IAES International Journal of Artificial Intelligence (IJ-AI)

IAES International Journal of Artificial Intelligence (IJ-AI), ISSN/e-ISSN 2089-4872/2252-8938 publishes articles in the field of artificial intelligence (AI). The scope covers all artificial intelligence (AI) and machine learning (ML) areas and theirs applications in the following topics: neural networks; fuzzy logic; simulated biological evolution algorithms (like genetic algorithm, ant colony optimization, etc); reasoning and evolution; intelligence applications; computer vision and speech understanding; multimedia and cognitive informatics, data mining and machine learning tools, heuristic and AI planning strategies and tools, computational theories of learning; technology and computing (like particle swarm optimization); intelligent system architectures; knowledge representation; bioinformatics; natural language processing; multiagent systems; supervised learning; unsupervised learning; deep learning; big data and AI approaches; reinforcement learning; and learning with generative adversarial networks; etc. This journal is indexed in Scopus and all published papers since 2018 issues were included in scopus.com.

Focus and Scope

The IAES International Journal of Artificial Intelligence (IJ-AI), ISSN/e-ISSN 2089-4872/2252-8938 covers all topics of artificial intelligence and soft computing and their applications, including but not limited to:

• Neural networks
• Reasoning and evolution
• Intelligent search
• Intelligent planning
• Intelligence applications
• Computer vision and speech understanding
• Multimedia and cognitive informatics
• Data mining and machine learning tools, heuristic and AI planning strategies and tools, computational theories of learning
• Technology and computing (like particle swarm optimization); intelligent system architectures
• Knowledge representation
• Bioinformatics
• Natural language processing
• Automated reasoning
• Logic programming
• Machine learning
• Visual/linguistic perception
• Evolutionary and swarm algorithms
• Derivative-free optimisation algorithms
• Fuzzy sets and logic
• Rough sets
• Simulated biological evolution algorithms (like genetic algorithm, ant colony optimization, etc)
• Multi-agent systems
• Data and web mining
• Emotional intelligence
• Hybridisation of intelligent models/algorithms
• Parallel and distributed realisation ofintelligent algorithms/systems
• Application in pattern recognition, image understanding, control, robotics and bioinformatics
• Application in system design, system identification, prediction, scheduling and game playing
• Application in VLSI algorithms and mobile communication/computing systems

Principal Contact
Prof. Dr. Eugene Yu-Dong Zhang
Editor-in-Chief, IJ-AI
Chair in Knowledge Discovery and Machine Learning
Associate Fellow of Higher Education Academy
IEEE Senior Member
ACM Senior Member

Contact
F26 Informatics Building
Department of Informatics
University of Leicester, University Road,
Leicester, LE1 7RH, UK
Email: ijai@iaesjournal.com

IAES International Journal of Artificial Intelligence (IJ-AI)

Dr. Aida Mustapha
Universiti Tun Hussein Onn Malaysia, Malaysia

Dr. Choong Seon Hong
Kyung Hee University, Korea, Republic of

Dr. Chunguo Li
Henan University of Science and Technology, China

Dr. D. Jude Hemanth
Karunya University, India

Dr. Dhiya Al-Jumeily
Liverpool John Moores University, United Kingdom

Dr. Farhad Soleimanian Gharehchopogh
Hacettepe University, Turkey

Dr. Floriano De Rango
University of Calabria, Italy

Dr. Gloria Bordogna
Institute for Electromagnetic Sensing of the Environment, Italy

Dr. Honghai Liu
University of Portsmouth, United Kingdom

Dr. Ibrahim Kucukkoc
Balikesir University, Turkey

Dr. Igor Kotenko
Saint-Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Russian Federation

Dr. Iickho Song
Korea, Republic of

Dr. Imam Much Ibnu Subroto
Universitas Islam Sultan Agung, Indonesia

Dr. Iztok Fister Jr.
University of Maribor, Slovenia

Dr. Javier Gozalvez
Miguel Hernandez University of Elche, Spain

Dr. Jingjing Wang
Tsinghua University, China

Dr. John S. Vardakas
Iquadrat Informatica S.L., Spain

Dr. Karan Veer
DR BR Ambedkar National Institute of Technology, India

Dr. Liang Yang
Hunan University, China

Dr. Lin X. Cai
Illinois Institute of Technology, United States

Dr. Magdi S. Mahmoud
King Fahd University of Petroleum and Minerals, Saudi Arabia

IAES International Journal of Artificial Intelligence (IJ-AI)
Vol 11, No 3 September 2022

Table of Contents

❏   210

Estimation of standard penetration test value on cohesive soil using artificial neural network without data normalization

Soewignjo Agus Nugroho, Hendra Fernando, Reni Suryanita
Department of Civil Engineering, Riau University, Pekanbaru, Riau

Article Info	ABSTRACT
	Artificial neural networks (ANNs) are often used recently by researchers to solve complex and nonlinear problems. Standard penetration test (SPT) and cone penetration test (CPT) are field tests that are often used to obtain soil parameters. There have been many previous studies that examined the value obtained through the SPT test with the CPT test, but the research carried out still uses equations that are linear. This research will conduct an estimated value of SPT on cohesive soil using CPT data in the form of end resistance and blanket resistance, and laboratory test data such as effective overburden pressure, liquid limit, plastic limit and percentage of sand, silt and clay. This study used 242 data with testing areas in several cities on the island of Sumatra, Indonesia. The developed artificial neural network will be created without data normalization. The final results of this study are in the form of root mean square error (RMSE) values 3.441, mean absolute error (MAE) 2.318 and R^2 0.9451 for training data and RMSE 2.785, MAE 2.085, R^2 0.9792 for test data. The RMSE, MAE and R^2 values in this study indicate that the ANN that has been developed is considered quite good and efficient in estimating the SPT value.

Corresponding Author:

Soewignjo Agus Nugroho
Department of Civil Engineering, Engineering Faculty, Riau University
Kampus Bina Widya, HR. Soebrantas Street KM 12.5, Pekanbaru, Riau, Indonesia
Email: nugroho.sa@eng.unri.ac.id

1.   INTRODUCTION

Soil investigation is the first step that must be taken when building a construction. The soil investigation method depends on the soil condition and the function of the building. The standard penetration test (SPT) and the cone penetration test (CPT) are frequently used test methods at field sites. Usually investigations with soil investigations in the laboratory to obtain further parameters of the mechanical and physical properties of the soil. The SPT and CPT tests have their respective advantages and disadvantages. In the CPT test, only soil pressure data were obtained in the form of qc and fs values but no visual soil was obtained and the maximum depth of the test was 20 m. While the SPT data obtained soil samples and the depth of the test can be tens of meters, but the test data obtained is only soil hardness (number of blows for penetration of 30 cm). Therefore, this study was conducted to obtain the concept of a formula approach to be able to see the correlation of the SPT value from CPT data so that it can predict the soil strength (shear strength) at a depth of more than 20 m.

In recent years, artificial neural networks have attracted a lot of research interest in solving a problem that is complex and has a nonlinear nature. An artificial neural network (ANN) is an information processing system that has characteristics similar to a biological neural network [1]. Therefore, this study will estimate the value of SPT using artificial neural network capabilities. The artificial neural network

architecture consists of an input layer, an output layer and a hidden layer. Commonly used activation functions are binary sigmoid, bipolar sigmoid and linear functions. The learning algorithm that is most often used and effective in solving complex problems is the backpropagation algorithm. To find the best performing network, trial and error is carried out on the network architecture, activation functions and training parameters. The best artificial neural network model is obtained based on the smaller the error rate and the correlation coefficient value is close to 1. Until the time this paper was published, ANN still received extensive attention by researchers and continues to be developed. In the geotechnical field, there have been many studies using the capabilities of this ANN. Related researches such as soil composition [2], soil classification [3], soil compaction [4], bearing capacity [5], unit weight [6], shallow foundation bearing capacity [7]–[9], estimated settlement in shallow foundations [10]–[12], preconsolidation stress [13], electrical resistivity of soil [14], deformation of geogrid-reinforced soil structures [15], tunnel boring machine performance [16], estimating cohesion of limestone samples [17] and many other related studies.

In estimating the value of SPT, artificial neural networks have also been widely used by previous researchers such as predicting the value of N-SPT using the general regression neural network [18] method at a location in Izmir, Turkey. In this study using input data in the form of the percentage of gravel, sand, silt and clay. From the research, it was found that the value of R^2 was 0.9738, root mean square error (RMSE) 0.04, mean absolute error (MAE) 0.01 in the training data, while the test data obtained the values of R^2 0.9348, RMSE 0.08 and MAE 0.05. Another similar study is predicting N-SPT values based on CPT data at study locations in Dubai and Abu Dhabi, United Arab Emirates [19]. In this study, using input data in the form of end resistance value (qc), blanket resistance (fs) and soil effective pressure. From this study, the results obtained in the form of R value 0.95 and MAE 2.88.

The research conducted by [18], [19] had several shortcomings that could be complemented by other researchers. In research [19], estimating the value of SPT is only based on the percentage of gravel, sand, silt and clay, while in research [18] only estimates the value of SPT based on the values of qc, fs and overburden effective pressure. In fact, the value of soil density is influenced by many variables both from the mechanical and physical properties of the soil. Due to the large number of variables that affect soil density, conventional correlation is considered less effective in estimating SPT values. This problem is what prompted the author to conduct research to estimate the value of SPT using the ability of ANNs by combining the thoughts of previous researchers, namely using input variables in the form of tip resistance (qc), sleeve resistance (fs) obtained from CPT and laboratory data in the form of effective overburden pressure, liquid limit, plastic limit, percentage of sand, silt and clay on the cohesive soil.

In general, the data used in artificial neural networks will normalize the data or transform the data into a range of values according to the activation function used. For example, if you use the binary sigmoid activation function, then the data must be normalized by transforming the data into a range of 0 to 1 or if use the bipolar sigmoid activation function, the data will be normalized by transforming the data into the range -1 to 1. This method sometimes experiences difficulties because at times perform normalization, the data is not normalized to normal. Therefore, this research will carry out the process of developing an artificial neural network without normalizing the data. This research is expected to develop an artificial neural network without normalizing data with a low error rate, thus facilitating further research.

2. RESEARCH METHOD
2.1. Research model

In this study, the method used is to conduct direct research in the field and then process the research results using ANN software. Field tests carried out are SPT and CPT. The UDS sample obtained from the SPT test was carried out in the laboratory to obtain data on soil properties, both physical and mechanical properties of the soil. All data that has been obtained is then processed in Microsoft Excel software first for grouping data based on training data and test data. After grouping the data, the next step is to develop an artificial neural network using the ANN application. The research was conducted to obtain ANN with a smaller error value. In general, the research methodology can be seen in Figure 1.

2.2. Data collection

Data collection is an activity that aims to find the data needed in the research process in order to achieve the research objectives. In this study, the data needed is data on SPT, CPT and laboratory tests results. In SPT testing, the data required is the SPT value. In the CPT test, the data required is the value of end resistance (qc) and sleeve resistance (fs). In laboratory testing, the data required is the value of effective overburden pressure (σ'0), the percentage of sand and fine grains. These data were obtained from the Laboratory of Soil and Rock mechanics, Civil Engineering department, University of Riau. The data obtained are the results of testing in several areas on the island of Sumatra, Indonesia, including the provinces of Riau,

Riau Islands, Jambi, South Sumatra, West Sumatra and North Sumatra from 2005 to 2020. Statistics of all data can be seen in Table 1.
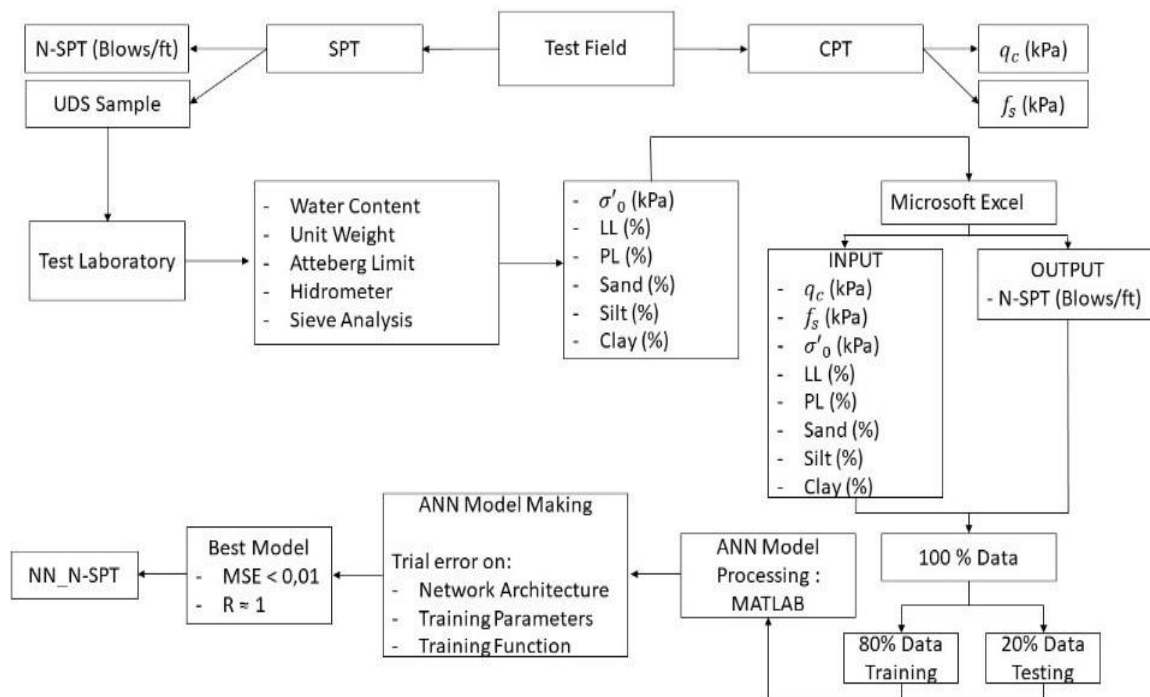
Figure 1. Research flow chart

Table 1. Collected data statistics

Variable	q_c (kPa)	f_s (kPa)	(σ'_0)(kPa)	Liquid Limit (%)	Plastic Limit (%)	Sand (%)	Silt (%)	Clay (%)	N-SPT (blows/ft)
Max	24525	426.106	422.105	87.210	51.700	71.440	96.050	95.630	60
Min	98.100	0.100	23.490	16.940	12.650	0.070	2.970	0.010	1
Mean	3028.281	83.289	157.609	48.976	27.271	11.920	36.512	51.399	11.612

2.3. Design of artificial neural network (ANN) model

The process of making ANNs is done by dividing the data into training data and test data. As much as 80% of the data is used as training data and 20% of the data will be used as test data. Training data is the data used to train the network by entering input data and output data. While the test data is data used to test the performance of the network being developed.

The design of the neural network model to be developed is adjusted to the purpose and nature of the data used. To predict the value of SPT that requires a relatively large amount of data input, the most appropriate method used is to create a network with a multilayer and backpropagation algorithm and supervised learning methods. Multilayer network architecture is the most appropriate solution for network models with large amounts of data and relatively complex problems. Multilayer network architecture consists of 3 layers, that is:

a. Input layer, this layer consists of several neurons whose number is adjusted according to the input pattern or variable.
b. Output layer, this layer consists of neurons whose number is in accordance with the desired output pattern or variable.
c. Hidden layer, this layer is between the input layer and the output layer, one or more hidden layers is determined based on a trial process and the number of neurons in the hidden layer is also determined based on a trial process to find the best performing network.

To get the network with the best performance, several trial variations can be carried out, that is:

a. Variations in network architecture (number of hidden layers and number of neurons in hidden layers).

b. Variations on training functions (trainlm, traincgb, traingd, traingdm, traingda, traingdx, trainrp, traincgf, traincgp, or other training functions that have been provided).
c. Variations in the activation function (bipolar sigmoid, binary sigmoid or linear function).
d. Variations in training parameters (number of epochs, learning rate, goals, and validation checks).
The training process can be stopped if you have found a network with the best performance, namely a network with a smaller error value and an R value that is closer to 1.

2.4. Testing the artificial neural network (ANN) model

To measure the accuracy and performance of the neural networks developed in producing SPT values, this study uses the RMSE and MAE values and the R^2 value. The best performance is indicated by the small RMSE and MAE values and the R^2 value that is close to 1. To calculate the RMSE and MAE values, the following equation is used:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(f_i - y_i)^2} \tag{1}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|f_i - y_i| \tag{2}$$

Where, RMSE is root mean square error, MAE is mean absolute error, f_i is original value, y_i is forecast result value, and n is amount of data.

2.5. Comparison of ANN with conventional equations

The best artificial neural network model that has been obtained is then compared with conventional equations by several previous studies in determining the SPT value. Several previous studies in determining the value of the SPT can be seen in Table 2. Where Kc is the ratio between qc and N-SPT or Kc=qc/NSPT (in MPa), N is the SPT value, D50 is the grain diameter that passes 50% filter while FC is the fines content.

Table 2. The value of Kc is based on several studies

Reference	Kc (MPa)	Notes
[20]	0.77	Sand
	0.70	Silty Sand
	0.58	Sandy Silt
[21]	0.438	Sand (Canada, Japan, Norwagia, China and Italy): D50=0.35+-0.23 mm
[22]	0.508	Clean Sand dan sandy silt, FC=3%-35%
	0.568	Sweden Sand
	0.367	Clay, Silty Clay and Silt
[23]	0.423	Sandy Silt, silt-sand
	0.529	Clean Sand dan Clayey Sand
	0.374	Sandy Clay, Silty Sand, Silty Clayey Sand
	0.572	Gravelly Sand, Coarse Sand and Sand-Gravel
[24]	0.37	Clay dan silty sand (Tanzania): D50=0.38 mm
[25]	0.43	Victoria Sand
	0.427	Silty Sand
[26]	0.337	Sandy Silt
	0.319	Silty Clay
	0.291	Clay

3.   RESULTS AND DISCUSSION
3.1. Results of the model making stage

From the research process, it was found that the best performance artificial neural network in estimating SPT values was a network with 2 hidden layer network architecture, 16 neurons in 1st hidden layer and 8 neurons in 2nd hidden layer. The training function used was traincgb. Network architecture can be seen in the Figure 2. In Figure 3 you can see the accuracy value of the network performance. The value of R training is 0.97053, R validation is 0.99052, R Test is 0.95974 and R All is 0.97216.

3.2. Weights and bias

Based on the best ANN model obtained, then the weight and bias values are also obtained. This value can be used as a multiplier of a network. Tables 3 to 7 are the weight and bias values of the developed network model.
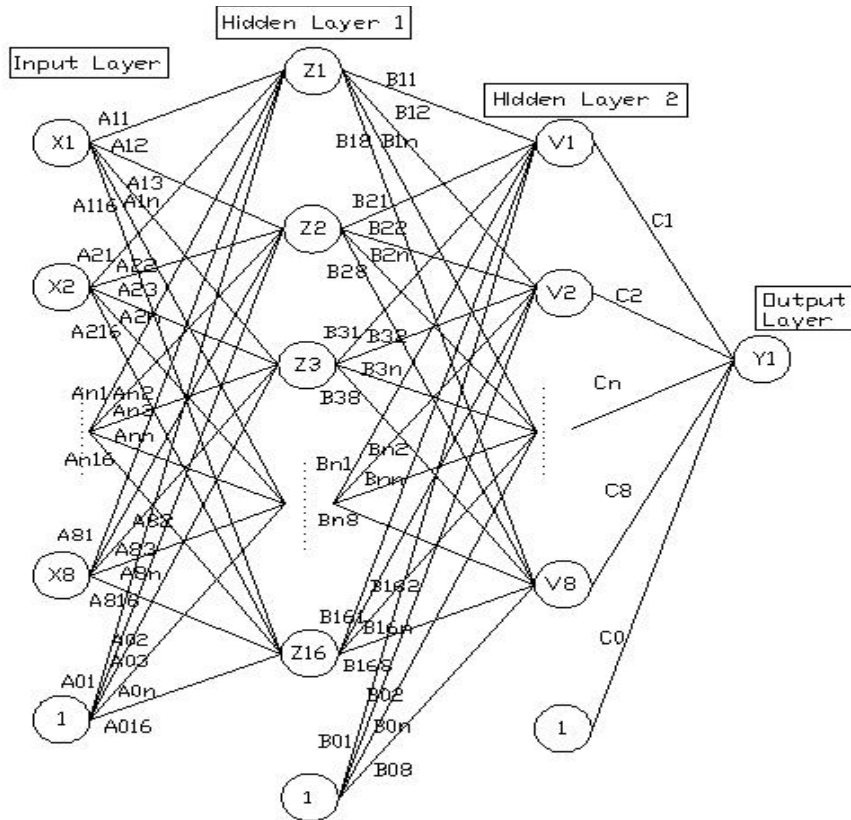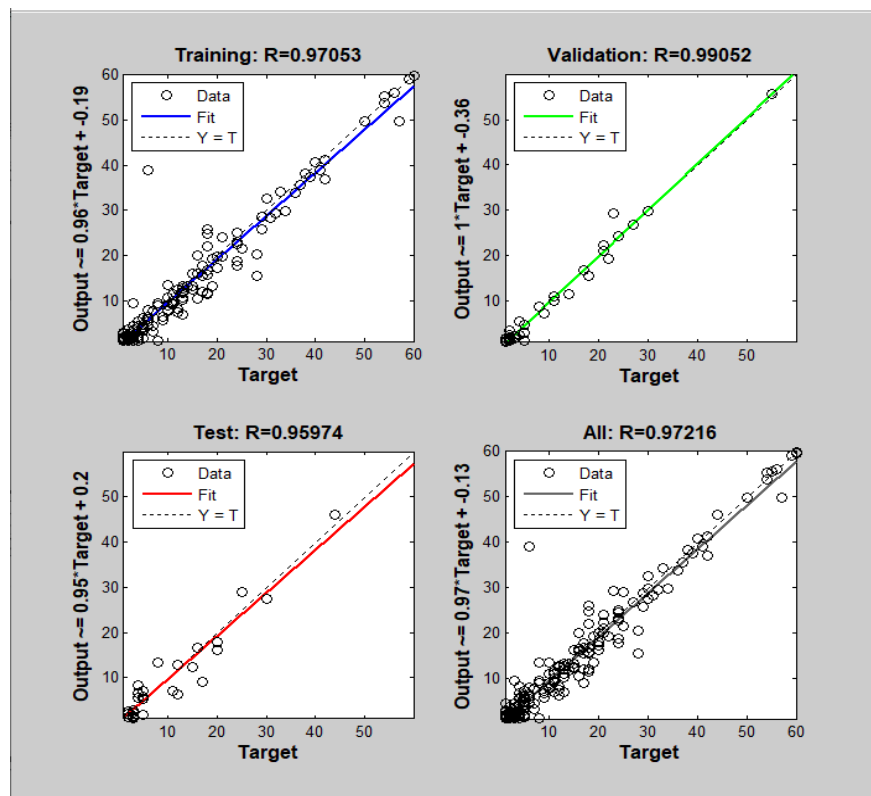
Figure 2. ANN architecture

Figure. 3. ANN regression

Table 3. Weights (A11-A816) from input layer to 1st hidden layer, as shown in Figure 2

A11	-1.321	A21	-0.543	A31	0.870	A41	-0.437	A51	0.207	A61	0.849	A71	-0.88	A81	0.778
A12	0.216	A22	-1.091	A32	-0.952	A42	0.541	A52	1.337	A62	-0.75	A72	0.752	A82	0.692
A13	-0.861	A23	0.067	A33	-0.556	A43	0.869	A53	-1.087	A63	0.893	A73	-0.819	A83	1.204
A14	-2.197	A24	-1.088	A34	0.532	A44	-0.365	A54	0.594	A64	1.970	A74	-1.584	A84	-0.600
A15	-0.308	A25	0.844	A35	-1.050	A45	0.438	A55	1.413	A65	-0.10	A75	-0.576	A85	0.810
A16	-0.183	A26	-0.920	A36	0.154	A46	0.381	A56	1.215	A66	-0.57	A76	0.364	A86	0.448
A17	-1.637	A27	-0.267	A37	2.367	A47	-2.290	A57	-1.130	A67	0.393	A77	-0.731	A87	-0.449
A18	-0.158	A28	0.744	A38	-0.923	A48	-0.089	A58	-0.387	A68	-0.14	A78	-1.479	A88	1.244
A19	-0.360	A29	1.173	A39	1.561	A49	-0.841	A59	-0.768	A69	-1.34	A79	-1.062	A89	-0.317
A110	-0.263	A210	1.563	A310	0.784	A410	0.667	A510	1.613	A610	0.602	A710	-0.002	A810	1.000
A112	-0.354	A212	2.228	A312	0.030	A412	-0.330	A512	0.996	A612	0.220	A712	-0.151	A812	1.957
A113	0.461	A213	-0.301	A313	1.092	A413	-0.798	A513	1.720	A613	-0.27	A713	-0.892	A813	0.633
A114	1.028	A214	0.596	A314	1.156	A414	-0.510	A514	-0.416	A614	0.543	A714	-1.501	A814	0.628
A115	1.117	A215	-0.110	A315	0.785	A415	-0.442	A515	-1.692	A615	-0.67	A715	-0.790	A815	0.380
A116	1.486	A216	-0.873	A316	0.369	A416	-0.216	A516	-0.524	A616	-0.86	A716	0.516	A816	-0.943

Table 4. Bias (A01-A016) from input layer to 1st hidden layer, as shown in Figure 2

A001	A002	A003	A004	A005	A006	A007	A008
1.819	-1.805	1.816	1.245	0.960	-1.559	1.158	-0.763
A009	A010	A011	A012	A013	A014	A015	A016
0.109	0.111	0.185	0267	-1.091	0.931	1.828	1.561

Table 5. Weights (B11-B168) from 1st hidden layer to 2nd hidden layer, as shown in Figure 2

B11	-0.103	B12	-0.040	B13	-0.839	B14	0.664	B15	-0.387	B16	-0.58	B17	-0.680	B18	-0.143
B21	0.232	B22	1.031	B23	-0.241	B24	-0.168	B25	0.505	B26	0.277	B27	0.464	B28	-0.299
B31	-0.425	B32	0.107	B33	-0.536	B34	1.112	B35	0.335	B36	0.920	B37	-0.291	B38	0.585
B41	0.380	B42	0.230	B43	-0.835	B44	-0.079	B45	-0.277	B46	-1.19	B47	-0.542	B48	-2.204
B51	0.658	B52	-0.309	B53	-0.958	B54	-0.531	B55	0.224	B56	1.203	B57	0.480	B58	0.098
B61	-0.237	B62	1.157	B63	-0.292	B64	0.543	B65	0.292	B66	-0.39	B67	0.216	B68	0.783
B71	-0.290	B72	1.169	B73	-0.555	B74	0.120	B75	-0.083	B76	0.465	B77	-0.227	B78	0.508
B81	-0.166	B82	0.612	B83	0.420	B84	-0.152	B85	0.589	B86	0.938	B87	-0.282	B88	0.533
B91	-0.183	B92	0.763	B93	0.763	B94	0.792	B95	-0.407	B96	-0.41	B97	0.439	B98	1.688
B101	-0.143	B102	-1.310	B103	-0.207	B104	0.311	B105	1.616	B106	0.524	B107	-0.384	B108	-1.572
B111	0.557	B112	-0.874	B113	0.178	B114	-0.887	B115	-0.175	B116	-0.42	B117	0.502	B118	-1.304
B121	-0.212	B122	-0.035	B123	-0.706	B124	1.606	B125	0.172	B126	-0.69	B127	-0.900	B128	0.067
B131	-0.549	B132	-0.208	B133	0.378	B134	-0.593	B135	1.056	B136	-0.06	B137	0.697	B138	-0.171
B141	-0.432	B142	-0.617	B143	0.176	B144	-0.695	B145	0.294	B146	0.45	B147	-0.091	B148	1.183
B151	-0.031	B152	-0.641	B153	0.143	B154	0.011	B155	0.544	B156	-0.48	B157	-0.030	B158	0.966
B161	-0.491	B162	-0.336	B163	-0.383	B164	-0.679	B165	0.548	B166	-0.57	B167	-0.115	B168	2.041

Table 6. Bias (B01-A08) from 1st hidden layer to 2nd hidden layer, as shown in Figure 2

B01	B02	B03	B04	B05	B06	B07	B08
1.789	-1.458	0.769	-0.520	-0.257	-0.664	-1,019	-1.286

Table 7. Weights (C1-C8) and bas (C0) from 2nd hidden layer to output layer, as shown in Figure 2

C1	C2	C3	C4	C5	C6	C7	C8	C0
-0.231	-2.029	-1.121	-1.783	1.209	-1.206	0.497	1.823	0.134

3.3. Results of the model testing stage

The ANN that has been developed is then simulated to estimate the SPT value. This simulation is done using input data on training data and input data on test data. Furthermore, the SPT output value from the artificial neural network is compared with the original SPT value to obtain the RMSE and MAE values. RMSE and MAE values from the simulation results can be seen in Table 8. To get the R^2 value, the predicted SPT value data compared with the original SPT value is displayed in a linear regression graph. This linear regression graph can be seen in Figure 4. In Figure 4(a) is a linear regression line on the training data and in Figure 4(b) is a linear regression line on the test data.

