AgNPs) colloid was prepared by green synthesis method. Optical absorption properties of AgNPs colloid was explored varying with the reducing agents concentration (dry and fresh natural dyes) 1% up to 5%. Upon increasing reducing agents concentration, the optical absorption peak positions of colloidal AgNPs were red-shifted and then the absorption peak intensity decreased. It may be due to the larger nanoparticles formation of colloidal silver nanoparticles. The average size of colloidal silver nanoparticles (AgNPs) was obtained from the field emission scanning electron microscopy (FESEM) micrograph. The obtained average size of silver nanoparticles with reducing agents (dry and fresh flower extract) concentration are ~ 52 nm and ~ 69 nm for concentration 1%. This observation is agreed with above speculation (red-shift of the absorption peak). So, in the second study, the average crystalline nanoparticles size was calculated from the XRD images by using Debye Scherrer equation. According to the above results, the average nanoparticles size is approximately ~ 52 nm for dry flower extracts and ~ 68 nm for fresh flower extracts (agrees with our FESEM results). The stability of AgNPs was also studied with optimum reducing agent concentration 1%. In this ageing study, the stability of natural dyes is over 5 day and 10 day for dry and fresh reducing agent of Caribbean Trumpet flowers. The extra peak obtained at 2θ nearly equal to 82° may be due to the bio-organic phase crystallization over silver nanoparticles surface.

ACKNOWLEDGMENTS

We would like to thank Dr Tin Tun Aung, Acting Rector, Pakokku University. We are also thankful to Dr Win Win Ei (Pro-rector) and Dr Khin Khin Htoot, (Pro-rector), Pakokku University. We would like to express my gratitude to Dr Htun Win (Professor and Head), Dr Win Mar (Professor) and Dr Win Myint Kyu (Professor), Department of Physics, Pakokku University, for their kind permission to give us the opportunity to do research.

References

[1] X.W. Guoyun*et al.*, 2014. Journal of Nanoscience.

[2] T. Markvart*et al.*, 2003. Journal of Apply Physics.

[3] H. Hahn *et al*., 1997.Nanostruct Mater.

[4] D.R. Duran N *et al*., 2012. Synthesis of Silver Nanoparticles.

[5] F.M. Al-Khateeb*et al.*, 2018. Apply Physics and Material Science.

[6] A.C. Chan *et al.*, 2005. Journal OfPhysical Chemistry.

[7] M.L. Almeida *et al., 1983.* Journal of Crystal Growth.

Authors Biography

First Author – Daw Yin Yin Htwe

Demonstrator
Department of Physics
Pakokku University

Second Author – Daw Htet Htet
Demonstrator
Department of Physics
Pakokku University

Third Author – Dr Thaung Hlaing Win
Lecturer
Department of Physics
Pakokku University

FINETUNING WHISPER FOR DOMAIN SPECIFIC MULTILINGUAL ASR APPLICATION

Sai Myat Htoo[1], and Ei Ei Moe[2]

[1,2] Department of Information Science, University of Technology (Yatanarpon Cyber City)
[1]saimyathtoo23@gmail.com, [2]eieimoe1989@gmail.com

ABSTRACT

As the demand for accurately converting spoken words to text grows, technology for Automatic Speech Recognition (ASR) has advanced significantly. This study focuses on OpenAI's Whisper, a new ASR model that works with multiple languages. A detailed approach is employed to analyze the design, performance, and potential improvements. English, Korean, and Japanese are chosen for this study because they are widely used and important globally. English is chosen because it's used in many areas like business and education worldwide. Korean is included because it's becoming more important in East Asia, especially in entertainment and technology. Japanese is also considered because of Japan's influence in technology and culture. Utilizing OpenAI's Whisper small model for the fine-tuning process, the best Word Error Rate (WER) is achieved after training on speech materials: 12.0% for English, 21.4% for Japanese, and 23% for Korean. This study aims to gauge Whisper's effectiveness across diverse languages, ultimately contributing to the refinement of multilingual ASR technology.

KEYWORDS

Automatic Speech Recognition, OpenAI Whisper, Speech To Text, Mel-Frequency Cepstral Coefficients, Word Error Rate, Transformer Architecture, Self Attention, Multi-Head Attention

1. INTRODUCTION

The human voice is a unique communication tool rich with emotion and nuance. Converting this complexity into written text is challenging, but Speech-to-Text (STT) technology effectively bridges the gap. STT transforms spoken language into text, greatly enhancing how exchange information in the digital age.

Speech-to-text (STT) technology has a broad impact across various fields. It enables real-time transcription in professional settings like meetings and lectures, aiding collaboration and knowledge sharing. For those with hearing impairments, STT provides essential communication access. In language learning, it helps with pronunciation practice and offers immediate feedback. Content creators use STT for transcription, generating subtitles, and documenting interviews.

Despite its potential, STT faces significant obstacles to widespread adoption. Challenges like noise intrusion from background sounds, traffic, and environmental disruptions can distort speech, causing transcription errors. Additionally, managing multiple speakers requires algorithms that can distinguish overlapping voices and accurately parse individual utterances.

The diverse dialectal nuances and accents present another formidable hurdle. Human languages exhibit a range of pronunciations and intonations that often transcend rigid grammatical conventions. To genuinely democratize access to information, STT must navigate the intricacies of accents and dialects with precision, acknowledging the

inherent diversity of human communication.

Beyond technical challenges, ethical considerations cast shadows on the horizon. Privacy concerns arise, as the training data for STT algorithms often comes from real-world recordings, prompting questions about data ownership, consent, and potential biases within the models. Additionally, the growing reliance on STT for communication and information access requires a thoughtful evaluation of its impact on social interactions and linguistic evolution.

This system contributes to the field of automatic speech recognition (ASR) by fine-tuning OpenAI's Whisper model for domain-specific, multilingual applications across English, Japanese, and Korean using the Common Voice 16.0 dataset. By leveraging this dataset, the model adapts to a variety of speakers and dialects, improving its generalization across languages. Fine-tuning process focuses on domain-specific tasks, allowing the ASR system to accurately transcribe context-sensitive speech in specialized fields such as business, healthcare, and education. Additionally, this system enhances cross-linguistic adaptability, demonstrating Whisper's ability to handle typologically diverse languages while maintaining high transcription accuracy. Benchmarking the fine-tuned model's performance against standard ASR systems, highlighting significant improvements in word error rate (WER) and transcription quality.

The rest of the paper is organized as follows: literature review is described in section 2. Background theory is shown in section 3. Whisper model is presented in section 4. Experimental results are shown in section 5. Finally, conclusion is given in section 6.

2. LITERATURE REVIEW

In 2020, Eva Sharma, Guoli Ye, and Wenning Wei et. al created a new training method that mixed real speech with synthetic speech generated by a specific TTS model trained on the target keywords. They suggested a new biasing technique within the RNN transducer to focus on keywords in both real and synthetic speech during training[6]. This technique helped the model learn keyword patterns that worked across different speaking styles and acoustic environments.

In 2020, Kevin Chu, Leslie Collins, and Boyla Mainsah employed a comprehensive approach to enhance speech intelligibility across diverse environments. Automatic Speech Recognition (ASR) utilized the Kaldi toolkit, training a GMM-HMM acoustic model with MFCC features and optimizing language models and pronunciation dictionaries. The Speech Synthesis Model, powered by the Merlin synthesizer, integrated duration and acoustic models trained on neural networks to generate speech waveforms with precise linguistic and spectral characteristics. A rigorous Listening Test Procedure engaged twenty native English speakers to evaluate speech intelligibility under varied conditions, including anechoic and reverberant settings, and synthesized speech derived from ASR-generated word sequences [8]. These methods collectively advanced our understanding and interventions for speech clarity in real-world scenarios.

In 2023, Xianrui Zheng, Yulan Liu, and Deniz Gunceler et. al used an RNN-T model to process data end-to-end. The system explored three methods to enhance the model, particularly for recognizing new words, while maintaining accuracy with existing ones. The first method blended synthetic and real data, dynamically selecting from both sources during training to better retain real audio characteristics. The second method involved keeping certain parts of the model unchanged, known as Encoder Freezing, which helped improve outcomes compared to altering the entire model [12]. Lastly, the system investigated Elastic Weight Consolidation (EWC), a technique that stabilizes the model and reduces errors when

encountering new words by addressing differences between real and synthetic data.

3. Background Theory

Converting spoken words into written text is a complex challenge that has significantly advanced Speech-To-Text (STT) systems, enhancing human-computer interaction. STT systems use acoustic models to extract features from raw audio, considering variations in tone, pronunciation, and noise. Techniques like Mel-Frequency Cepstral Coefficients (MFCCs) and spectrogram analysis help represent audio signals for further processing. Language models then interpret these features into coherent word sequences using grammar, vocabulary, and context, traditionally employing statistical methods like n-grams.

Transformer-based architectures have further advanced STT by capturing long-range dependencies between words, improving accuracy and fluency [7]. These deep neural networks analyze entire audio sequences simultaneously, enhancing performance in handling complex sentences, disfluencies, and overlapping speech [5]. Training [13] these models require substantial data, and manually aligning audio with transcripts is costly. Weak supervision, using loosely aligned or automatically generated transcripts, allows models to learn from larger datasets, improving resilience and adaptability.

As global discourse spans many languages, STT technology is increasingly embracing a multilingual approach. Advanced models now incorporate language identification and cross-lingual transfer techniques, breaking down language barriers and promoting global communication. As STT technology matures, addressing its societal impact and ethical considerations, such as data privacy, potential biases, and influence on communication patterns, is crucial [5]. Ensuring responsible development and deployment [3] practices is essential for fostering an inclusive and ethical future in human-computer interaction.

4. Whisper Model

OpenAI's Whisper has emerged as a robust player in the field of speech-to-text (STT) conversion, captivating both researchers and users with its multilingual capabilities and exceptional accuracy. To gain a comprehensive understanding of its abilities, it is essential to explore the theoretical foundations and intricately designed architecture that drive its performance.

4.1. Architectural Pillars

Whisper's architecture can be envisioned as a grand palace of sound, where each element plays a vital role in converting spoken words into written text.

4.1.1. Transformer Foundation

Central to the structure is the transformer, a deep learning architecture renowned for its capacity to capture extensive connections between words. Unlike analyzing isolated audio snippets, the transformer examines the entire audio sequence, fostering a nuanced comprehension of context and sentence structure. Figure 1 shows the overview of Transformer Architecture.

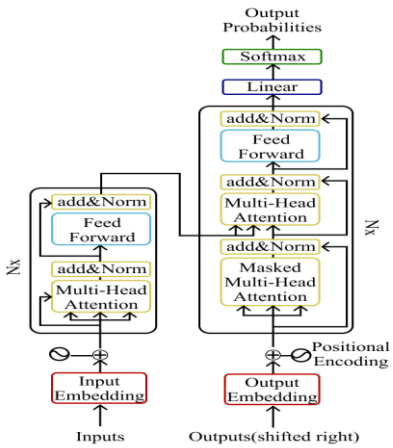


Figure 1. Transformer Architecture

4.1.2. Encoders & Decoders

Within transformer architecture, specialized sub-modules play crucial roles. Encoders analyze the acoustic features from raw

audio signal, examining complex interactions of frequencies and amplitudes. Decoders then use these features to generate the corresponding sequence of words in the target language [11]. Figure 2 shows overview of Whisper Encoder-Decoder Architecture.

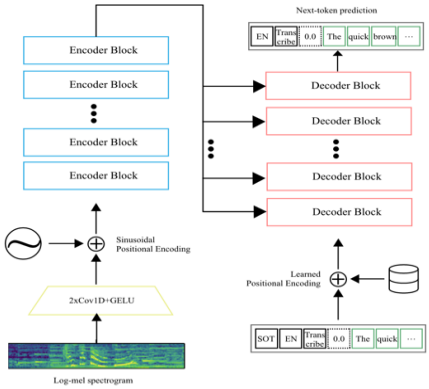


Figure 2. Overview of Whisper Encoder-Decoder Architecture

4.1.3. Weak Supervision Wisdom

Training Whisper relies on an extensive dataset of weakly labelled audio-text pairs, meaning text annotations may not perfectly align with audio segments. This approach allows Whisper to harness the richness of human speech data without meticulous manual labelling, enhancing scalability and adaptability.

4.2. Step-by-Step Symphony

Embark on a journey through Whisper's operation, observing the transformation of acoustic whispers into written words.

4.2.1. Acoustic Feature Extraction

The process begins with the raw audio signal. Whisper extracts meaningful features like Mel-Frequency Cepstral Coefficients (MFCCs) [9], capturing the essence of the audio for further stages as shown in Figure 3.

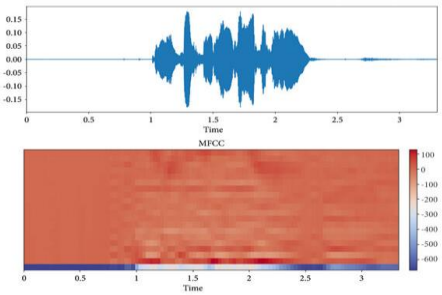


Figure. 3. Male happy speech data and its corresponding MFCC [1]

4.2.2. Encoder Analysis

These features are fed into the transformer's encoders, which delve into the acoustic patterns and relationships within the sequence.

4.2.3. Decoder Decoding

Armed with insights from the encoders, the decoders use their knowledge of language and context to sequentially construct a coherent text sequence.

4.2.4. Language Identification

Whisper can manage over 60 languages. An embedded module automatically identifies the predominant language in the input audio, guiding the decoding towards the appropriate linguistic context.

4.2.5. Continuous Refinement

Whisper employs methods like spectral augmentation to enhance its understanding of noisy and varied acoustic environments. It engages in continuous learning strategies, enabling ongoing adaptation and performance improvement with new data and languages.

4.3. Visualizing the Magic

Some key aspects of Whisper's workings are

4.3.1. Transformer Attention Mechanism

The encoders and decoders dynamically shift attention from one feature to another, assembling the puzzle of meaning [10].

This interaction [4] can be visualized with arrows connecting acoustic elements to corresponding words in the output text.

4.3.2. Weak Supervision and Data Diversity

Imagine an extensive library with books in diverse languages, some with precisely aligned chapters and others with loosely annotated content. This metaphor illustrates the abundance of weakly labelled data Whisper learns from, showcasing its ability to extract knowledge from imperfect sources [2].

4.4. The Future Beckons

Whisper's journey is just beginning, with vast potential ahead. It can enhance accessibility for the hearing-impaired, enable multilingual communication, and transform content creation. However, ethical considerations around data privacy and potential biases must be handled wisely to ensure this technology benefits humanity inclusively and fairly.

5. EXPERIMENTAL RESULT

5.1. DataSet

The Common Voice 16.0 dataset is a large collection of recordings in various languages, totaling 30,328 hours of MP3 audio paired with transcriptions. It includes additional speaker information such as age, gender, and accent, with 19,673 hours verified for accuracy. Covering 121 languages, it includes English, Japanese, and Korean for fine-tuning language tasks. The specifics of the dataset are outlined in Table I.

The English dataset (411k rows) is divided into validated (16.4k rows), invalidated (111k rows), train (114k rows), test (16.4k rows), and other (154k rows). Validated data has been reviewed and upvoted for quality, while invalidated data has been downvoted. Training and test sets consist of high-quality reviews, and the other category includes unreviewed data.

The Japanese dataset (180k rows) includes validated (6.09k rows), invalidated (12.8k rows), train (9.62k rows), test (6.09k rows), and other (146k rows). The Korean dataset (2.99k rows) is split into validated (235 rows), invalidated (237 rows), train (401 rows), test (282 rows), and other (1.83k rows).

Table 1. Statistics for Three Languages From Common Voice Dataset

Class	Validated	Invalidated	Train	Test	Other	Total
English	16.4k	111k	114k	16.4k	154k	411k
Japan	6.09k	12.8k	9.62k	6.09k	146k	180k
Korea	235	237	401	282	1.83k	2.99k
Total	22725	124037	124021	22772	301830	5939390

5.2. Experiment Setup

The experiment was conducted on Kaggle using a 16GB P100 GPU. PyTorch served as the main deep learning framework, with the Hugging Face Transformers library for model integration. Data was sourced from the Common Voice API. OpenAI Whisper was selected for fine-tuning. Data preparation involved processing datasets from Common Voice. Training took place on Kaggle with 10,000 iterations, aiming to refine OpenAI Whisper for improved multilingual speech recognition.

5.3. Performance Matrix

The performance of the speech recognition system was evaluated using the Word Error Rate (WER), a common metric in Automatic Speech Recognition. WER is based on the Levenshtein distance and is calculated at the word level. The

computation of WER involves the following formula:

$$WER = \frac{S+D+I}{N} = \frac{S+D+I}{S+D+C} \quad (1)$$

where

S is the number of substitutions,
D is the number of deletions,
I is the number of insertions,
C is the number of correct words,
N represents the total number of words in the reference transcript $(N = S + D + C)$.

Equation (1) offers a comprehensive assessment of recognition errors, accounting for substitutions and deletions relative to the total number of words in the reference transcript. The process of calculating with sample data is as follow:

- Prediction: What is Artificial intelligence and its applications?
- References: Wha is Artificial intelligent and is application?

WER = (2 + 2+ 0) / (2 + 2+ 3) = 4/7

=0.5714285714285714

5.4. Experiment Result

In this experiment, the primary objective was to refine OpenAI Whisper, an advanced Automatic Speech Recognition (ASR) model, utilizing Common Voice datasets in English, Japanese, and Korean. The fine-tuning process in Google Colab involves several steps:

5.4.1. Prepare Environment

Several well-known Python tools will be used to adjust the Whisper model. Datasets will be acquired and set up for training. Transformers and Accelerate libraries will be employed for handling and training the Whisper model. The soundfile package will be necessary for working with audio files, and model performance will be measured using jiwer. Progress tracking will be facilitated through TensorBoard, and a demo of the improved model will be created using Gradio. It is advisable to save the model as it is trained. During training, model checkpoints are uploaded directly to the Hugging Face Hub, this offers:

- Version control: Ensures no checkpoint is lost during training.
- Tensorboard logs: Keeps track of important metrics during training.
- Model cards: Documents what the model does and its use cases.
- Community: Makes it easy to share and collaborate with others.

Linking the notebook to the Hub is simple; just enter the Hub authentication token when prompted. The Hub authentication token can be found at https://huggingface.co/settings/tokens.

5.4.2. Load DataSet

Common Voice 16.0 is a dataset where people record sentences from Wikipedia in various languages, including English, Japanese, and Korean, spoken in the USA, Japan, and South Korea, respectively. It includes validated, training, and test subsets used for fine-tuning ASR models. To access the dataset, visit the Hugging Face Hub page for Common Voice (mozilla-foundation/common_voice_16_0) and accept the terms of use. The first 100 samples can be explored and filtered by language as shown in Figure 4.

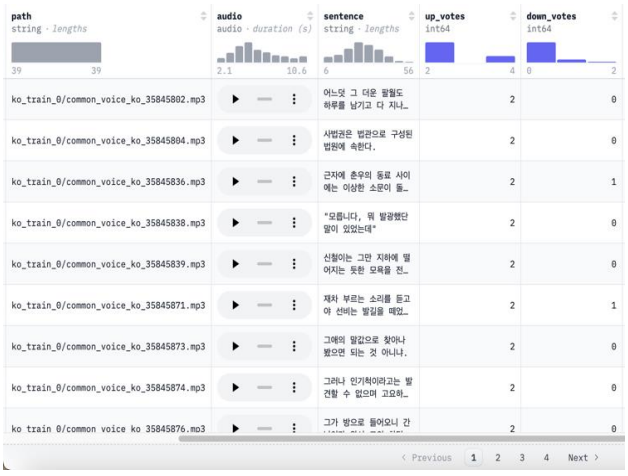


Figure 4. Sample Korea Dataset

The dataset features narrated speech with diverse speakers and recording quality. For English, approximately 114k training rows and 16.4k test rows are available. Japanese data includes 9.64k training rows and 6.09k test rows, while Korean has 401 training rows and 282 test rows. Fine-tuning the model will use on audio and sentence attributes, ignoring additional metadata like accent and locale to ensure the model is trained on essential components for accurate speech recognition.

5.4.3. Prepare Feature Extractor, Tokenizer and Data

To set up the ASR pipeline, three main components are needed:

- Feature extractor: This component processes the raw audio inputs.
- Model: It performs the sequence-to-sequence mapping.
- Tokenizer: This post-processes the model outputs into text format.

In Transformers, the Whisper model comes with its own feature extractor and tokenizer, named WhisperFeatureExtractor and WhisperTokenizer, respectively. Describe the details of each component in the next sections.

5.4.3.1. Load WhisperFeatureExtractor

The Whisper feature extractor processes speech inputs for automatic speech recognition (ASR). It works with one-dimensional arrays that represent speech signals, sampled at 16kHz to match the model's requirements. This ensures the audio inputs are consistent and prevents performance issues. The feature extractor standardizes all audio samples in a batch to 30 seconds by padding shorter samples with zeros and truncating longer ones. This uniformity eliminates the need for an attention mask. It then converts these standardized audio arrays into log-Mel spectrograms, which show the audio's frequency content over time. Whisper uses these spectrograms for further processing. Figure 5 shows overview of Whisper Encoder-Decoder Architecture.

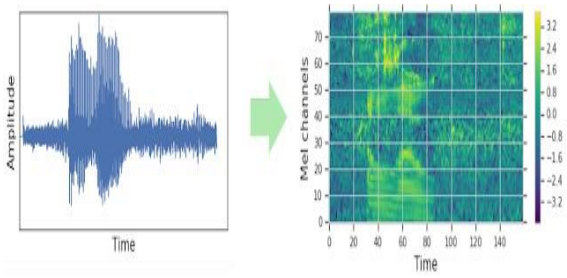


Figure 5. Conversion of sampled audio array to log-Mel spectrogram

5.4.3.2. Load WhisperTokenizer

The Whisper tokenizer converts model outputs into readable text. It's pre-trained on 96 languages and can be used directly without additional training. It includes special tokens for language and task identification, making it suitable for multilingual ASR tasks. To verify its correct functioning, a sample transcription can be encoded and decoded, ensuring the process works for English and two other languages.

5.4.3.3. Combine To Create A WhisperProcessor

To streamline the process, combine WhisperFeatureExtractor and WhisperTokenizer into a single class called WhisperProcessor. This class will handle both audio preprocessing and tokenization, simplifying the training process by managing only the processor and the model.

5.4.3.4. Prepare Data

To prepare the one-dimensional input audio array and its transcription for the Whisper model, the audio's sampling rate must be downsampled from 48kHz to 16kHz. This is achieved using the dataset's cast_column method, which resamples audio samples dynamically upon initial loading without altering the original audio. Once the first audio sample from the Common Voice dataset is reloaded, it is resampled to 16kHz, reducing the array to one amplitude value for every three original values, ensuring compatibility with the Whisper model.

To process the data, the function loads and resamples the audio using batch["audio"], with Datasets handling the resampling on the fly. The feature extractor then computes the log-Mel spectrogram input features from the audio array. Finally, the transcriptions are encoded into label IDs using a tokenizer. This function is applied to all training examples using the .map method, preparing the data for fine-tuning the Whisper model.

5.5. Training and Evaluation

With the data prepared, the training pipeline can be initiated using the Trainer. The process includes defining a data collator to prepare pre-processed data as PyTorch tensors for the model, and setting up evaluation metrics, specifically using the word error rate (WER), which requires a compute_metrics function. Additionally, a pre-trained model checkpoint needs to be loaded and configured correctly for training. Training arguments must also be defined to guide the Trainer in setting up the training schedule. Once the model is fine-tuned, it will be evaluated on the test data to ensure accurate transcription of speech in English, Japanese, and Korean.

5.5.1. Define a Data Collator

The data collator for a sequence-to-sequence speech model like Whisper handles input_features and labels independently. The input_features are initially padded to 30 seconds and converted to fixed-dimension log-Mel spectrograms, then batched into PyTorch tensors using the feature extractor's .pad method with return_tensors='pt'. Since these inputs already have fixed dimensions, no additional padding is required. For the labels, sequences are padded to the maximum length within the batch using the tokenizer's .pad method, and padding tokens are replaced with -100 to exclude them during loss computation. The start of the transcript token is also removed from the beginning of the label sequence, as it will be appended later during training. By utilizing the WhisperProcessor, which combines the feature extractor and tokenizer, the data collation process is streamlined.

5.5.2. Evaluation Metrics

For evaluation, the Word Error Rate (WER) metric, a standard measure for ASR systems, is used. This metric is loaded from the evaluate library and calculated via the compute_metrics function. The function replaces -100 in the label IDs with the pad_token_id to correctly handle padded tokens, decodes the predicted and label IDs to strings, and computes the WER between predictions and reference labels. These steps ensure accurate data handling and evaluation, facilitating effective model training and assessment for ASR tasks.

5.5.3. Load a Pre-Trained Checkpoint

Proceed to load the pre-trained Whisper small checkpoint. This can be easily done using Transformers. The Whisper model has token ids known as forced_decoder_ids, which are set as model outputs before autoregressive generation begins. These token ids control the transcription language and task for zero-shot ASR. However, for fine-tuning, these ids will be set to None, as the model will be trained to predict the correct languages (English, Korea, and Japan) and task (transcription).

Additionally, there are tokens known as suppress_tokens, which are completely suppressed during generation, meaning their log probabilities are set to -inf so they are never sampled. These tokens will be overridden to an empty list, indicating that no tokens are suppressed during fine-tuning.

5.5.4. Define the Training Arguments

In the final step, training arguments are defined to set up the training process. Key parameters include output_dir, which specifies the local directory for saving model weights and serves as the repository

name on the Hugging Face Hub. The generation_max_length sets the maximum number of tokens generated during evaluation. Intermediate checkpoints are saved and uploaded to the Hub every save_steps training steps, while evaluations are conducted every eval_steps steps. The report_to parameter determines where to save training logs, with options including "azure_ml", "comet_ml", "mlflow", "neptune", "tensorboard", and "wandb"; the default is "tensorboard" for logging to the Hub. These arguments, along with the model, dataset, data collator, and compute_metrics function, are passed to the Trainer to initialize the training process. The whole process is described in Figure. 6.

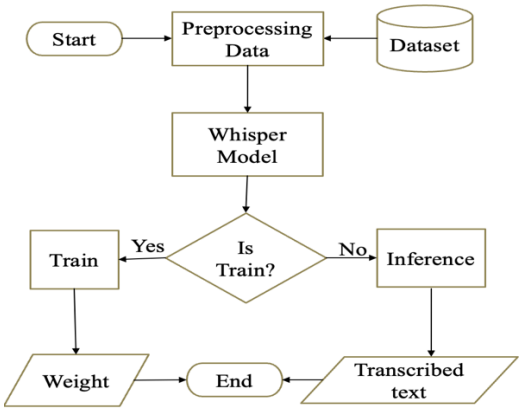


Figure 6. System Flow of Fine-Tuning Whisper Model

5.5.5. Training

This command trainer.train() will start the training loop, during which the model will be fine-tuned on the provided dataset using the specified training arguments. Training the model typically spans between 5 to 10 hours, depending on the GPU's capabilities or the one assigned in Google Colab. If a CUDA "out-of-memory" error occurs during training initialization, steps can be taken to address it.

Firstly, consider gradually decreasing the per_device_train_batch_size, reducing the amount of data processed in each batch. This adjustment may help alleviate memory constraints. Secondly, gradient_accumulation_steps can be utilized to compensate for the reduced batch size. This parameter accumulates gradients over multiple batches before performing a weight update, effectively simulating a larger batch size without increasing memory usage. Validation loss over steps for selected three languages are shown in Figure. 7 ,8 and 9.

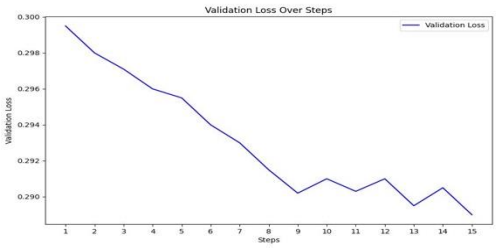


Figure 7. Validation loss over steps for English

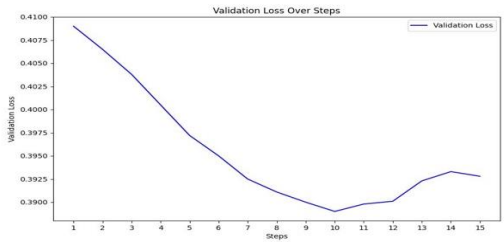


Figure 8. Validation loss over steps for Japan

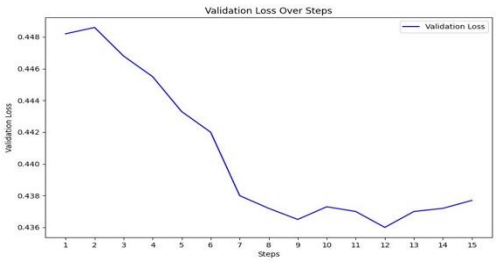


Figure 9. Validation loss over steps for Korea

Best Word Error Rate (WER) achieved after training speech material English is 12.0% and 21.4% for Japanese, 23% for Korean as shown in Figure. 10.

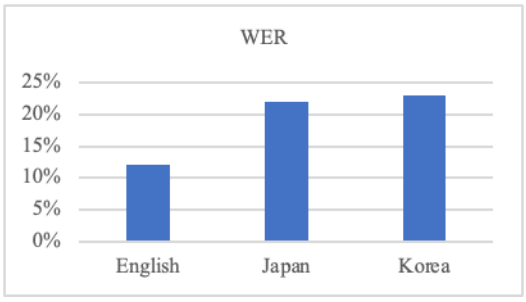


Figure 10. Word Error Rate for three languages

This experiment has demonstrated the effectiveness of finetuning the Whisper model for a domain-specific multilingual ASR application. Utilizing the Whisper Small model, remarkable Word Error Rates (WER) are achieved. Despite constraints imposed by using a small model, necessitated by hardware limitations such as those encountered on an M1 MacBook Pro with limited GPU and CPU performance, expectations and obtained impressive WER results are exceeded. This highlights the versatility of the Whisper model in accommodating diverse languages and its capacity to adapt to domain-specific ASR tasks, even under resource constraints.

6. CONCLUSIONS

The demand for accurate and efficient speech-to-text conversion has spurred notable advancements in Automatic Speech Recognition (ASR) technology. This study delved into OpenAI's Whisper, a cutting-edge multilingual ASR model constructed through a large-scale weak supervision framework. Through a comprehensive analytical approach, this system meticulously explored the model's intricate architecture and functionalities. It conducted a detailed performance analysis across diverse audio datasets, shedding light on its capabilities and limitations. Furthermore, the study systematically investigated potential enhancements and real-world applications of Whisper. The findings underscore the significance of Whisper in meeting the evolving needs of speech recognition technology and its promising prospects for future development and deployment in practical settings.

REFERENCES

[1] Abeer Talib Ali, Mohammed Zakariah, Prashant Kumar Shukla et.al (2022) "Human-computer interaction for recognizing speech emotions using multilayer perceptron classifier", *Journal of Healthcare Engineering*, pp5.

[2] Alec Radford, Jong Wook Kim, Tao Xu et.al (2022) "Robust speech recognition via large-scale weak supervision", *International Conference on Machine Learning*, pp2-5.

[3] Ankur Parikh, Oscar Täckström, Dipanjan Das et.al (2016) "A decomposable attention model," *In Empirical Methods in Natural Language Processing*, pp2249–2255.

[4] Ashish Vaswan, Noam Shazeer, Niki Parmar et.al (2017) "Attention is all you need," *In the Neural Information Processing Systems (NeurIPS) conference*, pp. 2–7.

[5] Dan Hendrycks, Xiaoyuan Liu, Eric Wallace et.al (2020) "Pretrained transformers improve out-of-distribution robustness,"*in ACL*, pp2744–2751.

[6] Eva Sharma, Guoli Ye, Wenning Wei, Rui Zhao et.al (2020) Adaptation of RNN transducer with text-to-speech technology for keyword spotting," *In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp7484-7488.

[7] Jianpeng Cheng, Li Dong, and Mirella Lapata (2016) "Long short-term memory-networks for machine reading," *in Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp551–561.

[8] Kevin Chu, Leslie Collins, Boyla Mainsah (2020) "Using automatic speech recognition and speech synthesis to improve the intelligibility of cochlear implant users in reverberant listening environments," *In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp 6929-6933.

[9] Manasi Kattel, Araju Nepal, Ayush Kumar Shah et.al (2019) "Chroma feature extraction," *in Proceedings of the Conference: Chroma Feature Extraction using Fourier Transform.*

[10] Romain Paulus, Caiming Xiong, and Richard Socher (2018) "A deep reinforced model for abstractive summarization," *in*

Proceedings of the International Conference on Learning Representations, arXiv preprint arXiv:1705.04304.

[11] Rosana Ardila, Megan Branson, Kelly Davis et.al (2020) "Common voice: massively-multilingual speech corpus," *in Proceedings of the Twelfth Language Resources and Evaluation Conference*, pp4218–4222.

[12] Xianrui Zheng, Yulan Liu, Deniz Gunceler, et.al (2021) "Using synthetic audio to improve the recognition of out-of-vocabulary words in end-to-end asr systems," *In IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp.5674-5678.

[13] Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos (2017) "A structured self-attentive sentence embedding," arXiv preprint arXiv:1703.03130.

COMPARATIVE ANALYSIS OF HASH ALGORITHMS PERFORMANCE IN DECENTRALIZED MESSAGING PLATFORM

Ko Ko Min[1], Hein Tun[2], Aung Ko Latt[3], and Nay Lynn[4]

[1,2,3,4] Department of Computer Technology, Higher Education Centre (Pyin Oo Lwin)
[1]kokokomin51.ct@gmail.com, [2]heintun.ct.dept@gmail.com,
[3]aungkolatt33049@gmail.com,[4]naylynn@gmail.com

ABSTRACT

Decentralized messaging platforms built on blockchain technology rely heavily on cryptographic hash functions to ensure secure and tamper-resistant storage of messages. The selection of an optimal hash algorithm is critical for the performance and scalability of these systems. This study provides a comprehensive comparative analysis of the performance characteristics of various hash algorithms suitable for use in decentralized messaging applications. We evaluate metrics such as hash rate, and throughput to identify the most efficient algorithms for different deployment scenarios. Our findings offer valuable insights for developers seeking to optimize the performance of blockchain-based messaging systems while maintaining robust security guarantees.

Keywords

Blockchain technology, Decentralized messaging, Cryptographic hash functions, Secure Hash Algorithms, Communication

1. INTRODUCTION

Blockchain technology has significantly altered communication as well as information transactions, which has led to the development of decentralized messaging applications which offer enhanced privacy, security, and resistance to censorship. These platforms rely extensively on cryptographic hash functions to ensure the integrity and authenticity of messages exchanged among users. The characteristic of the issue is of crucial significance to assure the integrity of the message. Researchers have to perform an analysis of the performance characteristics of hash algorithms appropriate for decentralized messaging for the purpose of making informed choices during the development and implementation of these systems.

This paper aims to provide a comprehensive analysis of the performance of various hash algorithms within the context of decentralized messaging platforms. We will thoroughly examine the fundamental concepts of hash algorithms, exploring their essential characteristics and analyzing different types, such as MD-5, Secure Hash Algorithms (SHA-1) and Secure Hash Algorithms (SHA-2 256,384 and512). By comparing the performance of these algorithms using appropriate metrics, we aim to identify the most suitable decisions for secure and efficient communication in decentralized messaging environments. This research will provide invaluable insights for developers and users alike, enabling them to make informed decisions about the choice of hash algorithms in their respective applications.

2. THEORETICAL BACKGROUND

2.1. Overview of decentralization

Decentralized messaging systems are an innovative way of communication that employs peer-to-peer (P2P) networks to enhance security and privacy. Decentralized platforms enable immediate interactions between users, reducing the possibility of security breaches and illegal access, as opposed to conventional messaging systems that rely on servers that are centralized. These platforms utilize blockchain

technology to generate irreversible records of messages, ensuring the integrity and authenticity of the data. Decentralized messaging systems typically utilize encrypted messaging, which ensures secure communication by protecting the identities of users and evidence from unauthorized access and modification. Decentralized messaging platforms are seen as necessary for user data privacy and digital communication trust [1].

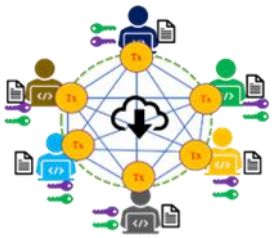


Figure 1. Nature of Decentralization

2.2. Cryptographic hash functions

Cryptographic hash functions are mathematical algorithms that map data of arbitrary size (called the "message") to a fixed-size string of bits (called the "hash value" or "message digest"). These functions must possess crucial properties such as preimage resistance, second preimage resistance, collision resistance, determinism, and efficiency [2].

2.2.1. Message Digest (MD-5)

The MD5 algorithm generates a 128-bit hash value by digesting input data of any size, which acts as a unique identity allowing users to verify the integrity of the content. It is widely employed to ensure the integrity of data received from the server throughout the file transfer process. Users are able to verify the integrity of the file they have downloaded by comparing the hash of it to the hash value tracked on the server. The process includes padding the input data, setting four 32-bit variables to constants, processing it in 512-bit blocks via bitwise operations and modular additions, and combining the four variables' results to get the hash value. The systematic approach used produces a unique hash that is crucial to ensuring the integrity of data [3].

MD5 uses four functions, each of which takes three 32-bit words as inputs and outputs one 32-bit word as a result. They apply the logical operators and, or, not and xor to the bits. The data of the 4 buffers, mainly A, B, C, and D, have been merged with the words of the input via the four auxiliary functions, denoted as F, G, H, and I. There are a total of four rounds, with each round containing 16 basic steps. The figure below represents one operation.

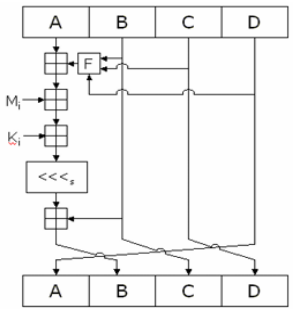


Figure 2. Principle of MD-5

Figure 2 represents the use of the auxiliary function F to the 4 buffers (A, B, C, and D) with the message word M_i and the constant K_i. The term "<<<s" represents a binary operation that shifts the bits of a value to the left by s positions. After all rounds are executed, the buffers A, B, C, and D contain the MD5 digest of the initial input.

2.2.2. SHA-1

The SHA-1 algorithm gets a message as input, that may have a maximum length of 2^{64} bits and produces a 160-bit message digest given output. The message gets processed by the compression function in blocks of 512 bits. Each block has been divided into sixteen 32-bit words, which represent M_t for t = 0, 1, ..., 15. The compression function possesses four rounds, each round comprised of a series of twenty stages. An entire repetition of the SHA-1 algorithm has 80 stages, utilizing a block size of 512 bits with a 160-bit linking variable, which produces the final 160-bit hash [4].

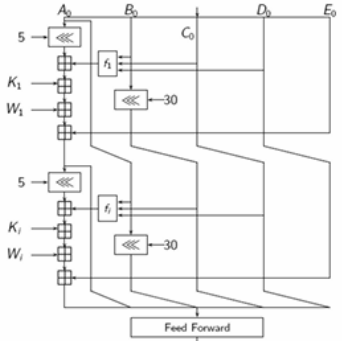


Figure 3. Working process of SHA-

2.2.3. SHA-2

The SHA-2 standard includes several hash algorithms, including SHA-2(256), SHA-2(384), and SHA-2(512), that are utilized to generate an easily understood summary (message digest) of encrypted data. The length of the generated messages digest changes from 256 to 512 bits, according to the hash function applied in each instance. Figure 4 represents the block diagram of the SHA-2 family algorithms. The hash function execution can be separated into two different stages [5]. In the preprocessing stage, the message is padded into a multiple of 512 or 1024 bits then parsing the padded message into N message blocks $B^0, B^1, ..., B^{(N-1)}$, where block size is 512 or 1024 bits. And in the hash computation stage, for each message block B_i, beginning with the message's schedule W_t, the next processes (1-4) are carried out to calculate the hash values H_0^i to H_7^i for the i^{th} block [5].

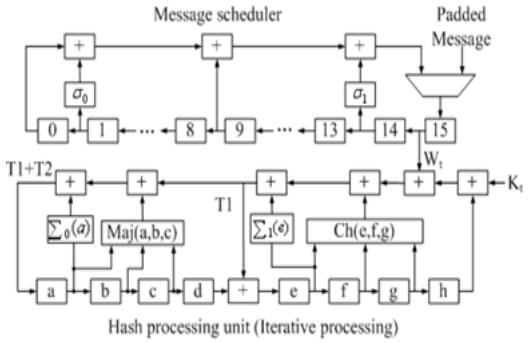


Figure 4. Working process of the SHA-2 family algorithm

3. LITERATURE REVIEW

Rashid et al. conducted a comparison study of hash algorithms to assess their effectiveness in ensuring secure data deduplication in cloud storage. The algorithms MD5, SHA-1, SHA-256, and SHA-512 have been evaluated based on components like hash value size, collision resistance, and processing time. The findings showed that SHA-256 and SHA-512 presented better results in both collision resistance and hash value size, but MD5 and SHA-1 showed faster computation time [6].

A study by Pandya performed a comparative analysis of the efficiency of Blake3, SHA-256, and SHA-512 hash algorithms over various hardware platforms. The researchers evaluated metrics such as hash rate, throughput, and memory utilization. The results indicated that Blake3 exhibited better results in terms of throughput and latency compared to SHA-256 and SHA-512. However the strength of this benefit differed based on the specific hardware and size of the input [7].

Chirag et al. proposed a blockchain-based decentralized messaging app to prevent misinformation and enhance security in educational systems. Their DMApp simply enables authorized individuals to exchange messages that are both safe and resistant to manipulation. The decentralized nature and authenticity features enable the identification of the original author of any misleading data. The authors highlighted the possible uses of blockchain and smart contracts in enhancing reliability and safety in decentralized messaging platforms [8].

5. RESULTS AND DISCUSSION

The implementation was carried out using Javascript, React, and Solidity and provided the performance results of hash algorithms. This language provides established classes for several algorithms, such as MD5, SHA-1, SHA256, SHA384, and SHA512 for decentralized messaging platforms. The performances of algorithms are calculated on the basis of Execution time, and throughput. Execution time refers to the duration an algorithm takes to process a given amount of data. The execution time of the algorithms in a decentralized messaging platform can be estimated by using the following formula:

$$T = \frac{C \times B}{F}$$

Where,

C = Number of cycles required to Number of cycles required to compute algorithm for a single byte

B = Total number of bytes to be hashed

F = Clock frequency of the processor (in Hz)

T = Execution time (in seconds)

Throughput generally refers to the rate at which data is processed, indicating the efficiency of each algorithm. The throughput

of the algorithms in a decentralized messaging platform can be calculated by using the following formula:

$$Throughput = \frac{TotalBytesHashed}{T}$$

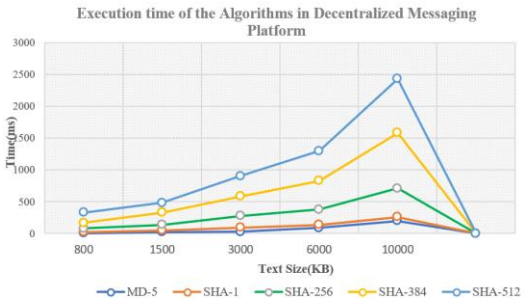


Figure 5. Execution time of the algorithms

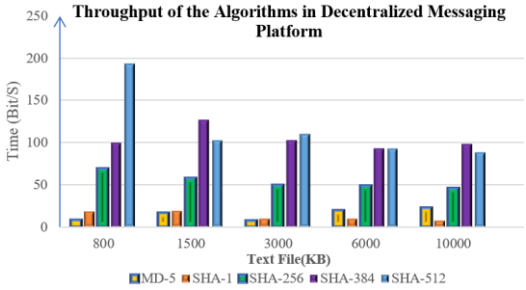


Figure 6. Throughputs of the algorithms

In our system, various sizes of text file inputs are used to find the performance such as execution time with milliseconds, and throughout with bit per second as shown in Figures 5 and 6. The text files from 1000 kb to 10000 kb are given as input to each algorithm one by one. Figures 5 and 6 represent the throughput of various hash algorithms MD5, SHA-1, SHA-256, SHA-384, and SHA-512 used in a decentralized messaging platform, measured in bits per second (Bits/S) across different text file sizes (KB). MD5 and SHA-1 exhibit relatively low execution times, indicating their faster processing speeds. However, they are less secure than the SHA-2 family algorithms. SHA-256 offers a balanced performance with moderate execution times, making it a suitable choice for applications requiring both security and efficiency. SHA-384 and SHA-512, while providing the highest level of security, have the longest execution times, particularly noticeable with increasing data sizes. This indicates that these algorithms require more computational resources, resulting in longer processing durations.

Figure 6 shows that SHA-256 and SHA-512 have the highest throughput, particularly for smaller text sizes (800 KB and 1500 KB), making them more efficient in handling data compared to other algorithms. As the text file size increases, the throughput for all algorithms tends to decrease, but SHA-256 and SHA-512 still maintain relatively high performance. MD5 and SHA-1, while faster in terms of processing time, show significantly lower throughput, indicating that despite their speed, they process less data per second due to their simpler cryptographic functions. This data suggests that SHA-256 and SHA-512 are more suitable for applications requiring higher data processing rates and stronger security, such as in decentralized messaging systems where data integrity and speed are crucial.

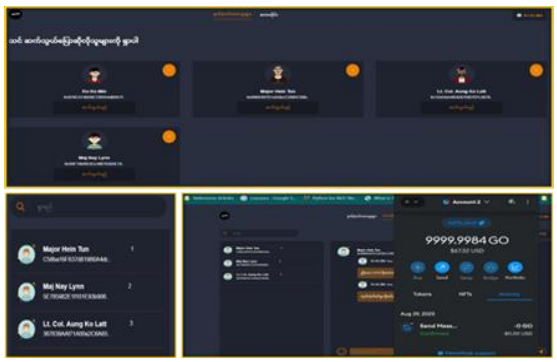


Figure 7. Decentralized Messaging Platform

5. CONCLUSIONS

MD-5 and SHA-1 in our messaging platform have high throughput due to their simpler cryptographic operations. However, these algorithms are considered insecure for critical applications due to vulnerabilities like hash collisions. On the other hand, SHA-256, SHA-384 and SHA-512 provide much stronger security but at the cost of higher execution times and lower throughput, making them less suitable for real-time applications. SHA-256 offers a balance between performance and throughput, making it a preferable choice for decentralized messaging platforms. Future work should focus on addressing scalability issues and exploring the integration of other cryptographic techniques to further enhance the security and efficiency of decentralized sharing systems. By continuing to innovate in this field, we can build more robust and trustworthy decentralized networks that meet the growing demands of our digital world.

ACKNOWLEDGEMENTS

I would like to express our deepest gratitude to Dr. Hein Tun, Dr. Aung Ko Lattand Dr. Nay Lynn for their invaluable guidance and support throughout this research progress. Their expertise in cryptography provided critical insights that significantly enhanced the quality of our work. And then, we would like to express our sincere thanks to all compassionate individuals who helped directly and indirectly during this work.

REFERENCES

[1] Gebhardt, Luisa, Marc Leinweber, Florian Jacob, and Hannes Hartenstein. "Grasping the Concept of Decentralized Systems for Instant Messaging." In Proceedings of the 17th Workshop in Primary and Secondary Computing Education, pp. 1-6. 2022.

[2] Mittelbach, Arno. "Hash combiners for second pre-image resistance, target collision resistance and pre-image resistance have long output." In Security and Cryptography for Networks: 8th International Conference, SCN 2012, Amalfi, Italy, September 5-7, 2012. Proceedings 8, pp. 522-539. Springer Berlin Heidelberg, 2012.

[3] Dhany, Hanna Willa, Fahmi Izhari, Hasanul Fahmi, Mr Tulus, and Mr Sutarman. "Encryption and decryption using password-based encryption, MD5, and DES." In International Conference on Public Policy, Social Computing and Development 2017 (ICOPOSDev 2017), pp. 278-283. Atlantis Press, 2017.

[4] Wang, Guoping. "An efficient implementation of SHA-1 Hash function." In 2006 IEEE International Conference on Electro/Information Technology, pp. 575-579. IEEE, 2006.

[5] Sklavos, Nicolas, and Odysseas Koufopavlou. "Implementation of the SHA-2 hash family standard using FPGAs." The Journal of Supercomputing 31 (2005): 227-248.

[6] Yeung, Mason. "SendingNetwork: Advancing the Future of Decentralized Messaging Networks." arXiv preprint arXiv:2401.09102 (2024).

[7] Pandya, Marut. "Performance Evaluation of Hashing Algorithms on Commodity Hardware." arXiv preprint arXiv:2407.08284 (2024).

[8] Jani, Chirag, Raaj Anand Mishra, and Anshuman Kalla. "Secure Blockchainized Decentralized Messaging Application (DMApp) for Educational Institute." Software Impacts 16 (2023): 100494.

Biography

Ko Ko Min obtained his Bachelor of Computer Science (B.C.Sc) from Defence Services Academy in 2008. Subsequently, he earned a Master of Engineering Technology (M.E.Tech) from the Moscow Institute of Aviation (MAI), Russia in 2014 and a Master of Computer Science and Engineering (M.C.Sc.E) degree from Korea National Defence University (KNDU) in South Korea in 2020. Currently, he serves as an assistant in the Department of Computer Science and also attending a Ph.D. in the 19th batch specializing in Computer Technology at the Higher Education Center of the Defence Services Academy.

Dr. Hein Tun received his B.C.Sc (Computer Science) from Defence Services Academy in 2003. He graduated M.E(Tech) in 2008 and also earned Ph.D from St. Petersburg State Marine Technical University (SMTU), Russia in 2012. He also earned M.P.A from International University of Japan (IUJ), Japan in 2020. He has served as an assistant lecturer at the Department of Computer Technology in Defence Services Academy.

Dr. Aung Ko Latt earned his B.C.Sc (Computer Science) from Defence Services Academy in 2001. He earned M.E(Tech) from MIET(Russia) in 2006 and graduated Ph.D(Computer Science) from Defence Service Academy in 2009. He has served as a Professor at the Department of Computer Technology in Defence Services Academy.

Dr. Nay Lynn obtained his Bachelor of Computer Science (B.C.Sc) from Defence Services Academy in 2006. He earned M.E.Tech from SouthWest State University (S.W.S.U) in 2011 and Ph.D from Kaluga State University (KSU), Russia in 2020. He has served as an assistant lecturer at the Department of Computer Technology in Defence Services Academy.

COMPARATIVE STUDY OF STRONG AND SOFT REDUCING AGENT CONCENTRATIONS ON SIZE EVOLUTION OF SILVER NANORODS

Thiriwin[1], Myint Myint Kyi[2], and Thaung Hlaing Win[3]

[1,2] Department of Engineering Physics, Technological University (Magway)
[3]Department of Physics, Pakokku University
[1]thiriwin1984@gmail.com,[2]maymoe17416@gmail.com,[3]thaung.naing001@gmail.com

ABSTRACT

The purpose of this experiment is to conduct synthesis of silver nanoparticles and explore their shape stability that affects optical property (referred to as localized surface plasmon resonance (LSPR). Silver nanorods have successfully synthesized by seed-mediated growth method. In this study, materials were used silver nitrate ($AgNO_3$) as a precursor, polyvinylpyrrolidone (PVP) as a capping agent and cetyltrimethylammonium bromide (CTAB) forms a bilayer structure around silver nanorods. The focus of this study was mainly placed on the effect of strong reducing agent ($NaBH_4$) and capping agent concentration on the size and optical extinction of nanorods by UV-vis spectrophotometry and field emission scanning electron microscopy (FESEM).

KEYWORDS

nanorods, reducing agent, UV-Vis, FESEM.

1. INTRODUCTION

Nanostructures have been extremely attractive because of their unique properties (e.g., optical, electronic, magnetic and chemical properties) and potential applications in nanoscience and nanotechnology. Important scientific and technological advances based on understanding, control of properties, process at the scale of atoms, molecules (nanometer scale) are taking place in laboratories around the world. Nanoparticles are usually the particles of size 1nm to 100 nm [1]. Physicochemical properties of nanoparticles such as mechanical, chemical, magnetic, optical or electrical properties are different compared to bulk materials for the volume to surface ratio.

In nanotechnology, nanorods are morphology of nanoscale objects. Each of their dimension ranges from 1-100 nm. One-dimensional (1D) metal nanostructures, such as nanorods, nanowires and nanotubes, have been a focused issue of intensive research because of their unique electrical, optical, magnetic, catalytic and other properties being distinctly different from their bulk counterparts. These fascinating properties of nanomaterials strongly depend on size, shape of nanoparticles, their interactions with stabilizers and surrounding media and also on the manner of their preparation [2]. Among the known nanoparticles, silver (Ag) is widely studied because of its characteristic optical and catalytic properties.

Preparation of metal nanostructures such as gold or silver of various shapes is very attractive for applications in plasmonics and sensorics. Plasmonic photovoltaics use plasmonic effects inherent to metallic nanostructures to enhance the power conversion efficiency of photovoltaic devices. In particular, gold and silver nanoparticles (NPs) strongly absorb light in the visible region as a result of the surface plasmon resonance. The resonance wavelength depends on the nanoparticle size and shape, aggregation state as well as the dielectric

constant of the surrounding medium [3]. The optical properties are related to the interaction between the metal conduction electrons and the electric field component of the incident electromagnetic radiation, which in the case of a few metals, such as gold and silver, leads to strong, characteristic absorption bands in the visible and near-infrared region of the spectrum, but not observed in the bulk material. Nowadays, the silver nanostructure research is currently an area of intense scientific research, due to a wide variety of potential applications in fields such as mechanics, chemical industries, electronics, energy science and catalysis [4].

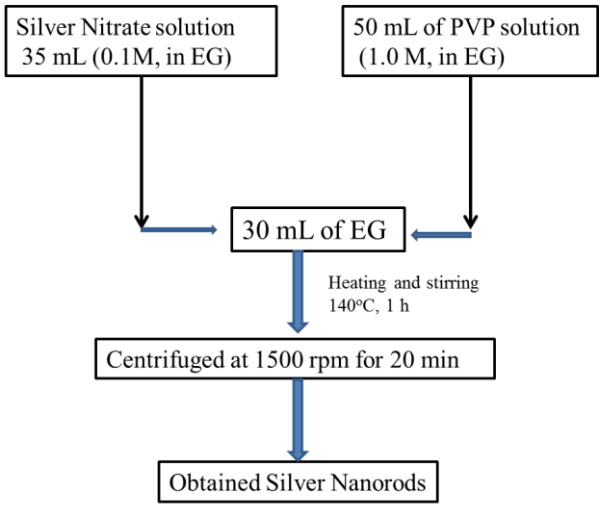
The various methods have been actively explored for the synthesis of 1D nanomaterials, including template method, seed-mediated growth method, wet chemical method, chemical reduction method and polyol reduction method. For instance, silver nanorods have been synthesized by seed-mediated growth method. The seed-mediated method is a method to produced metal colloids by heating technique using salt and a solvent like ethylene glycol (EG) and distilled water (DI). The capping agents are generally used to stable the particle and play a role as shape control through selectively binding the facets of nanocrystal [5]. In this experimental technique, the temperature is the most important parameter in controlling nanorod growth due to the kinetic control of the reaction.

The main objective of the present work is to investigate the effect of strong reducing agent on optical extinction and size of Ag NRs. The UV-vis spectrophotometry shows the optical extinction spectra of the Ag NRs. In the case of silver nanorod, the plasmon resonance splits into two modes: one is longitudinal mode and another one is transverse mode. The resonance frequencies as well as the width of the plasmon absorption band depend on the nanoparticles size. The field emission scanning electron microscopy (FE-SEM) images show the morphologies and sizes of Ag NRs and highly depends on the aspect ratio (i.e. length divided by width) of the nanorods.

2. PREPARATION OF SILVER NANORODS

2.1 Mechanism for the Formation of Silver Nanoparticles

The formation of silver nanorods by seed-mediated growth method involves at least two steps: first, the preparation of small size spherical silver nanoparticles (silver seed solution), and second, (silver growth solution) growth of the prepared spherical particle in surfactant (ethylene glycol) media. In the first step, silver nanoparticles with diameters on the order of 6 nm were formed by chemical reduction of $AgNO_3$ with a strong reducing agent such as $NaBH_4$ in the presence of sodium citrate to stabilize the nanoparticles. Silver nanorods (AgNRs) were synthesized by reduction of silver nitrate with EG in the presence of Ag seeds and PVP. The procedure was briefly described as follows: 30 mL of EG was refluxed in a three-necked round-bottom flask at 140° C (equipped with magnetic stirring bar). Silver nitrate solution measuring 35 mL (0.1M, in EG) and 50 mL of PVP solution (1.0 M, in EG) were simultaneously injected drop wise into the hot solution. The reaction mixture was heated for 1 h at 140°C. And then, reaction mixture was cooled to room temperature, diluted with acetone, and centrifuged several times at 1500 rpm for 20 min. A precipitate remained, and removal of the supernatant. The flow charts and preparation steps of the Ag NRs are shown in Figure 1(a-b).



(a)

Figure 1 (a-b). The flow charts and preparation steps of the Ag NRs

3. CHARACTERIZATION METHODS

The characterization methods consist of UV-vis spectrophotometry, scanning electron microscopy (SEM) and cyclic voltammetry (CV).

3. 1 UV-Vis Spectrophotometry

UV-vis spectrophotometry is a technique to measure absorbance of the sample in the visible region. The UV-vis absorption spectra were measured at room temperature using the GENESYS 10S UV-vis spectrophotometry, Figure 2. The scanning ranges of Ag NRs were recorded from 300 nm to 1000 nm. A glass substrate was used as baseline scanning before the measurement. Baseline measurement was made with distilled water as reference solution.

3.2 Scanning Electron Microscopy (SEM)

Silver nanoparticles solution was transferred onto the silicon substrate by a transfer dip-coating method. Ag NPs solution will run over the wafer relatively, producing thin film of Ag NPs. Scanning Electron Microscopy (Agilent 8500 SEM) in Figure 3 is an instrument, which is used to observe the surface morphology and size of Ag NPs at higher magnification, higher resolution and depth of focus compared to an optical microscope. We described the size of the silver nanoparticles using Image J software from the SEM micrographs. To perform these measurements, we used dip-coating method. Firstly, a thin film was washed in DI water to remove the excess surfactant and dispersed in PIA solution. After that, this thin film was dipping into the Ag NPs solution and dried in air. The morphological features of metal NPs assembled on Si substrate by SEM. The shape of nanostructures could be observed clearly under SEM.

Figure 2. UV-vis Spectrophotometry (GENESYS 10S)

Figure 3. Scanning Electron Microscopy (SEM)

4. RESULTS AND DISCUSSION

In this research, the synthesis and characterization of silver nanorods (Ag NRs) of the present work.

4.1 Effect of Strong Reducing Agent ($NaBH_4$)

In the case of silver nanorods, electron oscillation can occur in one of two directions (along the short and long axes) depending on the polarization of the incident light. The excitation of the surface plasmon oscillation along the short axis induces an absorption band in the visible region at wavelength similar to that of silver nanospheres /nanopraticles, referred to as the transverse surface plasmon resonance (TSPR) band. The excitation of the surface plasmon oscillation along the long axis induces a much stronger absorption band in the longer wavelength region, referred to as the longitudinal surface plasmon resonance (LSPR) band. While the transverse band is insensitive to the size of the

nanorods, the longitudinal band is blue-shifted with decreasing aspect ratio (length/width). The morphology and aspect ratio of the silver nanorods strongly depend on the reducing agent concentration. We studied the effect of strong reducing agent on transverse surface plasmon resonance (TSPR) and longitudinal surface plasmon resonance (LSPR) band properties of Ag NRs.

Figure 4 shows the absorption spectra of Ag NRs with different soft reducing agent concentrations (PVP-1% up to 8%). This figure shows a broad absorption band one at λ_{max} = 420 nm and another at 570 nm which is due to the oscillation of conduction band electrons as the surface plasmon resonance. In this case, the broad absorption band exhibits that the final product should be the presence of both silver nanorods and dispersive particles. Upon increasing PVP concentration, both LSPR and TSPR peak increased with absorbance values 0.185 up to 0.190. However, LSPR and TSPR peak intensity significantly increased which can be attributed to higher surface charge density of Ag NRs induced by PVP, thus expecting the smaller diameter of Ag NRs. The highest maximum absorption peak was observed at 8% of PVP concentrations.

Figure 5 shows the absorption spectra of Ag NRs in ethylene glycol solvent with different strong reducing agent concentrations ($NaBH_4$-1% up to 8%). This figure shows a broad absorption band one at λ_{max} = 420 nm and another at 550 nm which is due to the oscillation of conduction band electrons as the surface plasmon resonance. Upon increasing $NaBH_4$ concentration, both LSPR and TSPR peak increased with absorbance values 0.236 up to 0.243. However, LSPR and TSPR peak intensity significantly increased which can be attributed to higher surface charge density of Ag NRs induced by $NaBH_4$, thus expecting the smaller diameter of Ag NRs. The highest maximum absorption peak was observed at 8% of $NaBH_4$ concentrations.

Figure 6 shows the absorption spectra of Ag NRs in ethylene glycol solvent with different reducing agent concentrations (AA-1% up to 8%). This figure shows a broad absorption band one at λ_{max} = 430 nm and another at 556 nm which is due to the oscillation of conduction band electrons as the surface plasmon resonance. Upon increasing AA concentration, both LSPR and TSPR peak increased with absorbance values 0.63 up to 0.78. However, LSPR and TSPR peak intensity significantly increased which can be attributed to higher surface charge density of Ag NRs induced by AA, thus expecting the smaller diameter of Ag NRs. The highest maximum absorption peak was observed at 8% of AA concentrations.

Figure 7 shows the comparative study of optical absorption spectra of Ag NRs with different reducing agent concentrations 8 % (PVP and $NaBH_4$, AA). According to the above results, the highest optical absorption peak was observed with AA reducing agent. The optimum condition of silver nanorods was observed at concentration 8 % and reducing agent (AA).

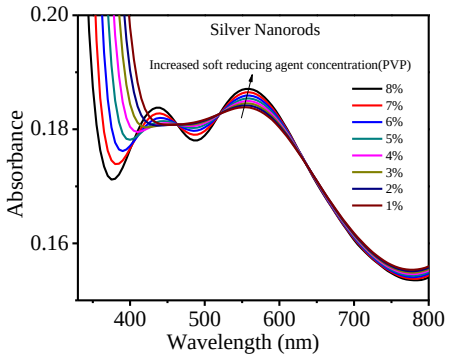


Figure 4. Optical absorption spectra of Ag NRs with different soft reducing agent concentrations AA (1mM-8mM)

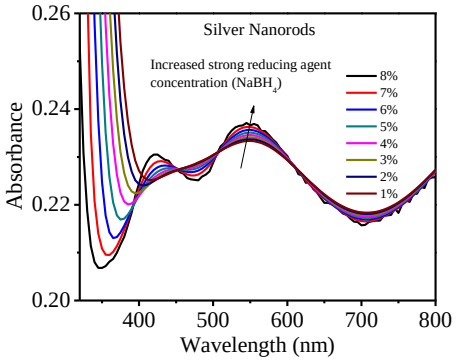


Figure 5. Optical absorption spectra of Ag NRs with different reducing agent concentrations $NaBH_4$ (1mM-8mM)

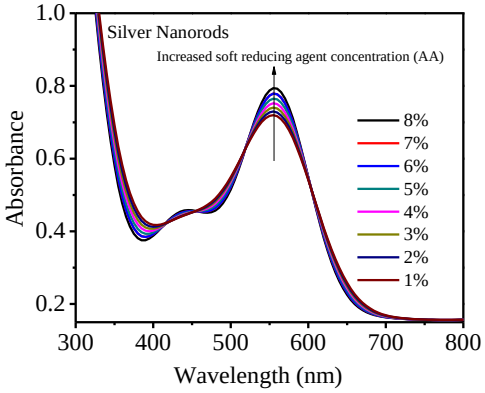


Figure 6. Optical absorption spectra of Ag NRs with different reducing agent concentrations AA (1mM-8mM)

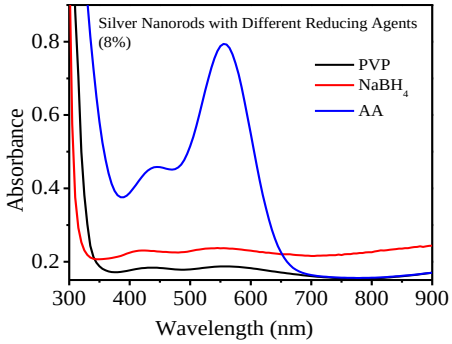


Figure 7 Optical absorption spectra of Ag NRs with different reducing agent concentrations 8% (PVP and $NaBH_4$, AA)

4.2. Size Analysis of Silver Nanorods

In order to determine the size of Ag NRs, field emission scanning electron micro spectroscopy (FESEM) images were acquired. Figure 8 (a-c) shows FESEM images of Ag NRs with reducing agent concentration 8 % (PVP and $NaBH_4$).The obtained sample possesses many amorphous particles and small rods. When the molar ratio was increased 8 %, the sample possessed a number of silver nanorods with a random orientation; the diameter and length of rods were uniform. The average diameter of nanorods was observed at 950 nm with PVP and 800 nm for $NaBH_4$ and 780 nm for AA. According to the FESEM results, the smaller diameter of AgNRs was observed with AA. The results observed from FESEM agrees well with our speculation that increased absorption band (LSPR) of NRs with AA would produce smaller diameter of Ag NRs.

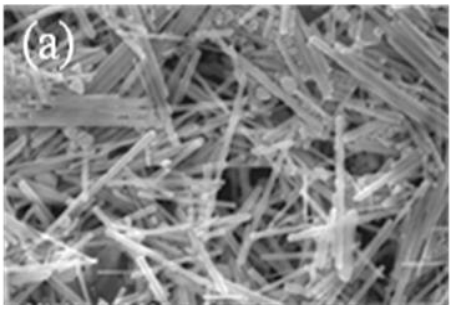


Figure 8. (a) FESEM images of Ag NRs with reducing agent PVP (8%).

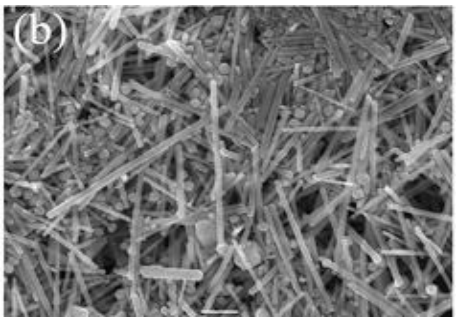


Figure 8. (b) FESEM images of Ag NRs with reducing agent $NaBH_4$ (8%).

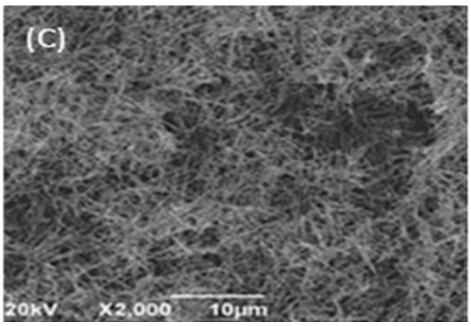


Figure 8. (c) FESEM images of Ag NRs with reducing agent AA (8%).