Table 8. Measure of accuracy ANN

Observation	Training data	Testing data
RMSE	3.278	2.012
MAE	1.783	1.328

The next step to compare the effectiveness of using artificial neural networks in estimating SPT values, estimation using conventional correlation by [23], [24], [26] was also carried out. The use of this correlation results in the RMSE and MAE values in Table 9. The R^2 value can be seen in the Figures 5(a) and (b) to Figures 7(a) and (b).

(a)   (b)
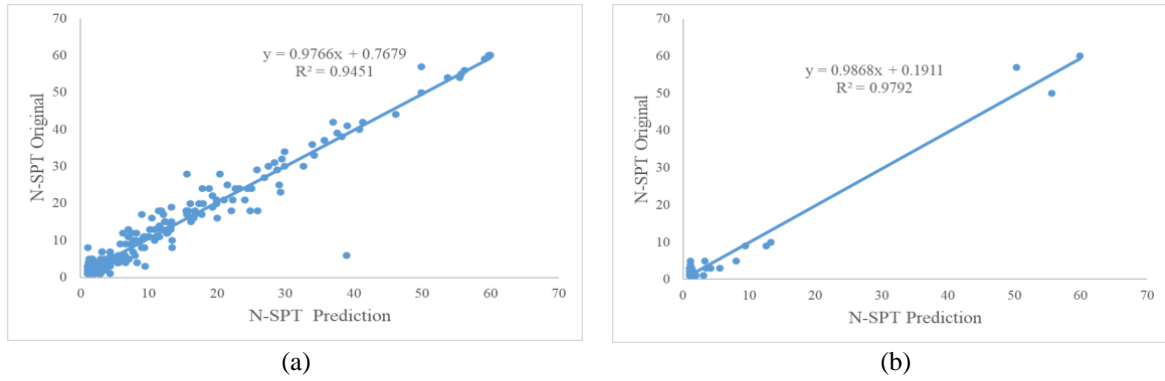
Figure 4. N-SPT prediction using ANNs (a) training data and (b) testing data

Table 9. Prediction of SPT value using conventional correlation

Research	Training data		Testing data	
	RMSE	MAE	RMSE	MAE
[23]	9.005	5.001	7.964	2.936
[24]	8.787	4.928	7.833	3.001
[26]	8.327	4.908	7.658	3.259

(a)   (b)

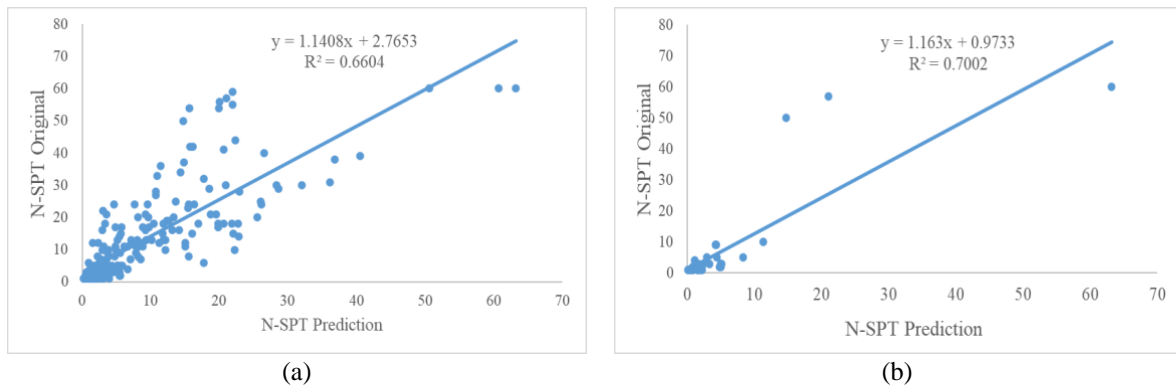Figure 5. SPT value estimation using correlation by [23] (a) training data and (b) testing data

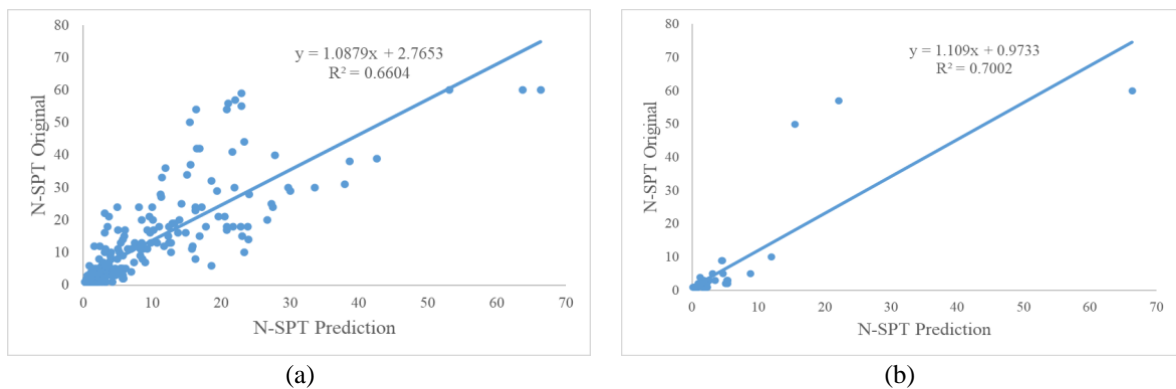(a)   (b)

Figure 6. SPT value estimation using correlation by [24] (a) training data and (b) testing data

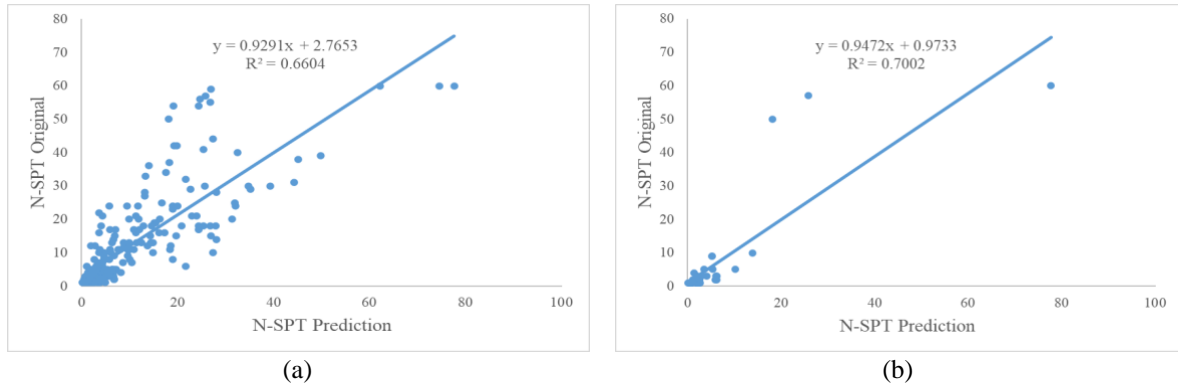(a)   (b)

Figure 7. SPT value estimation using correlation by [26] (a) training data and (b) test data

3.4. Design chart based on the best model

Figures 8(a) and (b) are a design chart between the estimated SPT value and the original SPT value. On this graph, a linear regression line is drawn between the estimated SPT value and the original SPT value through calculations with artificial neural networks or calculations using conventional correlation by [23], [24], [26]. Based on this graph, it can be seen that the estimation results of SPT values using ANNs give better results than the other three conventional correlations. Estimating the value of SPT using ANN produces a correlation coefficient (R^2) that is closer to 1 (red linear regression line) compared to the other three correlations, that is the R^2 value on the training data 0.9451 and on the 0.9792 test data.

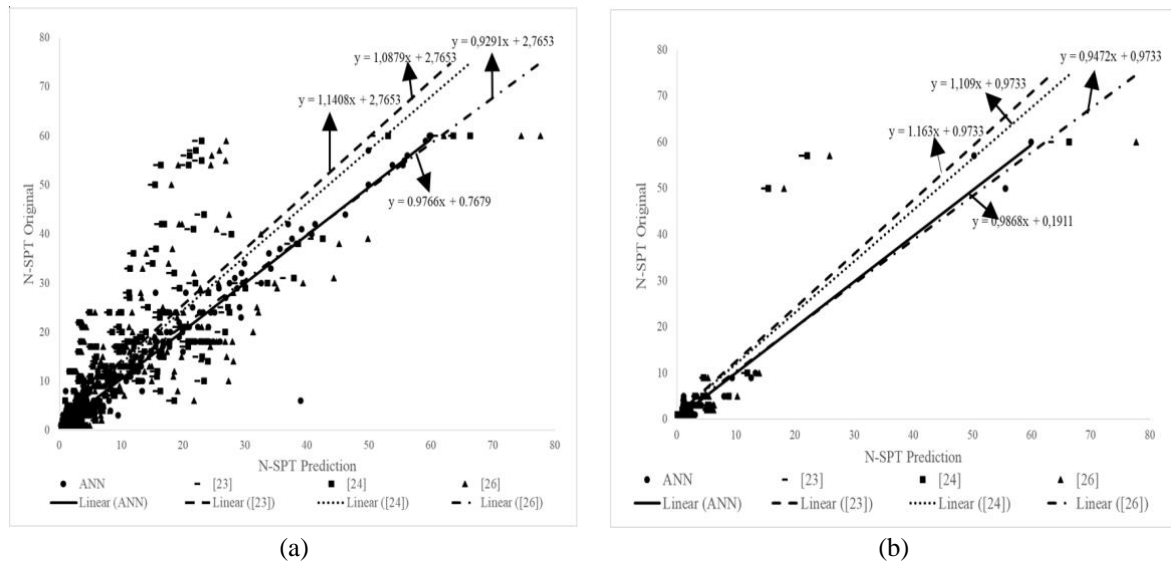(a)   (b)

Figure 8. Design chart N-SPT predicition VS N-SPT original (a) training data (b) testing data

Figures 9(a) and (b) are design chart between tip resistance (q_c) value and SPT value (N-SPT). In this graph, a combination of the relationship between qc and SPT values is displayed in the original data, the estimated data using an artificial neural network and the estimated data using conventional correlation by [23], [24], [26]. From this graph, it is clear that the linear regression line of the correlation between the q_c value and the SPT value estimated by ANN almost coincides with the linear regression line of the relationship between the q_c value and the SPT value in the original data. This means that the estimation results using ANN are almost close to the original value. Table 10 is a verification of the estimated data using an artificial neural network and using conventional correlation by [23], [24], [26]. It can be seen that the estimation results using an artificial neural network are almost close to the original value or have a small error value compared to using conventional correlation.
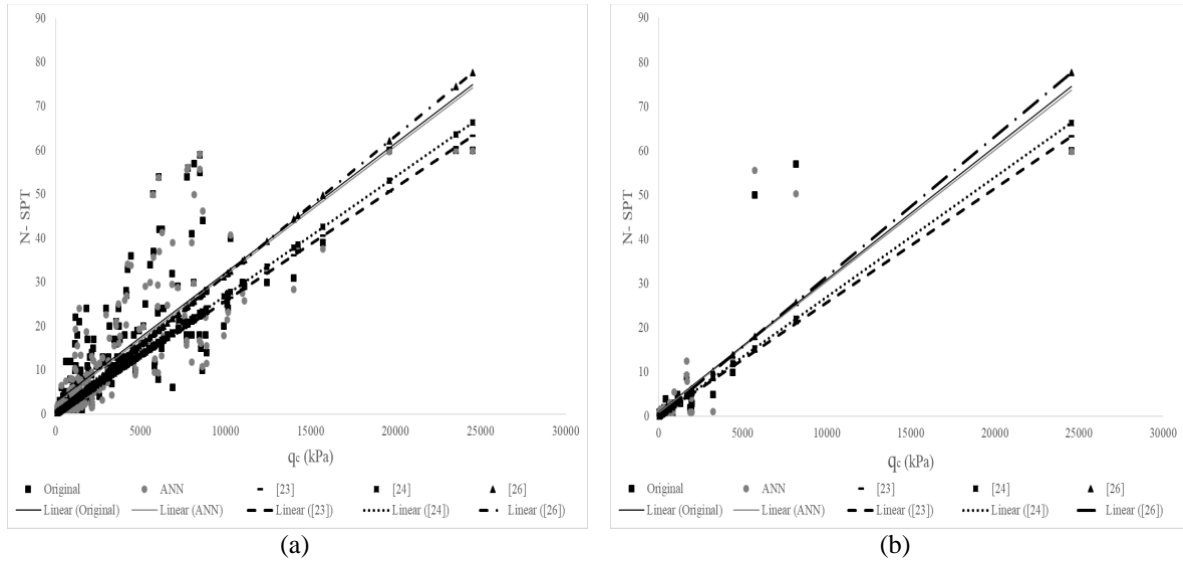
(a)   (b)

Figure 9. Design chart q_c VS N-SPT (a) training data and (b) test data

Table 10. Verification of SPT value estimation data with ANN and conventional correlation

No	qc (KN/m2)	fs (KN/m2)	σ'0 (KN/m2)	Liquid Limit (%)	Plastic Limit (%)	Sand (%)	Silt (%)	Clay (%)	Original N-SPT (blows/ ft)	ANN N-SPT (blows/ ft)	[23] N-SPT (blows/ ft)	[24] N-SPT (blows/ ft)	[26] N-SPT (blows/ ft)
				INPUT						OUTPUT			
1	837.623	63.388	199.150	60.90	39.10	0.080	49.48	50.440	1	1.003	2.159	2.264	2.651
2	2256.300	135.939	86.870	42.520	22.510	54.660	8.540	36.800	5	4.867	5.815	6.098	7.140
3	8632.800	303.920	302.100	49.540	27.460	0.800	12.850	86.350	10	10.797	22.249	23.332	27.319
4	5477.250	179.850	171.250	58.500	27.600	0.560	28.480	70.960	16	16.497	14.117	14.803	17.333
5	5165.341	74.455	252.900	18.500	15.200	53.140	29.890	16.970	20	19.832	13.313	13.960	16.346
6	6005.121	153.456	271.185	71.270	32.570	2.020	18.260	79.720	24	24.404	15.477	16.230	19.004
7	8115.545	98.100	319.755	58.200	30.380	1.900	11.830	86.270	30	29.832	20.916	21.934	25.682
8	4227.218	420.046	328.755	49.140	29.020	3.420	46.320	50.260	33	34.153	10.895	11.425	13.377
9	10277.862	237.955	131.170	73.170	33.420	0.240	9.850	89.910	40	40.741	26.486	27.778	32.525
10	8647.892	426.106	131.215	75.070	30.350	3.360	8.860	87.780	44	46.131	22.288	23.373	27.367
11	5715.162	157.960	185.430	76.530	32.960	0.120	94.350	5.530	50	49.850	14.730	15.446	18.086
12	7776.446	183.372	138.305	61.890	34.400	2.180	90.380	7.440	56	56.127	20.042	21.017	24.609
13	23544.00	98.100	176.500	72.000	42.400	1.180	49.690	49.690	60	59.931	60.680	63.632	74.506

4. CONCLUSION

ANN without data normalization is well developed in this study. The data used in this study were obtained from several locations on the island of Sumatra, Indonesia with a total of 244 data consisting of SPT, CPT and laboratory test data. This study uses input variables consisting of the value of tip resistance (q_c), blanket resistance (f_s), effective soil overburden pressure, liquid limit, plastic limit and percentage of sand, silt and clay. Meanwhile, the output variable is the SPT value. Based on the results of the research conducted, the network with the best performance is a network using network architecture with 2 hidden layers, 16 neurons in 1st Hidden Layer and 8 neurons in 2nd hidden layer, training function is traincgb, activation function is bipolar sigmoid and learning algorithm is backpropagation algorithm. This ANN model is said to be more effective in estimating the SPT value because it has a smaller error value than using conventional correlation. In the training data, the RMSE value for ANN was 3.278, MAE 1.783 and R^2 0.9451, while in the test data, the RMSE for ANN was 2.012, MAE 1.328, R^2 0.9792. Therefore, based on

this research, artificial neural networks without data normalization can be applied to other studies that have complex and nonlinear equations both in the geotechnical field and in other fields.

ACKNOWLEDGEMENTS

REFERENCES

[1] H. Bolouri, "Book review: fundamentals of neural networks- architectures, algorithms, and applications: L. Fausett," *The International Journal of Electrical Engineering and Education*, vol. 32, no. 3, pp. 284–285, Jul. 1995, doi: 10.1177/002072099503200320.

[2] P. U. Kurup and E. P. Griffin, "Prediction of soil composition from CPT data using general regression neural network," *Journal of Computing in Civil Engineering*, vol. 20, no. 4, pp. 281–289, Jul. 2006, doi: 10.1061/(ASCE)0887-3801(2006)20:4(281).

[3] C. Reale, K. Gavin, L. Librić, and D. Jurić-Kaćunić, "Automatic classification of fine-grained soils using CPT measurements and artificial neural networks," *Advanced Engineering Informatics*, vol. 36, pp. 207–215, Apr. 2018, doi: 10.1016/j.aei.2018.04.003.

[4] J. Jayan and N. Sankar, "Prediction of compaction parameters of soils using artificial neural network," *Asian Journal of Engineering and Technology*, vol. 3, no. 4, 2015.

[5] T. Gnananandarao, V. N. Khatri, and R. K. Dutta, "Prediction of bearing capacity of H shaped skirted footings on sand using soft computing techniques," *Archives of Materials Science and Engineering*, vol. 2, no. 103, pp. 62–74, Jun. 2020, doi: 10.5604/01.3001.0014.3356.

[6] G. Straż and A. Borowiec, "Estimating the unit weight of local organic soils from laboratory tests using artificial neural networks," *Applied Sciences*, vol. 10, no. 7, Art. no. 2261, Mar. 2020, doi: 10.3390/app10072261.

[7] Y. L. Kuo, M. B. Jaksa, A. V. Lyamin, and W. S. Kaggwa, "ANN-based model for predicting the bearing capacity of strip footing on multi-layered cohesive soil," *Computers and Geotechnics*, vol. 36, no. 3, pp. 503–516, Apr. 2009, doi: 10.1016/j.compgeo.2008.07.002.

[8] D. Padmini, K. Ilamparuthi, and K. P. Sudheer, "Ultimate bearing capacity prediction of shallow foundations on cohesionless soils using neurofuzzy models," *Computers and Geotechnics*, vol. 35, no. 1, pp. 33–46, Jan. 2008, doi: 10.1016/j.compgeo.2007.03.001.

[9] P. Provenzano, S. Ferlisi, and A. Musso, "Interpretation of a model footing response through an adaptive neural fuzzy inference system," *Computers and Geotechnics*, vol. 31, no. 3, pp. 251–266, Apr. 2004, doi: 10.1016/j.compgeo.2004.03.001.

[10] M. A. Shahin, H. R. Maier, and M. B. Jaksa, "Settlement prediction of shallow foundations on granular soils using B-spline neurofuzzy models," *Computers and Geotechnics*, vol. 30, no. 8, pp. 637–647, Dec. 2003, doi: 10.1016/j.compgeo.2003.09.004.

[11] M. A. Shahin, M. B. Jaksa, and H. R. Maier, "Neural network based stochastic design charts for settlement prediction," *Canadian Geotechnical Journal*, vol. 42, no. 1, pp. 110–120, Feb. 2005, doi: 10.1139/t04-096.

[12] Y.-L. Chen, R. Azzam, and F.-B. Zhang, "The displacement computation and construction pre-control of a foundation pit in Shanghai utilizing FEM and intelligent methods," *Geotechnical and Geological Engineering*, vol. 24, no. 6, pp. 1781–1801, Dec. 2006, doi: 10.1007/s10706-006-6807-6.

[13] S. Çelik and Ö. Tan, "Determination of preconsolidation pressure with artificial neural network," *Civil Engineering and Environmental Science*, vol. 22, no. 4, pp. 217–231, Dec. 2005, doi: 10.1080/10286600500383923.

[14] B. Alsharari, A. Olenko, and H. Abuel-Naga, "Modeling of electrical resistivity of soil based on geotechnical properties," *Expert Systems with Applications*, vol. 141, Art. no. 112966, Mar. 2020, doi: 10.1016/j.eswa.2019.112966.

[15] E. Momeni, A. Yarivand, M. B. Dowlatshahi, and D. J. Armaghani, "An efficient optimal neural network based on gravitational search algorithm in predicting the deformation of geogrid-reinforced soil structures," *Transportation Geotechnics*, vol. 26, Art. no. 100446, Jan. 2021, doi: 10.1016/j.trgeo.2020.100446.

[16] M. Koopialipoor, A. Fahimifar, E. N. Ghaleini, M. Momenzadeh, and D. J. Armaghani, "Development of a new hybrid ANN for solving a geotechnical problem related to tunnel boring machine performance," *Engineering with Computers*, vol. 36, no. 1, pp. 345–357, Jan. 2020, doi: 10.1007/s00366-019-00701-8.

[17] M. Khandelwal *et al.*, "Implementing an ANN model optimized by genetic algorithm for estimating cohesion of limestone samples," *Engineering with Computers*, vol. 34, no. 2, pp. 307–317, Apr. 2018, doi: 10.1007/s00366-017-0541-y.

[18] Y. Erzin and Y. Tuskan, "Prediction of standard penetration test (SPT) value in Izmir, Turkey using radial basis neural network," *Celal Bayar Üniversitesi Fen Bilimleri Dergisi*, Jun. 2017, doi: 10.18466/cbayarfbe.319912.

[19] B. Tarawneh, "Predicting standard penetration test N-value from cone penetration test data using artificial neural networks," *Geoscience Frontiers*, vol. 8, no. 1, pp. 199–204, Jan. 2017, doi: 10.1016/j.gsf.2016.02.003.

[20] N. Akca, "Correlation of SPT–CPT data from the United Arab Emirates," *Engineering Geology*, vol. 67, no. 3–4, pp. 219–231, Jan. 2003, doi: 10.1016/S0013-7952(02)00181-3.

[21] P. Mayne, "In-situ test calibrations for evaluating soil parameters," in *Characterisation and Engineering Properties of Natural Soils*, vol. 3–4, Taylor & Francis, 2006, pp. 1601–1652.

[22] S. M. Ahmed, S. W. Agaiby, and A. H. Abdel-Rahman, "A unified CPT–SPT correlation for non-crushable and crushable cohesionless soils," *Ain Shams Engineering Journal*, vol. 5, no. 1, pp. 63–73, Mar. 2014, doi: 10.1016/j.asej.2013.09.009.

[23] A. Shahri, C. Juhlin, and A. Malemir, "A reliable correlation of SPT-CPT data for southwest of Sweden.," *electronic Journal Of Geotechnical Engineering*, 2014.

[24] M. I. Lingwanda, S. Larsson, and D. L. Nyaoro, "Correlations of SPT, CPT and DPL data for sandy soil in Tanzania," *Geotechnical and Geological Engineering*, vol. 33, no. 5, pp. 1221–1233, Oct. 2015, doi: 10.1007/s10706-015-9897-1.

[25] M. D. dos Santos and K. V. Bicalho, "Proposals of SPT-CPT and DPL-CPT correlations for sandy soils in Brazil," *Journal of Rock Mechanics and Geotechnical Engineering*, vol. 9, no. 6, pp. 1152–1158, Dec. 2017, doi: 10.1016/j.jrmge.2017.08.001.

[26] M. Alam *et al.*, "Empirical SPT-CPT correlation for soils from Lahore, Pakistan," *IOP Conference Series: Materials Science and Engineering*, vol. 414, no. 1, p. 012015, Sep. 2018, doi: 10.1088/1757-899X/414/1/012015.

BIOGRAPHIES OF AUTHORS

Soewignjo Agus Nugroho ⓘ 🔍 SC Ⓟ is a Graduate from Gadjah Mada University Majoring in Geotechnical Engineering. Head of Soil Mechanics Laboratory Universitas Riau 2012 to 2014. Coordinator of geotechnical engineering research since 2010 at civil engineering department at same office. He is currently as senior researcher and associate professor with expert experience in soil investigation and machine learning. Recent decades research on slope stability, geo-disaster, soil stability, improvement and machine learning. He can be contacted at email: nugroho.sa@eng.unri.ac.id.

Hendra Fernando ⓘ 🔍 SC Ⓟ is a graduate of civil engineering from the University of Riau, Indonesia. He is currently as a research and assistant professor in geotechnical engineering. He can be contacted at email: hendra.fernando6188@student.inri.ac.id

Reni Suryanita ⓘ 🔍 SC Ⓟ is a currently as Associate Professor in Civil Engineering Department, Engineering Faculty, Universitas Riau, Pekanbaru, Indonesia. Her undergraduate education was completed in the Civil Engineering Department of Andalas University in 1996. Meanwhile, she got master's degree in 1998 in the Department of Civil Engineering, Bandung Institute of Technology. Completed Doctoral at Universiti Teknologi Malaysia in 2014. Her research interest includes artificial intelligent, bridge engineering, structure engineering and earthquake engineering. She can be contacted at email: reni.suryanita@eng.unri.ac.id.

Optimal economic dispatch using particle swarm optimization in Sulselrabar system

Marhatang[1], Muhammad Ruswandi Djalal[2]
[1]Energy Conversion Study Program, Department of Mechanical Engineering, State Polytechnic of Ujung Pandang, Makassar, Indonesia
[2]Energy Power Plant Study Program, Department of Mechanical Engineering, State Polytechnic of Ujung Pandang, Makassar, Indonesia

Article Info	ABSTRACT
	In this study, a particle swarm optimization (PSO) is proposed to optimize the cost of generating thermal plants in the South Sulawesi system. The study was con ducted by analyzing several methods using the lagrange and ant colony optimization (ACO). PSO algorithm converges on the 11th iteration algorithm with the lowest generation cost obtained, which is Rp129687962.17/hour. While the ACO algorithm converges on the 34th iteration with a generation cost of Rp131,473,269.39/hour. The results of optimization using PSO produce a total thermal power of 400.75 MW and losses of 26.15 MW. The PSO method is able to reduce the cost of generating the South Sulawesi system by Rp11,118,312.07/hour or 7.9%. While using the ACO method generates a generation cost of Rp131,473,269.39/hour to generate power of 400,812 MW with losses of 26,219 MW. The ACO method is able to reduce the cost of generating the South Sulawesi system by Rp9,333,004.9/hour or 6.62%. PSO algorithm provides the lowest cost calculation of generator compared with conventional methods and ACO smart methods. This is also shown in the calculation process, the PSO method completes calculations faster than the ACO method.

Corresponding Author:

Marhatang
Energy Conversion Study Program, Department of Mechanical Engineering, State Polytechnic of Ujung Pandang
Jl. Perintis Kemerdekaan KM. 10, Makassar 90245, Indonesia
Email: marhatang@poliupg.ac.id

1. INTRODUCTION

In a power plant center, good management is needed in regulating the loading and the amount of power that must be sent by the generator into the system. Operational management at the power plant is very important, especially in thermal plants that operate with fuel to drive turbines. Changes in the load on the electric power system will encourage additional fuel consumption per unit time in thermal power plants to produce electricity, this is commonly referred to as the input-output characteristics of the generator. The increasingly expensive fuel costs in thermal generators need to be optimized for use, so that minimum costs are obtained [1].

The Sulselrabar system operates at 150 kV and consists of 46 transmission lines. This system connects load centers in Sulselrabar such as the capital city of Makassar, Pare-Pare and several districts such as Maros, Pangkep, and others. The Sulselrabar system has 37 buses and consists of several thermal and non-thermal generators. For thermal generators there are 12 generating units and 4 hydro or non-thermal generating units. The Sulselrabar system is a large system so a comprehensive study is needed so that it can support the system's performancev [2].

In this research, a study on the South Sulawesi electricity system will be proposed, namely economic dispatch. In previous studies, previous economic dispatch studies have been conducted. Some of these studies produce a combination of economic loading for generating units in the South Sulawesi system. But the development of the system and also the emergence of several new optimization methods, we need a further study of economic dispatch.

Particle swarm optimization (PSO) method is an undeterministic method or smart method. PSO is an evolutionary computational technique, in which the population in the PSO is based on an algorithm search and begins with a random population called a particle. The application of PSO as an economic dispatch optimization method has been investigated by [3]–[7]. The study discusses the implementation of the PSO algorithm to solve economic dispatch problems. Previous research on economic dispatch in the South Sulawesi system has been conducted. In research [8] discusses economic dispatch using the PSO method. From the results of the study the generator data used did not cover all of the generators in the system, especially thermal plants. The Sulselrabar system has now developed with the addition of several thermal plants. In research [9], the implementation of the ant colony algorithm is explained as an optimization method. From the results of this study obtained the old ant colony computing process. In addition, the optimization results obtained are not so significant with the calculation process using the conventional lagrange method. In [4], [6] discusses the implementation of the PSO in calculating the cost of generation in a multi-engine system, and shows optimal results in tuning. In research [5] discusses the optimization of generation costs in small systems with a case study of a 9 bus system and 3 generators. Research on small scale systems has also been carried out by [10], who discusses the implementation of economic dispatch in microgrid systems.

This research will propose a new approach for optimization of generation costs in large multinational systems, especially in the South Sulawesi system. In this study, we consider transmission losses and the equality and inequality limits of the generator [3]. The final result of this research is the optimization of optimal power generation so that the cheapest generation costs are obtained.

2.   RESEARCH METHOD

Optimal operation of the generator must pay attention to equality constraints and inequality constraints. Equality constraint is the power limit generated by each generator which is equal to the total load requirement and transmission losses, expressed by the following (1) [11]. Loss coefficients can be considered constant for changes in the output power of each generator in the system.

$$\sum_{i=1}^{N} P_i = P_R + P_L \tag{1}$$

Where:

Pi   = Generator output power (MW)
PR   = Total load (MW)
PL   = Transmission losses (MW)

while the inequality constraint is the output power produced by the generator that must be greater than or equal to the minimum permitted power and less than or equal to the maximum permitted power [12]

$$P_{i\,min} \leq P_i \leq P_{i\,max} \tag{2}$$

$$P_L = \sum_{i=1}^{N} \sum_{j=1}^{N} P_i B_{ij} P_j + \sum_{i=1}^{N} B_{i0} P_j + B_{00} \tag{3}$$

where:

P_L: Losses.
B_{ij}: Losses coefficients.
P_i, P_i: Generator output
$B_{i\,0}$, B_{00}: Losses constant

2.1. Particle swarm optimization

Particle swarm optimization (PSO) is an artificial intelligence method that was discovered in 1995 [13], [14]. This algorithm works by adopting the movement behavior of a flock of birds or fish in search of food so that it can be applied to scientific and engineering research methods. The main advantages of the

PSO algorithm are simple algorithm structure, easy to use, easy to set parameters, and very good efficiency [10], [15]. The weight improvement function is determined by the following (4).

$$w(t) = (w_{max} - w_{\min})x\frac{iterasi_{max} - iterasi(t)}{iterasi_{max}} + w_{min}$$

(4)

Where:

W(t)	: Weight
W$_{max}$	: Maximum weight value
W$_{min}$	: Minimum weight value
Iter$_{max}$	: Maximum Iteration
Iter(t)	: Iteration

Inertia weight values are usually set between 0.4 and 0.9. The concept of inertia weight was developed by Shi and Eberhart in 1998 which inspired the modification of particle velocity and position using the adjustable inertia weight parameter. Velocity and particle position equation [16], [17]:

$$V_{ij}^t = \omega x V_{ij}^{t-1} + c_1 x r_1 x \left(P_{best_{ij}}^{t-1} - X_{ij}^{t-1}\right) + c_2 x r_2 x\left(G_{best_i}^{t-i} - X_{ij}^{t-1}\right)$$

(5)

For I=1,2,…, N$_D$; j=1,….,N$_{par.}$
Where:

t	: Calculate iteration
V_{ij}^t	: The ij dimension of the particle velocity at iteration t.
X_{ij}^t	: The ij dimension of the particle position in the iteration t.
ω	: Weight of inertia
c$_1$, c$_2$	: Positive acceleration coefficient
$P_{best_{ij}}^{t-1}$	: The ij dimension from the best position is reached until iteration t-1
$G_{best_{ij}}^{t-1}$	: Dimension I of all best positions is achieved until t-1 iteration
N$_D$	: Number of decision variables
N$_{par}$	: Number of swarm
r$_1$, r$_2$	: Random numbers are evenly distributed in the range [0,1]; the latest value is generated at any time.

PSO was developed based on the following model [18]:
a. When a bird approaches a food source, it will quickly send information to other birds.
b. After receiving the information, the other birds follow in groups.

PSO parameters used include [19]:
− Number of swarms=30
− Number of variables=16
− Maximum iteration=50
− Social constant=0.5
− Cognitive constant=0.01
− Inertia (w)=0.01

While the ant colony optimization (ACO) parameters include [20], [21]:
− Number of ants=10
− Max Iteration=100
− Alpha=0.9

Research begins by collecting system data. Then make a modeling of the Sulselrabar system to be integrated with the PSO algorithm. Then make PSO modeling in MATLAB software. Figure 1 shows the flowchart of the research conducted.

3. RESULTS AND ANALYSIS

In this study, the completion of economic dispatch uses several methods including the conventional Lagrange method, the ant colony optimization (ACO) method, and the proposed method, namely particle swarm optimization (PSO). The case study used is the Sulselrabar system. The algorithm performs computations to calculate the cheapest combination of thermal generation. In this study, 4 non-thermal power plants were maximized. The PSO algorithm works with the lowest cost generation objective function. The

solution begins by calculating the input-output characteristics of the generator with the following (6) [22], [23].
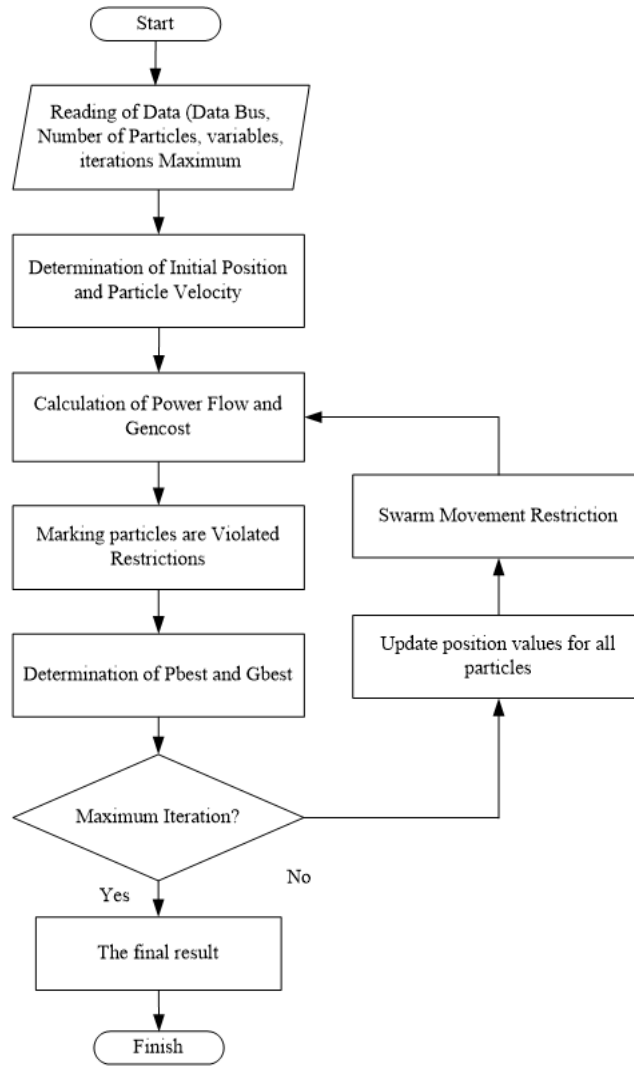
Figure 1. Particle swarm optimization flowchart

$$C_t = \sum_{i=1}^{n_g} \alpha_i + \beta_i P_i + \gamma_i P_i^2 \qquad (6)$$

To obtain stable generator performance, the operation of the generator should not exceed or be less than the generator capacity [24]. Therefore, the generator power production must be limited by the equality constraint as shown in (7). In addition, it must also pay attention to the limits of the inequality constraint as shown in (8) [25].

$$\sum_{i=1}^{n_g} P_i = P_D \qquad (7)$$

$$P_{i_{min}} \leq P_i \leq P_{i_{max}} \qquad (8)$$

3.1. Thermal generator input-output and fuel cost characteristics

The computational process begins by calculating the input-output characteristics of the thermal generator [26]. Then determine the fuel cost equation by multiplying the input-output equation by the price of the fuel used. The results of the calculation of the input output characteristics and cost functions are shown in Table 1 [9].

Table 1. IO characteristics and cost function

No	Unit	Input-Output Equation (Liter/Jam)	Input-Output Equation (Liter/Jam)
1	PLTD Pare-Pare	$714.0000+567.4000P-3.2941P^2$	$6211800+4936380P-28658.67P^2$
2	PLTD Suppa	$2070+178.6P+0.4P^2$	$18009000+1553820P+3480P^2$
3	PLTU Barru	$2805.6+251.6P-0.11976P^2$	$17675280+1585080P+754.488P^2$
4	PLTU Tello	$558+174.5P+1.375P^2$	$3515400+1099350P+8662.5P^2$
5	PLTD Agrekko/T.Lama	$771.975+160P+2.7397P^2$	$6716182.5+1392000P+23835.39P^2$
6	PLTD Sgmnsa	$617.625+477.25P-4.1667P^2$	$5373337.5+4152075P-36250.29P^2$
7	PLTD Arena/Jeneponto	$629.475+176.3P+4.8052P^2$	$5476432.5+1533810P+41805.24P^2$
8	PLTD Matekko/Bulukumba	$506.25+124.9P+9.4444P^2$	$4404375+1086630P+82166.28P^2$
9	PLTD Pajelasang/Soppeng	$432+66.2P+12.5P^2$	$3758400+575940P+108750P^2$
10	PLTGU Sengkang	$4418.89+38.0952P+0.021898P^2$	$27839000.000+240000.00P+137.9539P^2$
11	PLTD Malea/Makale	$165.75+409.5P+5.7692P^2$	$1442025+3562650P+50192.04P^2$
12	PLTD Palopo	$103.5+112.4P+50P^2$	$900450+977880P+435000P^2$

3.2. Analysis and discussion

The case study used is based on previous research, in which the settlement method uses an intelligent ant colony optimization (ACO) algorithm and the conventional Lagrange. Table 2 shows the real generation power and costs for the thermal unit of the South Sulawesi system at peak evening load before optimization and the comparison of the results of simulations carried out using the proposed method namely particle swarm optimization (PSO) algorithm, then compared with the ACO algorithm, and the Lagrange. The graph of convergence optimization of generation costs using PSO is shown in Figure 2.

Table 2. The complete optimization results

| Unit	Real		Lagrange		Ant Colony		PSO	
	P (MW)	Cost (Rp/hr)	P (MW)	Cost (Rp/hr)	P (MW)	Cost (Rp/hr)	P (MW)	Cost (Rp/hr)
1	20.1	9385464.873	19.40	9119159.49	18.50	8772640.01	10.50	5488737.86
2	62.2	12812016.72	31.98	7125923.05	60.03	12382534.57	59.73	12323912.16
3	44.7	10360370.52	44.00	10202568.76	41.40	9622921.45	33.55	7935567.73
4	29.7	4380719.96	19.80	2867857.65	17.80	2582845.65	28.38	4169183.92
5	19.3	4246022.69	19.00	4176875.82	21.88	4858396.36	18.39	4036545.83
6	12.3	5095955.36	27.60	9235658.65	29.20	9570548.02	19.08	7140357.97
7	19.6	5159900.95	23.86	6587284.55	13.36	3342993.46	18.66	4864960.35
8	9.0	2083951.36	6.30	1451132.36	11.94	2909265.78	9.35	2175760.65
9	15.1	3725118.15	14.56	3519839.04	11.52	2482548.48	11.24	2395951.36
10	192.9	79268321.18	184.38	76780078.63	166.10	71509642.44	188.35	77937456.65
11	3.5	1452615.24	3.730	1542902.63	3.520	1460445.24	2.02	885723.75
12	6.9	2835817.20	6.060	2280116.88	5.560	1978487.88	1.50	333803.93
Total	435.3	140806274.24	400.67	134889397.56	400.81	131473269.39	400.75	129687962.17
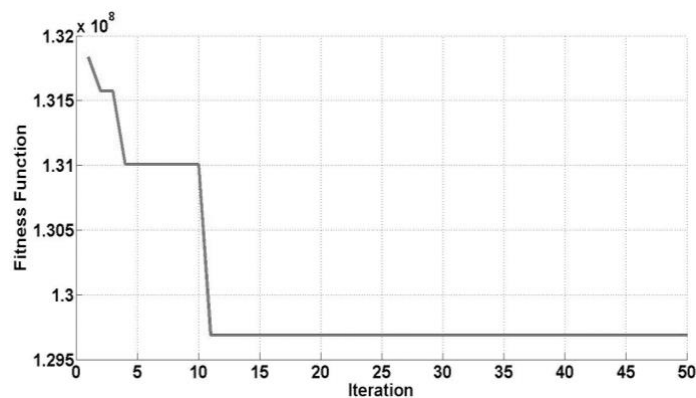
Figure 2. Graph of convergence optimization using particle swarm optimization

Table 2 is a preliminary study of the calculation of generator costs before being optimized in a case study of peak evening loads, the generation load charged to the thermal unit of 435.3 MW, with a total generation cost of Rp140,806,274.24. Losses generated before optimization were 26,299 MW. The total system load is 532.3 MW. 4 Hydro power plant units bear respectively: Bakaru 126 MW, Pinrang 0.3 MW, Sinjai 3.5 MW, Bili-Bili 7.1 MW. Furthermore, by using the proposed method that is using a smart method based on PSO algorithm, the generation results are more optimal than the smart method based on ACO algorithm. As another comparison method in this research also used the conventional Lagrange. The complete optimization results are shown in Table 2.

3.3. Analysis

In the condition before optimization, the total cost of generation is Rp140,806,274.24/hour with a power of 435.3 MW and losses of 26,299 MW. The first optimization is done using conventional lagrange and generates a generation cost of Rp134,889,397.56/hour with a power of 400.67 MW and losses of 28.352 MW. From the results of these calculations, the cost of generation can be reduced to Rp5,916,876.7/hour or 4.2% at night peak load. The most expensive generation costs are obtained from the Sengkang *Pembangkit Listrik Tenaga Gas* (PLTGU) unit, which in the system functions as a slack bus, which is Rp76,780,078,632/hour, with a power of 184,380 MW. While the cheapest generation costs are obtained from the Matekko *Pembangkit Listrik Tenaga Diesel* (PLTD) unit, which is Rp1,451,132,365/hour, with a power of 6.3 MW.

Next, use the ant colony optimization (ACO) method. From the calculation results [9], CO converges on the 34th iteration with a generation cost of Rp131,473,269.39/hour. From the computational results, ACO generates a generation cost of Rp131,473,269.39/hour with a power of 400,812 MW with losses of 26,219 MW. From these results, ACO was able to reduce the generation cost of Rp9.333.004.9/hour or 6.62%. The most expensive generating unit of the Sengkang *Pembangkit Listrik Tenaga Gas* (PLTGU) produces the most expensive thermal generation cost of Rp71,509,642,449/hour, with a power of 166,102 MW. While the cheapest thermal generating unit at the *Pembangkit Listrik Tenaga Diesel* (PLTD) Malea Makale plant is Rp1,460,445,245/hour, with a power of 3,520 MW.

Figure 2 is a graph of the convergence of calculations using PSO. The calculation process is carried out for 50 iterations, and in the 11th iteration the PSO algorithm has found the cheapest generation cost, which is Rp129,687,962.17/hour with a power of 400.75 MW and losses of 26.15 MW. From these results, PSO was able to reduce the generation cost of Rp11,118,312.07/hour or 7.9%. The Sengkang PLTGU generating unit produces the most expensive thermal generation costs, namely Rp77,937,456.65/hour, with a power of 188.35 MW. While the cheapest thermal generating unit at the Palopo PLTD plant is Rp333.803.93/hour, with a power of 1.5 MW.

From the results of the analysis, the PSO algorithm gives the cheapest generation cost calculation results compared to the conventional method and the smart ACO method. This is also shown in the calculation process, the PSO method is faster in completing the calculation than the ACO method. Losses produced by the PSO method are smaller than other methods. Figure 3 shows a comparison of the generation costs at night peak load for each method.
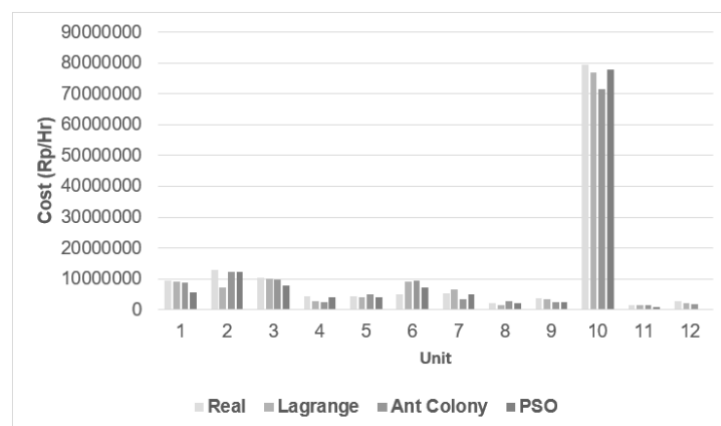

Figure 3. Comparison of the costs of generating peak evening loads

4. CONCLUSION

From the analysis results, the particle swarm optimization (PSO) computation process algorithm converges on the 11th iteration, where the PSO algorithm has found the cheapest generation cost of Rp129,687,962.17/hour. While the ant colony optimization (ACO) algorithm converges on the 34th iteration with a generation cost of Rp131,473,269.39/hour. From the results of the analysis using the conventional lagrange of generation costs Rp134,889,397.56/hour with active power 400.67 MW and losses of 28.352 MW. The ACO generates a total generation cost of Rp131,473,269.39/hour with active power of 400,812 MW and losses of 26,219 MW. Using the PSO method generates a total generation cost of Rp129,687,962.17/hour with active power 400.75 MW and losses of 26.15 MW. The lagrange is able to reduce the cost by Rp5,916,876.7/hour or 4.2%. Ant Colony is able to reduce the cost by Rp9,333,004.9/hour or 6.62%. PSO is able to reduce the cost by Rp11,118,312.07/hour or 7.9%.

REFERENCES

[1] M. R. Djalal, A. Imran, and I. Robandi, "Optimal placement and tuning power system stabilizer using participation factor and imperialist competitive algorithm in 150 kV South of Sulawesi system," in *2015 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, May 2015, pp. 147–152, doi: 10.1109/ISITIA.2015.7219970.

[2] M. Y. Yunus, M. R. Djalal, and M. Marhatang, "Optimal design power system stabilizer using firefly algorithm in interconnected 150 kV Sulselrabar System, Indonesia," *International Review of Electrical Engineering (IREE)*, vol. 12, no. 3, Jun. 2017, doi: 10.15866/iree.v12i3.11136.

[3] A. Mahor, V. Prasad, and S. Rangnekar, "Economic dispatch using particle swarm optimization: a review," *Renewable and Sustainable Energy Reviews*, vol. 13, no. 8, pp. 2134–2141, Oct. 2009, doi: 10.1016/j.rser.2009.03.007.

[4] H. M. Z. Iqbal, A. Ashraf, and A. Ahmad, "Power economic dispatch using particle swarm optimization," Jun. 2015, doi: 10.1109/PGSRET.2015.7312202.

[5] M. Murtadha Othman, M. Affendi Ismail Salim, I. Musirin, N. Ashida Salim, and M. Lutfi Othman, "Dynamic economic dispatch assessment using particle swarm optimization technique," *Bulletin of Electrical Engineering and Informatics*, vol. 7, no. 3, pp. 458–464, Sep. 2018, doi: 10.11591/eei.v7i3.1278.

[6] M. N. Alam, "State-of-the-art economic load dispatch of power systems using particle swarm optimization," Dec. 2018. *ArXiv:1812.11610.*

[7] V. K. Jadoun, N. Gupta, K. R. Niazi, and A. Swarnkar, "Nonconvex economic dispatch using particle swarm optimization with time varying operators," *Advances in Electrical Engineering*, vol. 2014, pp. 1–13, Oct. 2014, doi: 10.1155/2014/301615.

[8] S. Humena, S. Manjang, and I. C. Gunadin, "Optimization economic power generation using modified improved PSO algorithm methods," *Journal of Theoretical and Applied Information Technology*, vol. 93, no. 2, pp. 522–530, 2016.

[9] Tasrif, Suyono, Hadi, and R. Nur, "Economic dispatch in 150 KV sulselrabar electrical system using ant colony optimization," *IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE)*, vol. 13, no. 3, pp. 28–35, 2018, doi: 10.9790/1676-1303022835.

[10] D. McLarty, N. Panossian, F. Jabbari, and A. Traverso, "Dynamic economic dispatch using complementary quadratic programming," *Energy*, vol. 166, pp. 755–764, Jan. 2019, doi: 10.1016/j.energy.2018.10.087.

[11] G. Xiong *et al.*, "A novel method for economic dispatch with across neighborhood search: a case study in a provincial power grid, China," *Complexity*, vol. 2018, pp. 1–18, Nov. 2018, doi: 10.1155/2018/2591341.

[12] Z. Younes, I. Alhamrouni, S. Mekhilef, and M. Reyasudin, "A memory-based gravitational search algorithm for solving economic dispatch problem in micro-grid," *Ain Shams Engineering Journal*, vol. 12, no. 2, pp. 1985–1994, Jun. 2021, doi: 10.1016/j.asej.2020.10.021.

[13] M. Imran, R. Hashim, and N. E. A. Khalid, "An overview of particle swarm optimization variants," *Procedia Engineering*, vol. 53, pp. 491–496, 2013, doi: 10.1016/j.proeng.2013.02.063.

[14] G. Pereira, "Particle swarm optimization," *INESCID and Instituto Superior Tecnico*, vol. 15, 2011.

[15] M. F. Aranza, J. Kustija, B. Trisno, and D. L. Hakim, "Tunning PID controller using particle swarm optimization algorithm on automatic voltage regulator system," *IOP Conference Series: Materials Science and Engineering*, vol. 128, no. 1, Apr. 2016, doi: 10.1088/1757-899X/128/1/012038.

[16] M. Saini, A. M. Shiddiq Yunus, and M. R. Djalal, "Optimal PSS design using particle swarm optimization under load shedding condition," in *Proceedings - 2020 International Seminar on Intelligent Technology and Its Application: Humanification of Reliable Intelligent Systems, ISITIA 2020*, Jul. 2020, pp. 405–410, doi: 10.1109/ISITIA49792.2020.9163779.

[17] Z. Qi, Q. Shi, and H. Zhang, "Tuning of digital PID controllers using particle swarm optimization algorithm for a CAN-based DC motor subject to stochastic delays," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 7, pp. 5637–5646, Jul. 2020, doi: 10.1109/TIE.2019.2934030.

[18] H. M. Salman, A. K. M. Al-Qurabat, and A. A. Riyadh Finjan, "Bigradient neural network-based quantum particle swarm optimization for blind source separation," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 2, pp. 355–364, Jun. 2021, doi: 10.11591/ijai.v10.i2.pp355-364.

[19] M. I. Solihin, L. F. Tack, and M. L. Kean, "Tuning of PID controller using particle swarm optimization (PSO)," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 1, no. 4, 2011, doi: 10.18517/ijaseit.1.4.93.

[20] Y. Dhieb, M. Yaich, A. Guermazi, and M. Ghariani, "PID controller tuning using ant colony optimization for induction motor," *Journal of Electrical Systems*, vol. 15, no. 1, pp. 133–141, 2019.

[21] I. Chiha, N. Liouane, and P. Borne, "Tuning PID controller using multiobjective ant colony optimization," *Applied Computational Intelligence and Soft Computing*, vol. 2012, pp. 1–7, 2012, doi: 10.1155/2012/536326.

[22] K. Ma, C. Wang, J. Yang, Q. Yang, and Y. Yuan, "Economic dispatch with demand response in smart grid: Bargaining model and solutions," *Energies*, vol. 10, no. 8, p. 1193, Aug. 2017, doi: 10.3390/en10081193.

[23] J. Zhang, J. Zhang, F. Zhang, M. Chi, and L. Wan, "An improved symbiosis particle swarm optimization for solving economic load dispatch problem," *Journal of Electrical and Computer Engineering*, vol. 2021, pp. 1–11, Jan. 2021, doi: 10.1155/2021/8869477.

[24] N. M. Azkiya, A. G. Abdullah, and H. Hasbullah, "Economic dispatch and operating cost optimization for thermal power in 500 KV system using genetic algorithm (GA)," *IOP Conference Series: Materials Science and Engineering*, vol. 434, no. 1, Dec. 2018, doi: 10.1088/1757-899X/434/1/012013.