4.3 Stability Study of AgNPs (Reducing Agent concentration 8 %)

In order to examine the stability of AgNR colloids, we investigated the optical properties of colloidal AgNRs upon aging for 0-21 days. Figure 9 shows the absorption spectra of as-obtained and aged AgNRs colloid (reducing agent PVP concentration 8%). Upon ageing for 3 days up to 15 days, the color and peak position of AgNRs remained unchanged. The optical absorption peak intensity increased. After ageing up to 15 days, the optical absorption peak intensity decreased. This ageing study indicated that AgNP colloids were stable only up to 15 days for reducing agent concentration 8 %.

Figure 10 shows the absorption spectra of as-obtained and aged AgNRs colloid (reducing agent $NaBH_4$ concentration 8%). Upon ageing for 3 days up to 18 days, the color and peak position of AgNRs remained unchanged. The optical absorption peak intensity increased. After ageing up to 18 days, the optical absorption peak intensity decreased. This ageing study indicated that AgNR colloids were stable only up to 18 days for reducing agent concentration 8 %.

Figure 11 shows the absorption spectra of as-obtained and aged AgNRs colloid (reducing agent AA concentration 8%). Upon ageing for 3 days up to 15 days, the color and peak position of AgNRs remained unchanged. The optical absorption peak intensity slightly increased. Increased peak intensity can be attributed to the higher concentration of surface electrons in aged AgNR colloids. After ageing up to 15 days, the optical absorption peak intensity decreased. This ageing study indicated that AgNR colloids were stable only up to 15 days for reducing agent concentration 8 %.

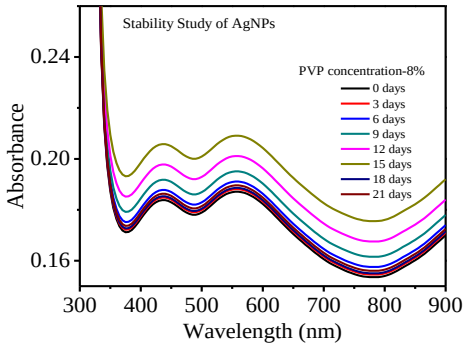


Figure 9. Absorption spectra of as-prepared and aged AgNPs colloid (reducing agent concentration 8 %)

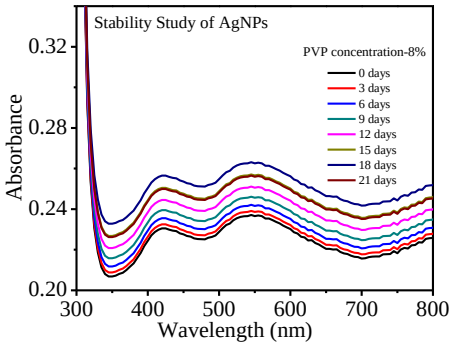


Figure 10. Absorption spectra of as-prepared and aged AgNPs colloid (reducing agent concentration 8 %)

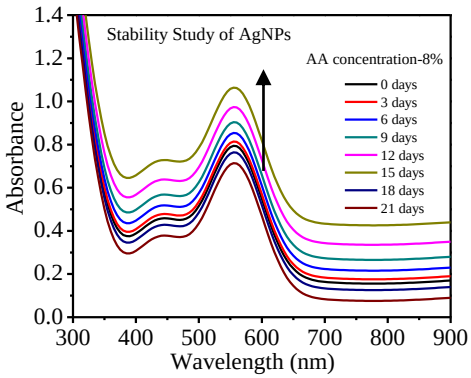


Figure 11. Absorption spectra of as-prepared and aged AgNPs colloid (reducing agent concentration 8 %)

5. Conclusion

For application of nanorods in photovoltaic device, the smaller diameter of nanrods was used. It can be attributed to the good light harvesting properties of silver nanorods. Silver nanorods were developed by seed- mediated growth method where the size and shape of nanostructures can be easily controlled by varying the concentration of reducing agents. In this work, the effect of strong and soft reducing agent on size and optical extinction of silver nanords was mainly investigated. Upon introducing strong reducing agent PVP, $NaBH_4$, AA (1% up to 8%), the optimum absorption peak was observed at molar ratios 8%. From FESEM images, the average diameter of nanorods was observed at 950 nm with PVP and 800 nm for $NaBH_4$ and 780 nm for AA. The stability of Ag NRs was observed at 15 day for PVP, AA and 18 day for reducing agents $NaBH_4$. This work demonstrates that, the reducing agent ($NaBH_4$) plays a key role in tuning the size and shape of nanostructures, thus improving the optical extinction properties.

Acknowledgements

Special thanks go to members of Materials Research Lab, Department of Physics, University of Mandalay, for their help in experimental work.

References

[1] J. Liu *et al.*, Colloids Surf. A, **302** (2007) 276.

[2] H.S. Shin *et al.*, J. Colloid Interf. **274** (2004) 89**.**

[3] J.I. Hussain *et al.*, Adv. Mat. Lett. **2** (2011) 188.

[4] S.L. Hsu *et al*., Int. C. Nan. Biosens, **2** (2011)

[5] S. Kpaoor, Langmuir, **14** (1998) 102.

Authors

Biography

1. First Author:

Daw Thiriwin

Lecturer

Department of Engineering Physics

Technological University (Magway).

2. Second Author:

Daw Myint Myint Kyi

Lecturer

Department of Engineering Physics

Technological University (Magway).

3. Third Author:

Dr Thaung Hlaing Win

Lecturer

Physics Department

Pakokku University.

EFFECT OF NATURAL DYES CONCENTRATION ON SIZE AND SURFACE PLASMON PROPERTIES OF COLLOIDAL SILVER NANOPARTICLES

Htet Htet[1], Yin Yin Htwe[2], and Htun Win[3]

[1,2,3] Department of Physics, Pakokku University
[1]htethtetphys0011.11@gmail.com, [2]yinyinpku777888@gmail.com,
[3]htunnwin@gmil.com

ABSTRACT

There is an increasing demand commercially for the bio synthesis of metallic nanoparticles, due to its catalytic property and wider application in medicine, pharmaceutical, agriculture. Colloidal silver nanoparticles (AgNPs) were synthesized by chemical reduction method. In this case, surface plasmon resonance (SPR) band, size and surface-morphological properties of silver nanoparticles were explored with different reducing agent concentration of Red Silk Cotton petal of flower (1:1, 1:2, 1:3, 1:4 and 1:5). Dry and Fresh Red Silk Cotton (*Bonmbax Ceiba)* petal of flower was used as a reducing agent to compromise the SPR band and size of AgNPs. According to the UV-Vis results, the optimum concentration was observed at ratio (1:2) for dry and (1:4) for fresh Red Silk Cotton petal of flower extracts. In FESEM results, the average sizes of colloidal silver nanoparticles are around ~ (50 -70) nm for different reducing agents concentration (1:1 – 1:2) for dry and (1:1 – 1:4) for fresh Red Silk Cotton petal of flower. The excess amount of reducing agent could not favored the smaller nanoparticles size. And then, Atomic Force Microscopy (AFM) study indicated the AgNPs coated- TiO_2 films exhibited single phase structure. The surface roughness of thin films was increased from ~ 50 nm up to ~ 70 nm due to the higher reducing agents concentration, the films surface is porous. In XRD analysis results, the reflection peaks are indexed as the fcc (111), (200), (220), and (222) planes, indicating that the silver is well crystallized.

KEYWORDS

Reducing agent, green synthesis, natural dyes, silver nanoparticles.

1. INTRODUCTION

Important scientific and technological advances based on understanding and control of properties and process at the scale of atoms and molecules (nanometer scale) are taking place in laboratories around the world. Nanostructures have been extremely attractive because of their unique properties (e.g., optical, electronic, magnetic and chemical properties) and potential applications in nano-science and nanotechnology. Nanoparticles are usually the particles of size 1 nm to 100 nm. Physicochemical properties of nanoparticles such as mechanical, chemical, magnetic, optical or electrical properties are different compared to bulk materials for the volume to surface ratio. Adjusting the stabilizer concentration and selecting the proper stabilizer agents, would result in the stability of colloidal nanoparticles

Recently, metallic nanoparticles (NPs) were introduced into organic photovoltaic (OPV) devices for highly improved light harvesting by utilizing the localized surface plasmonic resonances (LSPR) of metallic NPs. The influence of NPs on charge separation and transport inside high efficiency bulk hetero junction (BHJ) solar cells still needs to be explored in depth. The incorporation of silver nanoparticles exhibited the following advantages: broader light absorption enhancement than single NPs, enhanced exciton generation rate and dissociation efficiency, and increased charge carrier density and lifetime [2].

Commercially available dyes are sources of environmentally hazardous organic compounds that affect both the aquatic life and human health. Biological degradation of these dyes seems to be in effective due to its varied composition of organic compounds. This study was carried on to investigate the role of aqueous extract of Red Silk Cotton petal of flower in the green synthesis of colloidal silver nanoparticles. The synthesized nanoparticles were optimized for various parameters like temperature and concentration. The optimization was followed by characterization; the silver nanoparticles were taken for UV -Vis Spectroscopy analysis, Field Emission Scanning Electron Microscopy (FESEM) study, Atomic Force Microscopy (AFM) and X-Ray Diffraction (XRD) to characterize the synthesized particles. Thus the objective of current study, aims to understand the synergistic interaction of reducing agents concentration and SPR band with the Silver nitrate solution for the formation of silver nanoparticles.

2. EXPERIMENTAL WORK

In this section deals with the extraction of natural dyes, the green synthesis and characterization of colloidal silver nanoparticles (Ag NPs) with different Red Silk Cotton petal of flower of the present work.

2.1. Extraction of Natural Dyes from Red Silk Cotton Petal of Flower

Fresh Red Silk Cotton *(Bombax ceiba)* petals were collected from the Ngwe Taw Tarwomen's hall, Pakokku University. Fresh petals of flower washed thoroughly with tap water to remove the dust and dirt particles and drying in the sun. Dry Red Silk Cotton petals were crushed with the help of mortar and pestle and then this powder was preserved for experimental work.1% petals of flowers powder was mixed with 100 ml ethanol in 250 ml Borosil beaker and stirred for about 2 minutes. Similarly, the other concentrations (1:2, 1:3, 1:4 and 1:5) of petals powder were also mixed with each 100 ml ethanol. The extract solution was filtered through filter paper (pore size > 0.5µm), collected and kept at room temperature. And then Silver nanoparticles are synthesized from silver nitrate ($AgNO_3$) by using aqueous bio-extract of petals of dry and fresh Red Silk Cotton petal of flowers as reducing agent. The flowcharts of natural dyes extracted from dry and fresh Red Silk Cotton petal of flowers as shown in Figure 1.

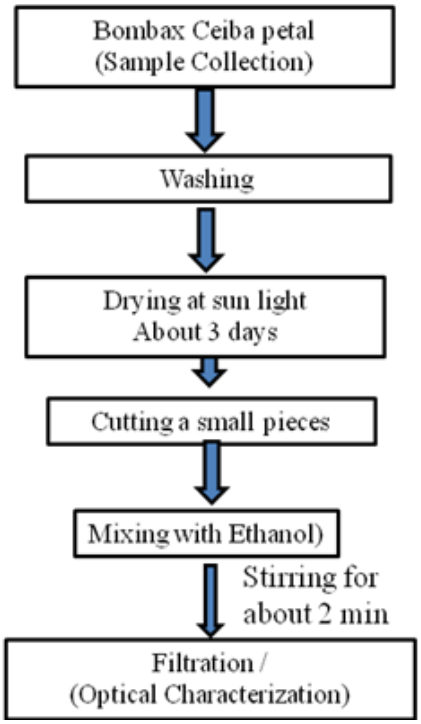


Figure 1. Flowcharts of natural dyes extracted from Red Silk Cotton petal of flower

2.2. Green Synthesis of Colloidal Silver Nanoparticles (AgNPs)

In the case of nano-metal particles, colloidal silver nanoparticle (AgNPs) solutions have been developed by a green synthesis method. For the preparation of silver nanoparticles (AgNPs) colloids, silver nitrate ($AgNO_3$) and natural dyes solution were used, as a metal salt precursor and reducing agents, respectively. (1 g) of dry petal of flowers was added to 50 ml of ethanol solvents while 0.3 g of $AgNO_3$ was added to 500 ml Brosil beaker. And then these two solutions are mixing and stirring for about 2 min. The color of the solution changed from yellow to brown reddish color, indicating the formation of colloidal silver nanoparticles. In another study, natural dyes concentration was changed (1:1 up to 1:5) while silver salt concentration (1 mM) was kept constant during this experiments. The flowcharts and experimental setup for the synthesis of colloidal AgNPs of the present work are shown in Figure 2 (a-b).

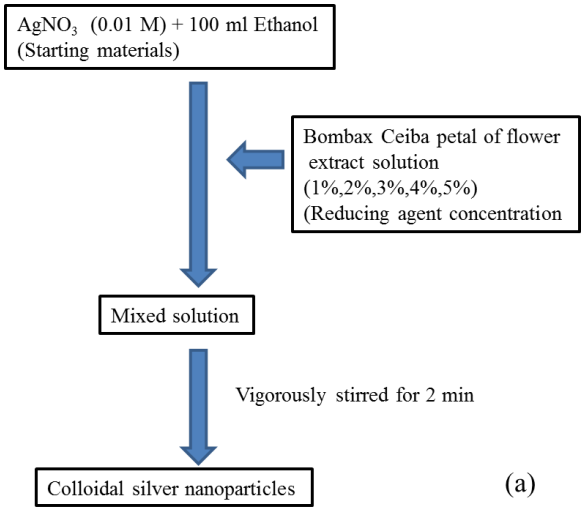


Figure 2. (a) Flowcharts of the preparation steps for colloidal AgNPs

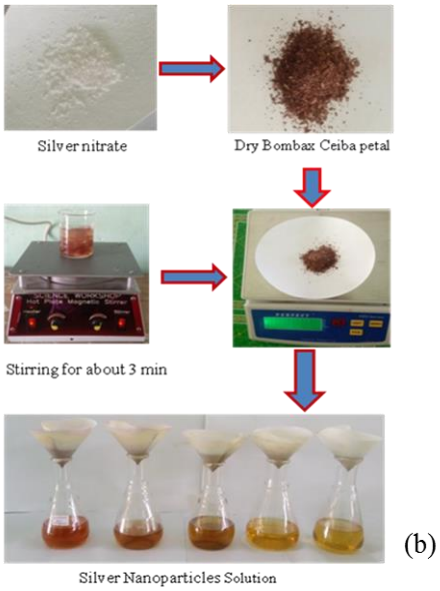


Figure 2. (b) Preparation steps of Colloidal AgNPs with different Dry and Fresh Red Silk Cotton petal of flower

2.3. Characterization of Colloidal Silver Nanoparticles

The characterization methods used are UV-vis spectrophotometry, field emission scanning electron microscopy (FESEM), Atomic Force Microscopy (AFM) and X- Ray Diffraction (XRD) respectively. UV-Vis Spectrophotometry is a technique to measure either absorbance or transmittance of the sample in ultraviolet and visible region. The surface Plasmon properties of colloidal nanoparticles were measured by UV-Vis spectrophotometer: GENESYS 10S UV-Vis (Materials Research Lab, Department of Physics, and University of Mandalay) as shown in Figure 3.

Silver nanoparticles solution was transferred onto the silicon substrates by a transfer dip-coating method. The surface morphological features of metal nanoparticles assembled on Si substrate by field-emission scanning electron microscopy: Agilent 8500 FE-SEM (Defence Services Science and Technology Research Centre (DSSTRC), Pyin Oo Lwin) as shown in Figure 4. The 8500 FE-SEM software runs on this computer and provides point-and-click operation, configuration and status monitoring of the 8500 FE-SEM. The images were obtained at an operating voltage of 1000 V and a current of 2.2 A. The sizes of silver nanoparticles were estimated from FESEM image by using “image j” analyzing software.

X-Ray Diffraction (XRD) is a non-destructive technique to measure the crystal structure of powder or film and the crystallite size (grain size) of the sample. The structural properties of AgNPs was investigated by X- ray diffractometer as shown in Figure 5 (RIGAKU Multiflex) using CuKα radiation (λ = 1.5418 Å). The 2θ scan range was from 30˚-70˚ with a step of 0.02˚.

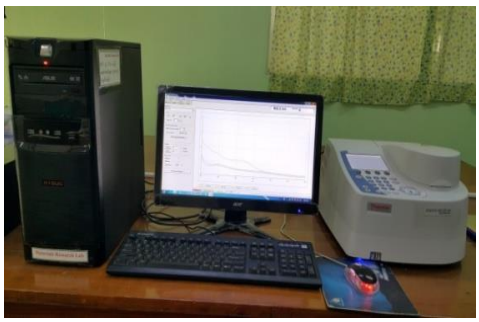


Figure 3. UV-Vis Spectrophotometry (GENESYS 10S)

Figure 4. Field Emission Scanning Electron Microscopy (Agilent 8500FE-SEM)

Figure 5. X-ray Diffraction (XRD)

3. RESULTS AND DISCUSSION

The optical absorption properties, size and structural of silver nanoparticles (AgNPs) were explored varying with the volume ratio of reducing agents (natural dyes) concentrations.

3.1. Optical Absorption Properties: Varying Reducing Agents Concentrations

The surface plasmon properties of colloidal AgNPs with Red Silk Cotton (*Banbax cebia*) petals of flowers were explored varying concentrations (1:1 up to 1:5). Figure 6 shows the surface plasmon band of colloidal AgNPs with different dry Red Silk Cotton petals of flowers concentration. The surface plasmon resonance (SPR) peak positions of AgNPs colloids are observed at 400 nm for natural dyes concentrations 1:1 up to 1:5. Upon increasing natural dyes concentrations 1:1 up to 1:2, the SPR maximum peak of colloidal AgNPs increased with absorbance values 0.63 up to 0.92. And then, upon increasing 2% up to 1:5, the SPR maximum peak deceased with absorbance values 0.45. From the above observation (increased peak intensity), it is attributed to the smaller nanoparticles size. This speculation is supported for the size estimation of nanoparticles. At the larger reducing agents concentrations, the surface plasmon peak of colloidal AgNPs is not strong enough for the light harvesting process.

The surface plasmon properties of colloidal AgNPs with fresh Red Silk Cotton petal were explored varying with natural dyes concentration (1:1 up to 1:5). Figure 7 shows the surface plasmon band of colloidal AgNPs with different reducing agent concentrations. The surface plasmon resonance (SPR) peak positions of AgNPs colloids are observed at 420 nm for all volume ratios. Upon increasing natural dyes concentrations 1:1 up to 1:5, the maximum SPR peak was observed at 1:4 and the maximum surface plasmon peak position remained unchanged. The optimum condition of AgNPs was observed at ratio 1:2 for dry and 1:4 for fresh Red Silk Cotton petal of flower extracts.

Figure 8 (a-b) shows the optical absorption spectra of colloidal AgNPs with different concentration of dry Red Silk Cotton petal of flower (1%, 2%) and fresh Red Silk Cotton petal of flower (1%, 4%).

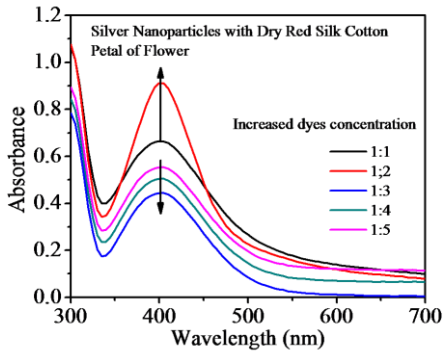


Figure 6. Optical absorption spectra of colloidal AgNPs with different concentration of dry Red Silk Cotton petal of flower

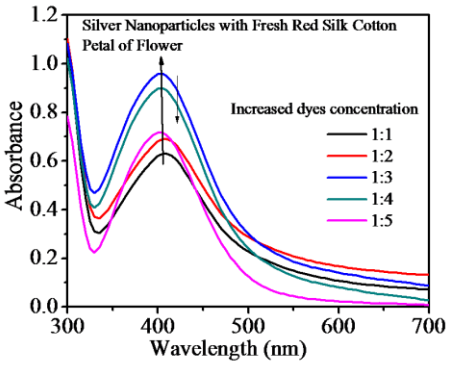


Figure 7. Optical absorption spectra of colloidal AgNPs with different concentration of fresh Red Silk Cotton petal of flower

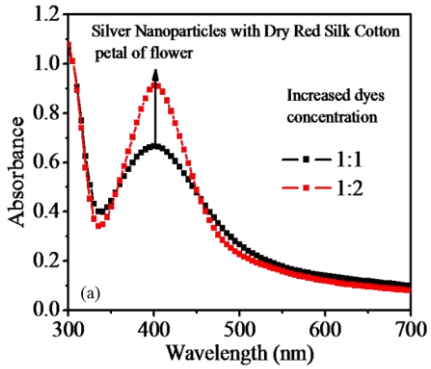


Figure 8. (a) Optical absorption spectra of colloidal AgNPs with different concentration of dry Red Silk Cotton petal of flower (1%, 2%)

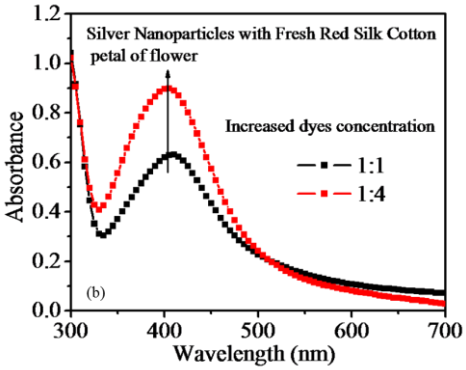


Figure 8. (b) Optical absorption spectra of colloidal AgNPs with different concentration of Fresh Red Silk Cotton petal of flower (1%, 4%)

3.2 Size Estimation: Varying Natural Dyes Concentrations

The surface features of AgNPs were measured by field emission scanning electron microscopy (FESEM) varying with natural dyes concentrations. Figure 9 (a-b) shows the FESEM micrograph of colloidal AgNPs (dry Red Silk Cotton petals of flower) with concentrations 1:1 and 1:2. The average sizes of colloidal AgNPs are ~ 63 nm for natural dyes concentrations 1:1 and ~ 51 nm for 1:2. The average size of AgNPs decreased with increasing reducing agents concentrations (natural dyes). Like the optical absorption result, the smallest average size of colloidal AgNPs was observed at the natural dyes concentrations 1:2.

Figure 10. (a-b) shows the FESEM micrograph of colloidal AgNPs (fresh Red Silk Cotton petals of flower) with concentrations of 1:1 and 1:4. The average sizes of colloidal AgNPs are ~ 58 nm for concentrations ratio 1:1 and ~ 49 nm for ratio 1:4. The average size of AgNPs decreased with increasing reducing agents concentrations (natural dyes). Like the optical absorption result, the smallest average size of colloidal AgNPs was observed at natural dyes concentrations 1:4.

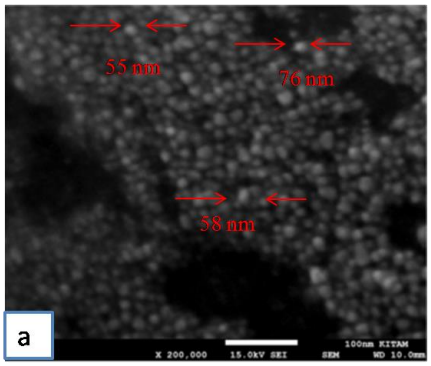


Figure 9. (a) FESEM images of colloidal AgNPs with different concentration of dry Red Silk Cotton petal of flower (1:1)

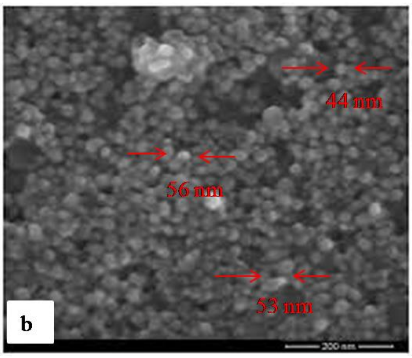


Figure 9. (b) FESEM images of colloidal AgNPs with different concentration of dry Red Silk Cotton petal of flower (1:2)

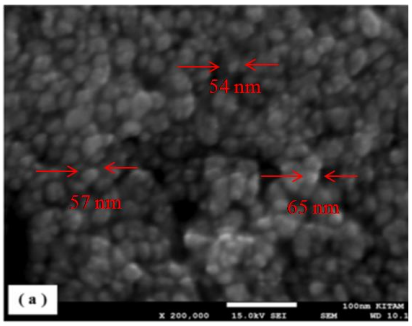


Figure 10. (a) FESEM images of colloidal AgNPs with different concentration of fresh Red Silk Cotton petal of flower (1:1)

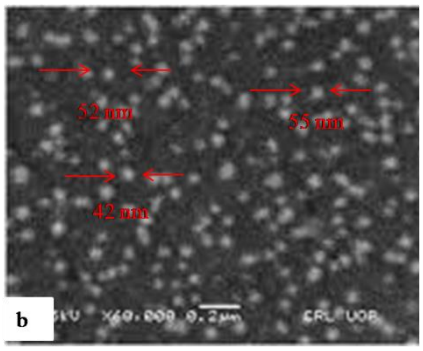


Figure 10. (b) FESEM images of colloidal AgNPs with different concentration of fresh Red Silk Cotton petal of flower (1:4)

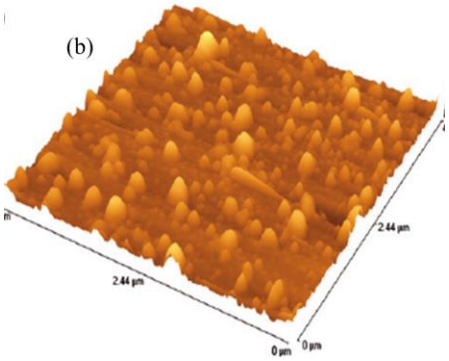
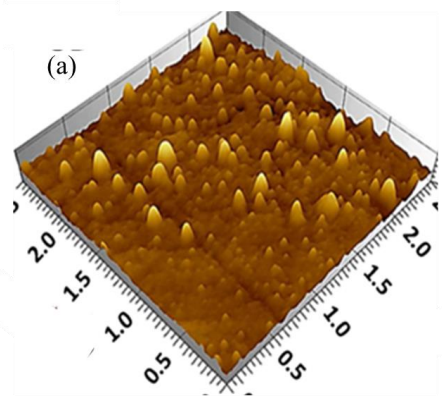


Figure 11. (a-b) AFM image of Silver Nanoparticles films with different dry Red Silk Cotton petal of flower 1:1 and 1:2.

3.3. Surface Roughness Properties of Silver Nanoparticles

The topographies of various surfaces have been studied in many fields due to the significant influence that surfaces have on the practical performance of a given sample. In viewing, the opportunity to enhance photovoltaic application, two samples having different reducing agents concentration. Figure 11(a-b) shows the AFM image of colloidal AgNPs with dry Red Silk Cotton petal of flower 1:1 and 1:2. AFM study indicated the entire AgNPs coated TiO_2 films exhibited single phase structure for all films. The surface roughness of thin films was increased from 50 nm up to 70 nm due to the higher reducing agents concentration, the films surface is porous. Figure 12 (a-b) shows the AFM image of colloidal AgNPs with fresh Red Silk Cotton petal of flower 1:1 and 1:4. The surface roughness of AgNPs thin films was increased from 60 nm (1:1) up to 75 nm (1:4) due to the higher reducing agents concentration, the AgNPs films surface is also porous structure.



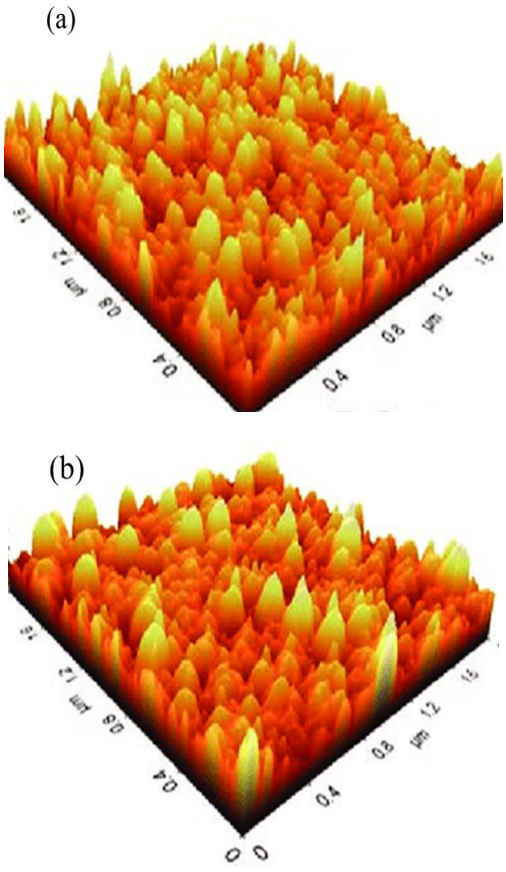


Figure 12. (a-b) AFM image of Silver Nanoparticles films with different fresh Red Silk Cotton petal of flower 1:1 and 1:4

3.4. Structural Characterization X-Ray Diffraction

The X-ray diffraction patterns of the silver films with Red Silk Cotton petal of flower concentration (1:1) for dry and fresh, presented in Figure 13. Figure shows the peak characteristics of metallic silver nanoparticles with dry and fresh Red Silk Cotton petal of

flower extract. The reflection peaks are indexed as the fcc (111), (200), (220), and (222) planes, indicating that the silver is well crystallized. Four different intense peaks cantered at 2**θ** values of 43.42, 48.4, 67.34, 77.23 indicating cubic phase of silver. There are no other peaks that no impurities are detected to indicate the formation of highly pure silver nanoparticles, which also confirm the raw material without adding any metal halide compound.

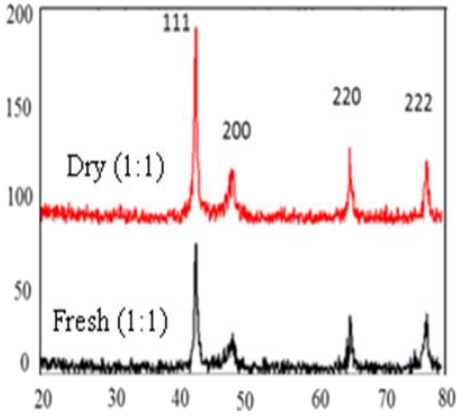


Figure 13. XRD patterns of Ag NPs with dry and fresh Red Silk Cotton petals of flower (1:1)

4. CONCLUSION

This plant based green synthesis of nonmaterial is the progress of eco-friendly method, for the fabrication of metal-based nanoparticles. Silver nanoparticles (AgNPs) colloid was prepared by green synthesis method. Optical absorption properties of AgNPs colloid was explored varying with the reducing agents concentration (1:1 up to 1:5). Upon increasing reducing agent (natural dyes) concentrations, the optimum ratio SPR peak intensity of colloidal AgNPs was observed at (1:2) for dry and (1:4) for fresh Red Silk Cotton petal of flower extracts. Increased SPR peak intensity may be due to the smaller nanoparticles formation. The average size of colloidal silver nanoparticles was obtained from the field emission scanning electron microscopy (FESEM) micrograph. The obtained average size of silver nanoparticles with dry Red Silk Cotton petal of flower 1:1 and 1:2 are ~ (63 nm - 51 nm) while ~ 58 nm and 49 nm for fresh extract concentrations 1:1 and1:4. This observation is agreed with above speculation UV-Vis results. Thus, selection of reducing agent concentrations is important to compromise the surface plasmon resonance band and the size of nanoparticles. The crystal structure of colloidal AgNPs was examined from the variation of surface plasmon resonance peak intensity. In AFM images, the surface roughness of AgNPs thin films was increased from ~ 50 nm up to ~ 70 nm both dry and fresh Red Silk Cotton petal of flower due to the higher reducing agents concentration, the films surface is porous. The reflection peaks are indexed as the fcc (111), (200), (220) and (222) planes, indicating that the silver is well crystallized. Four different intense peaks cantered at 2**θ** values of 43.42, 48.4, 67.34, 77.23 indicating cubic phase of silver. This proves that it could be a cost effective way to fabricate the organic - inorganic hybrid solar cells. In future, these biogenic fabricated nanoparticles could be employed in the development of light harvesting, drug and catalystics.

ACKNOWLEDGMENTS

We would like to thank Dr Tin Tun Aung, Acting Rector, Pakokku University. We are also thankful to Dr Win Win Ei (Pro-rector) and Dr Khin Khin Htoot, (Pro-rector), Pakokku University. We would like to express my gratitude to Dr Htun Win (Professor and Head), Dr Win Mar (Professor) and Dr Win Myint Kyu (Professor), Department of Physics, Pakokku University, for their kind permission to give us the opportunity to do research.

REFERENCES

[1] X.W. Guoyun *et al*., 2014. Journal of Nano science.

[2] T. Markvart *et al*., 2003. Journal of Apply Physics.

[3] H. Hahn *et al*., 1997.Nanostruct Mater.

[4] D.R. Duran N *et al*., 2012. Synthesis of Silver Nanoparticles.

[5] F.M. Al-Khateeb *et al*., 2018. Apply Physics and Material Science.