[25] B. Huang, C. Zheng, Q. Sun, and R. Hu, "Optimal economic dispatch for integrated power and heating systems considering transmission losses," *Energies*, vol. 12, no. 13, Jun. 2019, doi: 10.3390/en12132502.

[26] B. Dey, B. Bhattacharyya, and F. P. G. Márquez, "A hybrid optimization-based approach to solve environment constrained economic dispatch problem on microgrid system," *Journal of Cleaner Production*, vol. 307, Jul. 2021, doi: 10.1016/j.jclepro.2021.127196.

BIOGRAPHIES OF AUTHORS

Marhatang was born in Soppeng-Indonesia on November 17, 1974. He received bachelor degree from Electronic Engineering Polytechnic Institute of Surabaya (Surabaya, Indonesia), majors in Electrical Engineering in 2002. Then, master degree from Hasanuddin University (Makassar, Indonesia), majors in Electrical Engineering in 2012. His research about, Power Electronic, Renewable Energy, and Electrical Power System. Now, He is lecturer at State Polytechnic of Ujung Pandang (PNUP). He can be contacted at email: marhatang@poliupg.ac.id.

Muhammad Ruswandi Djalal was born in Makassar-Indonesia on March 11, 1990. He received bachelor degree from State Polytechnic of Ujung Pandang (Makassar, Indonesia), majors in Energy Generation engineering in 2012. Then, master degree from Sepuluh Nopember Institute of Technology, (ITS Surabaya, Indonesia), majors in Power System Engineering in 2015. His research about, Power System Operation and Control, Renewable Energy and Artificial Intelligent. Now, He is lecturer at State Polytechnic of Ujung Pandang (PNUP). He can be contacted at email: wandi@poliupg.ac.id.

Image fusion by discrete wavelet transform for multimodal biometric recognition

Arjun Benagatte Channegowda[1], Hebbakavadi Nanjundaiah Prakash[2]
[1]Department of Information Science and Engineering, Rajeev Institute of Technology, Hassan, Visvesvaraya Technological University, Belagavi, Karnataka, India
[2]Department of Computer Science and Engineering, Rajeev Institute of Technology, Hassan, Visvesvaraya Technological University, Belagavi, Karnataka, India

Article Info

ABSTRACT

In today's world, security plays a crucial role in almost all applications. Providing security to a huge population is a more challenging task. Biometric security is the key player in such type of situation. Using a biometric-based security system more secure application can be built because it is tough to steal or forge. The unimodal biometric system uses only one biometric modality where some of the limitations will arise. For example, if we use fingerprints due to oiliness or scratches, the finger recognition rate may reduce. In order to overcome the drawbacks of unimodal biometrics, Multimodal biometric systems were introduced. In this paper, new multimodal fusion methods are proposed, where instead of merging features, database images are fused using discrete wavelet transform (DWT) technique. Face and signature images are fused, features are extracted from the fused image, an ensemble classifier is used for classification, and also experiments are conducted for finger vein and signature images.

Corresponding Author:

Arjun Benagatte Channegowda
Department of Information Science and Engineering, Rajeev Institute of Technology, Visvesvaraya Technological University
Banglore-Manglore highway, Hassan, Karnataka, India
Email: bc.arjun@gmail.com

1. INTRODUCTION

The multimodal biometric systems give better results when compared to unimodal biometric systems. In multimodal biometric, more information is available compared to unimodal biometric. More than one biometric information is available in multimodal biometrics [1]–[3]. But designing the multimodal biometric system is a challenging task since to fuse the information of more than one biometric modalities, there will be a lot of compatibility issues. If two biometric modalities features are fused, then there may be chances of unwanted features stored as a template, or most important data may lose due to feature compatibility like data types. To overcome such limitations in multimodal biometrics raw image fusion is used, in the literature [4]–[6] we observed that discrete wavelet transform (DWT) image fusion techniques is not used on multimodal biometric recognition. The biometric fusion at the feature and sensor level is noticeable from the many research work [7]–[10]. In this paper new model is proposed where without extracting the features, raw images are fused using the discrete wavelet transform method, and experiments are conducted on Histogram oriented gradient and uniform local binary pattern features [11]. In discrete wavelets, transform images are decomposed into different levels, and new coefficients are generated. This

co-efficient is combined or merged to form a new co-efficient that holds the information of both images [12]. And in the next stage, inverse -discrete wavelet transform is applied to get back the fused image.

2.   RESEARCH METHODOLOGY

In discrete wavelets transform, images are fused by converting them into wavelets, as shown in Figure 1. In order to fuse the images, more than one image is required. These images can be either with the same resolution or a different resolution. Two different methods are used to fuse, as shown in Figures 1(a) and (b).

Figure 1(a) is used to fuse the images of the same resolution, and Figure 1(b) is used to fuse the image with a different resolution [12]. In the initial stage of sampling and registration, the technique is applied to make sure; images are of the same size and registered to the same sensor. After completion of sampling and registration process DWT is applied to images in order to generate coefficient values for very individual pixel of input image. These coefficients are merged to form new coefficient values. In the next step, inverse DWT is applied to a new co-efficient to obtain the fused image. The information of both input images and present in final fused image. Figure 1(a) shows the transformation of image fusion using wavelets for the same resolution image, and Figure 1(b) shows the transformation of image fusion using wavelets for different resolution images. The top left corner shows the levels of decomposition of an image [6], [12].

(a)   (b)
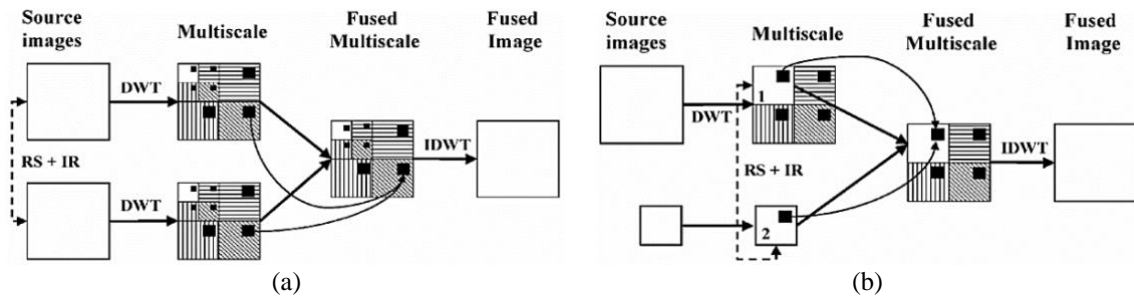
Figure 1. Discrete wavelets transform for (a) method 1 and (b) method 2

3.   MULTI-RESOLUTION DISCRETE WAVELET TRANSFORM (DWT) FOR PROPOSED WORK

This paper used multi-resolution wavelet fusion, face, signature, and finger vein images with different resolution images to combine. We used a combination of face and signature also finger vein and signature for experiments. Consider a face sample in multi-resolution wavelet transform shown in Figure 2, shows the stage 1 wavelet transform and Figure 3 shows the 1st, 2nd and 3rd degree decomposition levels of face image where low pass and high pass filters are applied for each row and column, and four sub-images are created. These four images represent the different decomposed degrees [6], [12].
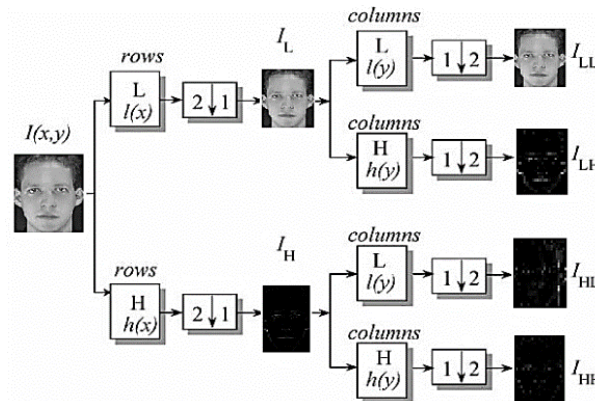
Figure 2. Stage 1 wavelet transform
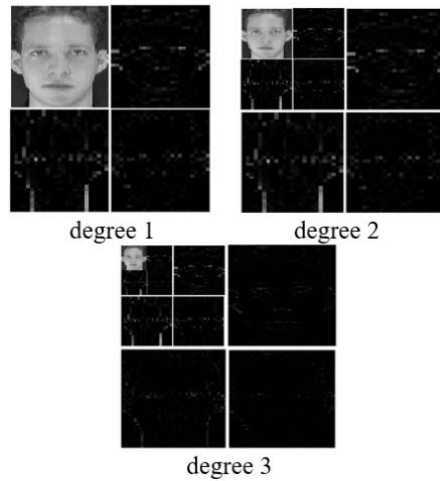
Figure 3. First, second, and third degree decomposition levels

After stage 1, with the 3-degree decomposition of the face image, Figure 3 shows the details of 3-degree wavelet decomposition, where the top left corner shows the decomposition degrees. After applying stage 1 transform, the inverse transformation of wavelet is applied to co-efficient, which are obtained as the output of stage 1. In stage 2, inverse DWT is applied, and the final merged image is obtained. The detailed steps are showed in Figure 4. Figure 5 gives the detailed steps of stage 1 and stage 2 transformations with notations.
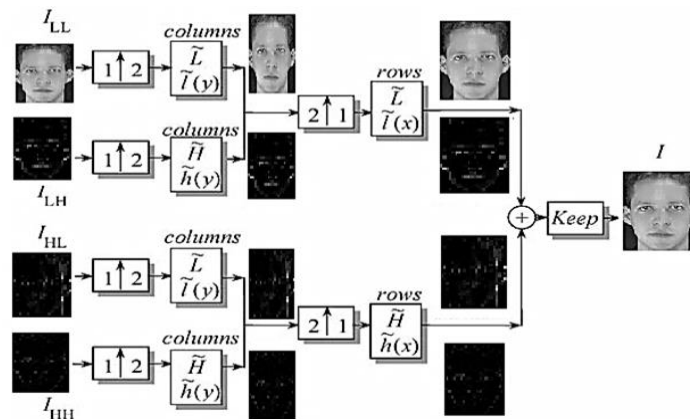
Figure 4. Stage 2 wavelet transform
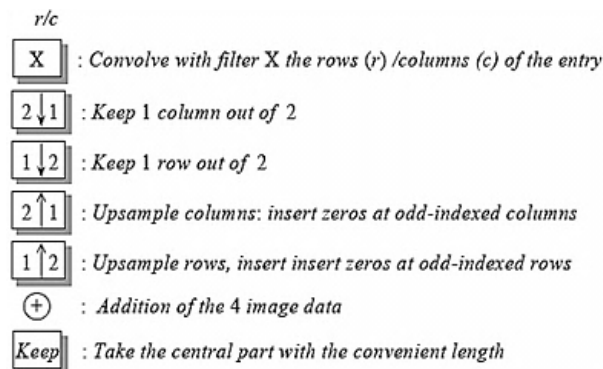
Figure 5. Steps of stage 1 and stage 2 wavelet transformation

4. DATABASE USED FOR EXPERIMENTATION

− Face database: american telephone and telegraph company (AT and T) laboratories Cambridge face database [13] is used, 40 persons, six samples of each individual is used for experimentations.
− Finger vein database: machine learning and data-mining lab, Shandong University (SDUMLA) finger vein database [14] is used. Six samples of the left index finger of the first 40 subjects are used for experimentations.
− Signature database: biometric research lab (ATVS) Madrid Spain's MCYT signature database [13] is used. In this paper, we considered the first 40 subjects/individuals with the first six genuine samples of each individual. Finally 720 samples of face, signature and finger vein are used for experimentation. The samples are mapped one to one for forty individuals. Mapping of data base is commonly used by all the researchers due to non availability of standard data bases, and also it has been showed that the performance of the fusion models will be same when compared with mapped data base and original data base of same person. Figure 6 shows the image samples of MCYT face, SDUMLA finger vein, and MCYT signature is shown in Figure 6(a), Figure 6(b), and Figure 6(c) respectively.

(a)   (b)

(c)
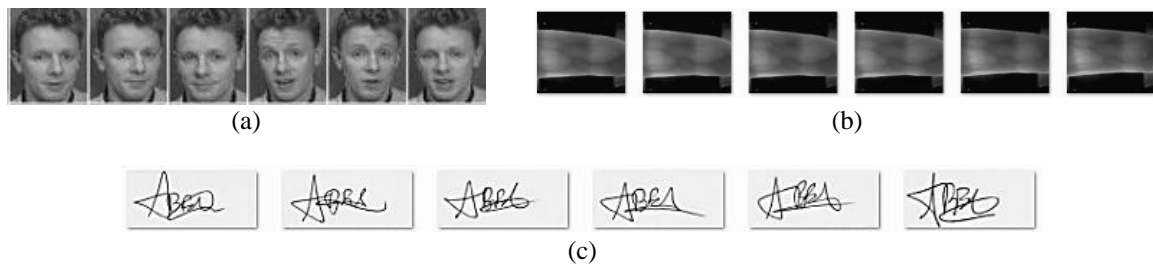
Figure 6. Image samples for (a) MCYT face, (b) SDUMLA finger vein, and (c) MCYT signature

5. PROPOSED IMAGE FUSION TECHNIQUE

In this paper, research work is carried out to increase the recognition rate of multimodal systems over unimodal systems by applying the wavelet fusion technique to multimodal biometrics. In the proposed method, we fused face and signature images directly by fetching the images from the standard database. Experiments are also explored on finger vein and signature database. The fusion of images using DWT for biometric recognition for the face with signature and finger vein with signature for the standard database is a novel approach.

5.1. Fusion of face images and signature images

In the proposed work, standard data samples or images of face and signature are fused using DWT, 240 samples of signatures are fused with 240 samples of face one to one, and new data samples are created. This new database with fused images has information from both signature and face biometric samples. The Figure 7 shows fusion of face and signature and Figure 8 shows the fusion of Finger vein and signature by using both DWT stage 1 and IDWT stage 2 of Figure 2 and Figure 4. The final output image contains a rich information source than the original images.

Figure 7. Wavelet fusion face signature   Figure 8. Wavelet fusion finger vein signature

5.2. Fusion of finger vein images and signature images

In the proposed work, standard data samples of finger vein and signature are fused using DWT, 240 samples of signatures are fused with 240 samples of finger vein one to one, and new data samples are created. This new database with fused images has information from both signature and finger vein biometric samples. Figure 8 shows the fusion of finger vein and signature by using both DWT stage 1 and IDWT stage 2 of Figure 2 and Figure 4. The final output image contains a rich information source than the original images.

5.3. Fusion of finger vein, signature and face

In the proposed work, standard data samples of finger vein, signature and face are fused using DWT, 240 samples of signatures are fused with 240 samples of finger vein and 240 samples of face one to one, and new data samples are created. This new database with fused images has information from signature, finger vein and face biometric samples. Figure 9 shows the fusion of finger vein, signature and face by using both DWT stage 1 and IDWT stage 2 of Figure 2 and Figure 4. The final output image contains a rich information source than the original images.
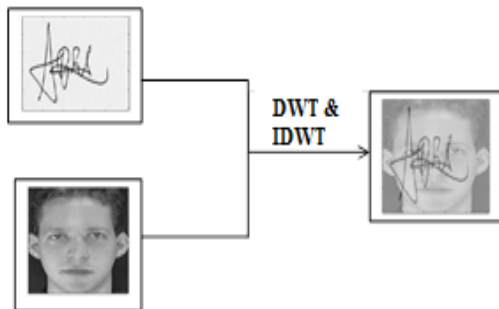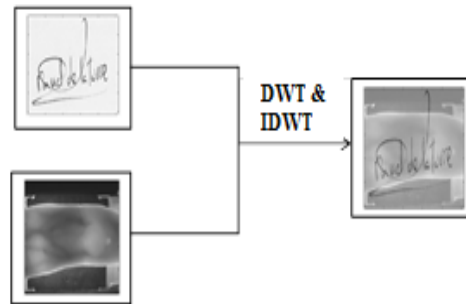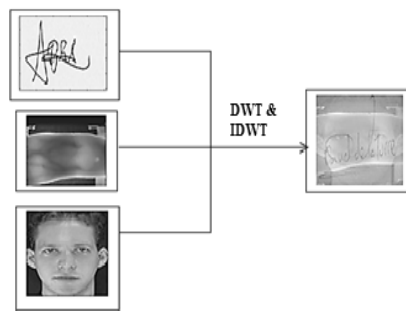

Figure 9. Wavelet fusion finger vein, face, and signature

6. PROPOSED FUSION MODEL

In this paper, image fusion is applied using the wavelet transformation technique, there are many wavelet filters available, and from the literature [15]–[20], it has been observed that the db2 filter is best suited for multi-resolution images. In this paper, we used the db2 wavelet filter for transformation. And it has been identified in the literature that more decomposition is loose the details and less decomposition will overlap; hence degree 5 decompositions is used for our work. We explored the experiments by combining face with signature and finger vein with the signature and finally combining all the three biometrics. All three multimodal biometric recognition models were explored in this paper. Figure 10 shows the proposed model for face, signature and finger vein biometric samples, and the same method is used to implement the face and finger vein model and face and signature model.
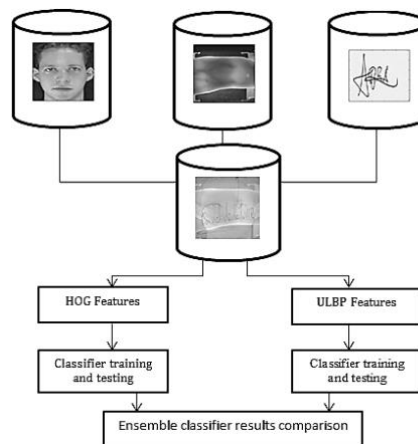
Figure 10. Proposed DWT fusion model

7. EXPERIMENTS AND RESULTS

We identified that image fusion using DWT is used for image enhancement in most of the research work from the literature. In this paper, we used the DWT image fusion technique for multimodal biometric recognition. And in multimodal biometric, not much work is explored on face and signature combination. As per our knowledge, combinations of finger vein and signature with DWT image fusion are novel approaches. All the experiments are conducted on standard databases with larger size data samples. Three experiments are conducted on the proposed model. First experiments on the face and signature biometric modalities were explored. And in the next stage, finger vein and signature biometric modalities have experimented and in last stage face, finger vein and signature are experimented. For the biometric samples of face, signature and finger vein, both unimodal and multimodal biometric investigation is conducted. I to XII represent the experiments in Tables 1 to 5.

7.1. Face and signature fusion experiments

Separate unimodal experiments are conducted on face and signature using uniform local binary pattern (ULBP) features and histogram of oriented gradient (HOG) features [21]–[26]. In the entire case, an ensemble machine learning classifier is used [26]. Unimodal experiment results are shown in Table 1.

Table 1. Unimodal biometric experiments

Unimodal	Biometric	Accuracy of Classifier
I	Face (ULBP)	74%
II	Face (HOG)	82%
III	Signature (ULBP)	85%
IV	Signature (HOG)	91%

In multimodal experiments, fused samples of face and signature are used. From the fused images, ULBP and HOG features are extracted. An ensemble classifier is used for classification [25], [26]. Multimodal experiment results are shown in Table 2.

Table 2. Multimodal experiments

Multimodal	Biometric	Accuracy of Classifier
V	Face Signature (ULBP)	92%
VI	Face Signature (HOG)	94%

7.2. Finger vein and signature fusion experiments

Separate unimodal experiments are conducted on finger vein and signature using ULBP features and HOG features. In the entire case ensemble, a machine learning classifier is used [25], [26]. Unimodal experiments result shown in Table 3.

Table 3. Unimodal biometric experiments

Unimodal	Biometric	Accuracy of Classifier
VII	Finger vein (ULBP)	88%
VIII	Finger vein (HOG)	82%

In multimodal experiments, fused samples of finger vein and signatures are used, from the fused images ULBP, and HOG features are extracted. An ensemble classifier is used for classification [26]. Multimodal experiments result shown in Table 4. Overall, twelve experiments are conducted in which proposed multimodal experiments outperforms over unimodal experiments.

Table 4. Multimodal experiments

Multimodal	Biometric	Accuracy of Classifier
IX	Finger vein Signature (ULBP)	94%
X	Finger vein Signature (HOG)	97%

7.3. Finger vein, face, and signature fusion experiments

After experimenting the combinations of face-signature and finger vein-signature, All the three biometrics are fused together to conduct new experiment, from the fused images ULBP, and HOG features are extracted. An ensemble classifier is used for classification [26]. Multimodal experiments result shown in Table 5.

Table 5. Multimodal experiments

Multimodal	Biometric	Accuracy of Classifier
XI	Finger vein, Face, and Signature (ULBP)	92.5%
XII	Finger vein, Face, and Signature (HOG)	95.8%

7.4. Comparision of results with previous research works

From the literature survey of the previous work [27]–[36], it has been observed that, very limited work is investigated by the researchers previously on the wavelet fusion for multimodal biometric recognition. The pervious work on finger vein and signature combination is done on cryptography [37], in which the author used only one sample for authentication. The pervious works were not explored on multimodal biometric recognitions. The standard database with larger data samples is not considered in the previous work. In this paper, we explored the work on larger standard databases for face, finger vein and signature combinations. We used 720 data samples of face, signature and finger vein. The Table 6 shows the performance of some latest multimodal biometric research work compared with the proposed system.

Table 6. Comparison with state of art work

Authors	Biometric Traits	Data Base Size	Accuracy/Recognition Rate	Error Rate
Mathkour *et al.* [38]	Fingerprints and Electrocardiogram	70 subjects, 720 samples	95.32%	4.68%
Kong *et al.* [39]	Finger vein, Fingerprint, and Face	50 subjects, 250 samples	93%	7.0%
Tharwat *et al.* [40]	Ear and Finger Kunckle	165 subjects, 6 samples	96.47%	3.53%
Arun *et al.* [7]	Fingerprint and Iris	750 subjects	94.8%	5.2%
Davies [29]	Ear and Face	56 subjects	96.8%	3.2%
Shen [30]	Palm print and Face	119 subjects	91.63%	8.37%
Proposed wavelet based fusion	Signature and Finger vein	40 subjects, 400 samples	97%	3.0%
Proposed wavelet based fusion	Signature and Face	40 subjects, 400 samples	94%	6.0%
Proposed wavelet based fusion	Face, Signature, and Finger vein,	40 subjects, 400 samples	95.8%	4.2%

The proposed method significantly reduced the storage space of the image samples when it is compared with the other state of art work. The combinations of Finger vein and signature using wavelet fusion technique shows significant reduction in error rate when compared with other state of art work shown in Table 6. The wavelet based fusion merges the two images of a different biometric characteristics of a person to single image, which significantly reduce the storage space. The proposed technique also showed the features extraction technique ULBP and HOG can be used to improve the recognition rates. The feature extraction techniques ULBP and HOG give the statistical features of the complete image. In the proposed work complete statistical information of samples is used, which is not done in the previous state of art work. The proposed multimodal biometric authentication schemes showed promising results and improved the classification performance rates, when considered recognition rate. With the comparison of pervious work in Table 6 it has been observed that the proposed model showed promising results in improving the biometric recognition rate.

8.   CONCLUSION

In this paper, we proposed the image fusion technique for multimodal biometric using wavelet transform. We introduced a new model where biometric images are fused using DWT, and these fused images can be used for recognition using machine learning algorithms by extracting HOG and ULBP features. We conducted experiments on the face with signature and finger vein with the signature, and face, signature and finger vein combinations. The fusion combinations of the proposed model, performed well over the unimodal biometric system. Both unimodal and multimodal experiments are conducted, and results are plotted. Twelve experiments are conducted on standard mapped databases. After investigation of all the experiments, it has been observed that wavelet-based image fusion with db2 wavelet filter and decomposition degree at 5 shows a better recognition rate in finger vein and signature for the selected database. And also

Histogram oriented gradient features showed better results over uniform local binary patterns in DWT image fusion for multimodal biometrics.

REFERENCES

[1] A. K. Jain, A. Ross, and S. Prabhakar, "An introduction to biometric recognition," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 1, pp. 4–20, 2004, doi: 10.1109/TCSVT.2003.818349.

[2] A. Ross, "An introduction to multibiometrics," in *European Signal Processing Conference*, 2007, pp. 20–24, doi: 10.1007/978-0-387-71041-9_14.

[3] A. Ross and A. Jain, "Information fusion in biometrics," *Pattern Recognition Letters*, vol. 24, no. 13, pp. 2115–2125, 2003, doi: 10.1016/S0167-8655(03)00079-5.

[4] K. Amolins, Y. Zhang, and P. Dare, "Wavelet based image fusion techniques - An introduction, review and comparison," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 62, no. 4. pp. 249–263, 2007, doi: 10.1016/j.isprsjprs.2007.05.009.

[5] M. K. Bhowmik, D. Bhattacharjee, M. Nasipuri, D. K. Basu, and M. Kundu, "Fusion of wavelet coefficients from visual and thermal face images for human face recognition - a comparative study," *International Journal of Image Processing*, vol. 4, no. 1, pp. 12–23, Jul. 2010, [Online]. Available: http://arxiv.org/abs/1007.0626.

[6] B. Jiang *et al.*, "A comparative study of wavelet-based image fusion with a novel fusion rule II," in *Proceedings - 5th International Conference on Instrumentation and Measurement, Computer, Communication, and Control, IMCCC 2015*, 2016, pp. 1304–1309, doi: 10.1109/IMCCC.2015.280.

[7] A. A. Ross, K. Nandakumar, and A. K. Jain, *Handbook of multibiometrics*, 1st ed., vol. 6. Boston: Springer Science and Business Media, 2006.

[8] M. O. Oloyede and G. P. Hancke, "Unimodal and multimodal biometric sensing systems: a review," *IEEE Access*, vol. 4. pp. 7532–7555, 2016, doi: 10.1109/ACCESS.2016.2614720.

[9] A. Ross and A. K. Janil, "Multimodal biometrics: An overview," in *12th European Signal Processing Conference*, 2004, pp. 1221–1224.

[10] M. I. Razzak, R. Yusof, and M. Khalid, "Multimodal face and finger veins biometric authentication," *Scientific Research and Essays*, vol. 5, no. 17, pp. 2529–2534, 2010.

[11] P. J. H. Prof. Matti Pietikäinen, "Image and video description with local binary pattern variants," *SpringerBriefs in Applied Sciences and Technology*, 2016.

[12] G. Pajares and J. Manuel de la Cruz, "A wavelet-based image fusion tutorial," *Pattern Recognition*, vol. 37, no. 9, pp. 1855–1872, Sep. 2004, doi: 10.1016/j.patcog.2004.03.010.

[13] J. Ortega-Garcia *et al.*, "MCYT baseline corpus: a bimodal biometric database," *IEE Proceedings: Vision, Image and Signal Processing*, vol. 150, no. 6, pp. 395–401, 2003, doi: 10.1049/ip-vis:20031078.

[14] Y. Yin, L. Liu, and X. Sun, "SDUMLA-HMT: a multimodal biometric database," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7098 LNCS, 2011, pp. 260–268.

[15] D. Suresha and H. N. Prakash, "Single picture super resolution of natural images using N-Neighbor Adaptive Bilinear Interpolation and absolute asymmetry based wavelet hard thresholding," in *Proceedings of the 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology, iCATccT 2016*, 2017, pp. 387–393, doi: 10.1109/ICATCCT.2016.7912029.