Authors Biography

First Author –Daw Htet Htet
Demonstrator
Department of Physics
Pakokku University

Second Author–Daw Yin Yin Htwe
Demonstrator
Department of Physics
Pakokku University

Third Author –Dr Htun Win
Professor
Department of Physics
Pakokku University

IOT-BASED REAL-TIME VEHICLE TRACKING AND BATTERY MONITORING SYSTEM FOR ELECTRIC VEHICLES

Nann Sandar Aye[1], Pa Pa Winn San[2], and Aye Min Soe[3]

[1,2,3]Department of Electronics, Technological University (Mandalay)
[1]nannsandaraye@gmail.com, [2]papawinsan.1@gmail.com,[3]ayeminsoeec@gmail.com

ABSTRACT

A new IoT-based application is in development to monitor electric vehicles (EVs) in real-time, tracking both their locations and battery monitoring system and displaying this information on Google Maps. The system is designed to deliver accurate and current data on vehicle positions and battery status, thereby enhancing monitoring and management capabilities. An ESP32 microcontroller processes and relays data from integrated sensors, including GPS for location tracking and a Battery Monitoring System (BMS) for monitoring voltage, current, and temperature. Using GSM internet and wireless networks, the system transmits data to the ThingSpeak API for storage and analysis. The Leaflet Map library, combined with Google Maps API, provides enhanced visualization of vehicle locations and routes. A web/mobile interface enables real-time monitoring, route tracking, and geofencing. This innovative system aims to make vehicle management more efficient, improve security, and maintain optimal battery performance, helping to advance sustainable transportation. The paper covers the benefits of using IoT technology in managing electric vehicles.

KEYWORDS

IoT, Location Tracking, ThingSpeak API, Battery Monitoring System, GPS

1. INTRODUCTION

Electric car manufactures are incorporating advanced technologies including Artificial Intelligence and IoT to enhance efficiency. Vehicles play a crucial role in transporting products or goods for many organizations, yet they face various operational challenges. Significant efforts are being made to replace combustion engines with electric motors, and integrating IoT can significantly streamline EV performance and monitor impacts. IoT not only improves city planning but also simplifies urban living. The car industry and transportation sector are experiencing rapid transformation, driven by advancements in IoT, increased computing power, and widespread connectivity, which also benefit the environment.

As the quantity of EVs grows, developing a robust EV charging infrastructure becomes essential. Fully charged EV batteries can deliver power back to the grid, allowing users to earn money. The State of Charge (SoC), measured using an ESP32 controller, transmits to the cloud. Users can monitor their battery status. Batteries have become to be the preferred form of electrical energy storage in EVs, and their importance in city transportation has surged, fostering societal and industrial growth. Given the widespread use of batteries for energy storage, accurately calculating SoC is crucial for future developments [1].

A system for tracking vehicles in actual time employs GPS technology to obtain vehicle locations, which are processed by a microcontroller and transmitted via GPRS technology. This data is displayed in real-time on a website map powered by Google Maps, enabling continuous vehicle monitoring. This seamless integration of IoT for vehicle tracking and battery monitoring illustrates the potential for enhanced operational efficiency and sustainability in the transportation sector.

1.1. Benefits of Electric Vehicles

Electric vehicles (EVs) present several significant advantages compared to conventional petrol or diesel cars. [6]

1.1.1. Less Expensive to Operate

One major benefit is the reduced operational costs, allowing EV owners to enjoy significantly lower running expenses. Figure 1 shows the benefits of electric vehicles.

Figure 1. Benefits of Electric Vehicles

1.1.2. More Environmentally Friendly

Many people opt for electric vehicles primarily due to their superior environmental friendliness compared to traditional cars. EVs devoid of an exhaust system, so they don't produce emissions like petrol or diesel vehicles. Gas-powered cars are significant contributors to greenhouse gas accumulation in the earth's atmosphere. Therefore, using electric vehicles helps reduce air pollution and promotes a healthier planet. [5]

1.1.3. Electricity is More Affordable than Gas

When comparing costs, the average Myanmar spends about 2800 kyats per mile operating a gasoline-powered car. This may seem minimal, but it contrasts sharply with the cost involved in driving an electric car, which is typically around 100 kyats per mile due to the lower cost of electricity compared to gasoline. Additionally, many EV owners charge their cars at home, and installing solar panels can further reduce costs, providing savings on both vehicle charging and home electricity.

1.1.4. Lower Maintenance Costs

Gasoline-powered cars require extensive maintenance, starting with regular oil changes and extending to numerous other tasks related to the combustion engine. Electric cars, on the other hand, don't need oil changes or any maintenance associated with a gas engine. Components related to combustion engines are absent in electric cars, reducing the need for many common repairs. Additionally, brakes used in electric vehicles typically last longer, further contributing to cost savings. When considering these factors, the financial advantages of choosing an electric car over a gasoline-powered vehicle become clear.

1.1.5. Electric Cars Are Generally Quieter

Residents in busy areas are well aware of how noisy traffic can be, especially during rush hour. Even vehicles with smaller engines can contribute significantly to the overall noise. In contrast, electric vehicles are almost silent. Their quiet nature has led some U.S. legislators to propose the installation of noise-making devices in EVs to alert pedestrians to their presence, ensuring safety while maintaining a peaceful environment. [6]

2. METHODOLOGY

2.1. Location Control System

A location control system determines the position of a vehicle using technologies like GPS and other radio navigation systems that operate via satellites and ground-based stations. Utilizing triangulation or trilateration methods, this system calculates the vehicle's precise location easily and accurately. [4] Details about the vehicle, including location, speed, and distance travelled can be displayed on digital maps through software accessed via the Internet. Additionally, data from the GPS unit can be stored and downloaded to a computer at a base station for later analysis. This system is essential for tracking vehicles in actual time and is gaining popularity among owners of expensive cars for theft prevention and recovery.

The location control system comprises current technological hardware and software components that enable vehicle tracking both online and offline modes. The system includes of three fundamental sections: the mobile vehicle unit, the fixed base station, and the firmware and software systems.

2.2. Location Tracking System

The Global Positioning System (GPS) has significantly influenced various applications involving positioning, navigation, timing, and monitoring. The United States, Department of Defence developed the Navstar GPS, a space-based navigation system that functions in all weather conditions, designed to meet the needs of U.S. military forces. This system accurately determines their position, velocity, and time within a common reference system, anywhere on or near the Earth continuously.

GPS transmits uniquely coded signals from satellites, which are then processed by a GPS receiver. This processing allows the receiver to precisely determine its position, velocity, and time within a unified reference framework, whether on or near the Earth. The system uses four satellite signals to compute three-dimensional positions and to correct the receiver's clock time offset.

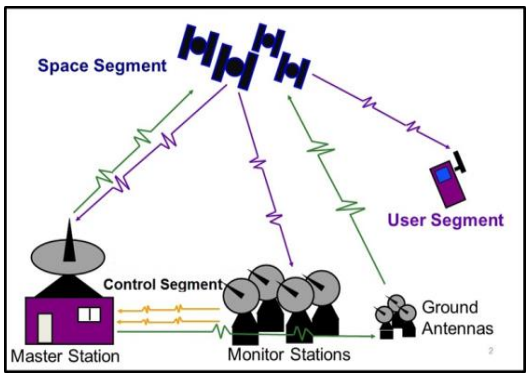


Figure 2. Three Segments of GPS

Figure 2 illustrates the three primary components of GPS, which are detailed below:

2.2.1. Space Segment

Figure 3 shows the GPS constellation of satellites is included in the Space Segment. The space vehicles (SVs) emit radio signals from orbit, as illustrated in Figure 4.

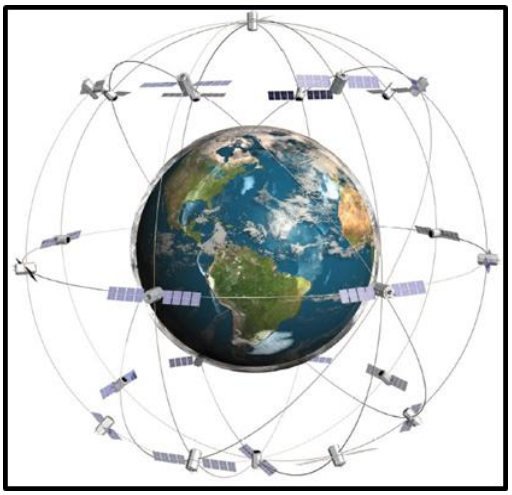


Figure 3. GPS Constellation

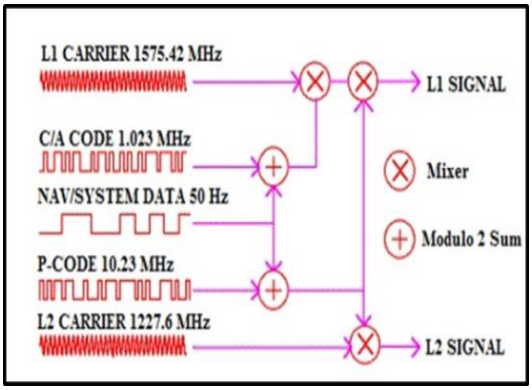


Figure 4. GPS Satellite Signals

2.2.2. Control Segment

The Control Segment comprises the global network of tracking stations. The primary Master Control facility is located at Schriever Air Force Base in Colorado, USA.[4]

2.2.3. User Segment

The User Segment comprises GPS receivers and user community GPS receivers process the signals from space vehicles (SVs) to provide estimates of position, velocity, and time.

The satellites are arranged across six orbital paths, following nearly circular trajectories

at an altitude of about 20,200 Kilometers above the Earth's surface. These orbits are tilted at an angle of 55 degrees relative to the equator and have orbital periods of approximately 11 hours and 58 minutes, which is equivalent to half a sidereal day.

2.2.4. Triangulation or Trilateration

The principle of trilateration underpins the impressive accuracy of GPS navigation. A GPS receiver determines the travel time of a satellite signal to Earth, a process that takes less than a tenth of a second. This time measurement is multiplied by speed of radio waves, which is about 300,000 km (186,000 miles) per second, to calculate the distance to the satellite.

This distance places the receiver at the surface of an imaginary sphere with its canter on the satellite. By repeating this process with utilizing signals from three additional satellites, the receiver's computer can identify the exact point where these four spheres intersect, thus determining the receiver's exact longitude, latitude, and altitude. (see Figure 5)

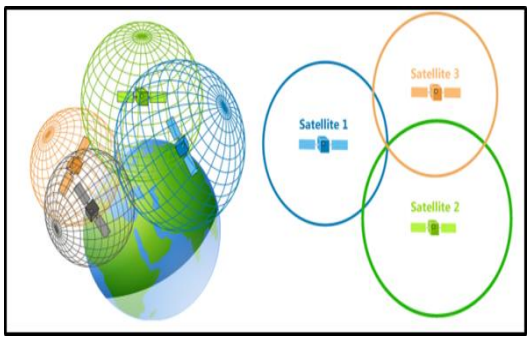


Figure 5. Triangulation or Trilateration

Although theoretically, three satellite signals should be sufficient for calculating a precise three-dimensional location, in practice, four satellites are used. This additional satellite helps to correct any timing errors in the receiver's clock, ensuring even greater accuracy.

In 2D trilateration, we use three known points.

Let:

(x_1, y_1) represents the coordinates of the first point.
(x_2, y_2) represents the coordinates of the second point,
(x_3, y_3) represents the coordinates of the third point,
d_1, d_2 and d_3 be the distances from the receiver to each of these points,
(x, y) represents the coordinates of the receiver.
The distances are given by:

$$d_1 = \sqrt{(x - x_1)^2 + (y - y_1)^2}$$

$$d_2 = \sqrt{(x - x_2)^2 + (y - y_2)^2}$$

$$d_1 = \sqrt{(x - x_3)^2 + (y - y_3)^2}$$

These equations can be solved simultaneously to determine the coordinates (x, y) of the receiver. Solving these equations typically involves using numerical methods or optimization techniques due to their non-linear nature.[4]

2.3. Battery Monitoring System

Battery Monitoring System (BMS) are integral to WBMS, Sensors attached to each cell measure voltage, current, and temperature, providing the necessary data to calculate SOC. within the system. This information is processed by a microcontroller and transmitted to a cloud platform for real-time monitoring and analysis. [2] They measure critical parameters such as:

(a) Voltage: Indicates the SOC and overall health.

(b) Current: Helps in understanding charging and discharging cycles.

(c) Temperature: Prevents overheating, a frequent cause of battery failure.

(d) State of Charge (SOC): Indicates the remaining battery capacity.

(e) State of Health (SOH): Assesses long-term usability and capacity degradation [5].

3. PROPOSED SYSTEM

Overview the proposed vehicle tracking and battery monitoring system is shown in Figure 6. The proposed control system consists of mainly five parts:

(a) GPS satellites,

(b) GSM communication,

(c) Base station Computer

(d) Battery Monitoring System and

(e) Vehicles fitted with the GPS receiver and GSM Module.

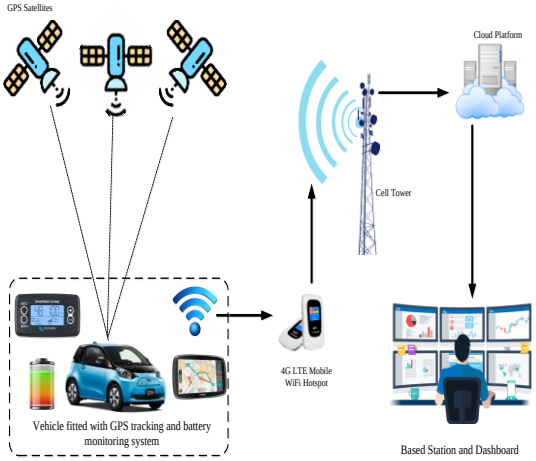


Figure 6. Proposed System Design

In the proposed system, the base station computer can simultaneously track the location of vehicle while monitoring their battery performance and health.

The system design of vehicle tracking and battery monitoring includes the following steps:

(a) The process begins with the satellite transmitting GPS signals, which are received by the GPS receiver in the vehicles.

(b) Upon receiving the signals, the GPS receiver calculates its position using triangulation.

(c) The accurate position determined by the GPS receiver is then sent to the base station computer via a wireless and GSM network.

(d) Additionally, the system simultaneously transmits battery data, including temperature, current, voltage, and humidity.

(e) The base station computer displays the vehicles' positions on a map and provides battery performance and health data.

In the proposed system, Real-time monitoring in electric vehicles (EVs) is essential for optimizing battery health and performance. By continuously tracking parameters such as temperature, voltage, and current, potential issues can be detected early, reducing the risk of catastrophic failures. For example, monitoring battery temperature can decrease the likelihood of thermal runaway incidents by up to 30%. This proactive approach not only enhances safety but also extends battery lifespan, ensuring that the vehicle remains efficient over time.

In addition to improving battery management, real-time monitoring plays a critical role in range optimization and energy management. By analysing driving patterns and environmental conditions, these systems provide dynamic range estimates, allowing drivers to make informed decisions about their trips. Research indicates that accurate range predictions can enhance driving efficiency by 15-20%. For instance, Tesla's energy management system adapts in real-time to optimize battery usage, contributing to an average range improvement of 5-10%. This capability helps prevent drivers from experiencing range anxiety, ultimately fostering greater confidence in EV technology.

Moreover, real-time monitoring significantly benefits fleet management and environ-mental sustainability. For commercial fleets, continuous tracking allows for efficient route optimization and reduces operational costs by up to 20%. By providing fleet operators with valuable data on vehicle status and performance, companies can maximize vehicle utilization and minimize downtime. Additionally,

informing drivers about their energy consumption can lead to eco-friendly driving practices, with studies showing that such feedback can reduce energy use by up to 15%. This not only contributes to lower operational costs but also helps decrease overall emissions, aligning with the growing emphasis on sustainability in transportation.

3.1. System Overview

The system's detailed design is outlined in the following parts is shown in Figure 7. The GPS module, voltage level sensor, current flow and temperature and humidity data are reading from the battery and the location of the vehicle are sent to ESP-32 microcontroller for processing. The processed data are wirelessly transmitted to a battery monitoring interface within the device, as depicted on the dashboard using a web-based GUI. Once the data transfer is finalized, the computer's battery monitoring interface will showcase the latest battery status information.

The established framework for battery monitoring also incorporates a web-based interface. This graphical user interface (GUI) tracks the locations of different battery monitoring devices and consequently monitors their conditions and location tracking of the vehicle. The interface is designed to handle scenarios where multiple battery conditions need simultaneous monitoring.

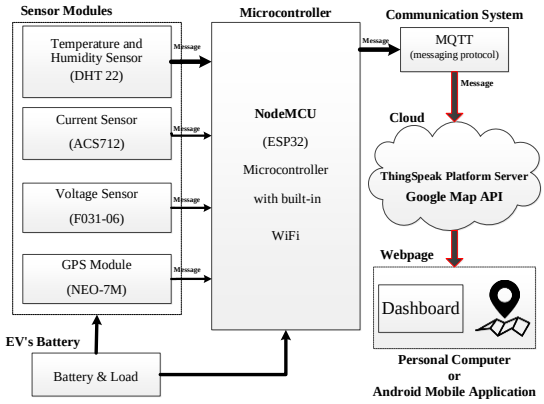


Figure 7. Block Diagram of Proposed System Design

Input sensors include Voltage sensor, Current sensor, DHT 22 (Temperature and Humidity Sensor), Neo7M GPS module and ESP32 Wireless Module.

(a) A voltage sensor used to monitor the voltage of a battery, providing real-time information about its state of charge. it can be employed to monitor the output voltage of power supplies in various electronic systems. By continuously measuring the voltage, the module can detect any fluctuations or deviations from the desired value.it can measure 0 to 25V DC Voltages.

(b) A current sensor board is based on the Allegro ACS712ELCTR-30A bi-directional hall-effect current sensor chip that detects positive and negative flowing currents in the range of minus 30 Amps to positive 30 Amps. It provides real-time data on current levels, which is essential for monitoring and controlling electrical systems.

(c) **DHT22 temperature and humidity sensor** uses a thermistor to measure temperature and capacitive humidity sensor to measure humidity. It has a single wire digital interface. DHT22 has higher precision and may replace the expensive imported SHT10 temperature and humidity sensor. It can measure the environment temperature and humidity to satisfy the high demand. The product has high reliability and good stability. The DHT22 temperature and humidity sensor can play a key role in Electric Vehicle (EV) systems, particularly in monitoring the environmental conditions the battery and other critical components. EV batteries are sensitive to temperature changes. High temperatures can accelerate battery degradation, while low temperatures can reduce efficiency. By integrating a DHT22 sensor, we can monitor ambient temperature and adjust battery

management systems accordingly. The sensor can alert the system to activate dehumidifiers or heat elements when needed.

(d) NEO-7M GPS satellite positioning module, is a high-performance GPS receiver designed for precise location tracking and navigation applications. It provides accurate positioning data by connecting to a network of GPS satellites, delivering information such as latitude, longitude, altitude, and speed. It offers enhanced sensitivity and quick satellite acquisition times. It supports multiple GNSS constellations, including GPS and GLONASS, which improves its accuracy and reliability. IoT, GPS data can be used to track the vehicle's location when remote diagnostics are needed. Fleet operators can remotely analyse the health of the vehicle's battery or other components based on its location. GPS could be used not only for tracking the vehicle's location on a web dashboard but also for offering valuable insights like battery efficiency based on route, distance to charging stations, and range prediction.

(e) ESP32 is a powerful microcontroller that significantly enhances the functionality of an electric vehicle (EV) monitoring system by offering dual-core processing, more memory, and advanced connectivity options such as Wi-Fi and Bluetooth. Its versatility allows seamless integration of multiple sensors for real-time tracking of key metrics like battery voltage, temperature, and location using GPS, while enabling efficient data transmission to cloud servers or local dashboards. The ESP32's power efficiency and security features make it ideal for managing energy consumption and ensuring encrypted communication. Its ability to host web server and support mobile app connectivity further enhances user interaction, providing real-time monitoring and control of EV systems.

3.2. System Flowchart of Battery Monitoring System

The following Figure 8 illustrates the flowchart showing the working principle of this proposed system. The system's source code flows the first step involves initializing the sensors connected to the EV's battery. These sensors are tasked with measuring crucial parameters such as voltage, current, temperature, and humidity. The gathered data is essential for evaluating the battery's state.

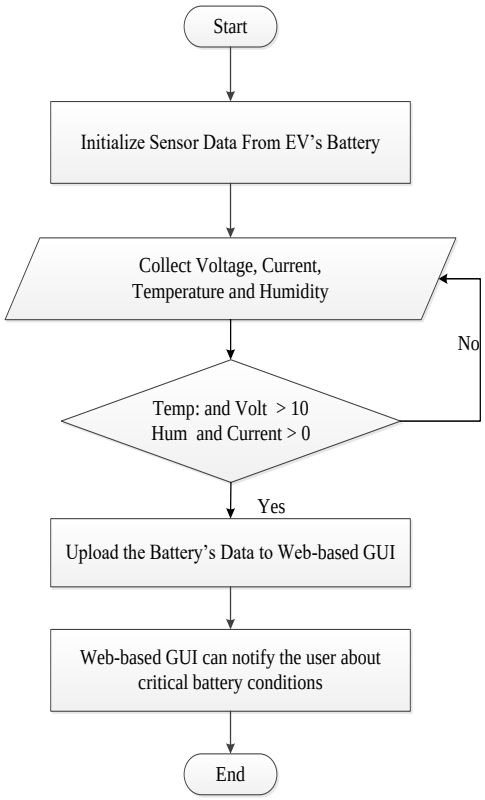


Figure 8. System Flowchart of Battery Monitoring System

The second step, this system actively collects real-time data from the sensors. This stage involves measuring the battery's voltage and current, as well as the ambient temperature and humidity using a DHT22 sensor. The collected data is then subjected to an evaluation process to determine if the battery's operating conditions meet

predefined criteria. Specifically, the flowchart checks whether the temperature exceeds a threshold of 10 °C and voltage exceed a threshold of 10 V and if both humidity and current are greater than zero. These conditions ensure safe and efficient EV operation by monitoring temperature to prevent thermal issues, voltage to avoid under-performance or damage, and humidity and current to maintain optimal environmental and operational standards.

If the battery status exceeds a predefined threshold, continuous monitoring occurs. The ESP32 then uses Wi-Fi to transmit this information to a web-based GUI, which allows for real-time monitoring and alerts users about the battery's condition, especially if there are any critical issues. This proposed system provides a clear and effective method for maintaining EV battery health, using real-time data and remote monitoring to ensure the battery operates safely and efficiently.

3.3. System Flowchart of Vehicle Tracking Unit

Figure 9 illustrates the system flow of the vehicle tracking unit in detail. Initially, the GPS receiver acquires real-time geolocation data, including latitude, longitude, and altitude, by communicating with multiple GPS satellites. This raw data is then transmitted to the ESP32 microcontroller, which processes and formats it into standardized messages suitable for further transmission. The ESP32 ensures that the location data is packaged correctly, including additional information such as timestamps and vehicle identifiers, if necessary. After formatting, the ESP32 sends the data wirelessly through a GSM router connected to a Wireless LAN network.

This data is uploaded to the ThingSpeak cloud platform, where it is processed and visualized in real-time. ThingSpeak's integration with Google Maps allows the base station to display the exact location of the vehicle on a map, enabling live tracking. This continuous flow of data ensures that the base station can monitor the vehicle's movements accurately and in real time, offering a reliable solution for vehicle tracking, fleet management, and remote monitoring applications.

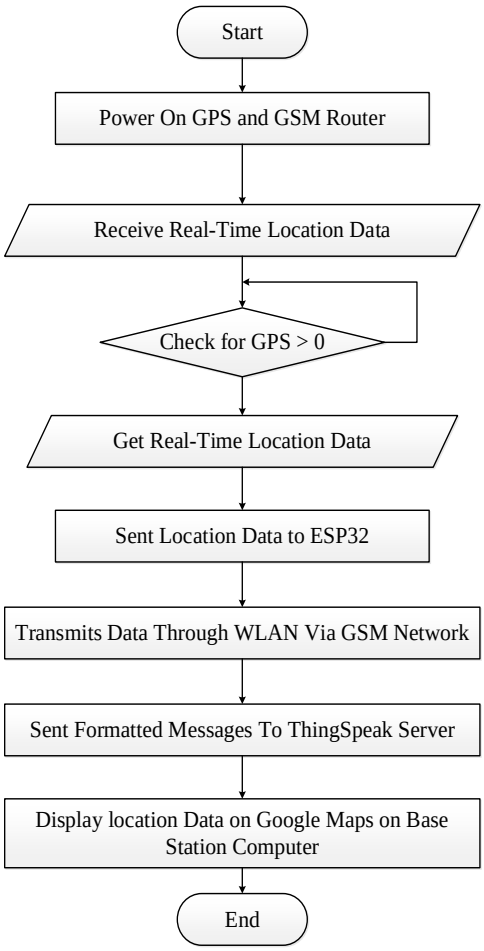


Figure 9. System Flowchart

In our proposed system, where GSM technology is used to share internet data like a router, the ESP32 microcontroller will communicate with the internet via the GSM module, which acts as a hotspot or gateway. The GSM module will establish a connection to the mobile network, providing internet access and enabling the ESP32 to transmit data, such as the vehicle's real-time GPS location, directly to platforms like ThingSpeak. This setup allows the vehicle tracking system to operate wirelessly and independently of external Wi-Fi networks, offering flexibility and mobility for remote monitoring, especially in areas with GSM coverage.

Using GSM technology for internet sharing in a vehicle tracking system has limitations, including inconsistent coverage in remote areas, slow data speeds, and high costs associated with mobile data plans. Latency can delay real-time updates, and continuous use may increase power consumption, impacting efficiency. Additionally, security vulnerabilities in GSM networks pose risks, making strong encryption essential to protect sensitive data.

3.4. System Implementation

The following Figure 10 illustrates the circuit diagram of Wireless Battery Monitoring and vehicle tracking system for electric vehicles, featuring an ESP32 microcontroller, 12V battery, and various sensors for monitoring temperature, voltage, humidity and current. The Neo7M GPS Module is connected for get location data. Additionally, the load model uses a motor driver module to control four motors.

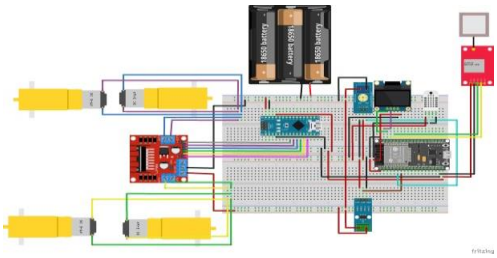


Figure 10. The circuit diagram of WBMS and Vehicle Tracking System for EV

Figure 11 illustrates the upper view of the proposed Wireless Battery Monitoring and vehicle tracking system for electric vehicles. This perspective showcases how each component, including GPS module, sensors and the ESP32 microcontroller, is positioned and interacts within the system.

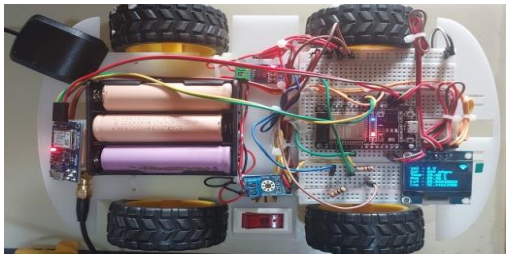


Figure 11. The Upper View of the Proposed Model

3.5. Implementation Process

The implementation process of a real-time battery monitoring and vehicle tracking system for connected WLAN networks encompasses various essential components and technologies. This section delineates the steps and methods involved in the implementation comprehensively.

Proposed battery monitoring and tracking system, several robust measures are implemented, particularly in the web interface and during communication with the ThingSpeak API. Data transmission is secured using HTTPS, encrypting the information in transit to prevent unauthorized access and eavesdropping. Authentication mechanisms, including unique write API keys for API requests, verify the identity of devices sending data, while sensitive user information in the web interface is protected through encryption and secure hashing algorithms for passwords.

In selecting the software for the EV battery monitoring and tracking system, Node.js, Leaflet, and Firebase were chosen for their specific strengths and alignment with project requirements. Node.js was selected for its non-blocking I/O model, which enables efficient handling of multiple simultaneous connections—crucial for a real-time application that requires constant data updates from various sensors.

Leaflet was preferred for its simplicity and flexibility in mapping; unlike more complex solutions like Google Maps, Leaflet offers an open-source platform that allows for easy integration and customization without incurring additional costs.

Firebase was chosen over traditional databases such as MySQL due to its real-time data synchronization capabilities, which provide instant updates to users regarding battery status and vehicle location.

Collectively, these software choices ensure a robust, scalable, and user-friendly system that effectively meets the needs of the project.[7]

3.5.1. Socket.IO

In this project, Socket.IO, a JavaScript library, is utilized to create a WebSocket connection linking the web application with the backend server. This library supports real-time, bidirectional communication, allowing the seamless exchange and updating of the vehicle's GPS coordinates [3].

3.5.2. Leaflet

Leaflet, an open-source JavaScript library, is employed to visualize the vehicle's location on the web application. This library offers mobile-friendly, interactive maps that are easy to use, highly flexible, and support customizable map styles.

3.5.3. Firebase

Firebase, a cloud-based platform, functions as the backend database for storing GPS coordinates and other pertinent data. It is chosen due to its real-time database capabilities, effortless integration with Node.js, and capacity to manage substantial data volumes.

3.5.4. Establishing an HTTP Connection

To initiate an HTTP session with the Neo 7M GPS module, Thingspeak read field data using Read API key must be executed. These Thingspeak read field data are sent to the web application via Wireless LAN to the base station computer, utilizing the ESP32 microcontroller using internet.

The workflow includes sending sensor data from the ESP32 microcontroller to the WLAN and GSM Router via Thingspeak Read API communication. This process allows the module to set up an HTTP session by connecting to the network, configuring the connection type, specifying the Read API Key and Channel ID, reading data from ThingSpeak, initializing the HTTP service, and defining the URL for data transmission. The session can be concluded using the Thingspeak Read API communication. [7]

In the context of connected of connected electric vehicles and GPS based location tracking technology, the backend server is essential for data management and processing. It acquires GPS Latitude and Longitude coordinates from the connected electric vehicle through the frontend web-based application. These coordinates are sent via an HTTP request and processed by the backend server, which utilizes technologies like Node.js and Express.js. Utilizing GPS data, the backend server performs real-time tracking and route optimization. Communication with the connected vehicle is facilitated using Socket.IO, ensuring seamless and efficient data exchange. This integration allows the backend this server to deliver real-time updates and notifications to users, guaranteeing precise tracking of the vehicle's location and movements.

3.5.5. Web Development

The frontend of the web application is created using HTML, CSS, and JavaScript. HTML structures the content, CSS styles the webpage, and JavaScript adds interactive functionality. Tailwind CSS, a utility-first framework, speeds up development by offering pre-defined utility classes for styling HTML elements, reducing the need for extensive custom CSS and class name generation. Next.js, a React-based framework for server-side rendering, leverages React and Node.js to help developers build scalable and optimized web applications. It offers features like server-side rendering, automatic code splitting, static site generation, and performance optimization. Node.js, an open-source JavaScript runtime environment, is selected for its efficient I/O capabilities and its ability to handle numerous connections simultaneously, making it ideal for developing both the web application and the backend server.

4. RESULTS AND DISCUSSION

In this research, the EV wireless battery monitoring and tracking system will undergo testing and simulation using a mini model powered by a 12V battery. This model will incorporate various sensors to measure key parameters such as voltage, current, and temperature, allowing for a practical evaluation of the system's performance in a controlled environment.

Simultaneously, real-world GPS tests will be conducted using an actual vehicle to assess the system's capability to track location accurately while driving. This dual approach of simulation and real-world testing will provide valuable insights into the system's functionality, reliability, and effectiveness in monitoring battery health and tracking vehicle movements, ensuring that it meets the necessary requirements for practical applications.

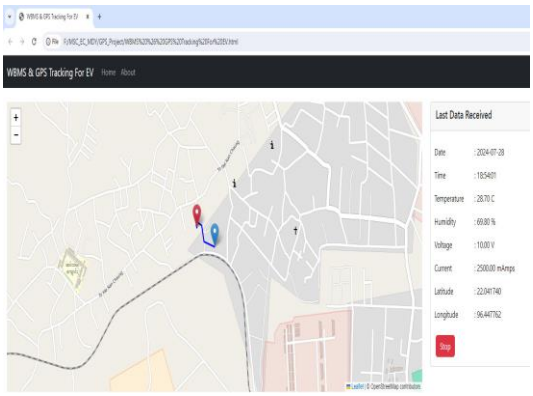


Figure 12 (a). Real-Time Vehicle Tracking on Web Application (Map Start Point)

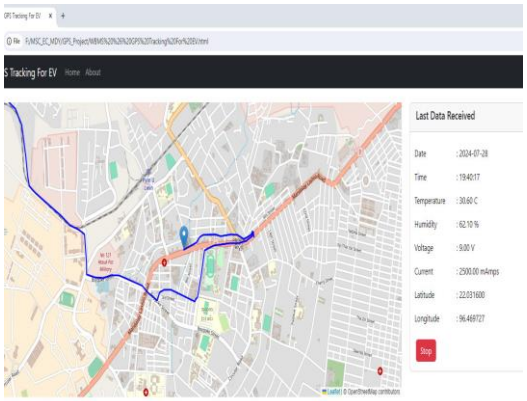


Figure 12 (b). Real-Time Vehicle Tracking on Web Application (Map Middle Point)

Figure 12 (c). Real-Time Vehicle Tracking on Web Application (Map End Point)

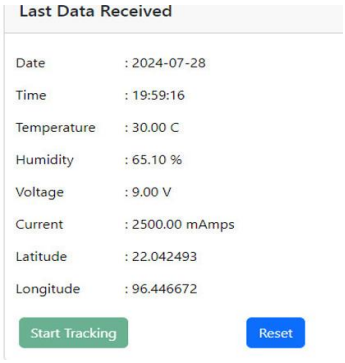


Figure 12 (d). Battery Condition Dashboard Display

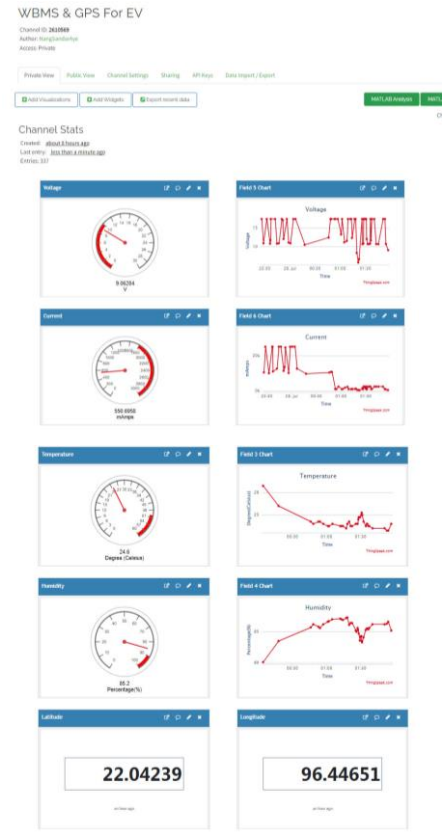


Figure 12 (e). Real-time Data Analysis of using Built in ThingSpeak Dashboard

Figure 12 (a, b, c, d, e) illustrates the real-time monitoring setup of the Wireless Battery Monitoring and vehicle tracking system for Electric Vehicles. The figure demonstrates the EV driving conditions and the web-based, real-time display of the EV's battery status on a computer dashboard. Before users can access the GUI shown in Figure 12 (e), they must first navigate through a secure login page. This login page is designed to ensure safe data handling by requiring users to authenticate themselves with a username and password. This security measure guarantees that only authorized users can view and manage the Wireless Battery Monitoring and vehicle tracking system for Electric Vehicles. The system also supports data sharing and allows guest accounts to view the ThingSpeak dashboard without modifying the data.

The left portion of the dashboard on the screen provides live updates on parameters, including battery voltage, current, temperature and humidity using gauge meter and latitude data. The right side of the dashboard demonstrates performance graphs and longitude data. These graphs are showcasing the effective use of a web-based interface for tracking and analysing the battery's condition. The gauge meters and graphs visually represent this data, allowing uses to monitor the battery's condition in real-time. The latitude and longitude data indicate the position of the electric vehicle and track its lane position.

Implementing the Wireless Battery Monitoring System and vehicle tracking system in electric vehicles can reduce overall vehicle weight and toxic gas emissions by optimizing battery usage and ensuring efficient operation. This system extends battery life, decreases the demand for frequent replacements, and lowers long-term maintenance costs. Real-time monitoring and predictive maintenance have the capability to detect issues early, preventing costly failures and minimizing downtime. Additionally, optimized charging cycles enhance energy efficiency, while the integration of energy harvesting technologies for sensors reduces dependency on the main battery, contributing to significant cost savings and environmental benefits.

By analysing real-time sensor data and applying advanced machine learning algorithms, the system can detect early signs of battery degradation and anomalies. Adoption this proactive strategy enables timely interventions, extending the lifespan of electric vehicle batteries and reducing unexpected downtimes. The integration of predictive maintenance not only enhances battery performance and reliability but also contributes significantly to more sustainable and cost-effective electric vehicle operations.

5. CONCLUSIONS

This initiative aimed to design and implement a Wireless Battery Monitoring System of electric vehicle and tracking system of electric vehicles, enabling real-time monitoring of battery performance. The project involved developing hardware and a web-based interface to monitor battery conditions. Key considerations included reliability, safety, and environmental sustainability, which enhance public perception and customer satisfaction. By integrating the WBMS, operators can improve EV reliability, operational efficiency, and public confidence in electric transportation, ultimately contributing to a more efficient and real-time monitoring and data analysis. The vehicle tracking system can help prevent theft, assess driver performance, and enhance safety by contributing to a sustainable transportation network.

EV battery monitoring and tracking using wireless technology hinges on several key factors. Real-time data transmission is essential for providing continuous insights into battery voltage, current, temperature, and location, while robust communication protocols like MQTT or HTTP ensure

efficient and secure data transfer. Scalability is crucial, allowing for the integration of additional sensors or features as needed. Energy-efficient components help minimize impact on the EV's overall battery life, and strong data security measures, such as encryption, protect sensitive information from unauthorized access.

A user-friendly interface for monitoring and real-time alerts for critical battery conditions enhances safety and usability. Additionally, geolocation services enable accurate tracking, and data analytics can reveal performance trends, optimizing usage and extending battery lifespan. Integrating these factors creates a comprehensive and reliable system for effective EV battery monitoring and tracking.

6. FUTURE SCOPE

Future work on the Wireless Battery Monitoring System (WBMS) and vehicle tracking system for electric vehicles will focus on integrating advanced algorithms and machine learning techniques to enhance accuracy and efficiency in detecting battery health issues. These models will analyse vast amounts of sensor data, recognize patterns in battery performance, and predict failures before they occur, enabling proactive maintenance and reducing downtime.

Additionally, integrating advanced sensors for precise monitoring, enhancing data security with robust encryption, and implementing real-time analytics will further improve WBMS and vehicle tracking system functionality. Efforts will also include scaling the system for larger EV fleets, exploring energy harvesting to power sensors. This will optimize routes and ensure safety. Standardizing protocols and improving the user interface for better data visualization will ensure a more efficient, reliable, and secure WBMS and vehicle tracking system for electric vehicles.

ACKNOWLEDGEMENTS

The author extends her heartfelt gratitude to her supervisor, Dr. Pa Pa Winn San, Head of the Department, for her exceptional supervision, patient guidance, invaluable advice, and continuous support throughout the research period. The author extends special thanks to her Co-supervisor, Dr. Aye Min Soe, for her valuable advice and guidance on this journal. Lastly, the author deeply thanks all of the teachers from the Department of Electronic Engineering, Technological University (Mandalay), and also be grateful to the colleagues who have contributed to the accomplishment of this work.

REFERENCES

[1] Renuka Modak; " *Wireless Battery Monitoring System for Electric Vehicle*", Department of Electronic Engineering, MIT Academy of Engineering, March 2021.

[2] S.Prabakaran, N.Ashok, D.Arunkumar And D.Ariharan, "Smart Battery Management System of Electric Vehicles using IoT Technology", (IJCRT), ISSN: 2320-2882, Vol. 11, Issue. 4, 2023.

[3] M H A Wahab, N I M Anuar, R Ambar, A Baharum, M S Sulaiman, S S M Fauzi, H F Hanafi, "IoT-Based Battery Monitoring System for Electric Vehicle", International research Journal Engineering & Technology, 2022.

[4] Global Navigation Satellite System (GNSS).

[5] N. M. Mane, D. V. Kale, S. B. Thombre and R. S. Piske, "IoT Based Battery Management System," International Research Journal Engineering & Technology (IRJET), Volume 10, May, 2023.

[6] https://e-amrit.niti.gov.in/benefits-of-electric -vehicles, "Benefits of Electric Vehicles", mobility-niti@nic.in.

[7] J.Nagendran, R.Vasanth, J.Tamizharasan, P.A.Anas and G.Swathisri, "IOT Based Battery Monitoring System in E-Vehicle ," International Journal of New Innovations in Engineering and Technology, Volume 24, Issue 1, March 2024.

Authors

Biography

My name is Nann Sandar Aye. I am 31 years old, I was born in Kyaukme on March 20, 1994. I am currently studying Master Electronic Engineering at the Technology University (Mandalay).

Dr. Pa Pa Winn San
B.E (Electronics), 2002, MTU;
M.E (Electronics), 2007, MTU;
Ph.D (Electronic Engineering), 2011, MTU
Professor and Head
Department of Electronic Engineering
Technological University (Mandalay)
Currently being interesting in the field of Communication & Control and taking supervisions on both undergrad and postgraduate students of this University.

Dr. Aye Min Soe
B.E (Electronics), 2007, TUM;
M.E (Electronics), 2011, MTU;
Ph.D (Electronic Engineering), 2015, MTU
Associate Professor
Department of Electronic Engineering
Technological University (Mandalay)
Currently being interesting in the field of Networking & Communication and taking supervisions on both undergrad and postgraduate students of this University.

COMPARATIVE ANALYSIS OF U-NET AND RESNET-ENHANCED U-NET ARCHITECTURES EVALUATING BINARY CROSS-ENTROPY AND FOCAL TVERSKY LOSS FUNCTIONS IN SATELLITE IMAGE SEGMENTATION

Myo Myat Thu[1], Phone Naing[2], and Kyaw Kyaw Lin[3]

[1,2,3] Department of Computer Science, Higher Education Center (Pyin Oo Lwin)
[1]nyimyatthu49@gmail.com, [2]kophonenaing7@gmail.com,
[3]kyawkyawlin@gmail.com

ABSTRACT

Roads are fundamental to transportation infrastructure and daily life, yet accurately identifying road features from high-resolution remote sensing images, such as those from Google satellite imagery, presents substantial challenges. Recent advancements in Convolutional Neural Networks (CNNs) have shown promising capabilities in detecting and segmenting roadways within these complex images. Among these advancements, the Residual Network (ResNet) architecture represents a significant breakthrough in deep learning by addressing the vanishing gradient problem, which is particularly common in very deep networks. This study conducts a comprehensive evaluation of various loss functions applied to a custom dataset derived from remote sensing data via ArcGIS Pro, using Google satellite imagery, to accurately extract roads in the Pyin Oo Lwin region. The experiments involve U-Net and a ResNet50-enhanced U-Net architecture, focusing on improving road segmentation accuracy with binary cross-entropy and focal Tversky loss functions. Through this approach, the study aims to enhance the precision of road feature extraction in high-resolution satellite images, with the potential for significant advancements in road network mapping and analysis.

Keywords

Convolutional Neural Networks (CNN), Loss Functions, Image Segmentation, U-Net, ResNet, Geographic Information System (GIS), Binary Cross-Entropy, Focal Tversky

1. INTRODUCTION

The extraction of roads from satellite images is a critical task with significant implications for navigation, disaster management, and urban planning. These images feature various types of roads, including highways, urban and rural roads, and byways. However, this task faces considerable challenges due to factors such as noise, occlusions, and the complex structure of road networks.

Currently, road extraction often relies on manual interpretation, which is time-consuming and susceptible to inconsistencies due to subjective judgments. Remote sensing images are further complicated by issues like terrain variations, shadows, and occlusions, making road extraction more difficult.

Recent focus has been on Convolutional Neural Networks (CNNs), which have garnered substantial attention and significantly advanced computer vision while addressing challenges in

natural language processing. CNNs find applications in various domains, including autonomous driving, remote sensing image analysis, and medical image processing. Noteworthy architectural innovations in this field include the Fully Convolutional Network (FCN) [2] and the U-Net [1].

Image segmentation can be defined as a classification task at the pixel level. An image comprises numerous pixels, and groups of these pixels define different elements within the image. The process of classifying these pixels into distinct elements is known as semantic image segmentation. Selecting the appropriate loss or objective function is crucial when designing deep learning architectures for complex image segmentation tasks, as it drives the learning process of the algorithm. Consequently, since 2012, researchers have experimented with various domain-specific loss functions to enhance the results for their datasets.

The satellite imagery used for road extraction is sourced from Google Satellite imagery and covers Pyin Oo Lwin region, which spans an area of 14.508206 square kilometres. The ground resolution of the image pixels is 10m/pixel.

2. METHODOLOGY

In this paper, our main goal is to train a custom dataset using specific loss functions within the U-Net and ResNet50 architectures to achieve accurate road segmentation from satellite images, with a particular focus on Google Satellite images.

2.1. GIS

Geographic Information System (GIS) is a robust technology that enables the capture, storage, analysis, and presentation of spatial data. It merges various types of geographical information—such as maps, satellite imagery, aerial photography, and survey data—into a unified platform. By integrating these diverse data sources, GIS forms a comprehensive database that offers a multifaceted perspective of geographic regions.

This combined information is georeferenced, meaning it is tied to specific geographic locations or coordinates on the Earth's surface, ensuring precise spatial representation. GIS technology allows users to effectively visualize, interpret, and understand spatial patterns and relationships. It supports a broad spectrum of applications, ranging from urban planning and environmental management to disaster response and resource allocation, making it an essential tool in numerous fields [3].

2.2 Image Segmentation

Image segmentation plays a critical role in visual understanding systems, aiming to produce dense predictions for images by assigning each pixel to a predefined class label (semantic segmentation), linking pixels to specific object instances (instance segmentation), or combining both approaches (panoptic segmentation). This process groups pixels with similar semantics into meaningful high-level concepts.

Segmentation is widely used in fields such as medical image analysis, video surveillance, and augmented reality. Various model architectures, ranging from Convolutional Neural Networks (CNNs) to Transformers, have been developed for semantic segmentation. However, achieving optimal performance in segmentation models depends on selecting the appropriate network architecture and objective function [10].

A significant area of research in image segmentation involves overcoming challenges such as class imbalance, limited datasets, and noisy, biased, and inconsistent labels. This is accomplished by developing robust loss functions that facilitate the joint optimization of model parameters.

2.3 U-Net Architecture

The U-Net architecture leverages the principles of fully convolutional neural networks to effectively capture context-based features and ensure accurate localization. It consists of two main components: the encoder and the decoder, which make up the first and second halves of the model, respectively. The encoder, also known as the contracting path, is responsible for extracting features from the input image. In the U-Net architecture, the encoder utilizes two consecutive blocks of 3x3 convolutions without padding, followed by a rectified linear unit (ReLU) activation function, and 2x2 max pooling with a stride of 2 for down-sampling. This down-sampling process encodes the input image into feature map representations at different levels, doubling the number of features at each step.

On the other hand, the decoder, or expansive path, forms the second part of the U-Net architecture. Its goal is to reconstruct the segmentation by mapping the discriminative features learned by the encoder back into the pixel space, thereby achieving higher resolution. Each step in the decoder involves up-sampling the feature map, followed by a 2x2 up-convolution that reduces the number of feature channels by half. Additionally, the feature maps cropped from the corresponding layers in the contracting path are concatenated with the up-sampled feature map. This is then followed by two consecutive 3x3 convolutions, each with ReLU activation. The U-Net model comprises a total of 23 convolutional layers.

Originally introduced in 2014 for the semantic segmentation of biomedical images, the U-Net model applies the concept of per-pixel classification. It is widely used in medical imaging to localize specific parts of an image, typically where abnormalities are present. In semantic segmentation, unlike classification, the input and output have the same shape, allowing for precise localization of objects or areas of interest. The architecture of U-Net is illustrated in Figure 1, and the flowchart of the road extraction system from satellite images is shown in Figure 2.

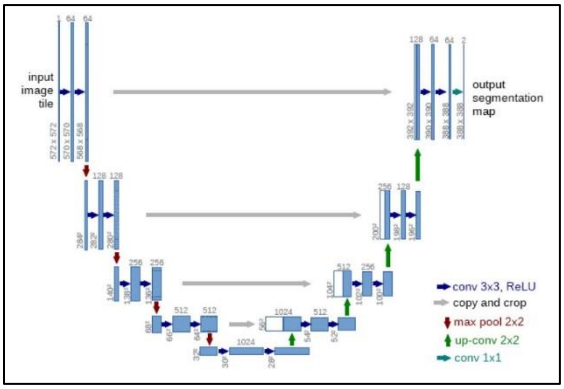


Figure 1. U-Net Architecture

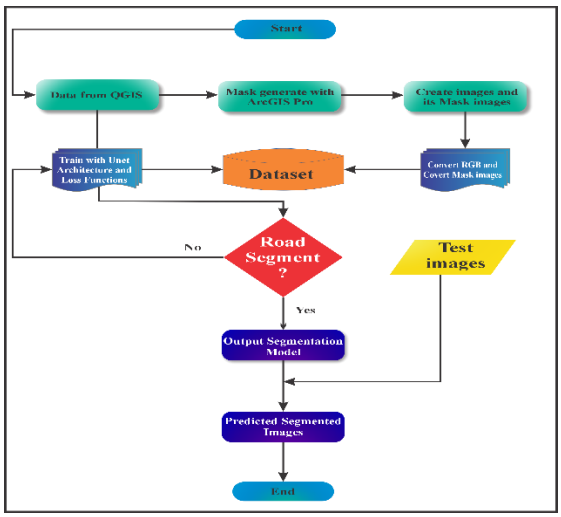


Figure 2. Flowchart of Satellite Images Segmentation System Design

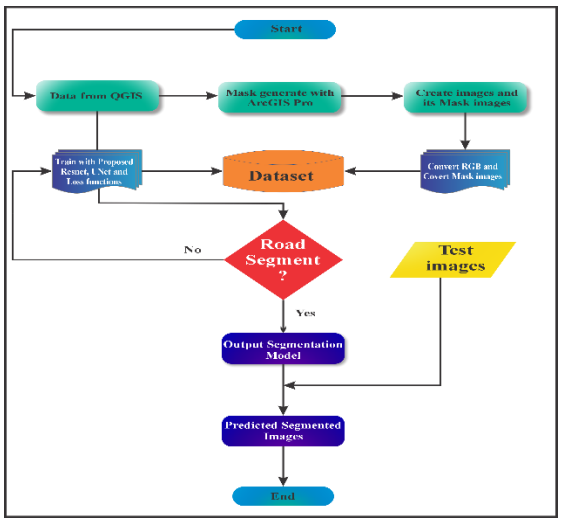


Figure 3. Flowchart of Satellite Images Segmentation System Design

2.4 Residual Network

The ResNet architecture, or Residual Network, represents a major advancement in deep

learning, particularly in addressing the critical issue of vanishing gradients. This problem, common in very deep networks, occurs due to the repeated multiplication of gradients during backpropagation, which causes them to decay exponentially and hampers effective learning. ResNet cleverly introduces "residual connections" or "skip connections," which allow information to bypass certain layers and flow directly to later ones [11]. This mechanism enables the network to learn "identity mappings," allowing it to effectively "skip" layers when necessary. By creating a direct path for gradients, ResNet successfully mitigates the vanishing gradient problem, facilitating the training of much deeper networks while preserving gradient flow. This innovation has had a profound impact on the field, enabling the creation of networks with hundreds or even thousands of layers and leading to significant performance gains in tasks such as image classification and object detection.

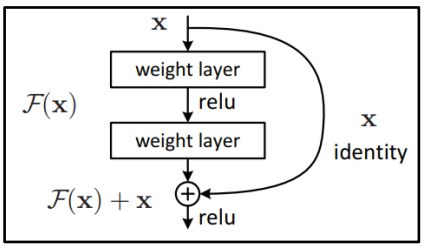


Figure 4. Residual-Block

2.5 Loss Functions

In deep learning, the loss function is a crucial element that measures how well a model's predictions align with the actual data, directing the optimization process during training[7]. The U-Net architecture, which is specifically tailored for image segmentation tasks, particularly benefits from a carefully selected loss function due to its requirement for accurate pixel-wise classifications.

2.5.1 Binary Corss-Entropy

Cross-entropy is defined as a measure of the difference between two probability distributions for a given random variable or set of events [4]. This metric is extensively used for classification tasks because it quantifies how well the predicted probability distribution matches the true distribution of the labels. In the context of image segmentation, which involves classifying each pixel in an image, cross-entropy is particularly effective. This is because segmentation can be seen as a form of pixel-level classification, where each pixel is assigned to a specific class. Therefore, using cross-entropy as a loss function helps in accurately classifying each pixel according to its true label, improving the overall segmentation performance.

$$BCE = -\frac{1}{N}\sum_{i=1}^{n}[y_i .* \log(p_i) + (1 - y_i) * log\,(1 - p_i)] \quad (1)$$

where, N is the number of samples in the dataset. y_i is the actual label for the i^{th} sample (0 or 1). p_i is the predicted probability that the i^{th} sample belongs to class 1.

2.5.2 Focal Loss

Focal loss is an enhanced version of the cross-entropy loss designed to assign different weights to easy and hard samples. In this context, hard samples are those that are misclassified with a high probability, indicating that the model struggles to predict them correctly. Conversely, easy samples are those that are correctly classified with a high probability, showing that the model finds them straightforward to predict. By assigning greater weight to hard samples and less to easy ones, focal loss balances their influence on the overall loss [5]. This approach ensures that the model focuses more on difficult examples, which can lead to improved learning and performance, especially in scenarios with class imbalances. For the hamming window, the equation is

$$FL(P_t) = -\alpha_t(1 - p_t)^{\gamma}\log(p_t) \quad (2)$$

where, p_t represents the predicted probability of the true class. α_t is the focusing parameter that is dynamically adjusted based on the true class label. It assigns higher weights to minority class samples to counteract the effects of class imbalance. γ is the modulating factor that reduces the loss contribution of well-classified examples (where p_t is close to 1).

2.5.3 Tversky index

The Tversky index is a metric used to quantify the similarity between two sets, often employed in tasks such as image segmentation [10]. It extends the concept of the Dice coefficient by introducing parameters that allow for the differential weighting of false positives and false negatives. For two sets P (predicted positives) and G (ground truth positives), the Tversky index $T_{\alpha,\beta}$ is defined as.

$$T_{\alpha,\beta} = \frac{|P \cap G|}{|P \cap G| + \alpha|P \setminus G| + \beta|G \setminus P|} \quad (3)$$

where, $|P \cap G|$represents the number of true positive instances where both P and G are positive. $|P \setminus G|$ represents false positive instances where the model predicts positive but ground truth is negative. $|G \setminus P|$ represents false negative instances where the model predicts negative but ground truth is positive. α and β are parameters that adjust the weights of false positives and false negatives, respectively.

2.5.4 Focal Tversky Loss

Similar to Focal Loss, which strategically down-weights easily classified or frequently occurring examples to enhance the model's focus on challenging instances, Focal Tversky Loss applies a similar principle [6]. It utilizes coefficients to specifically address difficult examples, particularly those associated with small Regions of Interest (ROIs). By incorporating these coefficients, the loss function aims to optimize the model's performance in learning and accurately segmenting intricate features or regions within the data, thereby improving overall segmentation quality, especially in scenarios where precise localization and identification of small ROIs are critical.

$$L_{FT} = (1 - T_{\alpha,\beta}(P, G)^{\gamma}) \quad (4)$$

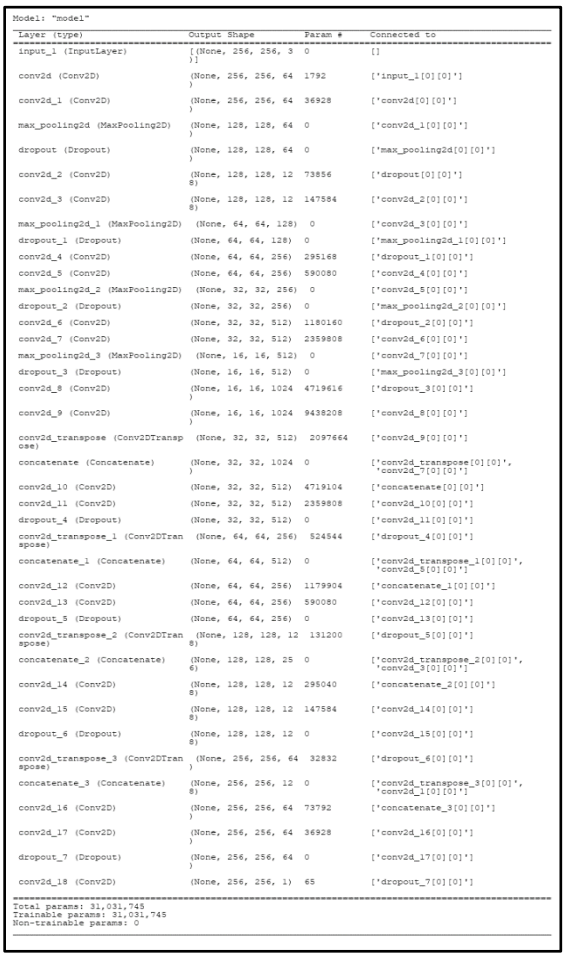
where, $T_{\alpha,\beta}$ indicates Tversky index, and γ can range from [1,3].

3. EXPERIMENTAL RESULTS

This section details an experiment designed to showcase a road extraction system using remote sensing data from Google Satellite images. The system was trained with specific loss functions and utilized the U-Net and ResNet50 architectures.

3.1. Dataset Preparation

The dataset used for training consisted of Google Satellite images, particularly from the Pyin Oo Lwin region, comprising 5,368 images with a resolution of 512x512 pixels. The training was performed on a system equipped with an AMD Ryzen 5880x processor and an NVIDIA GeForce RTX 12GB GPU. The images were split into training, validation, and test sets with a ratio of 70:20:10. For the training process, the resolution of the images was resized to 256x256 pixels.



Model: "model"

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 256, 256, 3)]	0	[]
conv2d (Conv2D)	(None, 256, 256, 64)	1792	['input_1[0][0]']
conv2d_1 (Conv2D)	(None, 256, 256, 64)	36928	['conv2d[0][0]']
max_pooling2d (MaxPooling2D)	(None, 128, 128, 64)	0	['conv2d_1[0][0]']
dropout (Dropout)	(None, 128, 128, 64)	0	['max_pooling2d[0][0]']
conv2d_2 (Conv2D)	(None, 128, 128, 128)	73856	['dropout[0][0]']
conv2d_3 (Conv2D)	(None, 128, 128, 128)	147584	['conv2d_2[0][0]']
max_pooling2d_1 (MaxPooling2D)	(None, 64, 64, 128)	0	['conv2d_3[0][0]']
dropout_1 (Dropout)	(None, 64, 64, 128)	0	['max_pooling2d_1[0][0]']
conv2d_4 (Conv2D)	(None, 64, 64, 256)	295168	['dropout_1[0][0]']
conv2d_5 (Conv2D)	(None, 64, 64, 256)	590080	['conv2d_4[0][0]']
max_pooling2d_2 (MaxPooling2D)	(None, 32, 32, 256)	0	['conv2d_5[0][0]']
dropout_2 (Dropout)	(None, 32, 32, 256)	0	['max_pooling2d_2[0][0]']
conv2d_6 (Conv2D)	(None, 32, 32, 512)	1180160	['dropout_2[0][0]']
conv2d_7 (Conv2D)	(None, 32, 32, 512)	2359808	['conv2d_6[0][0]']
max_pooling2d_3 (MaxPooling2D)	(None, 16, 16, 512)	0	['conv2d_7[0][0]']
dropout_3 (Dropout)	(None, 16, 16, 512)	0	['max_pooling2d_3[0][0]']
conv2d_8 (Conv2D)	(None, 16, 16, 1024)	4719616	['dropout_3[0][0]']
conv2d_9 (Conv2D)	(None, 16, 16, 1024)	9438208	['conv2d_8[0][0]']
conv2d_transpose (Conv2DTranspose)	(None, 32, 32, 512)	2097664	['conv2d_9[0][0]']
concatenate (Concatenate)	(None, 32, 32, 1024)	0	['conv2d_transpose[0][0]', 'conv2d_7[0][0]']
conv2d_10 (Conv2D)	(None, 32, 32, 512)	4719104	['concatenate[0][0]']
conv2d_11 (Conv2D)	(None, 32, 32, 512)	2359808	['conv2d_10[0][0]']
dropout_4 (Dropout)	(None, 32, 32, 512)	0	['conv2d_11[0][0]']
conv2d_transpose_1 (Conv2DTranspose)	(None, 64, 64, 256)	524544	['dropout_4[0][0]']
concatenate_1 (Concatenate)	(None, 64, 64, 512)	0	['conv2d_transpose_1[0][0]', 'conv2d_5[0][0]']
conv2d_12 (Conv2D)	(None, 64, 64, 256)	1179904	['concatenate_1[0][0]']
conv2d_13 (Conv2D)	(None, 64, 64, 256)	590080	['conv2d_12[0][0]']
dropout_5 (Dropout)	(None, 64, 64, 256)	0	['conv2d_13[0][0]']
conv2d_transpose_2 (Conv2DTranspose)	(None, 128, 128, 128)	131200	['dropout_5[0][0]']
concatenate_2 (Concatenate)	(None, 128, 128, 256)	0	['conv2d_transpose_2[0][0]', 'conv2d_3[0][0]']
conv2d_14 (Conv2D)	(None, 128, 128, 128)	295040	['concatenate_2[0][0]']
conv2d_15 (Conv2D)	(None, 128, 128, 128)	147584	['conv2d_14[0][0]']
dropout_6 (Dropout)	(None, 128, 128, 128)	0	['conv2d_15[0][0]']
conv2d_transpose_3 (Conv2DTranspose)	(None, 256, 256, 64)	32832	['dropout_6[0][0]']
concatenate_3 (Concatenate)	(None, 256, 256, 128)	0	['conv2d_transpose_3[0][0]', 'conv2d_1[0][0]']
conv2d_16 (Conv2D)	(None, 256, 256, 64)	73792	['concatenate_3[0][0]']
conv2d_17 (Conv2D)	(None, 256, 256, 64)	36928	['conv2d_16[0][0]']
dropout_7 (Dropout)	(None, 256, 256, 64)	0	['conv2d_17[0][0]']
conv2d_18 (Conv2D)	(None, 256, 256, 1)	65	['dropout_7[0][0]']

Total params: 31,031,745
Trainable params: 31,031,745
Non-trainable params: 0

Figure 5. Model Summary and Parameters of U-Net Architecture

3.2. Training Process

The road segmentation system for remote sensing images in the Pyin Oo Lwin region underwent training using specific datasets, including annotated ground truth images and corresponding mask images. To optimize computational efficiency, the training process operated on batches of 16 images. Additionally, a preprocessing step involved resizing the input images to dimensions of 256 x 256 pixels, aimed at mitigating computational resource requirements. During the training process, the validation loss is monitored using early stop function to prevent overfitting, and the training process is stopped when the validation loss stopped improving for 1000 epochs.

3.3. Evaluation Metric

After Evaluation metrics play a crucial role in assessing the performance of segmentation models. In this work, we have analyzed our results using the Dice Coefficient metric [12]. The Dice Coefficient, also known as the overlap index, measures the degree of overlap between the ground truth and the predicted output.

$$Dice\ coefficient = \frac{2|A \cap B|}{|A|+|B|} \quad (6)$$

Where |A| and |B| represent two sets, usually the binary masks. |A ∩ B| denotes the size of the intersection of the two sets. |A| and |B| represent the sizes of the individual sets. The Dice coefficient provides a value between 0 and 1, where 1 indicates a perfect match between the two sets. It measures the overlap or similarity between the predicted and ground truth segmentations. The figures 3,4,5 illustrate the training and validation loss curves obtained during the training process, as well as the Dice Coefficient metric. These visualizations provide insights into the model's performance over time, showcasing how the loss values decrease and how the Dice Coefficient improves, indicating better segmentation accuracy. Table 1 and table 2 show the results of Binary Cross-Entropy and Focal Tversky Loss along with their respective Dice Coefficient scores. In this table, the Focal Tversky Dice Coefficient metric score is higher than that of the other loss functions.

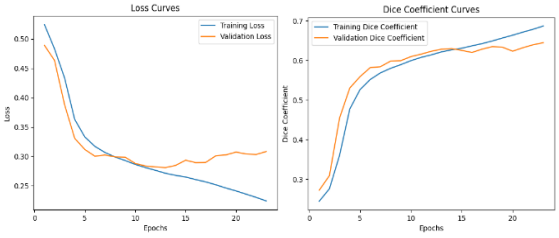


Figure 6. Training Results of Binary Cross-Entropy Loss

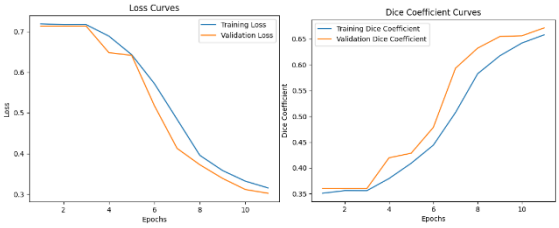


Figure 7. Training Results of Focal Tversky Loss

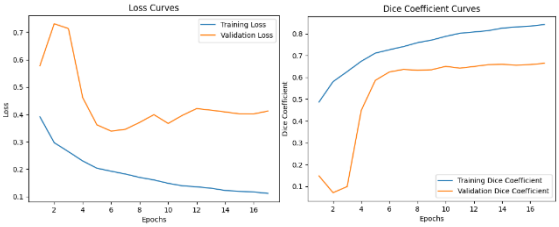


Figure 8. Training Results of BCE Loss with ResNet-Enhanced UNet

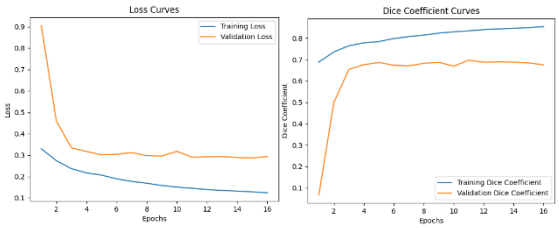


Figure 9. Training Results of Focal Tversky Loss with ResNet-Enhanced UNet

Table.1 Measurement Results of Loss and Dice Coefficient with U-Net

Loss Name		Dice Coefficient
Binary Cross Entropy	0.2772	0.6205
Focal Tversky	0.2924	0.6849

Table.2 Measurement Results of Loss and Dice Coefficient for ResNet-Enhanced U-Net

Loss Name		Dice Coefficient
Binary Cross Entropy	0.3428	0.6520
Focal Tversky	0.2686	0.7081

4. Experimental Results

U-Net and ResNet-Enhanced UNet architectures are trained using Binary Cross-Entropy (BCE) and Focal Tversky Loss. Following the training phase, inference is performed on the resultant models. The segmented images obtained from this inference process undergo a detailed comparative analysis. This analysis aims to evaluate and compare the performance of the different loss functions. Figure 6 showcases a comparative evaluation of segmented images produced using Binary Cross-Entropy and Focal Tversky Loss. From left to right, the figure highlights a comparison with ground truth images using various loss functions. It demonstrates the effectiveness of each loss function in accurately segmenting the images, providing insights into their respective strengths and weaknesses in handling the segmentation task. Focal Tversky Loss is the most optimized and closely matches the ground truth images for road width in the output segmented images.