[16] M. Tico, E. Immonen, P. Ramo, P. Kuosmanen, and J. Saarinen, "Fingerprint recognition using wavelet features," in *ISCAS 2001 - 2001 IEEE International Symposium on Circuits and Systems, Conference Proceedings*, 2001, vol. 2, pp. 21–24, doi: 10.1109/ISCAS.2001.920996.

[17] B. Aiazzi, L. Alparone, S. Baronti, and A. Garzelli, "Context-driven fusion of high spatial and spectral resolution images based on oversampled multiresolution analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 10, pp. 2300–2312, 2002, doi: 10.1109/TGRS.2002.803623.

[18] N. S. Kumar and C. Shanthi, "A survey and analysis of pixel level multisensor medical image fusion using discrete wavelet transform," *IETE Technical Review (Institution of Electronics and Telecommunication Engineers, India)*, vol. 24, no. 2, pp. 113–125, 2007, doi: 10.4103/02564602.10876590.

[19] L. Prasad and S. S. Iyengar, *Wavelet analysis with applications to image processing*, 1st ed. Boca Raton: CRC Press, 2020.

[20] H. Kaur, D. Koundal, and V. Kadyan, "Multi modal image fusion: comparative analysis," in *Proceedings of the 2019 IEEE International Conference on Communication and Signal Processing, ICCSP 2019*, 2019, pp. 758–761, doi: 10.1109/ICCSP.2019.8697967.

[21] A. B. Channegowda and H. N. Prakash, "Multimodal biometrics of fingerprint and signature recognition using multi-level feature fusion and deep learning techniques," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 22, no. 1, p. 187, 2021, doi: 10.11591/ijeecs.v22.i1.pp187-195.

[22] Z. Rustam, Y. Amalia, S. Hartini, and G. S. Saragih, "Linear discriminant analysis and support vector machines for classifying breast cancer," *IAES International Journal of Artificial Intelligence*, vol. 10, no. 1, pp. 253–256, 2021, doi: 10.11591/ijai.v10.i1.pp253-256.

[23] A. Ramachandran, A. Ramesh, A. Sukhlecha, A. Pandey, and A. Karuppiah, "Machine learning algorithms for fall detection using kinematic and heart rate parameters-a comprehensive analysis," *IAES International Journal of Artificial Intelligence*, vol. 9, no. 4, pp. 772–780, 2020, doi: 10.11591/ijai.v9.i4.pp772-780.

[24] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005*, 2005, vol. I, pp. 886–893, doi: 10.1109/CVPR.2005.177.

[25] B. C. Arjun and H. N. Prakash, "Multimodal biometric recognition: fusion of modified adaptive bilinear interpolation data samples of face and signature using local binary pattern features," *International Journal of Engineering and Advanced Technology*, vol. 9, no. 3, pp. 3111–3120, Feb. 2020, doi: 10.35940/ijeat.C6117.029320029320.

[26] B. C. Arjun and H. N. Prakash, "Feature level fusion of seven neighbor bilinear interpolation data sets of finger vein and iris for multimodal biometric recognition," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 2, pp. 1531–1536, 2020, doi: 10.30534/ijatcse/2020/95922020.

[27] K. Gupta, "Advances in multi modal biometric systems: a brief review," in *Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2017*, 2017, vol. 2017-January, pp. 262–267, doi:

10.1109/CCAA.2017.8229811.

[28] M. Ghayoumi, "A review of multimodal biometric systems: fusion methods and their applications," in *2015 IEEE/ACIS 14th International Conference on Computer and Information Science, ICIS 2015 - Proceedings*, 2015, pp. 131–136, doi: 10.1109/ICIS.2015.7166582.

[29] S. M. S. Islam, R. Davies, M. Bennamoun, R. A. Owens, and A. S. Mian, "Multibiometric human recognition using 3D ear and face features," *Pattern Recognition*, vol. 46, no. 3, pp. 613–627, 2013, doi: 10.1016/j.patcog.2012.09.016.

[30] L. Shen, L. Bai, and Z. Ji, "FPCode: an efficient approach for multi-modal biometrics," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 25, no. 2, pp. 273–286, 2011, doi: 10.1142/S0218001411008555.

[31] M. Singh, R. Singh, and A. Ross, "A comprehensive overview of biometric fusion," *Information Fusion*, vol. 52, pp. 187–205, 2019, doi: 10.1016/j.inffus.2018.12.003.

[32] R. Ryu, S. Yeom, S. H. Kim, and D. Herbert, "Continuous Multimodal Biometric Authentication Schemes: A Systematic Review," *IEEE Access*, vol. 9. pp. 34541–34557, 2021, doi: 10.1109/ACCESS.2021.3061589.

[33] L. Kondapi, A. Rattani, and R. Derakhshani, "Cross-illumination Evaluation of Hand Crafted and Deep Features for Fusion of Selfie Face and Ocular Biometrics," 2019, doi: 10.1109/HST47167.2019.9032976.

[34] G. Goswami, P. Mittal, A. Majumdar, M. Vatsa, and R. Singh, "Group sparse representation based classification for multi-feature multimodal biometrics," *Information Fusion*, vol. 32, pp. 3–12, 2016, doi: 10.1016/j.inffus.2015.06.007.

[35] D. Yaman, F. I. Eyiokur, and H. K. Ekenel, "Multimodal age and gender classification using ear and profile face images," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, 2019, vol. 2019-June, pp. 2414–2421, doi: 10.1109/CVPRW.2019.00296.

[36] N. Poh and S. Bengio, "Database, protocols and tools for evaluating score-level fusion algorithms in biometric authentication," *Pattern Recognition*, vol. 39, no. 2, pp. 223–233, 2006, doi: 10.1016/j.patcog.2005.06.011.

[37] A. Nandhinipreetha and N. Radha, "Multimodal biometric template authentication of finger vein and signature using visual cryptography," 2016, doi: 10.1109/ICCCI.2016.7479963.

[38] R. M. Jomaa, H. Mathkour, Y. Bazi, and M. S. Islam, "End-to-end deep learning fusion of fingerprint and electrocardiogram signals for presentation attack detection," *Sensors (Switzerland)*, vol. 20, no. 7, 2020, doi: 10.3390/s20072085.

[39] Y. Xin *et al.*, "Multimodal feature-level fusion for biometrics identification system on IoMT Platform," *IEEE Access*, vol. 6, pp. 21418–21426, 2018, doi: 10.1109/ACCESS.2018.2815540.

[40] A. Tharwat, A. F. Ibrahim, and H. A. Ali, "Multimodal biometric authentication algorithm using ear and finger knuckle images," in *Proceedings - ICCES 2012: 2012 International Conference on Computer Engineering and Systems*, 2012, pp. 176–179, doi: 10.1109/ICCES.2012.6408507.

BIOGRAPHIES OF AUTHORS

Arjun Benagatte Channegowda 🆔 � 🆂🅲 🅿 received a B.E. in CSE from VTU, Karnataka, India, and M.Tech. in Industrial Automation and Robotics from VTU Belagavi, Karnataka. India. Currently, he is pursuing a Ph.D. in Computer Science and Engineering at RIT research center affiliated to Visvesvaraya Technological University. He is currently working in ISE Department at RIT, Hassan as Assistant Professor and Head. He has 2 years of HCL industry and 11 years of teaching experience. He has published 2 scopus indexed papers and many conferences and international journals papers. His reseach interest in biometrics, image processing, and machine learning. He can be contacted at email: bc.arjun@gmail.com.

Dr. Hebbakavadi Nanjundaiah Prakash 🆔 � 🆂🅲 🅿 received a B.E. in ECE from Mysore University, and he received M.Tech. in Electronics Instrumentation from NIT Warangal, and received Ph.D. in Computer Science from Mysore University. He is currently working in CSE Department at RIT, Hassan as Professor and HOD. He has thiry years of teaching experience. He published IEEE transactions during his research work. He also published many papers in international journals and conferences. His research interest in biometrics, signature analysis, and artifical intelligence. He can be contacted at email: prakash_hn@yahoo.com.

❏   238

Comparison of classifiers using robust features for depression detection on Bahasa Malaysia speech

Nik Nur Wahidah Nik Hashim[1], Nadzirah Ahmad Basri[2], Mugahed Al-Ezzi Ahmad Ezzi[1], Nik Mohd Hazrul Nik Hashim[3]

[1]Department of Mechatronics Engineering, Faculty of Engineering, International Islamic University Malaysia, Gombak, Malaysia
[2]Department of Psychiatry, Kulliyyah of Medicine, International Islamic University Malaysia, Jalan Hospital, Kuantan, Pahang, Malaysia
[3]Graduate School of Business, Universiti Kebangsaan Malaysia, UKM Bangi, Selangor, Malaysia

Article Info

ABSTRACT

Early detection of depression allows rapid intervention and reduce the escalation of the disorder. Conventional method requires patient to seek diagnosis and treatment by visiting a trained clinician. Bio-sensors technology such as automatic depression detection using speech can be used to assist early diagnosis for detecting remotely those who are at risk. In this research, we focus on detecting depression using Bahasa Malaysia language using speech signals that are recorded remotely via subject's personal mobile devices. Speech recordings from a total of 43 depressed subjects and 47 healthy subjects were gathered via online platform with diagnosis validation according to the Malay beck depression inventory II (Malay BDI-II), patient health questionnaire (PHQ-9) and subject's declaration of major depressive disorder (MDD) diagnosis by a trained clinician. Classifier models were compared using time-based and spectrum-based microphone independent feature set with hyperparameter tuning. Random forest performed best for male reading speech with 73% accuracy while support vector machine performed best on both male spontaneous speech and female reading speech with 74% and 73% accuracy, respectively. Automatic depression detection on Bahasa Malaysia language has shown to be promising using machine learning and microphone independent features but larger database is necessary for further validation and improving performance.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Nik Nur Wahidah Nik Hashim
Department of Mechatronics Engineering, Faculty of Engineering, International Islamic University Malaysia
53100, Gombak, Malaysia
Email: niknurwahidah@iium.edu.my

1.   INTRODUCTION

Major depressive disorder (MDD) has been reported as one of the psychiatric diagnosis in more than 90% of suicidal cases [1]. Globally, depressive disorder is one of the most common form of mental illness. With the rising number of cases on COVID-19 across the world, an epidemic of depression also emerges. World health organization (WHO) published the results of a survey on the impact of COVID-19 towards mental health [2]. The survey on 130 countries revealed factors that triggers mental health conditions or exacerbating existing ones are due to the death of family members, long period of isolation, loss of income and fear of getting infected. The pandemic is also disrupting the mental health and psychosocial services due to the use of mental health facilities as COVID-19 quarantine or treatment facilities and insufficient number

of redeployment of healthcare workers. Approximately 70% of countries have begun to adopt tele-therapy to overcome the in-person service disruption [2]. However, extensive research needs to be done on the early diagnosis of depression using technology to be able to reliably detect remotely those who are at risk.

Conventional methods are generally performed according to a series of questionnaires and rating scales that measure various aspects including thoughts, behaviours and symptoms that are evaluated by a trained clinician or self-reported measurements. The information gathering process is a time-consuming process that requires the trained clinician to perform physical examination to rule out organic conditions and maintain regular interactions with the patient in order to attain accurate assessment. Also, it relies heavily on patients' effort to cooperate in communicating their symptoms and problems, while they are emotionally and psychologically impaired. The reliability of self-reported assessment can also be compromised by patients' overfamiliarity and subjectivity when rating the severity of their symptom [3].

Despite extensive research on automatic depression detection (ADD), we are still far from achieving reliable alternative methods for assisting psychiatric diagnosis. Depression is multifactorial and multi-characteristics with both genetic and environmental factor as the common associated factors in the development of depression, hence the complexity and subjectivity of its diagnosis. Development of an alternative psychological assessment tool is proposed to identify patient's mental condition using speech technology. Subjects with MDD often shows psychomotor retardation which manifests as a slowing or loss of spontaneous movement, reactivity and flattening of emotional expression. Manifestations of psychomotor retardation also include slowed speech and impaired cognitive function [4].

The field of ADD using speech has its own challenges. Each language is unique both in its structure and in the way it reflects the culture of those who speak it. Therefore, in this field of research, there is no one common set of acoustic feature across all languages that can be used as a universal depression detection. Table 1 summarizes information of speech corpus used by previous researches for studying the correlation between speech and psychological impairment. One research has studied the cross-corpus generalization on verbal biomarkers from Australia, USA and Germany database as depression detection [5]. They reported on the need of observations across multiple dataset characteristics in the training part in order to achieve higher accuracy in testing stage. This shows that the model has to learn the language pattern/structure and also learn depression cue patterns from the acoustics parameters in all languages and thus, they are not generalizable.

Researchers have began using deep learning for ADD [6]–[8], but due to the limited availability of large-scale data, most studies in this field still uses machine learning techniques. Deep learning requires a large training data due to the huge number of parameters needed to be tuned by a learning algorithm and computation power for data processing. Previous studies in the field of ADD using machine learning classifiers have used binary logistic regression (LR) [9], [10], gaussian mixture model (GMM) [11]–[14], discriminant analysis [15], [16], support vector machine (SVM) [11], [17], [18], k-nearest neighbour (KNN) [17], gaussian naïve Bayes (GNB) [19] random forest (RF) [20] and decision tree [21]. The common classification method used for detecting depression in speech are SVM and GMM. A study on English language by [14] compared multiple classifiers and concluded that SVM and GMM showed the best classification performance.

This work proposes to investigate the possibility of speech as a biomarker for depression in one of the countries in Southeast Asia, which is Malaysia. The national language in Malaysia is Bahasa Malaysia and thus, we focus to further investigate ADD using Bahasa Malaysia language. This study fills in the research gap on Bahasa Malaysia speech ADD with machine learning classification using microphone independent features. Commonly, only one recording device is used to record the speech to prevent variability in the data. This is the first study that have specifically looked at the effectiveness of using microphone independent features for classification. The hypothesis for this work is that depression can be identified using Bahasa Malaysia speech. Our research question is, can microphone independent acoustic features be deployed to distinguish between depression and healthy speech in Bahasa Malaysia language? This work extends from our recent findings reported in [22] where we explored robust features that are not affected by varying recording devices. In this study, we proceeded with investigating the robust features for depressed and healthy Bahasa Malaysia speech classification. A novel contribution is the analysis of a new database gathered through online platform in Bahasa Malaysia language and the comparison of machine learning classifiers using microphone independent features.

Some limitations should be noted. Gathering speech input through online platform was very challenging due to the difficulty of speaking to a device and instead of a face-to-face conversation. We are also still taunted by the public stigma which involves the discriminatory attitudes to those with mental illness. People are still reluctant to share their data especially when it involves recordings of either audio or video. Male subjects are more reluctant to speak and express themselves compared to women. Thus, we were only able to obtain a small size database for male and an adequate number of samples for female.

Table 1. Summary of depression speech corupus used by previous researchers

1st Published (Name)	Language	Significant acoustic features	Subjects
Vanderbilt II Study France *et al*. (2000) [15]	English (USA)	Fundamental frequency, amplitude modulation	115: 59 Dep, 22 HR, 34 Nat
Moore, Clements, Peifer, and Weisser (2004) [11]	English (USA)	Glottal waveform	33: 15 Dep (6 M, 9 F), 18 Nat (9 M, 9 F)
Mundt, Snyder, Cannizzaro, Chappie, and Geralts (2007) [23]	English (USA)	Formants, pauses, vocalization and speaking rate	35: Dep (15 M, 20 F)
Cohn *et al.* (2009) [12]	English (USA)	Fundamental frequency and respond time	57: Dep (24 M, 34 F)
Oregon Research Institute Low, Maddage, Lech, and Allen (2009) [13]	English (USA)	MFCC	139: 68 Dep (49 F, 19 M) 71 Nat (27 M, 44 F)
Black dog Institute Alghowinem *et al*. (2012) [14]	English (Australia)	Fundamental frequency, energy, intensity, loudness, jitter, shimmer, Harmonic-to-noise-ratio (HNR), voice probability and quality, formants, and MFCCs.	80: 40 Dep, 40 Nat
Shankayi, Vali, Salimi, and Malekshahi (2012) [24]	Persian (Iran)	Pitch, energy, formant, glottal features	56 (Dep, Nat, Rem)
Cincinnati Children's Hospital Medical Centre Scherer, Pestian, and Morency (2013) [25]	English (USA)	Energy, fundamental frequency, peak slope	60: 30 HR, 30 Nat
Audio-Visual Depressive Language Corpus (AViD Corpus) Valstar *et al*. (2013) [26]	Germany (German)	Open corpus	292 files each containing a range of mix of vocal, free and read speech tasks
Audio-Visual Depressive Language Corpus (AViD Corpus) [27]	Germany (German)	Open corpus	150 files each containing a range of mix of vocal, free and read speech tasks
Distress Analysis Interview Corpus (DAIC-WOZ) [28]	English (USA)	Open corpus	76: Nat (49 M, 27 F) 31: Dep (14 M, 17 F)
University of Nottingham Solomon *et al*. (2015) [29]	English (UK) Local and international students	Energy, pitch, MFCC, articulation rate and total time.	17 (13 F, 4 M): 9 Dep, 8 Nat
Liu *et al.* (2015) [17]	Chinese (China)	N/A	300: 100 HR, 100 Dep, 100 Nat
Kiss *et al.* (2016) [18]	Hungary (Hungarian) And Italy (Italian)	Fundamental frequency, volume dynamics, formants, jitter, shimmer, rate of transient, pauses.	Hungarian: 54: Dep (19 M, 35 F) 73: Nat (29 M, 44 F) Italian: 22: Nat (11), Dep (11)
Azam *et al.* (2016) [16]	Malaysia (Bahasa Malaysia)	MFCC	13: Dep (6 M, 7 F) 20: Nat (10 M, 10 F)
Hashim *et al*. (2017) [30]	English (USA)	Timing-based features, MFCC and power spectral density	116: Dep (44 M, 72 F)
Pan *et al*. (2019) [9]	Chinese (China-female only)	Intensity, loudness, zero-crossing rate, MFCC, line spectral pair, voicing probability, fundamental frequency.	Database 1: 1132: (584 Dep, 548 Nat) Database 2: 904: (500 Dep, 404 Nat)

Abbreviation: Dep–depressed, HR–high risk suicidal, Nat–natural, Rem–remitted, PTSD–post traumatic stress disorder, M–number of males, F–number of females.

2.   DATABASE

This study was conducted during the first wave of COVID-19 in Malaysia. We gathered recordings of depressed and healthy subjects using online platform due to the limitation of visiting the hospitals and having a face-to-face data collection with subjects on site. All procedures performed in studies involving human participants were in accordance with the ethical standards and has been approved by the IIUM research ethical committee (IREC 2019-006). The database was divided based on gender and diagnostic groups of depressed and healthy. Subjects consisted of 43 depressed and 47 healthy were required to sign an

informed consent and to be neither under the influence of alcohol, toxicity, nor experiencing respiratory problems. Table 2 summarizes the database demographic characteristics.

Table 2. Summary of database characteristics

Demographic characteristics	Subject categorized as depressed (n=43)	Subject categorized as healthy (n=47)
Gender		
Female	32 (74.42%)	34 (72.34%)
Male	11 (25.58%)	13 (27.66%)
Age (mean ⬚ SD)	28.11±5.71	25.28±7.48
Age (range)		
≤20	1 (2.33%)	3 (6.38%)
21-25	14 (32.56%)	30 (63.83%)
26-30	8 (18.60%)	12 (25.53%)
31-35	12 (27.91%)	1 (2.13%)
>35	8 (18.60%)	1 (2.13%)
Marital status		
Single	29 (67.44%)	44 (93.62%)
Married	11 (25.58%)	3 (6.38%)
Divorced	3 (6.98%)	0
Death of spouse	0	0
Number of children		
none	33 (76.74%)	43 (91.49%)
1-2	5 (11.63%)	3 (6.38%)
3-4	5 (11.63%)	0
5 or more	0	1 (2.13%)
Academic qualification		
Secondary school	4 (9.30%)	1 (2.13%)
Diploma/Degree	19 (44.19%)	2 (4.26%)
Master	17 (39.54%)	39 (82.98%)
PhD	3 (6.98%)	5 (10.64%)
Working status		
Full-time	19 (44.19%)	6 (12.77%)
Part-time	2 (4.65%)	2 (4.26%)
Studying	9 (20.93%)	15 (3.19%)
Housewives	3 (6.98%)	1 (2.13%)
Not working	10 (23.26%)	23 (48.94%)
Working sector		
Government	11 (25.58%)	6 (12.77%)
Private	7 (16.28%)	5 (10.64%)
Self employed	4 (9.30%)	1 (2.13%)
Others	21 (48.84%)	35 (74.47%)
Health level		
Not healthy	8 (18.60%)	0
Moderate	19 (44.19%)	4 (8.51%)
Very healthy	16 (37.21%)	43 (91.49%)

Subjects were asked to fill in an online form that consists of three sections. The first section gathers the demographic information such as age, marital status, number of children, education level, working status, working sector and physical health. Subjects were also asked to report whether they have received a diagnosis of MDD by the hospital's psychiatrist. The second section requires the subject to fill in the Malay BDI-II and PHQ-9.

The final section requires the subject to send a voice note recording through the application called WhatsApp voice note. Two types of recordings were collected which are the spontaneous and reading speech. For spontaneous speech, subjects were asked to answer a question in Bahasa Malaysia about their situation in the past three days which includes any activities or feelings they have experienced. They were also asked about what situation that usually causes them to feel stressed. For reading speech, patients were asked to read a standardized Bahasa Malaysia passage called Cerita Datuk that is commonly used by speech therapists. The recordings received in the voice note were in OGG and MP4 format. Speech files were converted to WAV using the audio.online-convert.com at a sampling rate of 44.1 kHz and 32 bits per sample, with a mono channel. The recordings were normalized prior to the acoustic parameter extraction. Each audio signal was then divided into 20 second segments and acoustic features were extracted for each segments. The total number of 20 second segments for female reading speech, male reading speech and male spontaneous speech are 198, 73, and 156, respectively. We omit the dataset for female spontaneous speech due to poor classification performance. The subjects were categorized as depressed if he/she has a score of BDI-II of 20 or more, a score of PHQ-9 of 16 or more and also has been diagnosed as MDD formally by the hospital's

psychiatrist. Healthy subjects were chosen based on their score of minimal range (0-13) for BDI-II and mild range (0-5) for PHQ-9.

3. RESEARCH METHODOLOGY

3.1. Acoustic parameter extraction

Prior to this work, we performed an analysis on robust parameters to identify acoustic parameters that are robust towards multiple mobile recording devices [28]. The reason we conducted the analysis is because of our method of collecting the speech recordings remotely using subject's personal mobile devices with different microphone specifications. Timing-based parameters such as transition parameters (TP), interval length probability density function (ILpdf) and amplitude modulation statistics (AM) were found to be less prone to suffer from microphone variability with coefficient of variance (CoV) less than 30%. Mel-frequency cepstral coefficient (MFCC) parameters were the most affected by microphone variability but power spectral density (PSD) band 1 (0 to 500 Hz) and band 2 (501 to 1000 Hz) were shown to be robust with CoV less than 20%. Therefore, we decided to proceed with the parameters listed in Table 3. All the parameters were extracted using matrix laboratory (MATLAB) software.

Table 3. List of selected robust parameters

Parameter type	List of paramerters
Time-based	t11, t31, t33, t13
	v1, v2, sil1
	avgAM, covAM, varAM, rangeAM, maxAM
Spectrum-based	PSD1, PSD2

3.1.1. Transition parameters (TP)

Each speech signal is divided into a 40-ms frames and labeled as voiced (1), unvoiced (2) and silence (3). Audio speech signal is quasi-periodic for a length of approximately 20-ms to 40-ms. However, due computation time, we use the maximum 40-ms frame. Labels were then concatenated into one sequence of samples (labeled sample sequence). Knowing the future labels in the sequence, we set the states equal to the sequence (not hidden) and estimate the Markov model. The output will be an emission matrix which equals to an identity matrix and a 3-by-3 matrix of transition parameters that describes the probability of being in the current state (row) to the next state (column) within the whole sequence. After the analysis of robust parameter, only four parameters had CoV of less than 30% which are transition probability from voiced-to-voiced (t11), silence-to-voiced (t31), silence-to-silence (t33) and voiced-to-silence (t13).

3.1.2. Interval length probability density function (ILpdf)

The frequency of consecutive 2-ms frames for voiced and silence segments within one labeled sample sequence were extracted and re-distributed into histogram bins of up to five for voiced and ten for silence. Analysis of robust parameters reveled that 2-ms consecutive frames of voiced (v1), 4-ms consecutive frames of voiced (v2) and 2ms of consecutive frames of silence (sil1) were the most robust towards microphone variability with less than 20% CoV. We therefore proceeded with these parameters. The idea of TP and ILpdf parameters are to represent the probability and distribution of pauses, unvoiced sound and voiced sound within a speech signal.

3.1.3. Amplitude modulation (AM)

Each speech signal is squared in order to push half of the energy signal up to higher frequencies and the other half lower towards the mean. The squared signal is then sent through a low-pass filter to eliminate the high frequency energy in speech that are not significant and has low information. To maintain the correct scale, we then take the square root of the signal to reverse the scaling distortion that was due to the squaring in the earlier stage. Finally, the signal is downsampled to reduc e sampling frequency. Five robust statistical measurements were calculated from the envelope which are the average, coefficient of variance, variance, range and maximum.

3.1.4. Power spectral density (PSD)

Speech signal comprised of voiced, unvoiced, and short silence segments that are mixed. A set of third order band-pass filters was applied to each frame segment of the sampled signal. All unvoiced and silence terms were removed and only the voiced terms were collected and concatenated into one new signal. Four normalized equal bands of PSD were obtained using the method of Periodogram and trapezoidal

numerical integration for frequencies between 0 to 2000 Hz. The final band was removed due to linear dependent on the other three spectral energy bands. The PSD is calculated on every non-overlapping of 40-ms window frame and averaged over the total number of frames per speech signals.

3.2. Classifiers

Four classifiers were chosen to be reported which are the support vector machine, k-nearest neighbour, random forest, and extreme gradient boosting. These classifiers were determined based on the pairplot distribution on the respective dataset. The pairplot distribution for all dataset categories of both gender reading and spontaneous speech demonstrated overlapping and non-linear behaviour. Thus, from the observation, linear classifiers are not suitable. However, we proceeded to perform training and testing on linear regression and naïve Bayes for verification, and we confirmed our initial observation when the accuracy for these linear classifiers were significantly low and distributed around 0.5.

3.2.1. Support vector machine (SVM)

SVM was chosen as one of the classifiers because of the small number of dataset and it is also one of the most common method of binary classification with discrete target variable. Although SVM is a linear classifier, it can also perform a non-linear classification by changing the kernel model. The idea behind SVM is to find the best boundary or hyperplane that best separates parameters in a two-classes dataset. With N being the number of parameters, the separating hyperplane is an (N-1)-dimensional subspace. SVM will choose a decision hyperplane that maximizes the distance between support vectors (sample data points) to the hyperplane. Three main parameters that can be reading in SVM are kernel, C parameter and gamma.

The kernel functions such as linear, polynomials, radial basis function (RBF) and sigmoid measure the distance of support vectors that are transformed into high-dimensional space to make them more linearly separable when running the machine learning algorithm. The result will then be transferred back to the original input space such as class predictions. However, due to the transformation into higher dimensionality spaces, applying the kernels may cause overfitting. C and gamma can be used for regularization parameters. Gamma acts as the multiplier in the kernel functions where higher gamma value corresponds to a closer fit but higher chance of overfitting. C parameter (penalty/cost) controls the trade-off between increasing the distance between hyperplane to support vectors or minimizing misclassification in the training set. For a small value of C, penalty for misclassification is reduced and thus, higher margin of hyperplane will be chosen, which will result in greater misclassification.

3.2.2. K-nearest neighbour (KNN)

KNN is known to perform better with smaller number of features. KNN classifies a new sample point based on a similarity measure such as Euclidean distance or Manhattan distance. K-value denotes the number k-nearest neighbour's classes which are the voting class of the new sample data point. The label of the new sample point will be determined based on the most common label closest to most classes in the k sample data points. Two main parameters in KNN are the k-value and the weights. The performance of KNN is mostly affected by the k-value where a smaller k-value might cause the algorithm to be more sensitive towards outlier and a larger k-value might cause the algorithm to include too many sample points. To overcome this issue, a weighted-KNN can be applied. Two options for weight parameter are uniform and distance. Uniform weight indicates that all neighbours get an equally weighted vote, however, distance weight gives a higher weight using a kernel function such as inverse distance function to the nearest k-sample point and less weight to k-sample point that are further than the test sample point.

3.2.3. Random forest (RF)

RF is an ensemble method that trains several decision trees in parallel on multiple subsets of training dataset using different subset of parameters. The final decision combines the decision of each individual trees by means of voting, thus exhibits good generalization. It does not suffer from overfitting because it takes the average of all decisions. However, the process is time-consuming due to having multiple decision trees making predictions for the same input.

The main parameters that are available for tuning are the n-estimator, maximum features, maximum depth and feature importance (criterion). N-estimator is the number of decision trees in the forest. Maximum depth represents the depth of each tree in the forest where more splits correspond to capturing more information about the data. Maximum features determine the number of features to consider when deciding for the best split. The importance of each data feature is calculated using attribute selection indicator called information gain or gini impurity. It captures how much the classifier model's accuracy decreases when a data feature is dropped. However, for this work, we only used n-estimator and criterion because the initial procedure already performed feature reduction and due to the low number of features, we decided to proceed with a deeper split of trees.

3.2.4. Extreme gradient boosting (XGBoost)

XGBoost uses a gradient boosting framework for problems involving structured or tabular datasets. XGBoost includes boosting where models are built sequentially by minimizing errors from previous models while assigning higher weight (boosting) on the misclassified data points to influence higher performance model. It works on the principle of ensemble where it combines a set of weak learners such as a single split decision tree to improve the model accuracy. The errors in sequential models are minimized using gradient decent algorithm. The output is aggregated along the way instead of at the end of the process.

Multiple parameters can be tuned in XGBoost, however we focused on five parameters which are the learning rate, gamma, maximum depth, minimum child weight and column sample by tree. The learning rate corresponds to how fast the error is corrected sequentially from previous tree to the next which is calculated based on the difference between the two. Gamma reflects the minimal loss reduction that is required to make a split. Maximum depth, minimum child weight and column sample by tree are parameters used in gradient boost that controls depth of tree, minimum sum of weights for all observations and the fractions of columns to be randomly sampled, respectively.

3.3. Optimization

In machine learning, feature selection and hyperparameter tuning are two essential processes to improve the performance of the algorithm. Feature selection removes irrelevant features while increasing the performance of the classifier model and also reduces the number of features in order to reduce computational cost and preventing over-modelling. Hyperparameter tuning is an automatic optimization of machine learning parameters that are specifically performed on specific classifier, feature set and dataset. Different dataset often produces different optimal hyper-parameter settings.

Figure 1 shows the optimization procedure for identifying the best feature set and classifier model using a two-stage optimization process which are the exhaustive feature selection (EFS) and cross-validation grid Search hyperparameter tuning. The EFS algorithm is a wrapper method that uses brute-force evaluation of feature sets. The best feature is selected by optimizing the performance metric such as accuracy score, given a specific classifier. In the first stage, the EFS was executed using each classifier (LR, NB, SVM, KNN, RF and XGBoost) for 200 iterations with each iteration having a different train-test split dataset and using 5-fold cross-validation. The analysis performed in this work uses 70% training and 30% test data (70-30) split. This way, the feature set obtained is expected to be more generalizable. Each iteration outputs a feature set with the best accuracy score. The same feature set with best accuracy score that has a majority of five or more was then selected for validation using its respective classifier.
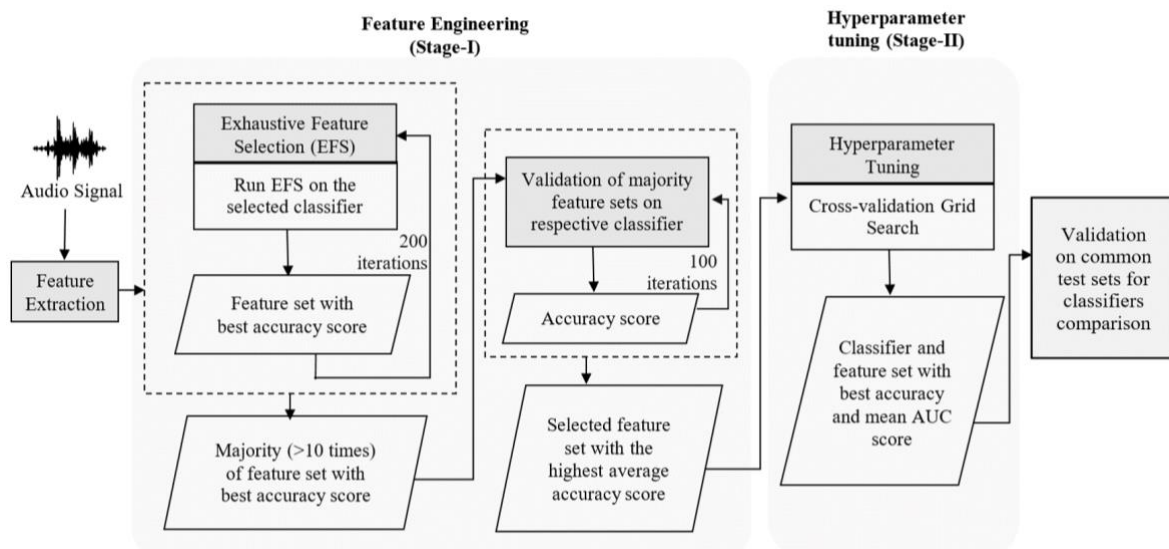

Figure 1. Optimization process using feature engineering and hyperparameter tuning

The performance of the selected feature sets was then validated by executing the classification for 100 iterations using its respective classifier with different test-train set. Each iteration outputs the accuracy

score and the classifier model with the feature set that produces a good performance of average accuracy score was selected for hyperparameter tuning. Finally, we tested on one common test set and the classifier with feature set with the best accuracy score and the best mean of area under the receiver operating characteristic (ROC) curve (AUC) was selected. The AUC was calculated for 10 folds cross-validation and the score ranges from 0 to 1 where 1 indicates a 100% correct classification. Different train-test set produce different accuracy scores, thus multiple iterations are required to obtain a suitable and generalizable model. The iterations are chosen based on computation power and time limitation but adequate to generate a normal distribution.

4.   RESULTS AND DISCUSSION
4.1. Comparison between spontaneous and reading speech
Based on Table 4, all feature sets consist of at least one time-based feature related to pauses and vocalization which demonstrates the significance of TP in distinguishing between the speech of depressed and healthy. We suspect that recognition of depression from spontaneous speech is cognitively more challenging as compared to reading speech due to the planning required for selecting proper words and controlling the articulatory motor. During data collection, subjects experienced difficulties in coming up with ideas on what to say to fulfil the recording minimum time of one minute. This might also be the effect of non-face-to-face interview. Talking to a device was considered too artificial for them to express information, articulate ideas and sharing emotions as compared to a face-to-face interaction. Thus, increasing the significance of the time-based features as compared to reading a standardized passage. This effect is prominently seen in the male spontaneous speech category where depressed subjects have more difficulty portraying and expressing their thoughts as opposed to healthy male subjects.

The feature values for the female spontaneous speech were predominantly categorized as depressed and thus, was not included in this report. Although the accuracy of the time-based feature classification was high with approximately zero false rate in the depressed group, however, all classifiers that we tried were also classifying majority of healthy speech as depressed, which shows no significant difference between the spontaneous speech of healthy and depressed subjects. This might indicate that Malaysian female, regardless of feeling depressed or not, have a strong inclination towards expressing themselves. When asked about the common condition that may cause them to feel stressful, judging the content of the speech and listening to their tone and rhythm of expression, healthy female subjects tend to express their stressful condition in detail, passionately and a noticeable degree of emotion.

4.2. Comparison of classifiers performance on depressed speech classifications
Table 4 shows the best classifier performance, selected features, and parameter values after performing feature selection and hyperparameter tuning. Classifiers with accuracy and AUC value of less than 0.6 were omitted and not listed in Table 4. The features were selected based on the majority times the feature set produces a high accuracy using EFS as compared to the other sets and produces high accuracy when trained and tested on the respective classifiers. On the other hand, the parameters were selected according to the grid search (GridSearchCV) best score. The test sets are considered balance with depressed to healthy ratio of 5:6 for male reading, 8:7 for female reading, and approximately 7:5 for male spontaneous speech.

When comparing classifiers performance on a balance dataset, appropriate combination of these metric; the f1 score, precision, recall, accuracy, and AUC score should be evaluated. In this work, precision addresses the following question: Of all those who we classified as depressed, how many were actually depressed? while recall addresses the following question: Of all those who are depressed, how many were correctly classified? However, precision and recall are often conflicting. Improving precision usually reduces recall and vice versa. Precision and recall depend on the threshold of the classifier model and the performance of the classifier would vary as the threshold changes.

In this study, the positive refers to the depressed speech and the negative refers to the healthy speech. It is essential to identify depressed subjects correctly and thus, for the model to be considered effective, its true positive rate should be high even at the expense of a few false positive being misclassified. However, a classifier model with a high balance of true positive and true negative would be ideal. Therefore, the classifier with a higher recall than precision would be preferable since it is more desirable to detect as many depressed speeches as possible. On the other hand, f1 score captures the balance between the precision and recall value. However, for this study, it is not necessary to maximize the f1 score considering that we may prefer a higher recall, but it is also important to have an f1 score in the upper range.

Accuracy captures the fractions of correctly assigned positive and negative classes. That means if our problem is highly imbalanced, we get a significantly high accuracy score by simply predicting that all observations belong to the majority class. A high accuracy could indicate a high percentage of true positive

but significantly low true negative as demonstrated in the performance of classifiers using female spontaneous speech. Lastly, the ROC curve is more effective in capturing the effect of all thresholds between 0 to 1. The ROC curve plots the true positive rate (sensitivity) with respect to the false positive rate (1-specificity). The area under the ROC curve is captured by AUC score. The higher the AUC score, the better the classifier model at predicting the depressed as depressed and healthy as healthy.

Figures 2 to 4 show the box plot distribution of classification performance for each category of speech signal. Table 5 shows the detailed classifier performance on the selected classifiers based on the analysis of the box plot distribution in Figures 2 to 4. Precision-Recall p-value indicates the significance between mean distribution of precision and recall. Based on a two sample mean t-test with equal variance, we accepted the classifier model with precision higher than recall if the p-value is larger than 0.05 due to the insignificant difference between the two-mean distribution.

Table 4. Selected feature set and classifier model with parameter values

Speech signal type	Classifier	Features	Classifier Parameters	Grid best score (Hyperparameter tuning)
Male Reading Speech (Reading passage)	SVM	t13	C: 100, gamma: 0.3, kernel: RBF	0.6301
	KNN	t33, psd1, psd2, avgam	n:2, weights: uniform	0.6575
	XGBoost	t33, sil1, psd1	min_child_weight: 3, max_depth: 10, learning_rate: 0.2, gamma: 0.2, colsample_bytree: 0.7	0.8103
	RF	t13, psd1, rangeam	criterion: entropy, n_estimator: 90	0.8259
Male Spontaneous Speech (Question and Answer)	SVM	sil1, psd2, covam	C: 50, gamma: 0.8, kernel: RBF	0.7078
	KNN	t11, t31, sil1, avgam	n: 4, weights: distance	0.7468
	XGBoost	t11, t31, sil1	min_child_weight: 7, max_depth: 3, learning_rate: 0.2, gamma: 0.4, colsample_bytree: 0.8	0.7403
	RF	t11, t31, sil1, varam	criterion: gini, n_estimator: 60	0.7273
Female Reading Speech (Reading passage)	SVM	t31, t33, v1	C: 100, gamma: 0.7, kernel: sigmoid	0.7222
	KNN	t13, t33	n: 8, weights: uniform	0.7121
	RF	t11, t33, sil1	Criterion: gini, n_estimator: 160	0.7272

Table 5. Detailed classification performance on the selected classification

Dataset	Classifier	Average Accuracy	Average AUC	Average f1 score	Average Precision	Average Recall	Precision-Recall p-value
Male Reading	RF	0.7000	0.7127	0.6688	0.7223	0.6859	0.3653
	XGB	0.6733	0.6834	0.6571	0.6777	0.6792	-
Male Spontaneous	RF	0.7629	0.7657	0.7811	0.8013	0.7754	0.0836
	XGB	0.7377	0.7366	0.7628	0.7728	0.7679	0.7320
	SVM	0.7368	0.7374	0.7616	0.7711	0.7662	0.7321
Female Reading	SVM	0.7020	0.7075	0.7022	0.7312	0.6952	0.1281
	RF	0.6970	0.6986	0.7094	0.7110	0.7201	-

Figure 2(a) compares the accuracy of the classifiers for male reading speech, followed by comparisons of precision in Figure 2(b), recall in Figure 2(c), f1 score in Figure 2(d), and AUC in Figure 2(e). Based on the accuracy and AUC box plot, RF has the highest value followed by XGB and KNN. Eventhough XGB's scores are slightly lower than 0.7, we include it Table 5 because of its consistency and similar performance to RF. With a larger dataset, we believe that XGB can improve its performance. However, KNN classifier's precision is significantly higher than recall, which can also be seen in the low range of f1 score. Thus, RF and XGB were selected as the best classifier on the male reading speech dataset.

Figure 3(a) compares the accuracy of the classifiers for male spontaneous speech, followed by comparisons of precision in Figure 3(b), recall in Figure 3(c), f1 score in Figure 3(d), and AUC in Figure 3(e). All classifiers show similar performance for accuracy. However, from the precision and recall plots, KNN can be excluded due to its precision distribution being significantly higher than recall (p-value>0.05) and a low f1 score as compared to others. This shows that KNN is better at predicting the healthy speech based on the higher precision value. Therefore, we included RF, XGB and SVM in the list of selected classifiers in Table 5.
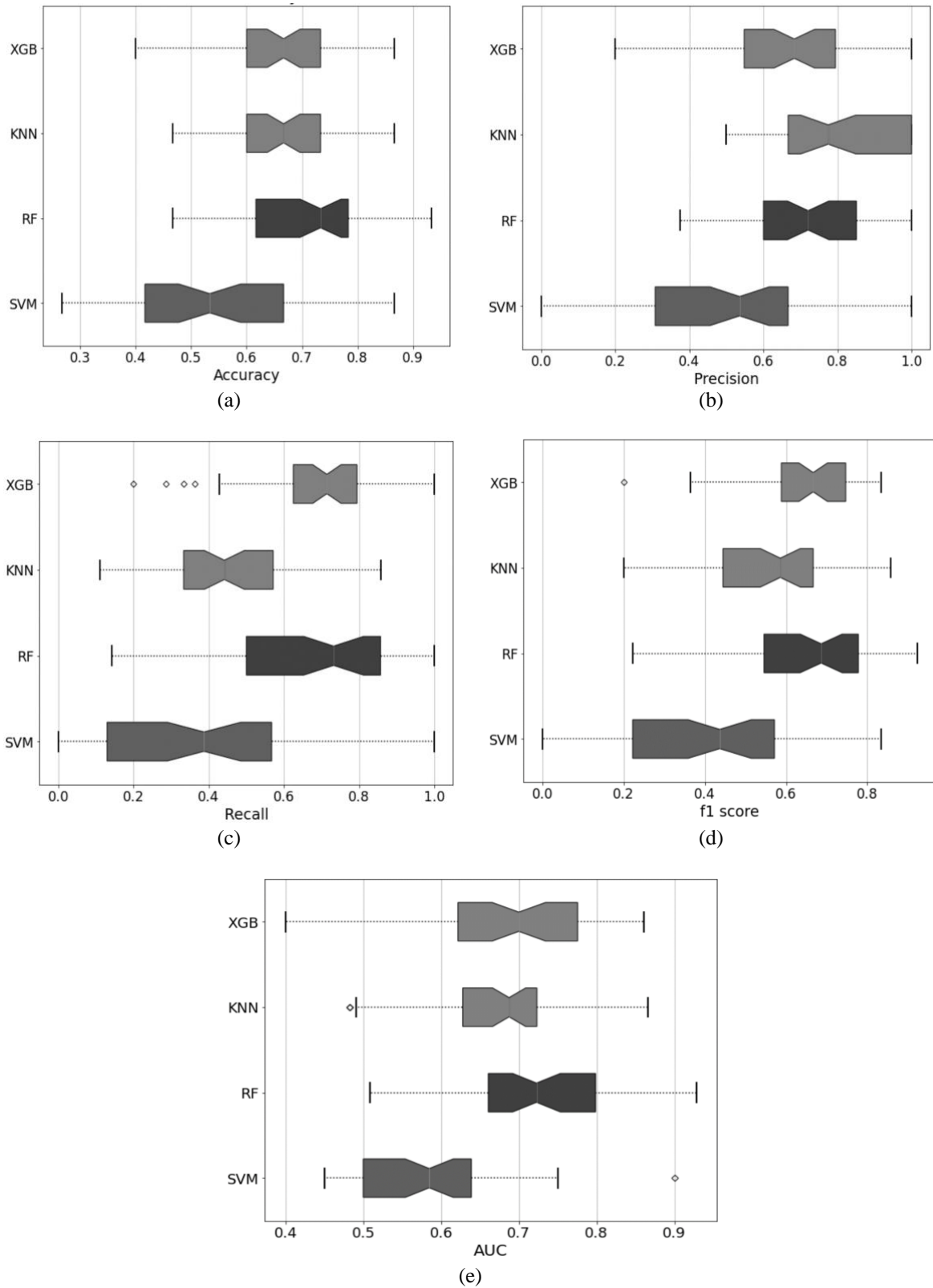
Figure 2. Classifiers performance on male reading speech with respect to (a) accuracy, (b) precision, (c) recall, (d) f1 score, and (e) AUC
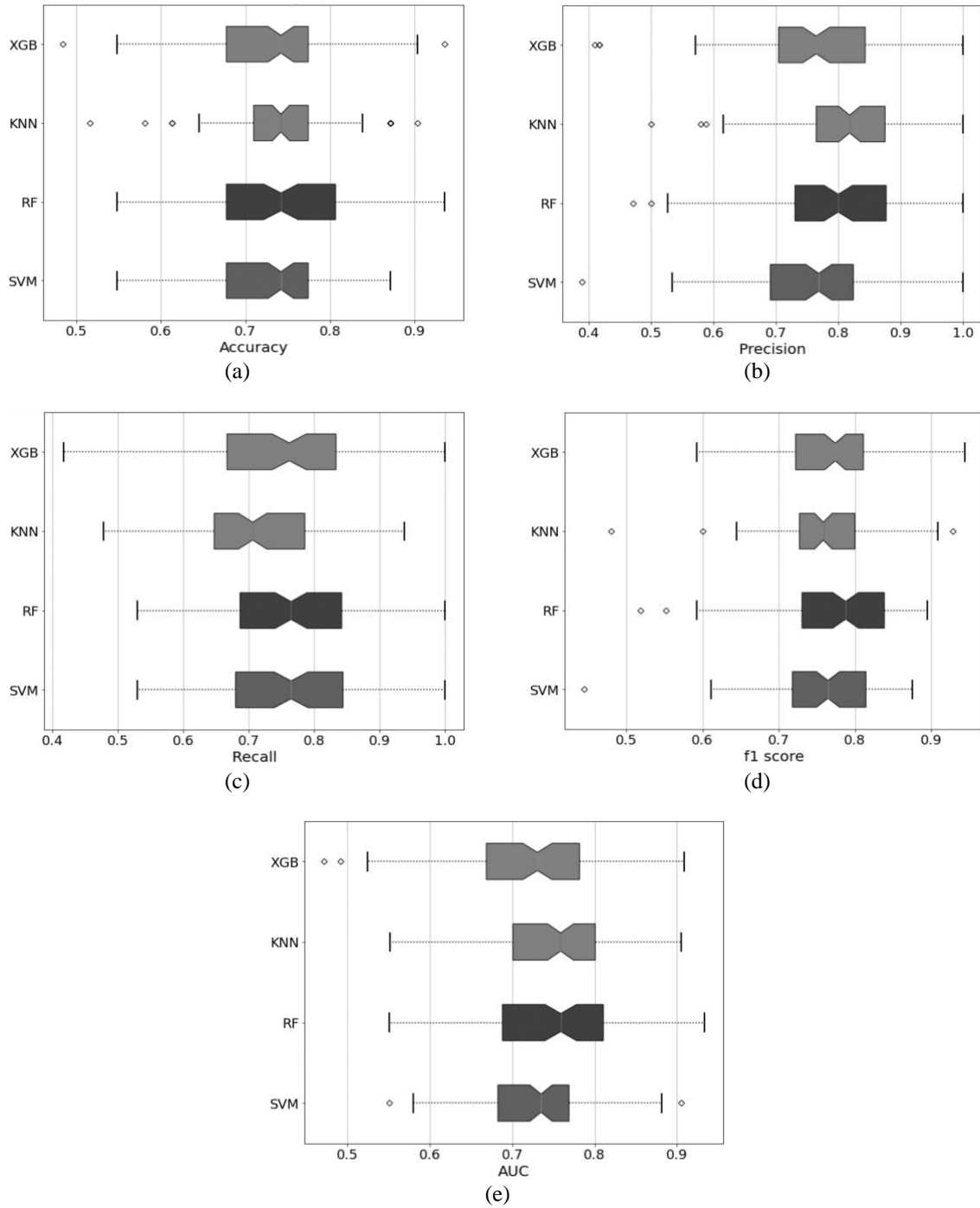
(a)

(b)

(c)

(d)

(e)

Figure 3. Classifiers performance on male spontaneous speech with respect to (a) accuracy, (b) precision, (c) recall, (d) f1 score, and (e) AUC

Figure 4(a) compares the accuracy of the classifiers for female reading speech, followed by comparisons of precision in Figure 4(b), recall in Figure 4(c), f1 score in Figure 4(d), and AUC in Figure 4(e). We excluded KNN based on its low performance on accuracy. KNN also displays low f1 score distribution and a significantly lower recall as compared to precision. Thus, using the robust features for female reading speech, the classification works best with SVM and RF.

Overfitting occurs when the model performs well on the training data but performs poorly on the validation set. To ensure our model was not overfitting, we obtained the training accuracy score and compare it with the validation accuracy. Avoiding overfitting was also considered with using a small number of

features by removing redundant features, integrating cross-validation during training and evaluating it over a number of iterations. However, it is best to re-check on the training accuracy and validation accuracy. If the accuracy for training data reaches 1.0, it means the model has memorized the training data instead of learning from the features. Also, a maximum of four features is an optimal dimension for this dataset size following the rule of thumb of one feature per 10 samples. Referring to Figure 1, we noticed that increasing the number of features beyond four does not significantly increases the performance of the classifier during the process in stage-I and it also significantly increases the computational time due to the method of exhaustive feature selection (EFS).
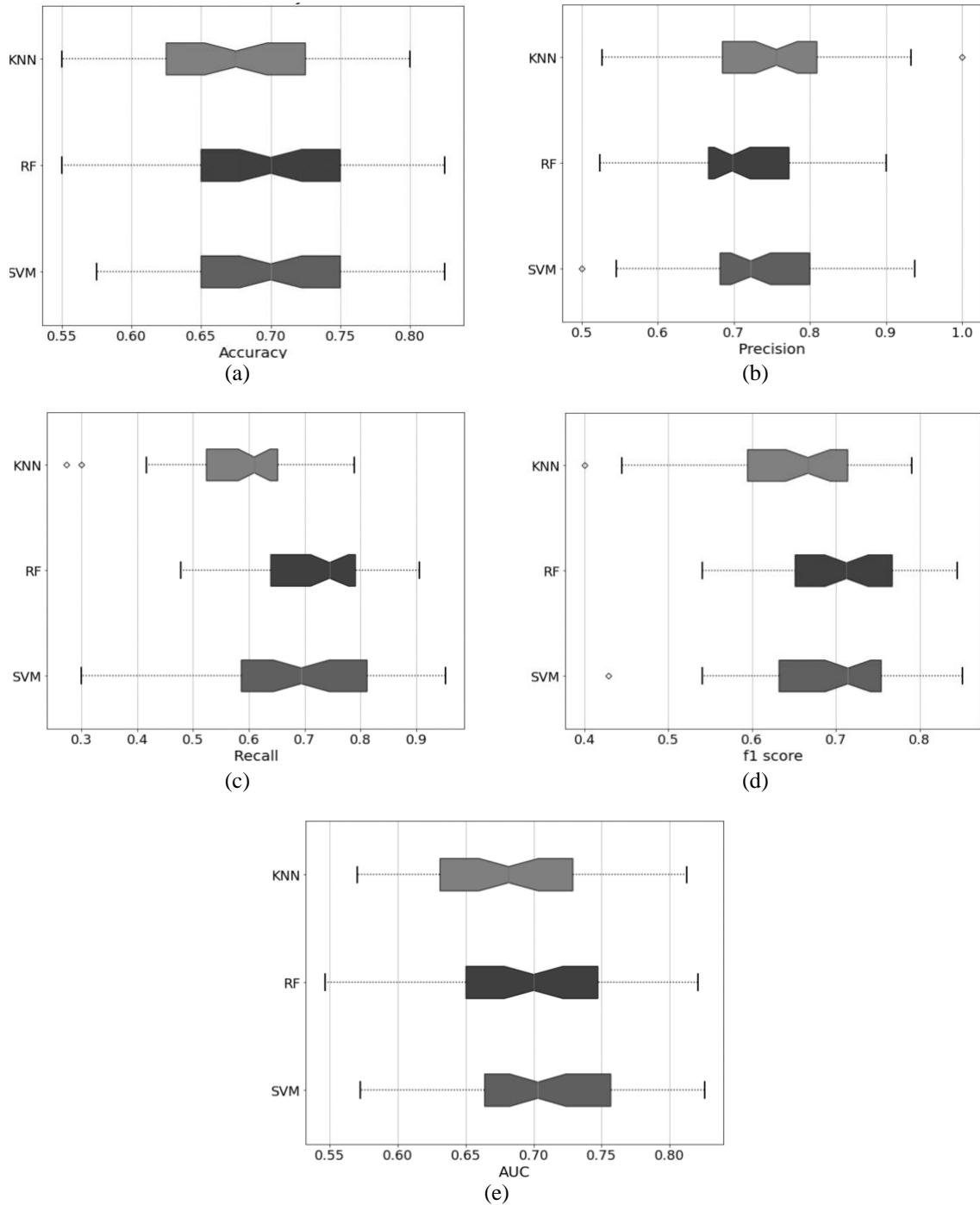

Figure 4. Classifiers performance on female reading speech with respect to (a) accuracy, (b) precision, (c) recall, (d) f1 score and (e) AUC

Table 6 shows the classification report for each category of speech signals. The classifications were analyzed by evaluating the performance on not just the accuracy, but also AUC, precision, recall and f1. Eventhough the performance is within the range of 70 to 80%, they showed consistency in all of classification variables, thus indicating that these are good models for the depression detection using Bahasa Malaysia language. However, with more data and information, we believe that the model can improve.