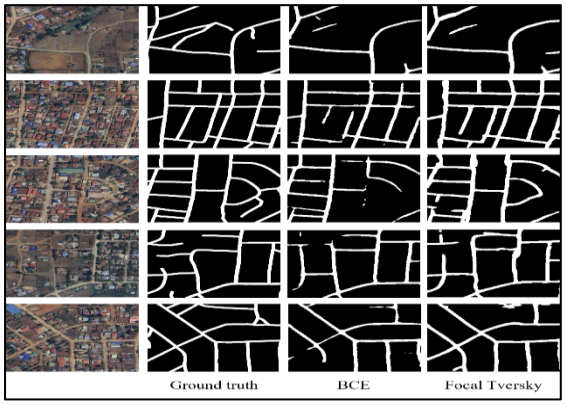


Figure 10. Comparsion of Segmented Images using Different Loss Functions using ResNet-Enhanced U-Net

5. Conclusion

In this study, we focus on road extraction from satellite images using U-Net and ResNet-Enhanced UNet architectures. The model is trained with Binary Cross-Entropy and Focal Tversky Loss. The primary objective is to segment roads from satellite images, particularly those obtained from Google Satellite. During the segmentation process, the system distinguishes roads from other elements within the satellite imagery. The results indicate that the Focal Tversky Loss yields the highest Dice Coefficient compared to the other loss functions, demonstrating its effectiveness in this application.

This superior performance suggests that the Focal Tversky Loss is particularly well-suited for handling the imbalanced nature of the segmentation task, especially when dealing with small and challenging regions of interest. The experiment successfully segments roads in both urban and rural areas, breaking down the images into meaningful segments that accurately represent the road networks. This demonstrates the system's capability to handle diverse environments and produce reliable segmentation results, making it a valuable tool for remote sensing applications and geographic information systems (GIS).

Acknowledgements

The author thanks his supervisor Dr. Phone Naning, Professor, Department of Computer Science, Higher Education Centre, Pyin Oo Lwin, who has brought him into a new territory of knowledge and Co-supervisors Dr. Myo Min Hein and Dr. Kyaw Kyaw Lin, Assistant Lecturer, Department of Computer Science, Higher Education Centre, Pyin Oo Lwin, for their great support and invaluable advices during the progress of this research. And then, the author would like to express his sincere thanks to all compassionate individuals who helped directly and indirectly during this research.

REFERENCES

[1] O. Ronneberger, P. Fischer, and T. Brox, "UNet: convolutional networks for biomedical image segmentation," in *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, Springer, Munich, Germany, October 2015.

[2] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, Boston, MA, USA, June 2015.

[3] Ershad Ali, "Geographic Information System (GIS): Definition, Development, Applications & Components", ResearchGate, March 2020.

[4] Ma Yi-de, Liu Qing, and Qian Zhi-Bai. Automated image segmentation using improved pcnn model based on cross-entropy. In Proceedings of 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing, 2004., pages 743–746. IEEE, 2004.

[5] TY Lin, P Goyal, R Girshick, K He, and P Doll´ar. Focal loss for dense object detection. arxiv 2017. arXiv preprint arXiv:1708.02002, 2002.

[6] Nabila Abraham and Naimul Mefraz Khan., "A novel focal tversky loss function with improved attention u-net for lesion segmentation", In 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), pages 683–687. IEEE, 2019.

[7] Shruti Jadon. A survey of loss functions for semantic segmentation. In 2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB), pages 1–7. IEEE, 2020.

[8] Reza Azad, Ehsan Khodapanah Aghdam, Amelie Rauland, Yiwei Jia, Atlas Haddadi Avval, Afshin Bozorgpour, Sanaz Karimijafarbigloo, Joseph Paul Cohen, Ehsan Adeli, and Dorit Merhof., "Medical image segmentation review: The success of u-net", arXiv preprint arXiv:2211.14830, 2022.

[9] Shervin Minaee, Yuri Boykov, Fatih Porikli, Antonio Plaza, Nasser Kehtarnavaz, and Demetri Terzopoulos., "Image segmentation using deep learning: A survey", IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(7):3523–3542, 2022.

[10] F Milletari, N Navab, SA Ahmadi, and V-net. Fully convolutional neural networks for volumetric medical image segmentation. In Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV), pages 565–571.

[11] He, K., Zhang, X., Ren, S., & Sun, J. (2015). "Deep Residual Learning for Image Recognition." *arXiv preprint arXiv:1512.03385*.

Authors Biography

Myo Myat Thu received the M.C.Sc degree in Information System from the Moscow State Mining University (MSMU), Russian Federation in 2011. He is now attending the Ph.D. 19th Batch at the Higher Education Center (HEC), Pyin Oo Lwin.

Dr. Phone Naing earned his Master's and Ph.D degree from National Research Nuclear University (MEPhI), (Russian Federation) in 2004 and 2007. Now he is a Professor of Department of Computer Science in HEC.

Dr. Kyaw Kyaw Lin received the M.I.C.Sc. and Ph.D. (IT) from the Moscow Institute of Electronic Technology (MIET), Russian Federation in 2008 and 2017. He has served as an assistant lecturer at the Department of Computer Technology at the Higher Education Center (HEC), Pyin Oo Lwin.

STUDYING IMPORTANCE OF ENCRYPTION AND TESTING SOME POPULAR TOOL

Zin Mar Oo[1], Theint Theint Htwe[2], and Myintzu Phyo Aung[3]

[1,2,3]Faculty of Information Science, University of Computer Studies (Mandalay)
[1]zmo.zinmaroo@gmail.com, [2]theinttheint1784@gmail.com,
[3]myitzu.mm@gmail.com

ABSTRACT

In the digital age, security is of paramount importance, especially for user's sensitive data and user credentials. As developers, it is needed to ensure that sensitive information, such as passwords, must be stored securely to prevent unauthorized access and potential breaches. One of the key techniques applied to enhance security is applying cryptographic techniques. Computing systems requiring cryptography tools are deeply ingrained into modern human lifestyles and business practices. Today, digital technologies are applied in every domain, including healthcare, security transportation, marketing, banking and education. As a result, data has become a vital asset. Cryptography is expanded for many areas of communication, securing e-commerce etc. Cryptography is used to transmit the message confidentially where no any trouble can happen between transmission. This paper study cryptography, types of encryption and encryption techniques, encryption tools and highlights the importance of encryption in information security.

KEYWORDS

Information Security, Cryptography, Confidentiality, Integrity

1. INTRODUCTION

Cryptography is a vital tool for safeguarding data sent over the Internet. It is the transformation of data into an unreadable format, only the intended recipient will be able to understand and use it. Cryptography, the art and science of secure communication and data protection, has experienced significant advancements in recent years. Strong encryption methods and secure protocols are now essential due to the rapid advancement of technology and the widespread usage of digital platforms.

Sensitive information such as codes, and all kinds of private data are potentially affected by external threats. [1]

Primitives of cryptographic are Symmetric Encryption, Message Authentication, Public Key Cryptography, Digital Signatures, Pseudo-random Number Generators.

Cryptography remains important to protecting data and users, ensuring confidentiality, and preventing cyber criminals from intercepting sensitive corporate information.

Cryptography is a science by which we can design strong encryption by applying complex mathematics. Attaining strong encryption means data hiding by which we can hide the secret data which can't be permitted by any third person for decryption.

The security services given by cryptography are Confidentiality, Data Integrity, Authentication, and Non-repudiation.

Confidentiality

Confidentiality refers to the ethical and legal need to protect sensitive information form unauthorised access or disclosure. Confidentiality pertains to using protective measure to prevent unauthorized individuals form gaining access to sensitive information [2].

Data Integrity

Security service that deals with identifying any alteration of data. The data gets modified by an unauthorized person intestinally or accidentally.

Authentication

It is an establishment of the identity of a user or computer system so that it may be trusted [3].

Non-repudiation

Security service that ensures that an entity cannot refuse ownership for a previous commitment or an action. [4]

3. ENCRYPTION TYPES

Encryption techniques can be categorized into three types. The categorization is done on the basis of the number of keys used to encrypt and decrypt information.

3.1 Secret Key Cryptography is an encryption method that uses one key for both encryption and decryption [6]. Example, advanced encryption standards (AES) and Caesar cipher

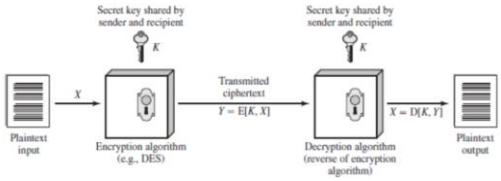


Figure 2. Secret Key Cryptography

3.2 Public Key Cryptography is an encryption method that uses two keys; one for encryption and another for decryption. There are two different applications for public key cryptography: data transmission and digital signatures [6]. Example, Diffie-Hellman and elliptic curve cryptography.

3.3 Hash functions is an encryption method that uses no keys. These encryptions are also called one-way transformations because there is no way to retrieve the message encrypted using a hash function [6]. Example, MD5, SHA1, SHA-3.

4. ORIGIN OF CRYPTOGRAPHY

Cryptography was in use during the era of antiquity and medieval times. During antiquity, cryptography was widely used in Mesopotamia, Egypt and ancient Greeks in various capacities. One of the ancient cryptographic was the non-standard hieroglyphs used during the 1900Bc [7]. The first documented instance of encryption was used by Roman Emperor Julius Caesar (100 Bc), Caesar cipher. In Caesar cipher, mono-alphabetic substitution, replacing individual letters with other letters for the purpose of encryption, is used for encryption.

In the Caesar cipher, each letter was replaced by the letter three places to the right of the letter in the alphabet, Figure 2.

A shift of four characters to the right would give us an encryption scheme such as A → E; B → F [6].

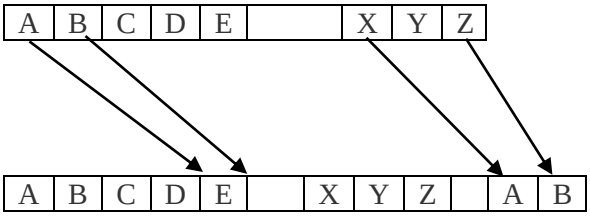


Figure 2. Caesar Cipher

5. CRYPTOGRAPHY TECHNIQUES

Mono Alphabetic Cipher

One of the earliest encryption methods is shift cipher. A cipher is a step method or algorithm, that converts plaintext to ciphertext (encryption) or ciphertext to plaintext (decryption). Caesar's shift cipher is known as mono-alphabetic substitution shift cipher [4].

Poly Alphabetic Cipher

In mono-alphabetic, for a given key the plain alphabet is fixed through out the encryption or decryption process means if 'A' is substututed by 'D'. So, all occurrence of A's in plain alphabet is substituted by D. But in Poly-alphabetic, substituted alphabet in plaintext may be different in places during encryption or decryption process [4].

5.1. Requirements for Good Encryption Technique

A strong encryption algorithm requires a strong encryption key, a strong mathematical algorithm, and a complex encryption process.

Strong encryption keys are passwords for encryption. The longer the password or the more complex the password, the more difficult it will be to guess. However, as binary numbers, encryption keys lack complexity and therefore require length. Most encryption algorithms require a minimum of 128 bits [8].

Strong Mathematical algorithms use the key to feed an algorithm made of simple mathematical processes. Current encryption algorithms use points on an ellipse, multiply large prime numbers, or implement exclusive OR (XOR) logical operations on portions of data as the basis for the algorithm.

A complex encryption process uses a complex combination of the encryption key and the mathematical algorithms on blocks of

data over multiple rounds of encryption. For example, The Blowfish algorithm uses simple XOR functions and performs four actions within each of the 16 rounds of encryption [8].

6. CRYPTOGRAPHY TOOLS

This session describes the overview of some cryptography tools.

6.1 Pretty Good Privacy (PGP)

Pretty Good Privacy (PGP) is a security program used to encrypt and decrypt email and authenticate email messages through digital signatures and file encryption. PGP works through a combination of cryptography, data compression, and hashing techniques. PGP uses the public key system in which every user has a unique encryption key known publicly and a private key that only they know.

Encryption

PGP provides up to date-up to date-end encryption for the wireless communication which prevents the nodes in transit from being able up updated view the records. As an end result, malicious sports along with eavesdropping, packet hijacking or guy-in-the-middle (MITM) assaults are averted.

Authentication

PGP can be used to up-to-date without difficulty establish at ease connection among up-to date which may be used up-to-date authenticate them and make sure handiest the legitimate parties are allowed updated talk in the network.

Key Control

PGP makes use of a robust key management machine up to data updated up-to-date personal keys used for encryption and decryption. This makes it difficult for attackers up to date between the non-public key up-to-date decrypt information at some point of conversation [9].

6.2 GNU Privacy Guard

GNUPG is a tool for secure communication. GNUPG uses public-key cryptography so that users may communicate securely. In a public-key system, each user has a pair of keys consisting of a private key and a public key. GNUPG uses a somewhat more sophisticated scheme in which a user has a primary keypair and then zero more additional subordinate keypairs [10].

6.3 VeraCrypt

VeraCrypt is one of the cryptography tools that is a widely used enterprise-grade system for Linux. macOS, and Windows operating systems. VeraCrypt provides automatic data encryption capabilities and partitions a network depending on specific hashing algorithms, location and volume size [11].

6.4 Bcrypt

The Bcrypt hashing algorithm is a hashing function created from Blowfish by two security researchers, Niels Provos and David Mazières. This hashing function has several advantages, using the original random salt (the salt is the order in which it is appended to the password to make it harder to brute force). Random salts also prevent lookup table creation. Brute force attacks are among the most frequently used when it comes to collecting password data and through a dictionary attack an attacker can match as few characters as possible. The Bcrypt Algorithm is resistant to brute force attacks [12].

7. APPLICATIONS OF ENCRYPTION TECHNIQUES AND ALGORITHMS

Cryptography and encryption methods have been used for centuries to prevent prying eyes from accessing secret messages. Strong encryption has become one of the most crucial cybersecurity practices for supporting modern Internet communications. Encryption algorithms convert original data in plain text to an encrypt message to ensure secure transmission. To safeguard sensitive data from cyber-attacks, several fields use encryption methods and approaches. The followings are some of the most typical uses for encryption:

1. Secure Communications: Email, instant messaging, and online transactions are all secured via encryption over the internet.
2. Cloud security: Data saved on the cloud, particularly private financial and health records, are protected via encryption.
3. Mobile security: Data held on mobile devices, such as smartphones and tablets, are protected via encryption.
4. Industrial Control systems security: Critical infrastructure, such as power plants and water treatment facilities,

are shielded from cyber-attacks via encryption. However, the application of the encryption techniques and algorithms in Table 3 needs to be better understood [13].

8.TESTING PASSWORD ENCRYPTION USING BCRYPT IN PYTHON

To store passwords in plain text is very vulnerable to attacks. It should be encrypted before it is stored or transmitted. This section describes the password hashing using Bcrypt in Python.

To install the Bcrypt tool in Python the following statement is used [14].

pip install bcrypt

Functions of Bcrypt

bcrypt.gensalt()	
bcrypt.hashpw()	
bcrypt.checkpw(password, hash)	

8.1 Password Hashing

In order to use bcrypt, bcrypt module is needed to import.

```
import bcrypt
```

```
import bcrypt
password='thisisasecrect'
bytes=password.encode('utf-8')
salt=bcrypt.gensalt()
hash=bcrypt.hashpw(bytes,salt)
print(hash)
```

The above python code generates the following output, hash of password "thisisasecrect.

```
b'$2b$12$Ec3S0aY83bFiBWntEGRvYeWIMJos6QLzjNTB8PaZ2ln8lG26fml1S'
```

8.2 Passwords Checking using Bcrypt in Python

Bcrypt can be used to check the password which was entered by the user is correct or not. bcrypt.check(password,hash) can be used for this purpose.

```
import bcrypt
password='thisisasecrect'
bytes=password.encode('utf-8')
salt=bcrypt.gensalt()
hash=bcrypt.hashpw(bytes,salt)
userPwd='p@ssw0rd'
userBytes=userPwd.encode('utf-8')
result=bcrypt.checkpw(userBytes,hash)
print(result)
```

The above python code for checking password will give the result "**False**" because the two passwords are not the same.

```
import bcrypt
password='thisisasecrect'
bytes=password.encode('utf-8')
salt=bcrypt.gensalt()
hash=bcrypt.hashpw(bytes,salt)
userPwd='thisisasecrect'
userBytes=userPwd.encode('utf-8')
result=bcrypt.checkpw(userBytes,hash)
print(result)
```

The above python code for checking password will give the result "**True**" because the two passwords are the same.

8.3 Encrypting and Decrypting Messages with Pretty Good Privacy

In order to Encrypt and Decrypt Messages with PGP Python, open-source python library named python-gnupg will be used. This library provides a Python inferface to GnuPG.It is needed to install and import the gnupg package and instantiate the gnupg class. To install and import and instantiate Python gnupg package, the following commands are used.

pip install python-gnupg
import gnupg
gpg=gnupg.GPG( )

Before the key is generated, the settings for the keys must be configured. After that public key is generated.

```
>>> input_data = gpg.gen_key_input(
... name_email = 'zinmar86615@gmail.com',
...  passphrase = 'thisisasecrect',
...  expire_date = '200d')
>>> keys = gpg.gen_key(input_data)
>>> public_keys = gpg.export_keys(keyids = keys.fingerprint)
>>> private_keys = gpg.export_keys(keyids = keys.fingerprint,
...                                secret = True,
...                                passphrase = 'thisisasecrect')
```

The generated (public and private) keys are written to files using follwing commands.

```
>>> with open('public_key.asc', 'w') as f:
...     f.write(public_keys)
...
995
>>> with open('private_key.asc', 'w') as f:
...                     f.write(private_keys)
...
1955
```

Messages can be encrypted using the generated keys.

```
>>> with open('public_key.asc', 'r') as f:
...                     public_key =f.read()
...
>>> public_key = gpg.import_keys(public_key)
>>> data = gpg.encruput('computer', public_key.fingerprints)
>>> print (data)
-----BEGIN PGP MESSAGE-----

hQEMAyD4JWaDYKHTAQf9FQIYNfZp8QepLlh7NCpYp4wzxu/y6aY8MvYHRXAOXDa0
uRAvqQ7Uld1L+ANv4DB7fn37BLTUGcW95rJdrsE6Ck2+6iv13Qljo4iU0zlEZfWL
U9zmJM1sbNiTcMUhSkfqXtjGq7oF4ySCyJlhuNcZuG9Y+WqGVZ/ANnEfB2RnIWq4
TVuLwmjVBmPksYqNWWwIBpSmuolIDZw5NQoYA65eIxBZCJs/iaoJdINXnjnvD302
8eAWRmp7oHX0voPkp1DWeHPF8v8Wdge4ny5Ldna+qqzVk5k8vpU5/ugnzx05HnIx
is6oIDUNvVNJte4+CSTdpuGVyFPBKTeptRqmiicetdRNAQkCEIXbt1RiVEBhvf1G
F/EeprPoRdDL00839WsUxTNL7E/DftGvOUKyRPw0t+Ze+gLzS9Cy8GRPCZff1ZMO
BcuTf7J6+2e3Q+R/4jo=
=q+j7
-----END PGP MESSAGE-----
```

Encrypted Messages can also be decrypted using these keys to get the original text.

```
>>> with open('private_key.asc', 'r') as f:
...         private_key = f.read()
...
>>> private_key = gpg.import_keys(private_key)
>>>
>>> decrypted_data = gpg.decrypt(data.data, passphrase ='thisisasecrect')
```

The following code display the decrypted original text.

```
>>> print(decrypted_data)
computer
>>>
```

9. CONCLUSION

Sensitive information such as passwords, data access codes, and all kinds of private data are potentially affected by external threats. Therefore, cryptographic algorithms are important element in many computer security services and applications. Many organizations use cryptography to secure the important information about its current working project by which no any third party can affect its data. Security mechanisms such as encryption tools and cryptographic mechanisms is important in protecting sensitive data and information in order to protect from unauthorized access and malicious attacks. In this paper, literature review of the cryptography tools and cryptography techniques are studied. Finally, password hashing in Bcrypt is demonstrated with Python.

REFERENCES

[1] M. Victor, et. al, (2023) *Cryptography: Advances in secure Communication ad Data Protection, E3S Web of Conference 399.*

[2] O. M. C. Osazuwa, Confidentiality, Integrity and Availability in Network Systems: *A Review of Related Literature, International Journal of Innovative Science and Research Technology, Volume 8, Issue 12, December 2023.*

[3] R. ma.Al-Amri et.al, Theoretical Background of Cryptography, Mesopotamian Journal of Cybersecurity, Volume 2023, pp 7-15.

[4] R. K. Choubey, A. Hashmi, *Cryptographic Techniques in Information Security, International Journal of Scientific Research in Computer Science, Engineering and Information Technology, Volume 3, Issue 1, 2018.*

[5] S.Sou et. al, *Encryption Technology in Information System Security, Advances in Computer Science Research, Volume 87, 3rd International Conference on Mechatronics Engineering and Information Technology, 2019.*

[6] M. Agrawal et. al, *Information Security and IT Risk Management, John Wiley & Sons Inc., 2014.*

[7] H. Li, Y. Wang, The History of Cryptography and Its Applications, International Journal of Social Science and Education Research, Volume 5 Issue 3, 2022.

[8] https://www.esecurityplanet.com/.

[9] *Sunil. MP, Exploring the Use of Pretty Good Privacy (PGP) in Wireless Network Security, 3rd International Conference on Smart Generation Computing, Communication and Networking, Karnataka, India, 2023.*

[10] The GNU Privacy Handbook

[11]https://www.veracrypt.fr/en/Documentation.html

[12] T.P. Batubara et. al, Analysis Performance BCRYPT Algorithm to Improve Password Security from Brute Force, Journal of Physics: Conference Series, 2020.

[13] S. A. Wadho et. al, *Encryption Techniques and Algorithms to Combat Cybersecurity Attacks: A Review, VAWKUM Transactions on Computer* Sciene, Volume 11, January- June 2023.

[14]https://www.geeksforgeeks.org/hashing-passwords-in-python-with-bcrypt/

Authors

Zin Mar Oo got her B.C.Sc(Hons.) and M.C.Sc degrees in University of Computer Studies (Pinlon) and University of Computer Studies (Meiktila). She worked as a teaching faculty at University of Computer Studies (Meiktila), University of Computer Studies (Monywa), and University of Computer Studies (Myitkyina). Currently, she works as a Lecturer at University of Computer Studies (Mandalay). Her selected publications are followings.

1. *Implementation of Software Agent for Agricultural Fulfilment System*, Proceedings of the second conference on Applied Information and Communication Technology
2. *Travel Package Recommendation System*, Journal of Innovative Research, Volume 1
3. *An Approach of Advice for Peasant Aid System*, Myanmar Universities' Research Conference, 2019
4. *Iris recognition Attendance System*, Scientific Journal of Innovative Research, Volume 1.
5. *M.P.Aung et.al, Design and Implementation of Shopping System Using MVC Software Architecture, Journal of Information Technology, Research and Innovation 2023, Volume 3, Issue 2.*

Theint Theint Htwe got her B.C.Sc (Hons.) and M.C.Sc degrees from University of Computer Studies (Pakokku) and University of Computer Studies, Yangon, respectively. She worked as teaching faculty at University of Computer Studies, Yangon, University of Computer Studies (Pakokku), University of Computer Studies (Mandalay), and University of Computer Studies (Loikaw). Currently, she works as a Lecturer at University of Computer Studies (Mandalay). Her selected publications are followings.

1. *Decision Support System for House Interior Design*, Proceedings of the Conference on Parallel and Soft Computing, 2009.
2. *Car Recommender System Based on Customer's Reviews Using Model Based Filtering, University Journal of Science, Engineering and Research, 2020.*
3. *M.P.Aung et.al, Design and Implementation of Shopping System Using MVC Software Architecture, Journal of Information Technology, Research and Innovation 2023, Volume 3, Issue*

Myintzu Phyo Aung (Ph.D) got her M.C.Sc and Ph.D (IT) degrees in 2008 and 2014 respectively from University of Computer Studies, Mandalay. She has over 18 years of experience in teaching on Information Science and Computer Science subjects. She served as a teaching faculty at University of Computer Studies, Mandalay, University of Computer Studies (Myitkyina) and Myanmar Institute of Information Technology. Currently she served as an Associate Professor at University of Computer Studies (Mandalay). She had published many International Conference Papers and Journals and gave presentations on her research work. The followings are her selected publications at International Conference and Journals.

1. *Construction of Finite State Machine for Myanmar Noun Phrase* (2013), MJIIT-JUC Joint International Symposium, Japan.
2. *Identification and Translation of Myanmar Noun Phrase to English* (2014), International Journal of Computational Linguistics and Natural Language Processing, India.
3. *New Phrase Chunking System for Myanmar Natural Language Processing* (2014), Applied Mechanics and Materials, Malaysia.
4. *Extraction of Myanmar Noun Phrase using Maximum Matching Approach (2015)*, International Workshop on Plasma Application and Hybrid Functionally Materials, USA.
5. *Constructing Myanmar Phrase Structure Grammar for Myanmar Noun Phrase Extraction* (2016), International Conference on Science and Engineering, Myanmar.
6. *Stochastic Context Free Grammar for Statistical Parsing of Myanmar Natural Language Processing (2018)*, International Conference on Computer Applications, Myanmar.
7. *Proposed Framework for Stochastic Parsing of Myanmar Language (2018)*, Springer Journal, Japan.
8. *Phrase Based Translation Model for Myanmar Language* (2019), Proceedings of Joint International Conference on Science, Technology and Innovation, Myanmar.
9. *M.P. Aung et.al, Design and Implementation of Shopping System Using MVC Software Architecture, Journal of Information Technology, Research and Innovation 2023, Volume 3, Issue 2.*

RECOMMENDATION SYSTEM BASED ON APACHE SPARK USING BIG DATA

Naw May Myat Noe[1], and Myintzu Phyo Aung[2]

[1,2]Faculty of Information Science, University of Computer Studies (Mandalay)
[1]mayherit8@gmail.com, [2]myitzu.mm@gmail.com

ABSTRACT

Recommendation systems are crucial in modern electronic commerce, addressing the challenge of finding desired information in the vast expanse of the Internet. These systems, a subset of information filters, analyze user interests and behaviors to intelligently recommend relevant products. By leveraging machine learning algorithms, recommendation systems forecast a user's preference for items based on interactions like ratings, reviews, and historical behavior. This approach personalizes recommendations to align with individual tastes. This system, built on Apache Spark, utilizes the Alternating Least Squares (ALS) matrix factorization method from Spark MLlib to handle recommendation requests efficiently and scale effectively. Specifically, the system applies collaborative filtering on the Amazon Beauty Products dataset, which features explicit user feedback. With a focus on training the model using ALS, this system achieves a Root Mean Square Error (RMSE) of 0.9, demonstrating its effectiveness in generating accurate recommendations.

KEYWORDS

Recommendation, Apache Spark, Alternating Least Square (ALS), Collaborative filtering, Root Mean Square Error (RMSE)

1. INTRODUCTION

In an era of everyone's daily life, internet technology has advanced, and it became an important part of humans. With increased technology between people and devices, the amount of data availability is growing in communication constantly. The mass of information can meet the information needs of Internet users; a serious challenge of processing information has to be handled [1]. Product recommendation systems have grown in popularity, playing a crucial role in e-commerce and social networks by predicting or suggesting products based on customer preferences. These systems use machine learning, particularly collaborative filtering and matrix factorization techniques like Alternating Least Squares (ALS), to enhance recommendation accuracy and manage over-fitting in sparse data. Platforms like Spotify, Netflix, and YouTube exemplify how recommendation systems are applied across various domains, from music and movies to app suggestions. Companies deploy these systems to improve user experiences and drive economic growth. This paper discusses a recommendation system built on Apache Spark using the ALS method, which significantly reduces Root Mean Square Error (RMSE) and improves prediction accuracy. It explores the theoretical background, methodology, system architecture, and experimental results, concluding with insights and future work.

2. RELATED WORK

In recent times to generate product recommendations all differing by small factors have been developed in many algorithms. Recommendation system contains simple algorithms so that it can recommend user with relevant items by collecting user data and discovering data patterns in the data set and filtering user related data. Li Xie, Wenbo Zhou, Yaosen Li proposed Application of Improved Recommendation System Based on Spark Platform in Big Data Analysis. This system used technology research of recommendation systems, and recommendation algorithm based on ALS model. This system proposed firstly, research on the existing collaborative filtering algorithm based on ALS and the big data distributed computing platform of Spark [1]. Ankit Maheshwari*, Anuradha Kumari, Anjali

Kumari, Neeraj Kumar and Nandini B.M discussed Movie Recommendation System Using Apache Spark. Author explains system implementation which provides outputs of movie recommendations at a very good speed because of the Spark's MLLib used [2]. Real-time News Recommendations using Apache Spark. The use of Apache Spark enables to run the recommender as a distributed system in a cluster ensuring the scalability of the approach [3]. Kavitha V K*, Dr.Sankar Murugesan proposed An Effective Book Recommendation System using Weighted Alternating Least Square (WALS) Approach. In this paper, demonstrated a significant reduction in RMSE when compared to standard RS, highlighting its effectiveness in enhancing the quality of book recommendations and a smaller RMSE value indicates that the system is more proficient at predicting user behavior, resulting in a more gratifying and individualized reading experience [4]. M. Chenna Keshava,P. Narendra Reddy, S. Srinivasulu, B. Dinesh Naik discussed Machine Learning Model for Movie Recommendation System. Author explained a privacy-retaining collaborative filtering method for binary facts referred to as a randomized reaction technique. Error rate is fined on five models. In this research, the best model is SVDpp with Test RMSE of 1.0675 [5].

3. RECOMMENDATION SYSTEM

Recommender system is defined as a tool that helps users in search something which is related to their tastes or as a strategy of choice for users under complex information environments. Recommender systems handle the problem of overload of information that users find, by providing them personalized and targeted content. For building this systems, distinct approaches have been developed. They can be collaborative filtering, content-based filtering or hybrid filtering. Type of recommendation system are shown in figure 1. The two most common recommender system techniques are: (1) collaborative filtering, and (2) content-based filtering. Collaborative filtering recommends items by identifying other users with similar tastes. On the other hand, Content-based filtering recommends elements that are similar in content to products that user appreciated in the past or matched to user attributes [6].

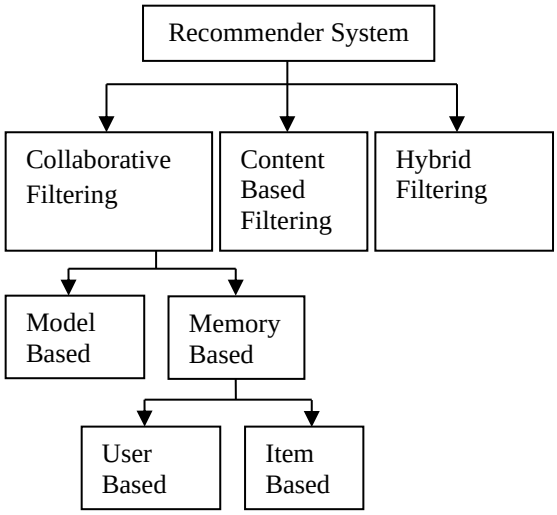


Figure 1. Types of Recommender System

3.1 Collaborative Filtering

Collaborative filtering is based on the users' future preferences. The goal of collaborative filtering is to predict a user's preferences. This is based on the feedback of similar users. These past user-item interactions, build through user-item matrix, are sufficient to find matches and to detect similarity between users and/or items and make predictions according on these estimated distances. To generate recommendation system, collaborative filtering uses a user-item matrix. User-item matrix defines the match data of m users by n items that contains the users' ratings over items. So each entry (i, j) in the matrix represents the interaction between user i and item j in following figure 2. Two items are considered to be similar if most of the users that have interacted with both of them did it in a similar way [6].

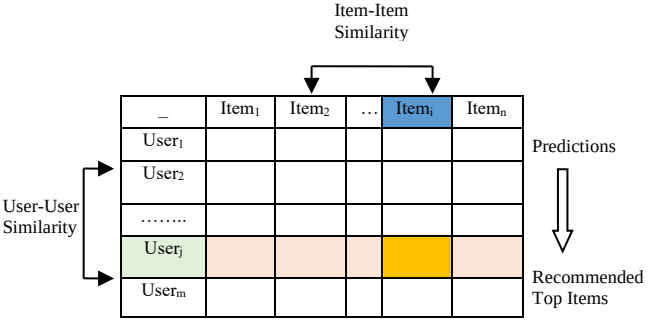


Figure 2. Collaborative Filtering Process

Collaborative filtering algorithm based on ALS is described as below.

$$W = (W_{tj})^{a \times b} \quad (1)$$

Where W_{tj} represents the rating matrix of items b by users a. The goal is to approximate this matrix W using a low rank matrix A such as:

$$A = JK^{\alpha} \quad (2)$$

Where:

$$J \epsilon C^{a \times d} \quad (3)$$

and

$$K \epsilon C^{b \times d} \quad (4)$$

These are the matrices to be learned and d represents the number of latent features in the model, typically, $d \ll s, s \approx \min(a, b)$, s represents the rank of the matrix, and the loss function is as follows:

$$L(J, K) = \sum_{tj}(W_{tj} - A_{tj})^2, \quad (5)$$

Where: $(W_{tj} - A_{ij})^2$ is a common error term in the low rank approximation. To find a way to solve the optimization arg $min_x\ L(x)$, which is to minimize the loss function. Formula can be re-writing as follows:

$$L(J, K) = \sum_{tj}(W_{tj} - J_t - \quad K_j^{\alpha})^2. \quad (6)$$

In order to prevent over-fitting, the second order regularization term is added to the about formula:

$$L(J, K) = \sum_{tj}(W_{tj} - J_t K_j^{\alpha})^2 + \lambda(\|J_t\|^2 + \|K_t\|^2) \quad (7)$$

If K is known, the ridge regression can be predicting each line of J, and vice versa. Therefore, the matrix K is fixed, then the derivative of t is taken; now obtain the following formula forU_t:

$$U_t = W_t K_{ut}^{\alpha}(K_{ut}^{\alpha} K_{ut} + \lambda n_{ut} I)^{-1}, t \in [1, a] \quad (8)$$

Where W_t represent the score vector for the item by user t; t represents the characteristic matrix formed by the characteristic vectors of the item evaluated by the user t. In the same way, the matrix J is fixed; then the derivative of K is taken; and obtain the following formula for K:

$$K_j = W_j^{\alpha} J_{aj}(J_{aj}^{\alpha} J_{aj} + \lambda b_{aj} I)^{-1}, j \in [1, b] \quad (9)$$

Where W_j represents the score vector for the item by user j; J_{aj} represents the characteristic matrix formed by the characteristic vectors of the items evaluated by the user j; b_{aj} represent the number of items evaluated by the user j; I is $d \times d$ unit matrix. Based on the ALS collaborative filtering algorithm is performed, namely the formulae (8) and (9) are called alternately to update J and K. Calculation processing ends if the result of the calculation is convergence or the number of iterations reaches the maximum value. Finally, the approximation matrix X is got to recommend items for users [1].