Table 6. Classification report

Data category	Male Reading			Male Spontaneous			Female Reading		
Classifier	RF (n_estimator=10)			SVM			SVM		
Classification report	Precision	Recall	f1	Precision	Recall	f1	Precision	Recall	f1
Healthy	0.75	0.75	0.75	0.70	0.76	0.73	0.74	0.70	0.72
Depressed	0.70	0.70	0.70	0.79	0.73	0.76	0.71	0.75	0.73
Macro average	0.72	0.72	0.72	0.74	0.75	0.74	0.73	0.72	0.72
Weighted average	0.73	0.73	0.73	0.75	0.74	0.74	0.73	0.72	0.72
Training accuracy	0.9411			0.7383			0.7225		
AUC score	0.7500			0.7949			0.7133		
Validation accuracy	0.7273			0.7447			0.7250		

Using the best classifier from the list in Table 5, we validated and trained the models with the respective feature set and classfier parameters on one selected data set with 70% training and 30% test. Figure 5 shows the Confusion matrix with each explanation Figure 5(a) shows for the male reading speech with random forest classifier, Figure 5(b) for the male spontaneous speech using random forest classifier and Figure 5(c) for female reading speech using support vector machine classifier.
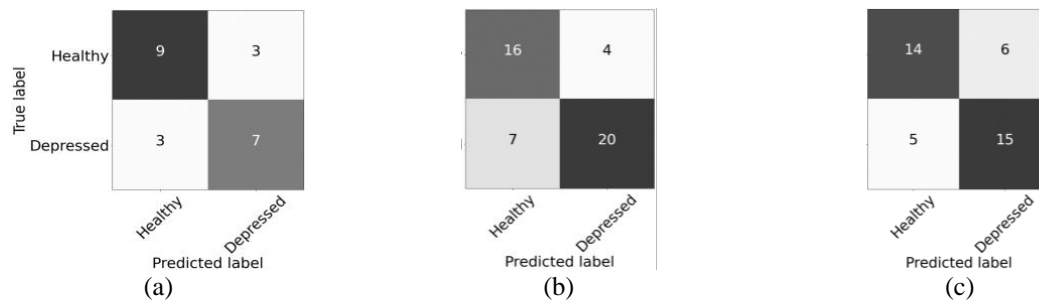

(a)　　　　　　　　　　　　(b)　　　　　　　　　　　　(c)

Figure 5. Confusion matrix for (a) male reading speech with random forest classifier, (b) Male spontaneous speech with random forest classifier, and (c) Female reading speech with support vector machine classifier

For male reading speech, we selected random forest but reduce the number of n_estimator to 10 since there was no significant change on the accuracy. For a tree-based models like RF, we can control the overfitting by tuning the parameters and we have demonstrated the slight difference of the hyperparameter tuning for this dataset. For the male spontaneous speech, RF, XGB and SVM performed equally, and therefore we selected the least complex model, which is SVM. For female reading speech, SVM outperform RF and therefore, we selected SVM as the model for this category.

Based on Table 1, depression identification using speech has been studied with multiple languages and so far, for Bahasa Malaysia, only a small database has been gathered and studied. The work in [16] shows the effectiveness of MFCC on classifying the depression and healthy speech. Based on the literature that shows ADD is dependent on language and the nature of this study's database collected using multiple recordings, we cannot compare our findings with the output reported in [16] because of the dependencies of MFCC on recording devices [28]. However, for a language independent based comparison, a simple classifier such as SVM has shown to be powerful in identifying depression speech [14], [20] which is similar to our findings.

A study using different languages such as English [31] and Japanese [32] have shown that timing-based feature is a promising biomarker for depression speech. Initial studies on pauses and silence found that pause time was longer for depressed patients than in healthy speech. Pause time shows correlation with the questionnaire scores where response time was longer for depressed patients than for healthy patients, and also response time became shorter as questionnaire score decreases [33]. In this work, timing-based features also

demonstrate the effectiveness as a feature to distinguish between depressed and healthy speech of Bahasa Malaysia in both male and female.

During the optimization process, there were two issues present pertaining to which procedure to perform first. Selecting the features will have an impact on the optimal hyperparameters and the selection of the hyperparameters will also affect the optimal feature set. This issue can be resolved by performing a unified feature selection and hyperparameter tuning. But, due to the computational time and cost, a unified procedure will be challenging to performed. We proceeded with performing the feature engineering as the first step before proceeding with hyperparameter tuning. Feature engineering is the process of transforming information into an underlying pattern that better represents the overall information but with minimal information, in this case, in the form of acoustic features. This process involves the feature selection. The hypothesis is that the classifier model can give better results if it has found a pattern that is more descriptive of the overall information. In this work, we have seen that the hyperparameter tuning does not significantly improve the performance of the classifier and thus, justifies the separating procedure of stage-I and stage-II. However, we still believe that hyperparameter tuning is crucial as they control the overall behaviour of the machine learning by searching for an optimal combination of hyperparameter that minimizes predefined loss function.

5.   CONCLUSION

The classifiers performance demonstrate the effectiveness of robust features for automatic depression detecting using Bahasa Malaysia language. There was no significant difference between the speech of depressed and healthy for the female spontaneous category, but RF classifier performed well for male reading speech and SVM performed well for male spontaneous and female reading speech. Depressive symptoms yield changes in the speech timing of pauses, even in standardize reading. It also shows that recordings through mobile applications or multiple recording devices is also possible, provided with appropriate robust feature sets. It is important to develop a system that can receive input from various microphone specifications such as a mobile app where users will provide speeh input using their personal mobile devices with its own microphone specification. With the output of this study, researchers in the field of ADD are not only limited to gathering data on site with a constant recording device but will also be able to use online platform. A study on a larger collected database will allow for verification on the effectiveness of reading versus spontaneous speech for both genders, and also able to implement ensemble machine learning and deep learning with multiple severity level classification.

REFERENCES

[1]   E. K. Mościcki, "Epidemiology of completed and attempted suicide: toward a framework for prevention," *Clinical Neuroscience Research*, vol. 1, no. 5, pp. 310–323, Nov. 2001, doi: 10.1016/S1566-2772(01)00032-9.
[2]   World Health Organization, *The impact of COVID-19 on mental, neurological and substance use services*. 2020.
[3]   L. Baer and M. A. Blais, Eds., *Handbook of clinical rating scales and assessment in psychiatry and mental health*. Totowa, NJ: Humana Press, 2010.
[4]   J. S. Buyukdura, S. M. McClintock, and P. E. Croarkin, "Psychomotor retardation in depression: Biological underpinnings, measurement, and treatment," *Progress in Neuro-Psychopharmacology and Biological Psychiatry*, vol. 35, no. 2, pp. 395–409, Mar. 2011, doi: 10.1016/j.pnpbp.2010.10.019.
[5]   S. Alghowinem, R. Goecke, J. Epps, M. Wagner, and J. Cohn, "Cross-cultural depression recognition from vocal biomarkers," in *Interspeech 2016*, Sep. 2016, pp. 1943–1947, doi: 10.21437/Interspeech.2016-1339.
[6]   Z. Huang, J. Epps, D. Joachim, B. Stasak, J. R. Williamson, and T. F. Quatieri, "Domain adaptation for enhancing speech-based depression detection in natural environmental conditions using dilated CNNs," in *Interspeech 2020*, Oct. 2020, pp. 4561–4565, doi: 10.21437/Interspeech.2020-3135.
[7]   N. S. Srimadhur and S. Lalitha, "An end-to-end model for detection and assessment of depression levels using speech," *Procedia Computer Science*, vol. 171, pp. 12–21, 2020, doi: 10.1016/j.procs.2020.04.003.
[8]   K. Schultebraucks, V. Yadav, A. Y. Shalev, G. A. Bonanno, and I. R. Galatzer-Levy, "Deep learning-based classification of posttraumatic stress disorder and depression following trauma utilizing visual and auditory markers of arousal and mood," *Psychological Medicine*, pp. 1–11, Aug. 2020, doi: 10.1017/S0033291720002718.
[9]   W. Pan *et al.*, "Re-examining the robustness of voice features in predicting depression: Compared with baseline of confounders," *Plos One*, vol. 14, no. 6, Art. no. e0218172, Jun. 2019, doi: 10.1371/journal.pone.0218172.
[10] H. Jiang *et al.*, "Detecting depression using an ensemble logistic regression model based on multiple speech features," *Computational and Mathematical Methods in Medicine*, vol. 2018, pp. 1–9, Sep. 2018, doi: 10.1155/2018/6508319.
[11] E. I. I. Moore, M. Clements, J. Peifer, and L. Weisser, "Comparing objective feature statistics of speech for classifying clinical depression," in *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2004, vol. 3, pp. 17–20, doi: 10.1109/IEMBS.2004.1403079.

[12] J. F. Cohn *et al.*, "Detecting depression from facial actions and vocal prosody," in *2009 3rd International Conference on Affective Computing and Intelligent Interaction and Workshops, ACII*, pp. 1–7, 2009.

[13] L.-S. A. Low, N. C. Maddage, M. Lech, and N. Allen, "Mel frequency cepstral feature and Gaussian Mixtures for modeling clinical depression in adolescents," in *2009 8th IEEE International Conference on Cognitive Informatics*, Jun. 2009, pp. 346–350, doi: 10.1109/COGINF.2009.5250714.

[14] S. Alghowinem, R. Göcke, M. Wagner, J. Epps, M. Breakspear, and G. Parker, "From joyous to clinically depressed: mood detection using spontaneous speech," in *Twenty-Fifth International FLAIRS Conference*, pp. 141–146, 2012.

[15] D. J. France and R. G. Shiavi, "Acoustical properties of speech as indicators of depression and suicidal risk," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 7, pp. 829–837, 2000.

[16] H. Azam *et al.*, "Classifications of clinical depression detection using acoustic measures in Malay speakers," in *2016 IEEE EMBS Conference on Biomedical Engineering and Sciences (IECBES)*, Dec. 2016, pp. 606–610, doi: 10.1109/IECBES.2016.7843521.

[17] Z. Liu *et al.*, "Detection of depression in speech," in *2015 International Conference on Affective Computing and Intelligent Interaction (ACII)*, Sep. 2015, pp. 743–747, doi: 10.1109/ACII.2015.7344652.

[18] G. Kiss, M. G. Tulics, D. Sztahó, A. Esposito, and K. Vicsi, "Language independent detection possibilities of depression by speech," in *Smart Innovation, Systems and Technologies*, pp. 103–114, 2016.

[19] D. Shin *et al.*, "Detection of minor and major depression through voice as a biomarker using machine learning," *Journal of Clinical Medicine*, vol. 10, no. 14, p. 3046, Jul. 2021, doi: 10.3390/jcm10143046.

[20] C. W. Espinola, J. C. Gomes, J. M. S. Pereira, and W. P. dos Santos, "Detection of major depressive disorder using vocal acoustic analysis and machine learning-an exploratory study," *Research on Biomedical Engineering*, vol. 37, no. 1, pp. 53–64, Mar. 2021, doi: 10.1007/s42600-020-00100-9.

[21] Z. Liu, D. Wang, L. Zhang, and B. Hu, "A novel decision tree for depression recognition in speech," *Electrical Engineering and Systems Science*, Feb. 2020.

[22] N. N. W. N. Hashim, M. A.-E. A. Ezzi, and M. D. Wilkes, "Mobile microphone robust acoustic feature identification using coefficient of variance," *International Journal of Speech Technology*, Aug. 2021, doi: 10.1007/s10772-021-09877-1.

[23] J. C. Mundt, P. J. Snyder, M. S. Cannizzaro, K. Chappie, and D. S. Geralts, "Voice acoustic measures of depression severity and treatment response collected via interactive voice response (IVR) technology," *Journal of Neurolinguistics*, vol. 20, no. 1, pp. 50–64, Jan. 2007, doi: 10.1016/j.jneuroling.2006.04.001.

[24] R. Shankayi, M. Vali, M. Salimi, and M. Malekshahi, "Identifying depressed from healthy cases using speech processing," in *2012 19th Iranian Conference of Biomedical Engineering, ICBME 2012*, 2012, no. December, pp. 242–245, doi: 10.1109/ICBME.2012.6519689.

[25] S. Scherer, J. Pestian, and L. Morency, "Investigating the speech characteristics of suicidal adolescents," in *ICASSP 2013-2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 709–713, 2013.

[26] M. Valstar *et al.*, "AVEC 2013," in *Proceedings of the 3rd ACM international workshop on Audio/visual emotion challenge*, Oct. 2013, pp. 3–10, doi: 10.1145/2512530.2512533.

[27] M. Valstar *et al.*, "AVEC 2014," in *Proceedings of the 4th International Workshop on Audio/Visual Emotion Challenge-AVEC '14*, 2014, pp. 3–10, doi: 10.1145/2661806.2661807.

[28] J. Gratch *et al.*, "The distress analysis interview corpus of human and computer interviews," *Proceedings of the 9th International Conference on Language Resources and Evaluation, LREC 2014*, 2014.

[29] C. Solomon, M. F. Valstar, R. K. Morriss, and J. Crowe, "Objective methods for reliable detection of concealed depression," *Frontiers in ICT*, vol. 2, Art. no. 5, Apr. 2015, doi: 10.3389/fict.2015.00005.

[30] N. W. Hashim, M. Wilkes, R. Salomon, J. Meggs, and D. J. France, "Evaluation of voice acoustics as predictors of clinical depression scores," *Journal of Voice*, vol. 31, no. 2, pp. 256.e1-256.e6, Mar. 2017, doi: 10.1016/j.jvoice.2016.06.006.

[31] M. W. Hashim, Nik Nur Wahidah Nik and R. S. Jared Meggs, "Analysis of timing pattern of speech as possible indicator for NearTerm suicidal risk and depression in male patients," in *2012 International Conference on Conference on Signal Processing Systems (ICSPS 2012)*, 2012, Art. no. 58, doi: 10.7763/IPCSIT.2012.V58.2.

[32] M. Yamamoto *et al.*, "Using speech recognition technology to investigate the association between timing-related speech features and depression severity," *Plos One*, vol. 15, no. 9, Art. no. e0238726, Sep. 2020, doi: 10.1371/journal.pone.0238726.

[33] P. hardy, R. Jouvent, and D. Widlöcher, "Speech pause time and the retardation rating scale for depression (ERD)," *Journal of Affective Disorders*, vol. 6, no. 1, pp. 123–127, Feb. 1984, doi: 10.1016/0165-0327(84)90014-4.

BIOGRAPHIES OF AUTHORS

Nik Nur Wahidah Nik Hashim (ID) (SC) (P) (Ph.D.) received her BS, MS, and PhD in Electrical Engineering at Vanderbilt University, Nashville, Tennessee, USA beginning 2005 to 2014. Currently an assistant professor in the Department of Mechatronics, Kuliyyah of Engineering at the International Islamic University Malaysia (IIUM). Her passion in research is studying the field of mental health through speech analysis. Other specific field of research interest include speech processing, machine learning, deep learning, digital signal processing and natural language processing. Email: niknurwahidah@iium.edu.my.

Nadzirah Ahmad Basri ⓘ 🔣 sc Ⓟ (Ph.D.) is currently working in the Department of Psychiatry, Kulliyyah of Medicine, International Islamic University Malaysia, Kuantan, Pahang. Nadzirah is a lecturer and a practicing Clinical Psychologist. She carries out psychological assessment and intervention to adults and children of varying mental disorders. She enjoys giving talks on mental health issues and is active working in non-governmental organisations advocating for mental health and academic issues. Email: nadzirahbasri@gmail.com.
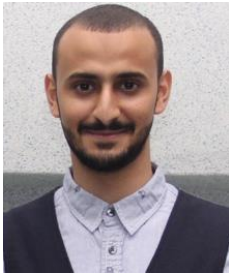
Mugahed Al-Ezzi Ahmed Ezzi ⓘ 🔣 sc Ⓟ is an R&D software manager at C&C I Motive Sdn. Bhd. His work focuses specifically on intelligent systems in automotive industries, including machine vision, machine learning, and deep learning. In 2018, he received his bachelor's degree with first-class honors in mechatronics engineering at International Islamic University Malaysia (IIUM). He is currently pursuing his master's degree in mechatronics engineering at IIUM, emphasizing speech analysis and mental health. Email: ezzi.mugahed@gmail.com.

Nik Mohd Hazrul Nik Hashim ⓘ 🔣 sc Ⓟ (Ph.D.) is an Associate Professor of Marketing at the Graduate School of Business, National University of Malaysia. He has more than twenty years of experience as an academician, including a casual teaching stint in Australia. He holds a PhD in Marketing from the University of Melbourne and specializes in Product Innovation. Nik has published in leading international journals, including Journal of Retailing and Consumer Services, Review of Managerial Science, The Service Industries Journal, Journal of Relationship Marketing, International Journal of Emerging Markets, Journal of Financial Services Marketing, European Journal of International Management, and Renewable Energy, and has co-authored a textbook; Product Management and Strategy published by McGraw-Hill. He is keen on knowledge sharing and believes that real projects or practical applications should be embedded in the teaching process. Email: nikhaz@ukm.edu.my.

Privacy preserving human activity recognition framework using an optimized prediction algorithm

Kambala Vijaya Kumar, Jonnadula Harikiran
School of Computer Science and Engineering, VIT-AP University, Amaravathi, India

Article Info

ABSTRACT

Human activity recognition, in computer vision research, is the area of growing interest as it has plethora of real-world applications. Inferring actions from one or more persons captured through a live video has its immense utility in the contemporary era. Same time, protecting privacy of humans is to be given paramount importance. Many researchers contributed towards this end leading to privacy preserving action recognition systems. However, having an optimized model that can withstand any adversary models that strives to disclose privacy information. To address this problem, we proposed an algorithm known optimized prediction algorithm for privacy preserving activity recognition (OPA-PPAR) based on deep neural networks. It anonymizes video content to have adaptive privacy model that defeats attacks from adversaries. The privacy model enhances the privacy of humans while permitting highly accurate approach towards action recognition. The algorithm is implemented to realize privacy preserving human activity recognition framework (PPHARF). The visual recognition of human actions is made using an underlying adversarial learning process where the anonymization is optimized to have an adaptive privacy model. A dataset named human metabolome database (HMDB51) is used for empirical study. Our experiments with using Python data science platform reveal that the OPA-PPAR outperforms existing methods.

Corresponding Author:

Vijaya Kumar Kambala
School of Computer Science and Engineering, Vellore Institute of Technology, VIT-AP University
Amaravathi, Vijayawada, Andhra Pradesh, India
Email: kvkumar@pvpsiddhartha.ac.in

1. INTRODUCTION

Video based surveillance has become an important computer vision application. It has plenty of applications in the real world. While video based surveillance in different domains is useful, it has potential risk in terms of privacy leakage. Therefore, many researchers contributed towards privacy preserving action recognition. Human action recognition is an important research area with rich set of methods with machine learning, deep learning and generative adversarial network (GAN) based models. Action recognition using deep learning, often supported by privacy preserving method, are explored in [1]–[6]. Lyu *et al.* [1] proposed a deep learning based method for privacy preserving framework with fair and decentralized approach. Rasim *et al.* [2] proposed a deep learning based model for privacy preserving approach to protect personal data. Weng *et al.* [3] proposed a deep learning model with blockchain for privacy protection. Lyu *et al.* [4] studied federated cloud models to achieve fair and privacy preserving approaches to solve problems. Kumar *et al.* [5] explored deep learning algorithms and resolution images besides spatial relationships to recognize human actions. Rajpur *et al*. [6] proposed a cloud-based service to achieve privacy preserving action recognition using deep convolution neural network (CNN) model.

There are many adversarial models that paved way for human action recognition. They are found in [7]–[12] to mention few. Wu *et al*. [7] proposed a privacy-protective-generative adversarial network (PP-GAN) with modules such as regulator and verificator. It ensures protection of privacy, structure similarity and utility of the approach. Debie *et al*. [8] proposed a privacy preserving GAN for classification of ECG data. Maximov *et al*. [9] proposed a GAN based system known as conditional identity anonymization generative adversarial network (CIAGAN) which supports anonymization and recognition of actions in image and video. In future, they intend to enhance it with full image anonymization. Martinsson *et al*. [10] proposed an adversarial representation learning model with efficient management of learnable parameters. Li *et al*. [11] used a pre-trained GAN based model for privacy protection. Shirai and Whitehill [12] proposed a GAN based model for recognition of faces.

From the literature, it is understood that there are plenty of deep learning based methods for action recognition. Similarly, there are many GAN based approaches used for human activity recognition. Many of the deep learning and GAN based methods are equipped with privacy preserving approaches to protect data. However, there is need for optimization of action recognition method with privacy budget optimization. To address this problem, we proposed an algorithm known as optimized prediction algorithm for privacy preserving activity recognition (OPA-PPAR) based on deep neural networks. It anonymizes video content to have adaptive privacy model that defeats attacks from adversaries. The privacy model enhances the privacy of humans while permitting highly accurate approach towards action recognition. The algorithm is implemented to realize privacy preserving human activity recognition framework (PPHARF). The visual recognition of human actions is made using an underlying adversarial learning process where the anonymization is optimized to have an adaptive privacy model. A dataset named HMDB51is used for empirical study. Our contributions in this paper are: i) we proposed a framework known as PPHARF that leverages action recognition model, privacy budget model and anonymization model for privacy preserving with adversarial setting; ii) we proposed an algorithm known as OPA-PPAR based on deep neural networks; and iii) we built an application to evaluate the PPHARF and the underlying OPA-PPAR algorithm using HMDB51 dataset.

The remainder of the paper is structured in: section 2 review different kinds of methods used for action recognition and privacy preservation. Section 3 presents the proposed method with underlying algorithm. Section 4 presents experimental results and evaluates the same. Section 5 concludes the paper and gives suggestions for future work.

2. RELATED WORK

Human action recognition is an important research area with rich set of methods with machine learning, deep learning and generative adversarial network (GAN) based models. Many privacy preserving deep learning techniques are explored by Boulemtafes *et al*. [13]. Malekzadeh *et al*. [14] proposed privacy preserving based approach that makes use of deep autoencoder. Lyu *et al*. [1] proposed a deep learning-based method for privacy preserving framework with fair and decentralized approach. Rasim *et al*. [2] proposed a deep learning-based model for privacy preserving approach to protect personal data. Weng *et al*. [3] proposed a deep learning model with blockchain for privacy protection. Yonetani *et al*. [15] investigated on security using doubly permuted homomorphic encryption (DPHE) which is meant for protecting high-dimensional data. Lyu *et al*. [4] studied federated cloud models to achieve fair and privacy preserving approaches to solve problems. Yang *et al*. [16] employed machine learning (ML) models for hyperspectral image classification. Du *et al*. [17] proposed deep learning models with privacy preserving and also approximate approach in computing. Jhonson *et al*. [18] focused on the real time style transfer using perception loss and super-resolution. He *et al*. [19] proposed a method for image recognition based on deep residual learning. Kuehne *et al*. [20] worked on the video database known as HMDB that is used for human action recognition.

Yun *et al*. [21] focused on human activity recognition using multiple instance learning and body pose features. He *et al*. [22] exploited deep residual networks with identity mapping. Szegedy *et al*. [23] investigated on deep convolutional networks with action recognition using pre-recorded videos. Leenes *et al*. [24] studied on the privacy issues associated with data protection Dai *et al*. [25] proposed a novel method towards human action recognition with privacy preserved. Kumar *et al*. [5] explored deep learning algorithms and resolution images besides spatial relationships to recognize human actions. Orekondy *et al*. [26] proposed a model for visual privacy advisor that improves privacy of the system. Pittaluga *et al*. [27] focused on motion reconstruction of videos by using different image descriptors. Dai *et al*. [28] used spatial resolution cameras and extremely low temporal resolutions for activity recognition and preserving privacy. Dosovitskiy and Brox [29] investigated on convolutional networks for inverting of visual representations. Lyu *et al*. [30] proposed collaborative deep learning models for human activity recognition. Weinzaepfel *et al*. [31] exploited local descriptors in images to arrive at reconstruction of images for visual quality. Ryoo *et al*. [32] used superstitious video recordings in order to recognize human actions

from extreme low-resolution videos. Mahendran and Vedaldi [33] explored on the visualization of CNNs by using natural pre-images. Wang *et al*. [34] used coded aperture videos for human activity recognition with privacy preserved.

Machot *et al*. [35] investigated on sensor data in order to discover unseen activities associated with human action recognition. Pittaluga and Koppal [36] used miniature vision sensors proposed privacy preserving optics to strike balance between utility of videos and privacy. It has many applications like motion tracking, depth sensing and blob detection. Pittaluga *et al*. [37] did similar kind of work. Zhang *et al*. [38] proposed a methodology to identify human activities associated with fall detection of elderly people. Sur *et al*. [39] on the other hand proposed a technique to characterize given target using MIMO radar. Rajpur *et al*. [40] proposed a cloud-based service to achieve privacy preserving action recognition using deep CNN model. Cheng *et al*. [41] used a deep learning approach for emotion recognition. Riboni and Bettini [42] provided an ontology-based approach towards context aware activity recognition supported by hybrid reasoning. Xu *et al*. [43] defined an architecture for human activity recognition with two-stream spatiotemporal networks fully coupled.

Zolfaghari *et al*. [44] proposed smart activity recognition framework (SARF) that helps in monitoring humans that promote ambient assisted living (AAL). Youn *et al*. [45] focused on prognostics and health management that involves sensing functions, reasoning, prognostics, and health management. Ciliberto *et al*. [46] proposed a 3D model to have action recognition with privacy preserved. Cippitelli *et al*. [47] used skeletal data collected from sensors to detect human actions. Wang *et al*. [48] studied on gender bias elimination while making deep image representations.

Wu *et al*. [7] proposed a privacy-protective-GAN (PP-GAN) with modules such as regulator and verificator. It ensures protection of privacy, structure similarity and utility of the approach. It has issues with different head poses of humans in terms of face recognition that needs further improvement. Debie *et al*. [8] proposed a privacy preserving GAN for classification of ECG data. Maximov *et al*. [9] proposed a GAN based system known as CIAGAN which supports anonymization and recognition of actions in image and video. In future, they intend to enhance it with full image anonymization. Martinsson *et al*. [10] proposed an adversarial representation learning model with efficient management of learnable parameters. Tseng and Wu [49] proposed GAN known as "privacy generative adversarial network (CPGAN)" which is a learning framework with adversarial settings. Jin *et al*. [50] proposed Asynchronous Interactive GAN while Li *et al*. [11] used a pre-trained GAN based model for privacy protection. Ma *et al*. [51] defined yet another GAN model known as fusion GAN which makes use of a game between generator and discriminator. Shirai and Whitehill [12] proposed a GAN based model for recognition of faces. Liu *et al*. [52] explored adversarial networks for accuracy enhancement and privacy quantification.