3.2 Model-based Filtering

Model based method is expressed by matrix for predicting, and the method is insert into Matrix for items not been seen by the user previously. The rating of unrated items can be done by using the data mining techniques to predict. This technique depends upon the basis of the information collected from the repository used to generate a model for recommending without making the use of dataset every time. Speed and scalability have the two factors. That factors are improved when this approach is used the appropriate algorithm. Furthermore, the prediction accuracy vale is increase. Matrix factorization, this technique is most commonly used in model-based technique [6].

3.3 Matrix Factorization

To identify the relationship between items and users' entities, collaborative filtering is the application of matrix factorization. Matrix Factorization is the most successful realization of latent factor models. This method is one of the most common methods for implementing model-based collaborative filtering-based algorithms. Matrix factorization describes a mathematical term. Which describe as linear algebra. The aim of matrix factorization is to approximate the actual ratings matrix S with the user matrix A and item matrix B, the following cost function needs to be minimized [7].

3.4 Apache Spark

Building a recommendation system involves leveraging Apache Spark's distributed architecture and MLlib library for large-scale machine learning. The pipeline begins with data ingestion from sources such as HDFS or cloud storage or local file system, followed by data transformation using Spark SQL and

DataFrame operations for cleaning and preparing the dataset (e.g., user-item interactions). Machine Learning library's Alternating Least Squares (ALS) algorithm is used in PySpark to build collaborative filtering models that can handle large-scale data across clusters. Spark's caching ensure efficient, reliable training, and the trained model can predict user preferences, with performance monitored through the Spark. [8]

3.5 Alternating Least Squares (ALS)

In matrix factorization recommenders, Alternating Least Squares (ALS) is one of the most popular algorithms. Apache Spark ML implemented the ALS algorithm for matrix factorization based collaborative filtering for explicit feedback. It is a two-step procedure of iterative optimization, in which in each iteration, it first fixes the item matrix A and then solves for the user matrix B, then fixes B and then solves for A [7][9]. ALS algorithm describe as below:

Algorithm: Alternating Least Squares (ALS)

Procedure ALS (A_u, B_i)
Initialization $A_u \leftarrow 0$
Initialization matrix B_i with random values
Repeat
    Fix B_i , solve A_u by minimizing's the objective
    Function (the sum of squired errors)
    Fix B_i by minimizing the objective function similarly
Until reaching the maximum iteration
Return A_u, B_i
End Procedure

4. PROPOSED SYSTEM

The recommendation algorithm based on ALS for collaborative filtering recommendation system using Apache Spark is proposed. For distributed big data processing, Apache Spark is a integrate analysis framework. Pyspark has been widely used to the popularity of big data in recent years. In Java, Scala, Python, and R is provided the high-level APIs. Apache spark can handle both batches as well as real-time analytics and data processing workloads [9]. The proposed system is Beauty Product recommender system based on ALS using Apache Spark. Proposed system is described as below:

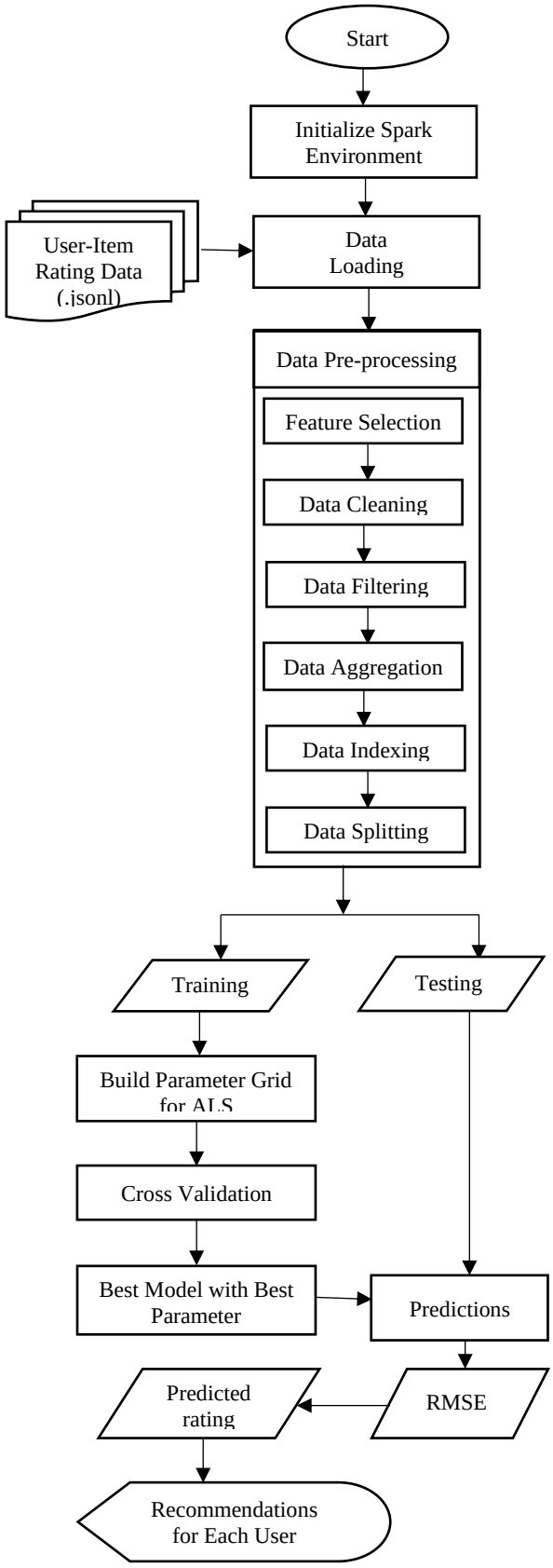


Figure 3. Proposed system

4.1 Dataset Set

The system use a large-scale Amazon Reviews dataset collected in 2023 [10]. Within this dataset, the system specifically utilizes the "Beauty Product.csv" file. The dataset includes user ID, product ID, product type, ratings, and timestamps and URL. This file consists of 701,528 rows and has a size of 311MB. The dataset uses the following data types.

Table 1. Root of Dataset

Field Name	Data Type	Descriptions
user_id	String	Unique IDs users
product_id	String	Unique IDs of products
rating	Integer	User ratings of the products that range from 1 to 5
Timestamp	Integer	Time of the review (UNIX format)
title	String	Title of the user review.
text	String	Text body of the user review.
images	list	Images that users post after they have received the product.
asin	String	ID of the product.
verified_purchase	bool	User purchase verification
helpful_vote	Integer	Helpful votes of the review

4.2 Creating the DataFrame

The beauty product recommendation system will run on Apache Spark. First, set up the Apache Spark framework and then load the beauty product reviews dataset from a JSONL file into a Spark DataFrame by using the spark.read.json() method to load a dataset into the Spark session. If the dataset has column names, header have to be given true. On the other hand, to automatically determine the data types of the columns, use the infer Schema argument.

4.3 Data Pre-processing

For analyzing Amazon product reviews using Spark, several pre-processing steps are assumed to prepare the data. In pre-processing step, firstly, select a subset of columns from the dataframe to focus on most relevant data which can improve the efficiency of subsequent processing steps. Relevant features such as user_id, product_id and rating have to be selected.

Table 2. Beauty Product Rating Dataframe

user_id	product_id	rating
AGKHLEW2SOWHNMFQIJGBECAF7INQ	B00YQ6X8EO	5
AE74DYR3QUGVPZJ3P7RFWBGIX7XQ	B097R46CSY	5
AGMJ3EMDVL6OWBJF7CA5RGJLXN5A	B00R8DXL44	4
AEYORY2AVPMCPDV57CE337YU5LXA	B08BBQ29N5	3

After that, clean the data in rating column. When there is missing values in rating column, the row has to be removed. Next, filter the rows with invalid rating and keep only ratings with the range of 1 to 5. Then, aggregate data to calculate the popularity of each product. All the records with the same row will be combined together. The products are then ranked based on their popularity to identify the most popular products in the dataset. Table 3 illustrates the formation of Popularity Data Frame.

Table 3. Popularity Data Frame

Product_id	**Popularity**
B085BB7B1M	1962
B0BM4GX6TT	1750
B07C533XCW	1513
B09X9BG4FC	1374
B00R1TAN7I	1372

Moreover, combine this Popularity DataFrame with the original DataFrame based on the product_id to filter for the top popular items. The filtered DataFrame after merging the data is shown in Table 4.

Table 4. Filtered Dataframe

Product id	user id	rating	description	Popularity
B085BB7B1M	AF4NIQPIZQ3S3…	3	Salux Nylon Japanese Beauty Skin Bath Wash Cloth/towel (3) Blue Yellow and Pink	1962
	AG6UDZJUG4I…	5		
	AEXTSZOMHU…	4		
	AHBDCC6PPJ6…	4		
	AHW2IGENKQ…	4		

Machine learning algorithms, including collaborative filtering techniques like Alternating Least Squares (ALS), require numerical input. Before starting the modelling process, convert all columns that will be used in the model to integer. String Indexer converts categorical string data (such as user_id and product_id) into numerical indices, making the data suitable for these algorithms. This requires numerical input, improving computational efficiency, ensuring consistent and reproducible data processing, and facilitating compatibility with machine learning libraries like Spark MLlib. So in this system, transformations of data applied using StringIndexer to convert userId and productId into indexed values, which are then converted to integer. Next, the number of items rated by each user is computed, and users who have reviewed fewer than 2 items are filtered out. To get higher accuracy, the filtered records are used.

Table 5. Filtered Dataframe

User ID	Product ID	Rated_Item
AE22PHL74EPEDWUU52ZLLDZE57EA	B07RB82J1D	2
	B0BBYDRKXP	
A02155413BVL8D0G7	B0089JVEPO	5
	B006ZA0A5Y	
	B005Z41P28	
	B0055MYJ0U	
	B00117CH5M	

Preparing for masking items is a critical step in developing a reliable and robust recommendation system. The number of items each user has rated is calculated and stored in the rated_item column. Only users with more than two ratings are considered. The highest number of items rated by a single user is 165. Next, 10% of the items rated by each user are selected to be masked for testing. This is done by multiplying the number of items rated by 0.1, and the result is stored in the mask_value column. Each item rated by a user is assigned a unique number, starting from 1, and this is stored in the item_number column.

Lastly, the data is split into training and testing sets based on the item ranks. The pre-processing steps in above, the system provide for data cleaning, correctly formatted, and appropriately filtered and transformed, making it suitable for training the Alternating Least Squares (ALS) model.

4.4. Alternating Least Squares Method (ALS)

In the implementation of recommender systems based on Collaborative Filtering, Spark has implemented ALS. This algorithm widely used and it approaches the recommendation problem as a factorization matrix, where the ratings given by users to different items (beauty products, in this case) are represented as a sparse matrix. ALS define three hyper parameters for model turning such as rank, maximum iteration and regularization parameter. Rank is defined the number of latent factors (also called dimensions) of the ALS. Maximum iterations define the maximum number of iterations that ALS can run during training and regularization parameter specifies the regularization term that controls the strength of the penalty to avoid over fitting [11].

Table 6. Parameter Grid of ALS

Parameter Grid	Range	Descriptions
Rank	[10,20,30]	Latent Factors
MaxIter	[10,15,20]	Maximum Iteration
RegParam	[0.1,0.5,0.15]	Regularization Parameters

4.5 Root Mean Squared Error (RMSE)

Regression model is used in machine learning to predict continuous outcomes. In determining the performance of regression models, several evaluation metrics are used. These metrics include Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), R-squared (R^2), Mean Percentage Error (MPE), and Mean Absolute Percentage Error (MAPE). A regression evaluator is created to evaluate the performance of the ALS model. RMSE is a more suitable metric for comparing the accuracy of different linear regression models [12]. RMSE calculated the sum of the squared differences between the predicted and original values, then divided by the number of observations and the square root of the result is taken to RMSE. The RMSE can be computed using the following formula:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=0}^{n} (a_{ri} - b_{ri})^2} \qquad (10)$$

Where, n is the number of user-item pairs in data, a_{ri} is the actual rating for user r and item i and b_{ri} is the predicted rating for user r and item i.

User	Item	Actual Rating	Predicted Rating
0	3971	5	4.2
1	97	5	4.5
2	868	4	4.1
3	3888	5	4.2
4	1025	5	4.5

First, calculate the square error for each row.

For User0, Item3971= $(5-4.2)^2$= 0.64
For User1, Item97 = $(5-4.5)^2$= 0.25
For User2, Item868 = $(4-4.1)^2$= 0.01
For User3, Item3888= $(5-4.2)^2$= 0.64
For User4, Item1025= $(5-4.5)^2$= 0.25

Calculate the Mean Squared Error (MSE)
$MSE = \frac{0.64+0.25+0.01+0.64+0.25}{5} = 0.358$

Calculate the RMSE

$$RMSE = \sqrt{0.358} \approx 0.598$$

For this system, using root mean square error, the above fit gives us an RMSE of 1.9. This is done by using cross-validation through PySpark.

5. EXPERIMENTAL RESULT

In the proposed system, the recommender system is trained using collaborative filtering with the Alternating Least Squares (ALS) algorithm. The performance of the beauty product recommendation is evaluated using the Root Mean Squared Error (RMSE) metric. The objective for recommendations is to minimize the RMSE value. The system is implemented on Apache Spark, and to load datasets in Python, various libraries such as pandas and PySpark used, depending on the format and size of the data [8][13]. In order to load a JSONL file into a Spark Data Frame, create a Spark Session first. Then, use Spark Session builder to initially define the application name in session. After the session is created, read the JSON file by calling spark.read.json() with the path to the file. To read CSV files, set the header to be true to indicate that the first row contains the column names. Dataset is loaded into Apache Spark Framework, following step built the proposed system.

Collaborative filtering in the context of Alternating Least Squares (ALS) is a technique used to build recommendation systems. It operates by leveraging user-item interactions to predict missing ratings or preferences. To use ALS, DataFrame is split into two separate DataFrames. Training data and testing data are depended on the condition involving the columns item_rank and num_items_to_mask. Because it is a common practice in recommendation systems to evaluate the performance of the model.

For model training data, DataFrame includes rows where the value of item_rank is greater than the value of num_items_to_mask. For model evaluation or testing data, DataFrame includes rows where the value of item_rank is less than or equal to the value of num_items_to_mask. The parameter grid is built to explore different combinations of rank,

maxIter, and regParam to find the best hyperparameters for the model. To train the best model, Regression Evaluator and Cross Validator are used. After the run the coding, the best model is generated as:

Table 7. Best Parameter Grid of ALS

Parameter Grid	Best Parameter
Rank	[10]
MaxIter	[20]
RegParam	[0.1]

Moreover, to perform cross validation, the Cross Validator object is created. It uses the parameter grid and evaluator to evaluate model performance. Data are divided into 5 folds: 4 for training data and 1 for data validation.

The evaluator will use the metric Root Mean Squared Error (RMSE) to calculate the error between the actual evaluations and the model predictions. RMSE calculated the sum of the squared differences between the predicted and original values, then divided by the number of observations and the square root of the result is taken to RMSE.

For this calculation, the best model has indicated as below and that the baseline performed better. The best model for ALS training data is rank 1, maximum iteration is 20 and regularization parameter is 0.015. By using root mean square error (RMSE), the above fit gives an RMSE of 0.92. This can be done by using cross-validation through PySpark.

6. CONCLUSION

The Apache Spark framework, which is quick and effective for distributed processing and it is supporting the many languages such as Scala, Python, and Java. In recommendation systems, the collaborative filtering recommendation algorithm based on ALS is one of most common algorithms using matrix factorization technique. It combines a lot of ratings data. In this study, first conducted a review of existing collaborative filtering algorithms based on the Alternating Least Squares (ALS) method and the Spark framework. The experimental results derived from this research were implemented on Apache Spark. The performance of the system demonstrated an improvement, attributable to the ALS-based algorithm, achieving a Root Mean Squared Error (RMSE) value of 1.9, with the optimal model exhibiting a rank of 1. The experimental outcomes indicate a favourable result, as the system's accuracy was evaluated using Spark's MLlib. In future work, the recommendation system will be trained with users click stream predictive analytics, deep collaborative filtering, and convolutional neural network-based recommendation system.

ACKNOWLEDGEMENTS

I would like to thank everyone who supported and helped for this system.

REFERENCES

[1]. Li Xie, Wenbo Zhou, Yaosen Li (2016) "Application of Improved Recommendation System Based on Spark Platform in Big Data Analysis" Print ISSN: 1311-9702; Online ISSN: 1314-4081; DOI: 10.1515/cait-2016-0092.

[2]. Ankit Maheshwari*, Anuradha Kumari, Anjali Kumari, Neeraj Kumar and Nandini B.M, (2018) "Movie Recommendation System using Apache Spark".

[3]. Jaschar Domann, Jens Meiners, Lea Helmers, and Andreas Lommatzsch, (2013)"Real-time News Recommendations using Apache Spark". http://www.aot.tu-berlin.de.

[4] Kavitha V K*, Dr.Sankar Murugesan, (2024) "An Effective Book Recommendation System using Weighted Alternating Least Square (WALS) Approach". International Journal of Advanced Computer Science and Applications (IJACSA).

[5] M. Chenna Keshava,P. Narendra Reddy, S. Srinivasulu, B. Dinesh Naik, (2020) "Machine Learning Model for Movie Recommendation System".International Journal of Engineering Research & Technology (IJERT) http://www.ijert.org ISSN: 2278-0181.

[6] F.Isinkaye, Y. Folajimi, and B. Ojokoh, (2015) "Recommendation systems Principles methods and evolution", Egyptian Informatics Journal, vol. 16, no. 3, pp. 261–273.

[7] Elsa Rachel Dementieva, Z K A Baizal*, Donni Richasdy, (2022) "Food and Beverage Recommendation in EatAja Application Using the Alternating Least Square Method Recommender System", Jurnal Media Informatika Budidarma, Volume 6, Nomor 4.

[8] Xiangrui Meng, Joseph Bradley, Burak Yavuz, (2016) "MLlib: Machine Learning in Apache Spark", Journal of Machine Learning Research 17.

[9] Subasish Gosh1, Nazmun Nahar, Mohammad Abdul Wahab3, Munmun Biswas, Mohammad Shahadat, Karl Andersson, (2021) “Recommendation System for E-commerce using Alternating Least Squares (ALS) on Apache Spark”.

[10]https://cseweb.ucsd.edu/~jmcauley/datasets.html#amazon_reviews

[11] https://www.databricks.com/glossary/what-is-apache-spark

[12]https://medium.com/analytics-vidhya/mae-mse-rmse-coefficient-of-determination-adjusted-r-squared-which-metric-is-better-cd0326a5697e

[13]https://medium.com/tech-blogs-by-nest-digital/introduction-to-pyspark-d89b92264484

Authors

Biography

First A. Daw Naw May Myat Noe was graduated B.C.Sc(Q) in 2019. Now, she prepares for (M.C.Sc-Thesis) and works at Faculty of Information Science in University of Computer Studies (Mandalay). She is interested in Big Data majored in Data Science.

Second B. Dr. Myintzu Phyo Aung

.

RICE LEAF DISEASE CLASSIFICATION WITH SVM CLASSIFIER

Than Than Htwe[1]

[1]Faculty of Information Science, University of Computer Studies (Mandalay)
[1]thanthanhtwe421@gmail.com

ABSTRACT

*Rice, being a staple food in many Southeast Asian countries, is vulnerable to various diseases that significantly affect yield. In Myanmar, where rice is a primary food source, the increasing incidence of insect attacks and foliar diseases threatens food security. This paper employs Convolutional Neural Networks (CNNs) for feature extraction and Support Vector Machines (SVMs) for classification. Utilizing two distinct datasets, the Plant Village Dataset and Rice Leaf Disease Images. The system is composed of four main phases namely image acquisition, pre-processing, feature extraction and classification. In image acquisition, benchmark dataset is used to evaluate the system. In pre-processing phase, input image is resized into 250*150 dimension and then median filter is applied. In feature extraction phase, convolutional neural network is used. Support Vector Machine (SVM) classifier is finally used for classifier. After evaluation with benchmark dataset, acceptable classification 98% result is achieved.*

Keywords

Rice leaf, SVM Classifier, Convolutional Neural Network

1. INTRODUCTION

Agriculture plays a pivotal role in the socio-economic development of nations, particularly in regions like Southeast Asia. In Myanmar, where rice serves as a primary food source, ensuring optimal rice production is imperative to meet the demands of a growing population. However, the agricultural sector faces numerous challenges, among which diseases significantly contribute to yield reduction. Addressing rice diseases is paramount to safeguarding food security and sustaining livelihoods.

Recent advancements in agricultural research emphasize the development of disease-resistant seed varieties using genetic technologies. Additionally, the integration of information technology, including remote sensing and image processing techniques, has revolutionized crop management practices. These technologies enable the early detection and management of diseases, thereby minimizing yield losses.

This paper focuses on leveraging image processing techniques to detect foliar diseases affecting rice plants. Specifically, it employs Convolutional Neural Networks (CNNs) for feature extraction and Support Vector Machines (SVMs) for disease classification. By analyzing two distinct datasets, the paper aims to provide insights into effective strategies for detecting and categorizing rice leaf diseases, thereby contributing to enhanced agricultural practices and food security.

In this system, Firstly, input image is resized into 250 x 150 specific format without losing the original information. The input image contains a lot of noise, it has missing values. Data pre-processing [11, 13, 14] involves removing noise, missing data, and organizing the data into an appropriate format to increase accuracy. It improves data quality. This step involves data cleaning. Data cleaning removes the noise contained in the data using median filter. And then CNN model [2, 16] is used to extract features from pre-processed data. After extracting the features, SVM model is applied to classify the types of diseases such as Bacterial blight, Blast, Brown spot and Tungro. Finally, the system measure the performance by using sensitivity, specificity, precision and accuracy.

2. LITERATURE REVIEW

Qi, H., [1] discussed the automatic identification of coconut foliar diseases using stack clustering technique. The proposed research was conducted on diseased leaves to identify four diseases of coconut leaves, in this

study, they combined deep learning models with conventional machine learning methods. Deep networks, such as ResNet50 and DenseNet121, performed the best in terms of data prediction. The highest accuracy of data augmentation is 97.59 percent. ResNet50 had the best signal performance when combined with the LR model.

The focus of [2] is to reduce the use of pesticides in agriculture while improving the quality and quantity of production. For feature extraction, they use image processing techniques, and for classification, they use SVM. To improve performance and provide better results, the model was combined with data augmentation. This paper aims to detect three main diseases of rice plants: bacterial leaf blight, brown spot, and leaf blight. The input to this model is the complete image to be processed, and the output is the disease that affected the plant. Suresha et al., [3] 2017, applied global threshing methods to extract the diseased part of rice leaves. In addition, the authors used a k-nearest neighbour (k-NN) classifier to identify rice blast and brown spot.

Khaing and Chit Su [6] proposed an automated system to classify four categories of rice-borne diseases such as leaf line, bacterial leaf blight (BLB), brown leaf spot, and fungal RLB diseases and bacterial. This process included preprocessing, image acquisition, journal review, classification, and compression of features. PCA (Principal Component Analysis), GLCM, and Color Grid-based moment were used to consolidate image color, statistics, and also structural features such as standard deviation, mean, variance, intensity, entropy, and correlation. Finally, it was classified according to diseases using SVM. This method achieved a performance of 72.70% in the original gray scale conversion and 90% in the modified gray scale conversion.

Eusebio L. Mique, Jr [7] evaluated an application that helped farmers detect rice diseases and also pests using image processing and CNN. It assessed pest control, farmers' knowledge of heterogeneous rice pests and diseases, the range of pests attacking rice fields, and provided information on how to effectively control and control them. Using image processing and CNN, an application was proposed to detect rice pests and even diseases. The main goal of this model is to achieve a training accuracy of 90.90%. A small cross entropy value means that the trained design could make an earlier prediction or classify the images with less error. With this advanced app, farmers can easily manage and manage rice pests.

Chia-Lin Chung et al [8] proposed a framework for the detection of “dumb crop” or Kanae disease in rice. Disease occurred most frequently when contaminated seeds were used. When the contaminated seeds were used, a pathogen called "Fusarium fujikuroi" began to spread over the rice fields. Therefore, screening of infected plants (rice) should be done at the actual stage of development. In this paper, machine vision is used to differentiate between healthy and infected seedlings at 3 weeks of age. Seedling incubator cultivation was carried out for three weeks. Flat scanners are used to obtain images of healthy and infected seedlings to measure their morphological and color characteristics. The SVM classifiers used distinguish between healthy and infected seedlings. SVM classifier uses GA for selection This function results in 87.9% accuracy and 91.8% positive prediction to differentiate between healthy and infected seedlings.

3. DATASET SPECIFICATION

In the system, two datasets [31, 32] are used to evaluate the classification results. They are “Rice Leafs Dataset” and “Paddy Doctor Dataset”. The detailed specifications of the dataset are described in the following table1. The model of diseases in Rice Leafs Dataset and Paddy Doctor Dataset are shown in figure1 and figure 2.

Table 1. Details of Rice Leaf Disease The Plant Village Dataset

Leaf Diseases	Label Name	Number of original images
Bacterial Blight	Label1	1584
Blast	Label2	1440
Brown Spot	Label3	1600
Tungro	Label4	1308
Total		**5932**

A. The Plant Village Dataset

The dataset contains 5932 images of diseased rice leaves, which include Bacterial Blight, Blast, Brown Spot, and Tungro species. Jpg format type is used in this dataset.

Figure 1. Simple Image for Each Label

B. Rice Leaf Disease Images Dataset

The dataset contains 10407 leaf images across ten classes namely nine diseases and healthy leaves. The detail diseases are shown as the following table. Jpg format type is used in this dataset.

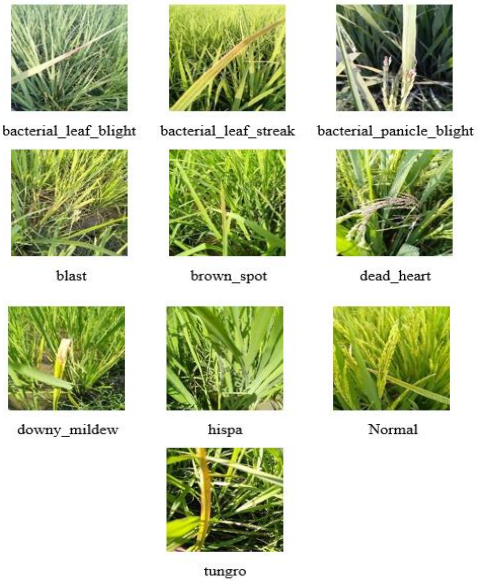


Figure 2. Simple Image for Each Label

Table 2. Details of Rice Leaf Disease Rice Leaf Disease Images Dataset

Leaf Diseases	Label Name	Number of samples
		Number of original images
Bacterial_leaf_blight	Label1	479
Bacterial_leaf_streak	Label2	380
Bacterial_panicle_blight	Label3	337
Blast	Label4	1738
Brown_spot	Label5	965
Dead_heart	Label6	1442
Downy_mildew	Label7	620
Hispa	Label8	1594
Tungro	Label9	1088
Normal	Label10	1764
Total		**10407**

4. IMAGE PROCESSING TASKS

A. Pre-processing Step

Firstly, input image is resized into 300 x 300 specific format without losing the original information. The input image contains a lot of noise, it has missing values. Data pre-processing involves removing noise, an appropriate format to increase accuracy. It improves data quality. This step involves data cleaning. Data cleaning removes the noise contained in the data using median filter.

Median filtering is a non-linear technique to remove noise from images. It is widely used because it is very effective in removing noise while preserving edges. The median filter works by going through the image pixel by pixel, replacing each value with the median value of the adjacent pixels. The neighbour model is called a "window", which slides, pixel by pixel, over the entire image. The median is calculated by first selecting all pixel values from the window in numerical order, and then replacing the pixel under consideration with the median (median) pixel value. The simple calculation of median filter is shown as the following figure.

Figure 3. Media Filter

B. Feature Extraction

Convolutional Neural Network (CNN): A convolutional neural network (CNN) [9] is an artificial network (CNN) that is often designed to extract features from data. CNN is specifically designed to reconstruct two-dimensional shapes with flexibility for translation, scaling, and other transformations. The calculation includes feature extraction. A CNN consists of several levels of convolution and subsampling followed by highly correlated output levels. It mainly consists of three types of layers, namely convolutional, pooling, and fully-connected layers.

Each image is divided into equally sized areas called blocks or patches. Convolutional layer of the network consists of several maps. Neurons in one map are forced to share the same weight. To obtain the activation value for each hidden unit, the weight connecting the path in the map must be multiplied by the input vector. This process is called Convolution.

Pooling basically works in two ways namely max pooling and average pooling. In both methods, a window of predetermined size is selected. In the Max pooling method, the largest activation value from the window values is taken into account. The average pooling value takes into account the average of the activation value of the window. In this study, max pooling concept is applied.

A fully connected layer resembles the way neurons are arranged in a traditional neural network. Therefore, every node in a fully connected layer is directly connected to every node in both the first and the next layer. After passing fully connected layer, features are achieved to classify the image.

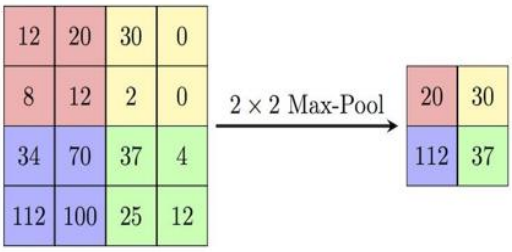


Figure 4. Process of Max Pooling

C. Pretrained Neural Network

Simply put, neural network pre-training [12] means pre-training the model for a single task or data set. Then use the parameters or model from this training to train another model on a different task or data set. This is an example of a head start from the bottom up. In this paper, resnet50 pre-trained neural network is applied.

	Label1	Label2	Label3	Label4	Total
Label1	158	1	0	0	159
Label2	0	143	0	0	143
Label3	0	0	160	0	160
Label4	0	0	0	131	131
Total	158	144	160	131	593

Figure 5. Confusion Matrix of the Plant Village Dataset

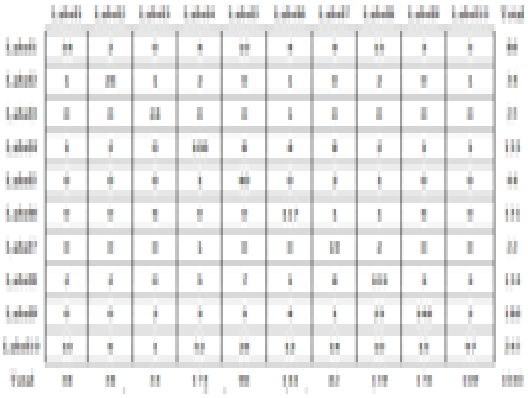


Figure 6. Confusion Matrix of Rice Leaf Disease Images Dataset

5. CLASSIFICATION TASKS

A. Support Vector Machines (SVMs)

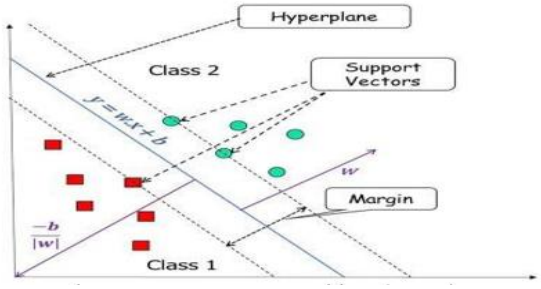


Figure 7. Support Vector Machines

SVM is a class of learning algorithms widely used for complex data classification tasks such as image classification and protein structure analysis. SVM is used in countless fields of science and industry, including biotechnology, medicine, chemistry, and computer science. It has also proved ideally suited for classifying large textual repositories, such as those found in almost all large organizations. The graphical representation of SVM is shown as the following figure.

6. PERFORMANCE

A. Confusion Matrix (CM)

The confusion matrix is an N x N matrix used to evaluate the performance of the classification model, where N is the number of target clusters. Result of confusion matrix is described as the following:

Label Number	**1**	**2**	**3**	**4**
1	3	0	5	26
2	1	2	13	18
3	0	0	34	0

4	1	0	17	16

B. Sensitivity (True Positive Rate)

Sensitivity [5] measures the number of correct occurrences found divided by all correct occurrences. The equation is:

$$Sensitivity = \frac{TP}{TP + FN} \quad (1)$$

The results for each dataset are described as the followingtable.

Table 3. Sensitivity for the plant village dataset

	Label1	**Label2**	**Label3**	**Label4**
TP Rate	99.4%	100.0%	100.0%	100.0%
FN Rate	0.60%	0.00%	0.00%	0.00%
Total	100.0%	100.0%	100.0%	100.0%

Table 4. Sensitivity value for rice leaf disease images dataset

Label	**TP Rate**	**FP Rate**	**Total**
label1	39.5%	60.5%	100%
label2	76.5%	23.5%	100%
label3	96.0%	4.0%	100%
label4	79.7%	20.3%	100%
label5	90.9%	9.1%	100%
label6	89.3%	10.7%	100%
label7	86.4%	13.6%	100%
label8	76.9%	23.1%	100%
label9	79.6%	20.4%	100%
label10	39.6%	60.4%	100%

C. Specificity (True Negative Rate)

Specificity measure the proportion of negative that are correctly identified as negative. The equations is:

$$Specificity = \frac{TN}{TN + FP} \quad (2)$$

Table 5. Specificity value for the plant village dataset

	Label1	**Label2**	**Label3**	**Label4**
TN Rate	100.0%	99.8%	100.0%	100.0%
FP Rate	0.0%	0.2%	0.0%	0.0%
Total	100.0%	100.0%	100.0%	100.0%

Table 6. Specificity value for rice leaf disease images dataset

Label	**TN Rate**	**FP Rate**	**Total**
label1	85.5%	1.5%	100%
label2	98.8%	1.2%	100%
label3	99.0%	1.0%	100%
label4	92.5%	7.5%	100%
label5	94.4%	5.6%	100%
label6	97.0%	3.0%	100%
label7	95.8%	4.2%	100%
label8	93.8%	6.2%	100%
label9	96.7%	3.3%	100%
label10	98.5%	1.5%	100%

D. Precision

Precision [5] the number of true positives divided bythe number of true positives plus the number of false positives. The equation is

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

Table 7. Precision value for the plant village dataset

	Label1	Label2	Label3	Label4
Precision	100.0%	99.3%	100.0%	100.0%

Table 8. Precision value for rice leaf disease images dataset

Precision	
label1	70.8%
label2	68.4%
label3	70.6%
label4	60.9%
label5	41.7%
label6	81.3%
label7	30.6%
label8	64.8%
label9	84.1%
label10	89.0%

E. Accuracy

In the classification task, classification accuracy is the number of true positives (TP) divided by the total number of elements belonging to the positive class. The equation is described as the following.

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \quad (4)$$

Accuracies for dataset 1 and dataset 2 are 99.80% and 68.70%.

7. CONCLUSION

In this system, image classification with multilevel SVM is used to develop an automated and accessible system. The most important diseases in rice leaves, such as late blight and early blight, are detected with minimal computational effort. CNN model using Transfer learning was designed to classify leaf diseases. A comparative analysis was performed to compare CNN with traditional classifiers in terms of accuracy, precision, and recall. The results show that this model is effective for disease detection and classification. The early stage of this system can decide the accurate disease of rice leaf. Therefore, if farmers use appropriate or proper pesticides for disease, they can take precaution and prevent the diseases and increase their product.

Paddy leaf disease classification is presented in this paper with two datasets. Mostly used feature extraction method namely neural network is applied. Neural network is one type of popular methods because it can detect distinct features than another method. But after analysing two datasets with same methods, it can find that dataset is also important portion to get better classification results. In the Plant Village Dataset, images are clearly and captured by mostly defected part. But in Rice Leaf Disease Images Datasets are mixed with background portion.

8. REFERENCES

[1] Qi, H., Liang, Y., Ding, Q., & Zou, J. (2021). Automatic identification of peanut-leaf diseases based on stack ensemble. Applied Sciences, 11(4), 1950.

[2] R Salini, A.J Farzana, B Yamini, (2021). Pesticide Suggestion and Crop Disease classification using Machine Learning , 63(5), 9015- 9023.

[3] Suresha, M., Shreekanth, K. N. & Thirumalesh, B. V. (2017). Recognition of diseases in paddy leaves using KNN classifier. 2nd International Conference for Convergence in Technology (I2CT), pp. 663-666.

[4] Sethy, P. K., Barpanda, N. K., Rath, A. K., & Behera, S. K. (2020). Deep feature based rice leaf disease identification using support vector machine. Computers and Electronics in Agriculture, 175, 105527. doi:10.1016/j.compag.2020.105527.

[5] Davis, Jesse, and Mark Goadrich. "The relationship between Precision- Recall and ROC curves." Proceedings of the 23rd international conference on Machine learning. 2006.

[6] Khaing War Htun and Chit Su Htwe, “Development of Paddy Diseased Leaf Classification System Using Modified Color Conversion”, International Journal of Software & Hardware Research in Engineering, vol. 6, no. 8, 2018.

[7] Mique Jr, Eusebio L and Thelma D Palaoag, “Rice Pest and Disease Detection Using Convolutional Neural Network”, International Conference on Information Science and System, pp. 147-151, 2018.

[8] Chia-Lin Chung, Kai-Jyun Huang, Szu-Yu Chen, Ming-Hsing Lai, Yu- Chia Chen and Yan-Fu Kuo, “Detecting Bakanae disease in rice seedlings by machine vision”, Computers and Electronics in Agriculture, vol. 121, pp. 404-411, 2016.

[9] Li, Zewen, et al. "A survey of convolutional neural networks: analysis, applications, and prospects." IEEE transactions on neural networks and learning systems (2021).

[10] https://data.mendeley.com/datasets/fwcj7stb8r/1

[11] https://www.kaggle.com/competitions/ paddy-disease-classification/data

[12] https://www.mathworks.com/help/deep learning/ug/pretrained-convolutional-neural-networks.html

WEATHER MNITORING SYSTEM BASED ON MACHINE LEARNING WITH RASPBERRY PI

Ngu War Tun [1], and Atar Mon [2]

[1,2] Faculty of Electronic Engineering, University of Technology (Yatanarpon Cyber City)

[1]nguwartun495@gmail.com,[2]atarmon@gmail.com

ABSTRACT

This paper presents a comprehensive weather monitoring system utilizing machine learning for accurate weather prediction. All of our daily activities are greatly impacted by the weather. Compiling data on various weather boundaries is important for both indoor and outdoor landscaping. The Raspberry Pi worker itself reviews this data, which comes from the information sensor, and stores it in text documents and CSV configurations. The system employs a Raspberry Pi integrated with DHT22 and BMP180 sensors to collect data on temperature, humidity, and atmospheric pressure. Collected data is stored in MongoDB and will be used to predict weather conditions using decision tree algorithms. Weather conditions are rainy, sunny, cloudy. It gets highest temperature 30.8℃ and the lowest temperature 30.6℃ within 10 minutes. The lowest humidity is 49%rh and the highest humidity is 71%rh. The highest pressure is 2641.37 Mb and the lowest pressure is 2640.97 Mb in each minute. This system aims to provide real-time weather updates and forecasts for various applications, including agriculture, and disaster management smart city implementations using machine learning.

KEYWORDS

MongoDB, Raspberry Pi, Weather Monitoring, Decision Tree

1. INTRODUCTION

Monitoring weather's condition plays an extensive role in every person's life. The condition of the environment has a significant impact on numerous other sectors, including agriculture, industry, construction, and more. However, the majority of the evaluated impact can be observed in industry and agriculture.

Weather monitoring systems are indispensable tools for tracking environmental changes and providing accurate weather forecasts. Traditional weather monitoring systems often involve high costs and limited customization options. The advent of the Internet of Things (IoT) has enabled the development of more accessible, cost-effective, and scalable solutions. This paper details the design, implementation, and future enhancements of an IoT-based weather monitoring system that leverages machine learning algorithms for improved predictive accuracy. The system is designed to be versatile, catering to various applications such as agricultural planning, disaster management, and urban planning.

Weather forecasting is an application of science and technology which has been growing as many important factors depending on the weather forecast.

2. SYSTEM DESIGN AND ARCHITECTURE

Figure 1 demonstrates the block-level diagram of the system. In the proposed system, an application has been developed on Raspberry Pi 4 and Arduino Uno. The real-time data is received from humidity-temperature and pressure sensors to predict the possibility of rain on the present day. The output of the Raspberry Pi is connected to the input of the Arduino Microcontroller. The data of Digital Humidity and Temperature (DHT 22) is processed in Raspberry Pi. The pressure values sensed by the Barometric Pressure Sensor (BMP 180) are sent to Arduino Microcontroller. The UV sensor sends the voltage and current to Arduino Microcontroller. Wind speed and direction sensor gives the wind speed and direction values to Arduino Microcontroller. Based on

machine learning techniques, the system architecture explains the general composition and constituent parts of meteorological situations. This system is made up of hardware and software components that work together to provide the intended functionality.

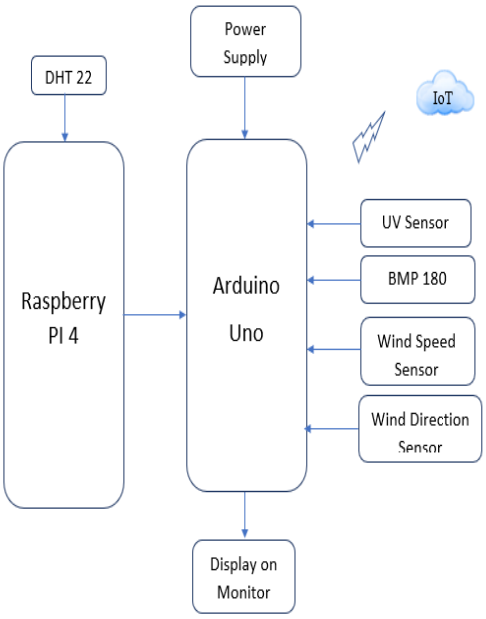


Figure 1. Block Diagram of the System

3. METHODOLOGY

Figure 2 displays this block diagram for the prediction system based on machine learning. The machine learning approach is applied as follow:

- Data Collection: Gather information for this system.
- Data Pre-processing: The most salient aspects of the data collection are extracted for additional usage after all of its attributes have been processed once. In this paper, data pre-processing uses handling missing.
- Training and testing using weather data of Machine Learning Model.

The collected data is then transmitted to a MongoDB database for storage and analysis. Python scripts are used to automate data collection and transmission, ensuring seamless operation. Decision tree algorithms will be employed to analyse the stored data and predict weather conditions. Decision trees are chosen for their simplicity and effectiveness in handling classification and regression tasks, making them suitable for weather prediction.

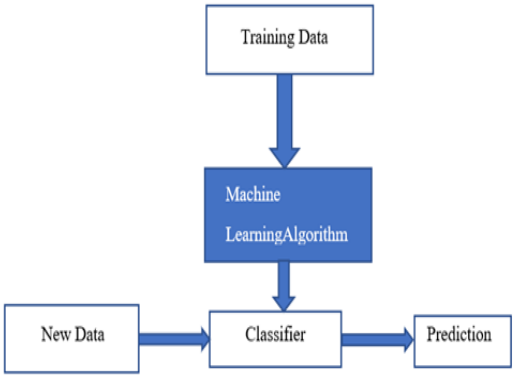


Figure 2. Machine Learning-Based Prediction System

3.1. Hardware Requirements

Raspberry Pi 4: A versatile single-board computer used for data collection and processing. It is capable of connecting sensors and other devices with its 40-pin expanded GPIO.

DHT22: A sensor for measuring temperature and humidity. It is used to measure both temperature and relative humidity in the environment. The sensor is known for its accuracy and reliability, providing precise data with a wide measurement range. In this system, DHT 22 sensor senses a temperature of 30.7 ℃ and humidity of 47.7 %rh at monitor. But it can be varied depending on the weather and region.

BMP180 Barometric Pressure Sensor: This high accuracy sensor monitors the barometric pressure or atmospheric pressure. The output of the sensor is digital. It utilizes the protocol for interconnected circuits. In this paper, the pressure value of 1006.716 Mb can be achieved by BMP 180 sensor.

UV Sensor: The strength or intensity of incident ultraviolet (UV) light is measured by UV sensors. Though its wavelengths are less than those of visible light, this type of electromagnetic radiation is still longer than x-rays. In this process, 4.44 V and 0.4 mA can be obtained by this sensor.

Wind Speed Sensor: The top three wind cups rotate due to the wind created by the airflow

and the internal sensor element is driven by the centre axis to produce an output signal that can be used to calculate wind speed. In north of west direction, the wind speed optimizes 4 km/h. But south of west direction, it enhances 6 km/h.

The data collection process begins with the Raspberry Pi gathering readings from the DHT22 and BMP180 sensors at regular intervals. These readings include temperature, humidity, and atmospheric pressure.

3.2. Software Requirements

Python: Python is a high-level programming language. The automatic memory management and dynamic type system are its salient features. Used for scripting data collection and processing.

MongoDB: MongoDB is a cross-platform, document-oriented database that is open source. A NoSQL database for storing sensor data.

Decision Tree Algorithm: Used for weather predication based on historical data. A decision tree is a flowchart- like tree- structure where each internal node denotes the feature, branches denote the rules and the leaf nodes denote the result of the algorithm.

The mathematical expression for entropy (H)

$$H(S) = \sum -p(c)log_2p(c) \quad (1)$$

S = The current (data) set for which entropy is being calculated

C = Set of classes in S, C = {yes, no}

p(c) = The proportion of the number of elements in class C to the number of elements in set S

The mathematical expression for Information Gain (IG)

$$IG(A,S) = H(S) - \sum p(t)H(t)IG(A,S) \quad (2)$$

where,

H(S) = Entropy of set S

p(t) = The proportion of the number of elements in t to the number of elements in set S

H (t) = Entropy of subset t

Different methods that improve system are described for handling missing or corrected sensor data in Table 1.

Table 1. Methods for Handling Missing to Improve System

No.	Methods	Contents
1	K-Nearest Neighbors (KNN) Imputation	The results of the dataset's closest and most comparable data points are used to impute missing data.
2	Machine Learning-Based Imputation (Random Forest)	Advanced machine learning models can be trained on the available data to predict missing values based on patterns and relationships between different variables.
3	Kalman Filter	A recursive algorithm that estimates the true state of a dynamic system by considering the measurement noise and uncertainties. It can smooth out noisy or corrupted sensor data by using past information.

The effectiveness of the Machine Learning-Based Imputation approach in addressing missing or corrupted sensor data is contingent upon the unique attributes of the data, the system's criticality, and the computational resources at hand.

Data collected by the sensors is transmitted wirelessly to a central server where it is stored in MongoDB. The system is designed to be scalable and energy-efficient, making it suitable for deployment in various environments, including remote areas.

- Sensor Calibration and Data Accuracy

To make sure that the data gathered is accurate, sensors are calibrated regularly. Calibration includes comparing sensor readings to established reference values and

adjusting as needed to address any differences found. The DHT22 and BMP180 sensors used in this system are known for their great dependability and accuracy, making them suitable for precise weather monitoring.

- Data Transmission and Storage

Wi-Fi is used to send the sensor-gathered data to the central server. The Raspberry Pi, equipped with a Wi-Fi module, ensures seamless data transmission. MongoDB is chosen for data storage due to its scalability and flexibility in handling large volumes of unstructured data. The database schema is designed to efficiently store time-series data, allowing for easy retrieval and analysis.

Case Study 1: Agricultural Applications

In an agricultural setting, accurate weather data is crucial for planning planting and harvesting activities. The weather monitoring system was deployed in a farm to provide current temperature data, humidity, and soil moisture levels. Farmers used this data to optimize irrigation schedules, resulting in improved crop yields and water conservation. The integration of IoT-based weather monitoring in agriculture demonstrated significant benefits, including reduced water usage and enhanced crop productivity.

Case Study 2: Disaster Management

The weather monitoring system was deployed in a disaster-prone area to provide early warnings for extreme weather conditions such as floods and storms. Authorities were able to lessen the impact of natural disasters on the local population by issuing timely notifications and putting evacuation plans into action because to the real-time data the system collected. The system's ability to provide accurate and timely weather information proved invaluable in mitigating the effects of extreme weather events.

4. TEST AND RESULT

Detailed insights into the data dependencies are provided by categorization. The primary outcome anticipated from this effort is the determination of whether or not rain is forecast on a given day based on sensor data, the graph illustrates how humidity and temperature affect rain.

The prediction of hourly and daily data is described as shown in Figure 3. Daily temperature, humidity and pressure are included in the data sets. The new data set for day 10.5.2024, which was created by averaging the values, is displayed in Figure 6.

	A	B	C	D	E
1	_id	timestamp	tempC	tempF	humidity
2	663dd1b6	2024-05-10T14:20:14.090Z	30.6	87.08	47.1
3	663dd1b9	2024-05-10T14:20:17.486Z	30.6	87.08	47.2
4	663dd1bc	2024-05-10T14:20:20.841Z	30.6	87.08	47.2
5	663dd1c0	2024-05-10T14:20:24.208Z	30.6	87.08	47.2
6	663dd1c3	2024-05-10T14:20:27.562Z	30.6	87.08	47.3
7	663dd1c6	2024-05-10T14:20:30.960Z	30.6	87.08	47.3
8	663dd1ca	2024-05-10T14:20:34.324Z	30.6	87.08	47.6
9	663dd1cd	2024-05-10T14:20:37.708Z	30.7	87.26	47.7
10	663dd1d1	2024-05-10T14:20:41.074Z	30.7	87.26	47.6
11	663dd1d4	2024-05-10T14:20:44.426Z	30.5	86.9	47.4
12	663dd1d7	2024-05-10T14:20:47.767Z	30.6	87.08	47.5
13	663dd1de	2024-05-10T14:20:54.444Z	30.6	87.08	47.3
14	663dd1e1	2024-05-10T14:20:57.789Z	30.6	87.08	47.3
15	663dd1e5	2024-05-10T14:21:01.170Z	30.6	87.08	47.3
16	663dd1e8	2024-05-10T14:21:04.512Z	30.6	87.08	47.3
17	663dd1eb	2024-05-10T14:21:07.847Z	30.6	87.08	47.3
18	663dd1ef	2024-05-10T14:21:11.183Z	30.6	87.08	47.3
19	663dd1f2	2024-05-10T14:21:14.572Z	30.7	87.26	47.4
20	663dd1f5	2024-05-10T14:21:17.937Z	30.7	87.26	47.4
21	663dd1f9	2024-05-10T14:21:21.305Z	30.7	87.26	47.4
22	663dd1fc	2024-05-10T14:21:24.684Z	30.6	87.08	47.4
23	663dd203	2024-05-10T14:21:31.369Z	30.6	87.08	47.4
24	663dd206	2024-05-10T14:21:34.731Z	30.6	87.08	47.4
25	663dd20a	2024-05-10T14:21:38.108Z	30.7	87.26	47.6

Figure 3. Derived Data Set from Data by Averaging for day 10.5.2024

Figure 4 shows the new data set of day 11.5.2024 in Pyinsrar village, Pyin Oo Lwin derived by averaging the values.

	A	B	C	D	E
1	_id	timestamp	tempC	tempF	humidity
2	663ed097	2024-05-11T08:2	26.2	79.16	66.4
3	663ed09c	2024-05-11T08:2	26.3	79.34	67.5
4	663ed0a7	2024-05-11T08:2	26.2	79.16	67.5
5	663ed0ab	2024-05-11T08:2	26.2	79.16	67.5
6	663ed0af	2024-05-11T08:2	26.2	79.16	67.5
7	663ed0b2	2024-05-11T08:2	26.2	79.16	67.5
8	663ed0b5	2024-05-11T08:2	26.3	79.34	67.5
9	663ed0b9	2024-05-11T08:2	26.2	79.16	67.6
10	663ed0bc	2024-05-11T08:2	26.2	79.16	67.4
11	663ed0bf	2024-05-11T08:2	26.2	79.16	67.4
12	663ed0c3	2024-05-11T08:2	26.2	79.16	67.4
13	663ed0c6	2024-05-11T08:2	26.2	79.16	67.4
14	663ed0ca	2024-05-11T08:2	26.3	79.34	67.4
15	663ed0cd	2024-05-11T08:2	26.2	79.16	67.4
16	663ed0d0	2024-05-11T08:2	26.4	79.52	67.5
17	663ed0d4	2024-05-11T08:2	26.2	79.16	67.3
18	663ed0d7	2024-05-11T08:2	26.3	79.34	67.4
19	663ed0da	2024-05-11T08:2	26.3	79.34	67.5
20	663ed0de	2024-05-11T08:2	26.3	79.34	67.4
21	663ed0e4	2024-05-11T08:2	26.2	79.16	67.2
22	663ed0e8	2024-05-11T08:2	26.3	79.34	67.8
23	663ed0eb	2024-05-11T08:2	26.3	79.34	67.6
24	663ed0ee	2024-05-11T08:2	26.2	79.16	67.5
25	663ed0f2	2024-05-11T08:2	26.3	79.34	67.5
26	663ed0f5	2024-05-11T08:2	26.2	79.16	67.3
27	663ed0f8	2024-05-11T08:2	26.4	79.52	67.4

Figure 4. Derived Data Set from Data by Averaging for day 11.5.2024 without Pressure Conditions

	A	B	C	D	E	F	G	H
1	_id	timestamp	tempC	tempF	humidity	pressure	pressure	rain
2	663ed097	2024-05-11T08:27:43	26.2	79.16	66.4	2651.763	1006.716	0
3	663ed09c	2024-05-11T08:27:48	26.3	79.34	67.5	2651.616	1007.083	0
4	663ed0a7	2024-05-11T08:27:59	26.2	79.16	67.5	2651.852	1006.165	0
5	663ed0ab	2024-05-11T08:28:03	26.2	79.16	67.5	2651.881	1006.716	0
6	663ed0afd	2024-05-11T08:28:07	26.2	79.16	67.5	2652	1006.257	0
7	663ed0b2	2024-05-11T08:28:10	26.2	79.16	67.5	2651.97	1006.899	0
8	663ed0b5	2024-05-11T08:28:13	26.3	79.34	67.5	2651.852	1006.257	0
9	663ed0b9	2024-05-11T08:28:17	26.2	79.16	67.6	2651.97	1006.991	0
10	663ed0bcc	2024-05-11T08:28:20	26.2	79.16	67.4	2651.704	1006.165	0
11	663ed0bfd	2024-05-11T08:28:23	26.2	79.16	67.4	2652.206	1006.073	0
12	663ed0c3	2024-05-11T08:28:27	26.2	79.16	67.4	2652.059	1006.716	0
13	663ed0c6	2024-05-11T08:28:30	26.2	79.16	67.4	2652.029	1006.808	0
14	663ed0ca	2024-05-11T08:28:34	26.3	79.34	67.4	2651.852	1006.532	0
15	663ed0cd	2024-05-11T08:28:37	26.2	79.16	67.4	2651.852	1006.899	0
16	663ed0d0	2024-05-11T08:28:40	26.4	79.52	67.5	2651.97	1006.44	0
17	663ed0d4	2024-05-11T08:28:44	26.2	79.16	67.3	2651.852	1006.716	0
18	663ed0d7	2024-05-11T08:28:47	26.3	79.34	67.4	2652.029	1005.155	0
19	663ed0da	2024-05-11T08:28:50	26.3	79.34	67.5	2651.881	1006.44	0
20	663ed0de	2024-05-11T08:28:54	26.3	79.34	67.4	2651.941	1006.808	0
21	663ed0e4	2024-05-11T08:29:00	26.2	79.16	67.2	2651.97	1005.981	0
22	663ed0e8	2024-05-11T08:29:04	26.3	79.34	67.8	2652.029	1006.44	0
23	663ed0eb	2024-05-11T08:29:07	26.3	79.34	67.6	2652.147	1005.981	0
24	663ed0ee	2024-05-11T08:29:10	26.2	79.16	67.5	2651.97	1006.44	0
25	663ed0f2	2024-05-11T08:29:14	26.3	79.34	67.5	2651.941	1006.349	0

Figure 5. Derived Data Set from Data by Averaging with Pressure Conditions

Figure 6 shows the data are achieve from measuring using decision tree of in machine learning algorithms. Temperature changes in each minute.

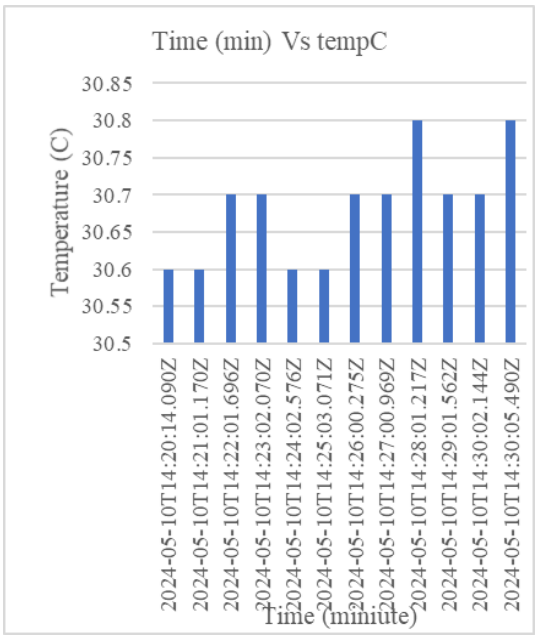


Figure 6. Temperature Changes in Each Minute from Decision Tree Algorithm Measure of this System

In this result, highest temperature is 30.8℃ and the lowest temperature is 30.6℃ within 10 minutes. From this result, temperature changes in every minute can be easily recorded. The data are measured in Pyinsar village, Pyin Oo Lwin.

Figure 7 shows humidity changes during 5th May 2024 in each minute. In this result, highest humidity is 50.5%rh and the lowest humidity is 47.1%rh within 10 minutes. In looking this result, suitable for agriculture.

In agriculture, accurate weather data is essential for optimizing irrigation schedules, planning planting and harvesting activities, and protecting crops from extreme weather conditions. The weather monitoring system can provide farmers with real-time data and accurate forecasts, enabling them to make informed decisions and improve crop yields.

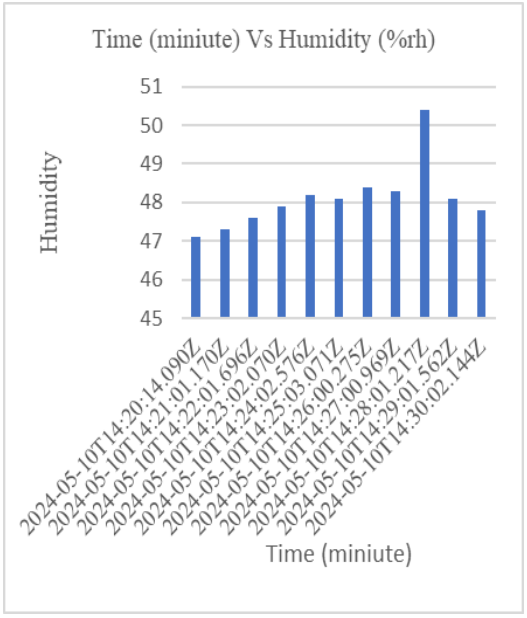


Figure 7. Humidity Changes in Each Minute from Decision Tree Algorithm Measure of this System

Figure 8 shows temperature changes in daily maximum temperature (℃) during 10th May 2024. In this result, highest temperature is 30.5℃ and the lowest temperature is 25.1℃ in daily recorded. In the same way, monthly temperatures can be recorded.

In smart cities, weather monitoring systems can enhance urban planning and management. Real-time weather data can be used to

optimize energy consumption, manage traffic flow, and improve public safety using solar system. The integration of IoT-based weather monitoring systems with other smart city technologies can lead to more efficient and sustainable urban environments.

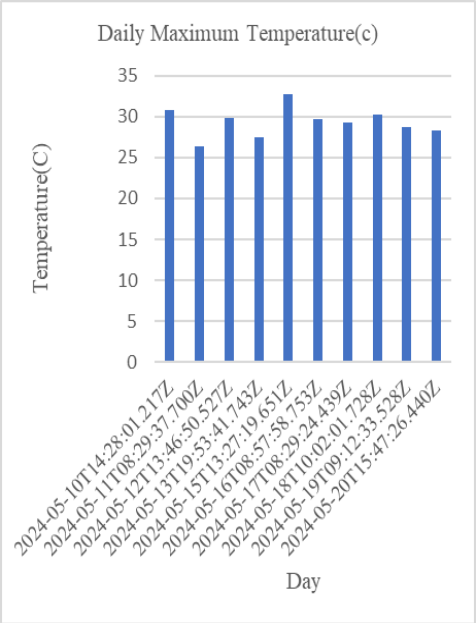


Figure 8. Temperature Changes in Daily Temperature

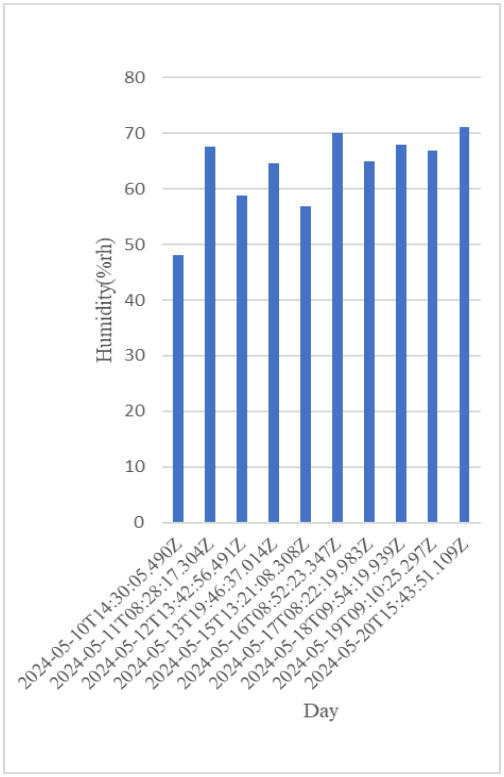


Figure 9. Humidity Changes in Daily from Decision Tree Algorithm

Daily Humidity changes is shown in Figure 9. In this figure, result shown the lowest humidity is 49%rh and the highest humidity is 71%rh. In the same way, monthly humidity can be recorded.

The data are achieved from measuring using decision tree of in machine learning algorithms as show in Figure 10. Pressure changes in each minute. In this result, highest Pressure is 2641.37 Mb and the lowest pressure is 2640.97 Mb in each minute. From this result, Pressure changes in every minute can be easily recorded using machine learning algorithms.

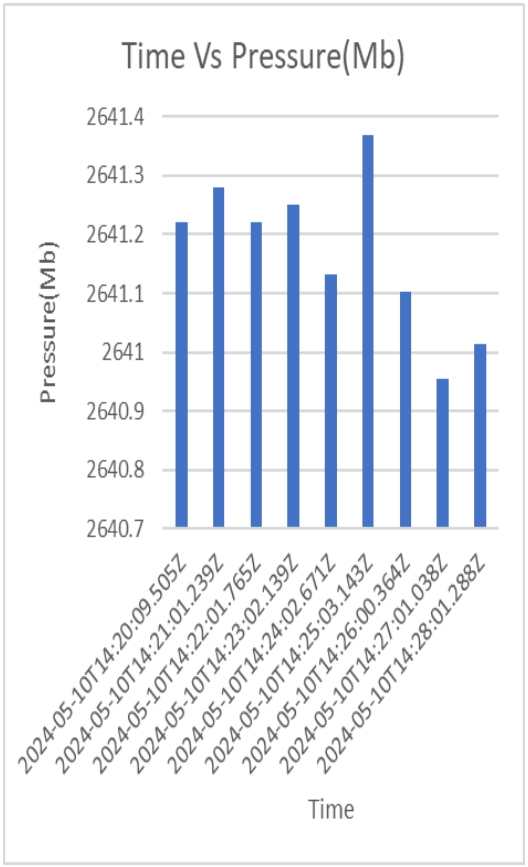


Figure 10. Pressure Changes in Each Minute from Decision Tree Algorithm Measure of this System

5. CONCLUSION

This paper presents a comprehensive weather monitoring system that leverages IoT and machine learning technologies to provide accurate and real-time weather data. In this paper, shown temperature and humidity in every minute and daily result using decision

tree method. The system's low cost, scalability, and customizability make it suitable for a wide range of applications. The successful implementation of this system demonstrates the potential of IoT and machine learning in revolutionizing weather monitoring and forecasting.

In this paper, the collection of data is measured in Pyinsar village, Pyin Oo Lwin. Using decision tree algorithm, it gets maximum and lowest temperature are 30.8℃ and 30.6℃ within 10 minutes. The lowest and highest humidity are 49%rh and 71%rh. It gets the highest pressure is 2641.37 Mb and the lowest pressure is 2640.97 Mb in each minute.

ACKNOWLEDGMENT

The author would like to express sincere appreciation to the Rector and Pro-Rector of University of Technology (Yatanarpon Cyber City) for kind Permission to prepare for this paper. The author would also like to give special thanks to my supervisor and all teachers in Faculty of Electronic Engineering and all who willingly helped the author throughout the preparation of the paper.

REFERENCES

[1] Aslani, M., Seipel, S., 2020. A fast instance selection method for support vector machines in building extraction. Applied Soft Computing 97, 106716.

[2] Aslani, M., Seipel, S., 2021. Efficient and decision boundary aware instance selection for support vector machines. Information Sciences 577, 579–598.

[3] Huiningsumbam, U., Jain, A., Verma, N., 2020. Artificial neural network for weather forecasting: A review, in: 2020 IEEE International Conference on Technology, Engineering, Management for Societal impact using Marketing, Entrepreneurship and Talent (TEMSMET), IEEE. pp. 1–7.

[4] Mishra, S., Bera, A., & Behera, J. K. (2022). Weather Monitoring System. *Dogo Rangsang Research Journal*, 12(5), 582-586.

[5] Wu, J., Yang, S., Yang, F., & Yin, X. (2021). Road Weather Monitoring System Shows High Cost-Effectiveness in Mitigating Malfunction Losses. *Sustainability*, 13(22), 12437.

[6] Xu, J., 2011. An extended one-versus-rest support vector machine for multi-label classification. Neurocomputing 74, 3114–3124.

[7] Zhang, W., Zhang, H., Liu, J., Li, K., Yang, D., Tian, H., 2017. Weather prediction with multiclass support vector machines in the fault detection of photovoltaic system. IEEE/CAA Journal of Automatic a Sinica.

Authors

Biography

Ngu War Tun is received B.E from Government Technological Collage (Shwe Bo) in 2017; currently working as an Assistant Lecturer in Faculty of Electronic Engineering at University of Technology (Yatanarpon Cyber City).