Wu *et al*. [53] proposed GAN model for visual recognition while preserving privacy. They used the concept of restarting and ensemble approaches to leverage performance. Roy and Boddeti [54] proposed a non-zero sum game with adversarial settings. Zhang *et al*. [55] used adversarial learning mechanism to reduce unwanted biases in ML applications. Cheid *et al*. [56] proposed a protocol named multi-party classification that helps in human action recognition with privacy preserved. Cheid and Challal [57] investigated on human activity based on sensor based on sensors and privacy preserving protocols. Oh *et al*. [58] proposed a faceless person recognition and investigated on its implications. From the literature, it is understood that there are plenty of deep learning-based methods for action recognition. Similarly, there are many GAN based approaches used for human activity recognition. Many of the deep learning and GAN based methods are equipped with privacy preserving approaches to protect data. However, there is need for optimization of action recognition method with privacy budget optimization. Towards this end a framework is proposed in this paper.

3. MATERIALS AND METHOD
3.1. Problem definition

Given a video dataset (raw videos captured), denoted as X, which is subjected to action recognition task T with a privacy budget. The dataset X has set of class labels denoted by y_T and the performance of task is evaluated using a cost function denoted as L_T. An existing supervised learning method for prediction of actions is denoted as f_T which is enhanced to support J_B which is a cost function for budget associated with privacy leakage and used to find privacy leakage. Smaller value of J_B indicates that the input data has less private information associated with it. Table 1 shows the notations used in the paper.

Provided X, define an anonymization function f_A^* which transforms X into anonymized X denoted as $f_A^*(X)$ and a new deep learning based action recognition model, denoted as f_T^* is derived. In the process, care is taken to ensure that the function of f_T is affected minimally. This dual goal is to be achieved is considered as an optimization problem expressed in (1). The cost function of privacy budget is dynamic in nature as it

depends on the runtime task. Therefore, (1) is redefined and expressed as in (2). A fixed structure neural network, denoted as f_B, is defined in order to have finite search space to solve the problem with ease. This modification is expressed in (3). In order to enhance performance of the deep learning model, we proposed an ensemble approach.

Table 1. Notations used in the paper

Notation	Description
f_A^*	Anonymization function optimized
f_T^*	The new or derived deep learning method
f_T	An existing deep learning method for prediction
L_T	Cost function
f_A	Anonymization function
X	Raw video dataset
y_T	Set of labels of X
J_B	Cost function for privacy budget to find privacy leakage
$f_A^*(X)$	Anonymized input data X
f_B	A privacy budget model. It is a fixed structure neural network
θ_A	Represents learnable parameters of f_A
θ_B	Represents learnable parameters of f_B
θ_T	Represents learnable parameters of f_T
H_B	Negative entropy

$$f_A^*, f_T^* = argmin \left[L_T\left(f_T(f_A(X)), y_T\right) + \gamma_B^J\left(f_A(X)\right) \right] \tag{1}$$

$$f_A^*, f_T^* = argmin_{(f_A, f_T)} \left[L_T\left(f_T(f_A(X)), y_T\right) + \gamma sup_{f_B \epsilon p} J_B\left(f_B(f_A(X)), y_B\right) \right] \tag{2}$$

$$f_A^*, f_T^* = argmin_{(f_A, f_T)} \left[L_T\left(f_T(f_A(X)), y_T\right) + \gamma max_{f_B} J_B\left(f_B(f_A(X)), y_B\right) \right] \tag{3}$$

3.2. Proposed framework

We proposed a framework named PPHARF which is crucial for accommodating the underlying mechanisms and algorithms to achieve the desired dual goal of the system which enhances the action recognition with privacy leakage prevention and keeps the capabilities of prediction algorithm maximal. The framework has different models involved. They are known as the action recognition model f_T^*, an optimised anonymization function f_A^* and a privacy budget model denoted as f_B. These models are implemented as deep neural networks with learnable parameters. The training of the entire model is made with combination of two loss functions namely L_T and J_B. The underlying training in the framework has a dual goal consisting of achieving optimized anonymization function f_A^* which filters private information prior to the actual task and also ensures that $f_A^*(X)$ is achieved without limiting functionality of action recognition model.

As presented in Figure 1, the learned anonymization module takes X as input and transforms it into anonymized video content that filters out private information and modifies it so as to ensure that the video content is useful for action recognition, but unique human identity cannot be achieved. The anonymized video is subjected to action recognition model which is denoted as f_T. It has its cost function denoted as L_T. In the same fashion, the anonymized video content is subjected to privacy budget module f_B where another cost function denoted as J_B. When both cost functions are combined to form of a loss function which controls the iterative process of the framework and ensures optimization of action recognition while preserving privacy. The anonymization model f_A is implemented as a frame level filter which is based on 2D convolutional network. The action recognition module is taken from [59] and reused it. The privacy budget model f_B is made up of ResNet. For the same of action recognition, the video is divided into number of frames (video clips) and each frame is uniquely identified.

A minimax problem associated with (3) is solved by considering different learnable parameters of the three models used in the framework. The learnable parameters θ_A, θ_B, and θ_T are associated with f_A, f_B and f_T respectively. In order to solve the minimax problem, we considered the notion of alternative minimization found in [60]. It is expressed as in (4)-(6). Then the two loss functions are optimized to solve the optimization problems expressed in (7) and (8). The (7) is the minimization problem while (8) is minimax problem. The former is used to have training of f_A and f_T while the latter is used to keep track of different parameters of privacy budget model. In order to solve two loss functions two parameter update rules are expressed in (9)-(12).
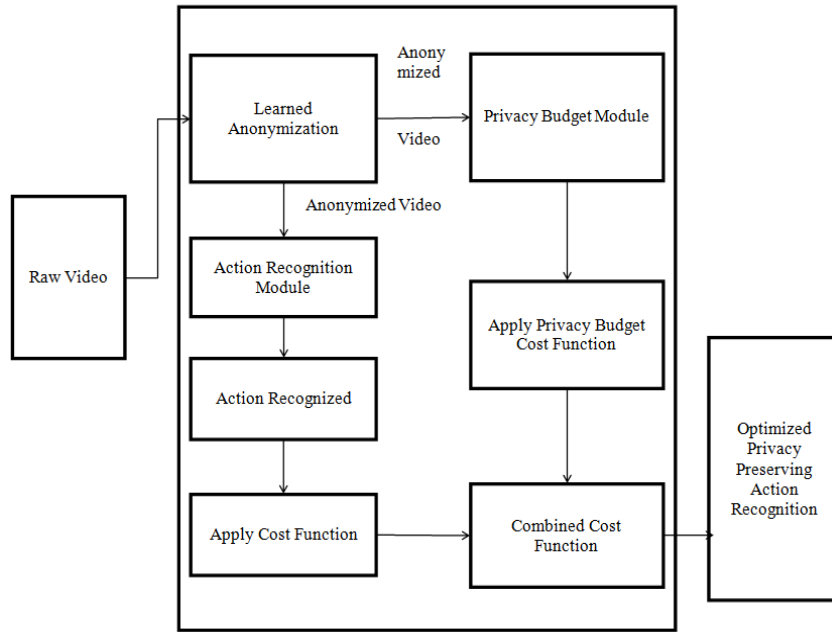
Figure 1. Proposed privacy preserving human activity recognition framework with adversarial setting

$$\theta_A \leftarrow \theta_A - \propto_A \nabla_{\theta_A}\big(L_T(\theta_A, \theta_T) - \gamma L_B(\theta_A, \theta_B)\big) \tag{4}$$

$$\theta_T \leftarrow \theta_T - \propto_T \nabla_{\theta_A} L_T(\theta_A, \theta_T) \tag{5}$$

$$\theta_B \leftarrow \theta_B - \propto_B \nabla_{\theta_B} L_B(\theta_A, \theta_B) \tag{6}$$

$$Q_A^*, Q_T^* = argmin_{(Q_A, Q_T)} L_T(\theta_A, \theta_T) \tag{7}$$

$$Q_B^*, Q_A^* = argmin_{\theta_B} argmax_{\theta_A} L_B(\theta_A, \theta_B) \tag{8}$$

$$\theta_A \leftarrow \theta_T \leftarrow \theta_A, \theta_T - \propto_T \nabla_{(Q_T, Q_A)} L_T(\theta_A, \theta_T) \tag{9}$$

$$j \leftarrow argmin_{i \in \{1, \dots K\}} L_B\big(\theta_A, \theta_B^i\big) \tag{10}$$

$$\theta_A, \theta_A + \propto_A \nabla_{\theta_A} L_B\big(\theta_A, \theta_B^j\big) \tag{11}$$

$$\theta_B^i, \theta_B^i - \propto_B \nabla_{\theta_B^i} L_B\big(\theta_A, \theta_B^j\big), \forall i \in \{1, \dots K\} \tag{12}$$

We found that (4) is instable which can be solved by considering negative entropy which is incorporated to have a new scheme as expressed in (13)-(15). With these optimizations, there is possibility of maximizing entropy that leverages performance. The (2) is further optimized with ensemble approach in the training process to improve model accuracy as expressed in (16). The ensemble model and optimized parameter settings are further improved with a scheme expressed in (17)-(19).

$$\theta_A \leftarrow \theta_A - \propto_A \nabla_{\theta_A}\big(L_T(\theta_A, \theta_T) - \gamma H_B(\theta_A, \theta_B)\big) \tag{13}$$

$$\theta_T, \theta_A \leftarrow \theta_T, \theta_A - \propto_T \nabla_{\theta_T, \theta_A} L_T(\theta_A, \theta_T) \tag{14}$$

$$\theta_B \leftarrow \theta_B - \propto_B \nabla_{\theta_B} L_B(\theta_A, \theta_B) \tag{15}$$

$$f_A^*, f_T^* = argmin_{(f_A, f_T)} \left[L_T\big(f_T\big(f_A(x)\big), y_T\big) + \gamma max_{f_B^i \in Pt} J_B\big(f_B^i\big(f_A(x)\big), y_B\big) \right] \tag{16}$$

$$\theta_A \leftarrow \theta_A - \propto_A \nabla_{\theta_A} \left(L_T + \gamma max_{\theta_B^i \in Pt} - H_B(\theta_A, \theta_B^j) \right) \qquad (17)$$

$$\theta_A \leftarrow \theta_T \leftarrow \theta_A \leftarrow \theta_T - \propto_T \nabla_{(Q_T, Q_A)} L_T(\theta_A, \theta_T) \qquad (18)$$

$$\theta_B^i, \theta_B^i - \propto_B \nabla_{\theta_B^i} L_B(\theta_A, \theta_B^j), \forall i \in \{1, \dots M\} \qquad (19)$$

With these optimizations, the proposed framework PPHARF is made more sophisticated in terms of human action recognition and preserving privacy that ensures non-disclosure of identity. With different modules in place, the framework operates in an iterative model in order to have better performance. With combined loss function it can realize the dual goal of the framework aforementioned.

3.3. The proposed algorithm

We proposed an algorithm known OPA-PPAR based on deep neural networks. It anonymizes video content to have adaptive privacy model that defeats attacks from adversaries. The privacy model enhances the privacy of humans while permitting highly accurate approach towards action recognition. The algorithm is implemented to realize PPHARF. The visual recognition of human actions is made using an underlying adversarial learning process where the anonymization is optimized to have an adaptive privacy model. A dataset named HMDB51 is used for empirical study.

Algorithm 1. Optimized prediction algorithm for privacy preserving activity recognition

```
Inputs:X, model learnable parameters such as θ_A, θ_B and θ_T
Output: Updated recognized actions map with privacy preserved
1.   Initialize frames vector F
2.   Initialize actions map R
3.   F←SplitVideo(X)
4.   For each frame f in F
5.      Repeat
6.         Apply learned anonymization model f_A on f
7.         Apply privacy budget model f_B on f
8.         Apply action recognition model f_T* optimized by f_A and f_B
9.   L_T←ComputeCostFunctionOfActionRecognition()
10.  J_B←ComputeCostFunctionOfPrivacyLeakage()
11.        loss function L ←L_T+J_B
12.        Use learnable parameters θ_A, θ_B and θ_T
13.        Get feedback for three models
14.     Until Convergence
15.     Update R
16.  End For
17.  Return R
```

As presented in Algorithm 1, it takes X and model learnable parameters such as θ_A, θ_B, and θ_T as inputs and produces an updated recognized action map with privacy preserved. In step 1, it initialized frames vector named F which holds frames (nothing but split films of video). Step 2 initializes actions map that will be updated iteratively and retuned on convergence. Step 3 splits given raw video into some frames. An iterative process is expressed in steps 4 through step 16. For each frame again, there is an iterative process that applies the two modules as given in step 6, step 7, and step 8 respectively. Two kinds of cost functions are computed in step 9 and step 10 respectively. These two cost functions are combined in step 11 to arrive at a combined loss function that is used in the training of the models in order to give feedback and continue process until convergence. Step 12 uses learnable parameters and step 13 gives feedback needed in the adversarial setting of the proposed framework. Step 7 returns final results that are obtained with privacy preserved.

4.   EXPERIMENTAL RESULTS

We proposed an algorithm known OPA-PPAR is evaluated using HMDB51 dataset. The results of OPA-PPAR is compared with that of our prior work named multi-task learning based hybrid prediction algorithm (MTL-HPA) and the state of the art method named gradient reversal layer (GRL) [61]. As shown in Figure 2, there are 51 action samples in HMDB51 dataset. Out of them 100 samples are used for empirical study in this paper. However, results are presented in this paper for 10 actions. They include climb, eat, jump, kiss, push, pushup, run, sit, smile and walk. As presented in Figure 3, the input images or frames are shown in left column and the action recognized and anonymized frame are shown in second and third columns

respectively. The experimental results are evaluated in terms of precision, recall and F1-Score. The performance values are obtained with human study on anonymized samples. The ground truth and prediction results of the action recognition methods are subjected to evaluation in terms of the measures.
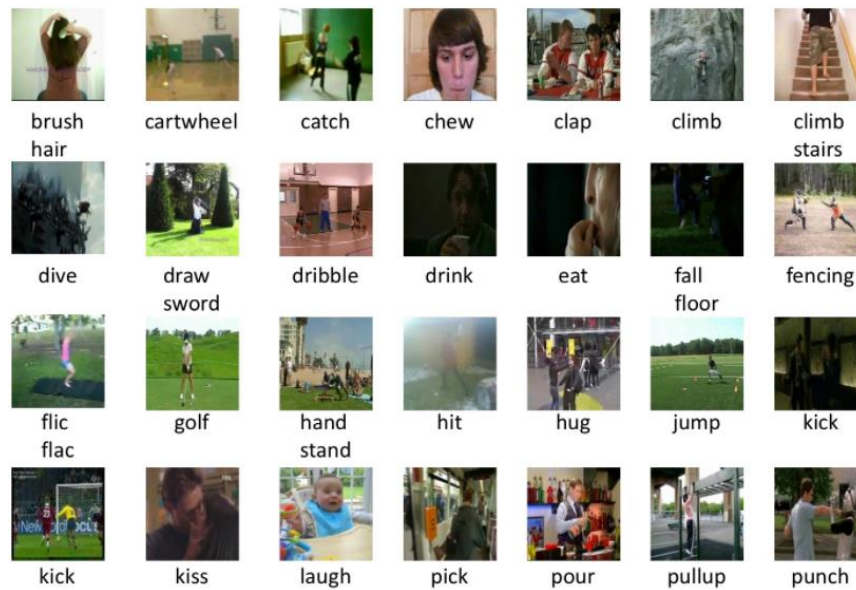

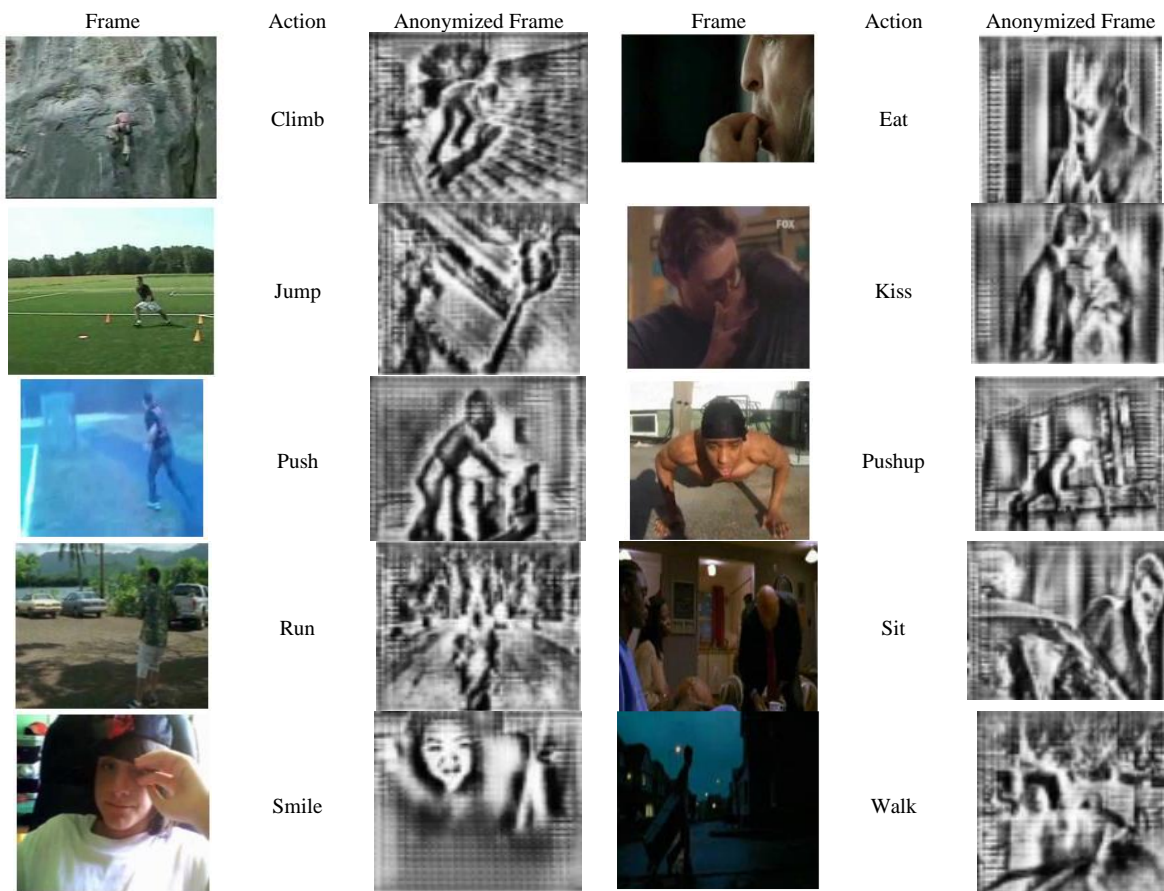Figure 2. Some human action samples present in HMDB51 dataset [39]

Figure 3. Experimental results for the selection actions

Figure 4 and Figure 5 show the performance comparison between the GRL vs. proposed method and MTL-HPA vs. the proposed method. In both the cases, the action recognition models are presented in horizontal axis and vertical axis shows the performance (%). Observations are made with 10 human actions. For each human action 100 experiments are made with the prototype made to demonstrate proof of the concept. Precision, recall and F1-score are computed based on ground truth and the results of the action recognition models. The final evaluation results are obtained with human study. The results revealed that the proposed method OPA-PPAR outperforms the existing methods known as GRL and MTL-HPA. The MTL-HPA showed significantly better performance over the baseline GRL method. The experimental results revealed that the proposed action recognition method not only preserves privacy and recognizes human actions but also has optimizations in terms of privacy budget and a combined loss function to guide the recognition process associated with the proposed framework. As presented in Table 2 and Table 3, the performance of the proposed method is compared with that of GRL and MTL-HPA.
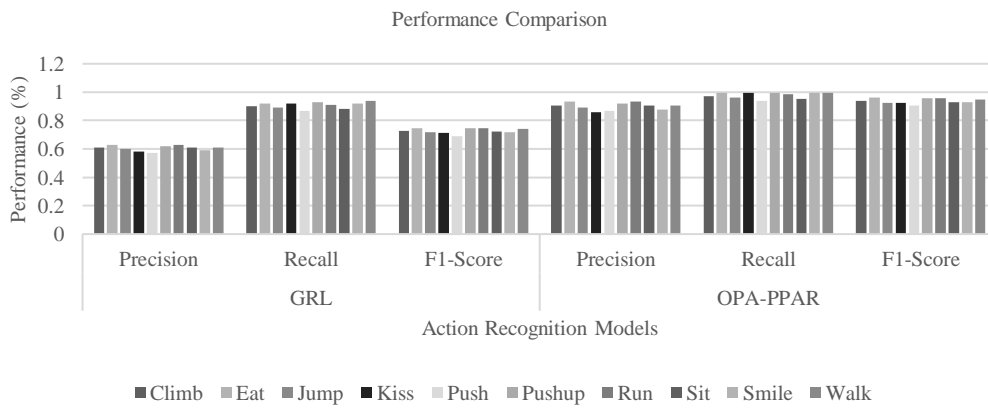

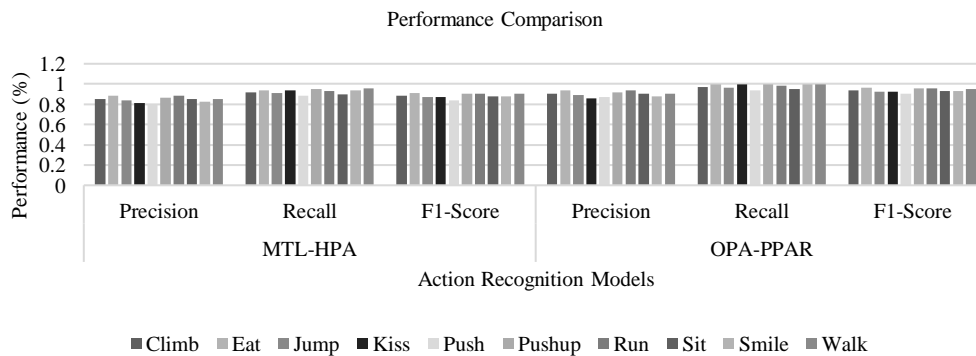Figure 4. Performance comparison of action recognition models GRL and OPA-PPAR

Figure 5. Performance comparison of action recognition models MTL-HPA and OPA-PPAR

Table 2. Results of the proposed method compared with that of GRL

Action	GRL Performance			OPA-PPAR Performance		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Climb	0.61	0.9	0.727152	0.90524	0.97308	0.937935
Eat	0.63	0.92	0.747871	0.93492	0.994704	0.963886
Jump	0.6	0.89	0.716779	0.8904	0.962268	0.92494
Kiss	0.58	0.92	0.711467	0.86072	0.994704	0.922874
Push	0.57	0.87	0.68875	0.86982	0.940644	0.903847
Pushup	0.62	0.93	0.744	0.92008	0.99603	0.95655
Run	0.63	0.91	0.744545	0.93492	0.983892	0.958781
Sit	0.61	0.88	0.720537	0.90524	0.951456	0.927773
Smile	0.59	0.92	0.71894	0.87556	0.994704	0.931337
Walk	0.61	0.94	0.739871	0.90524	0.997152	0.948976

Table 3. Results of the proposed method compared with that of our prior method MTL-HPA

Actions	MTL-HPA			OPA-PPAR		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Climb	0.854	0.918	0.884844	0.90524	0.97308	0.937935
Eat	0.882	0.9384	0.909326	0.93492	0.994704	0.963886
Jump	0.84	0.9078	0.872585	0.8904	0.962268	0.92494
Kiss	0.812	0.9384	0.870636	0.86072	0.994704	0.922874
Push	0.798	0.8874	0.840329	0.86982	0.940644	0.903847
Pushup	0.868	0.9486	0.906512	0.92008	0.99603	0.95655
Run	0.882	0.9282	0.90451	0.93492	0.983892	0.958781
Sit	0.854	0.8976	0.875257	0.90524	0.951456	0.927773
Smile	0.826	0.9384	0.87862	0.87556	0.994704	0.931337
Walk	0.854	0.9588	0.903371	0.90524	0.997152	0.948976

5. CONCLUSION AND FUTURE WORK

We proposed an algorithm known OPA-PPAR based on deep neural networks. It anonymizes video content to have adaptive privacy model that defeats attacks from adversaries. The privacy model enhances the privacy of humans while permitting highly accurate approach towards action recognition. The algorithm is implemented to realize PPHARF. The visual recognition of human actions is made using an underlying adversarial learning process where the anonymization is optimized to have an adaptive privacy model. A dataset named HMDB51 is used for empirical study. Our experiments with using Python data science platform reveal that the OPA-PPAR outperforms existing methods. It can be used in real world applications where PPHARF can fit seamlessly. The experimental results revealed that the proposed action recognition method not only preserves privacy and recognizes human actions but also has optimizations in terms of privacy budget and a combined loss function to guide the recognition process associated with the proposed framework. The proposed method paves way for further investigations in terms of optimizing the three models involved in the system.

REFERENCES

[1] L. Lyu *et al.*, "Towards fair and decentralized privacy-preserving deep learning with blockchain," *CoRR*, pp. 1–4, Jun. 2019, [Online]. Available: http://arxiv.org/abs/1906.01167.

[2] R. M. Alguliyev, R. M. Aliguliyev, and F. J. Abdullayeva, "Privacy-preserving deep learning algorithm for big personal data analysis," *Journal of Industrial Information Integration*, vol. 15, pp. 1–14, Sep. 2019, doi: 10.1016/j.jii.2019.07.002.

[3] J. Weng, J. Weng, J. Zhang, M. Li, Y. Zhang, and W. Luo, "DeepChain: auditable and privacy-preserving deep learning with blockchain-based incentive," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–1, 2019, doi: 10.1109/TDSC.2019.2952332.

[4] L. Lyu *et al.*, "Towards fair and privacy-preserving federated deep models," *IEEE Transactions on Parallel and Distributed Systems*, vol. 31, no. 11, pp. 2524–2541, Nov. 2020, doi: 10.1109/TPDS.2020.2996273.

[5] K. V. Kumar, D. J. Harikiran, M. A. R. Prasad, and U. Sirisha, "Privacy-preserving human activity recognition and resolution image using deep learning algorithms spatial relationship and increasing the attribute value in OpenCV," *International Journal of Advanced Science and Technology*, vol. 29, no. 7, pp. 514–523, 2020.

[6] A. S. Rajput, B. Raman, and J. Imran, "Privacy-preserving human action recognition as a remote cloud service using RGB-D sensors and deep CNN," *Expert Systems with Applications*, vol. 152, Aug. 2020, doi: 10.1016/j.eswa.2020.113349.

[7] Y. Wu, F. Yang, Y. Xu, and H. Ling, "Privacy-protective-GAN for privacy preserving face de-identification," *Journal of Computer Science and Technology*, vol. 34, no. 1, pp. 47–60, Jan. 2019, doi: 10.1007/s11390-019-1898-8.

[8] E. Debie, N. Moustafa, and M. T. Whitty, "A privacy-preserving generative adversarial network method for securing EEG brain signals," Jul. 2020, doi: 10.1109/IJCNN48605.2020.9206683.

[9] M. Maximov, I. Elezi, and L. Leal-Taixe, "CIAGAN: conditional identity anonymization generative adversarial networks," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 5446–5455, doi: 10.1109/CVPR42600.2020.00549.

[10] J. Martinsson, E. L. Zec, D. Gillblad, and O. Mogren, "Adversarial representation learning for synthetic replacement of private attributes," Jun. 2020, [Online]. Available: http://arxiv.org/abs/2006.08039.

[11] Q. Li, Z. Zheng, F. Wu, and G. Chen, "Generative adversarial networks-based privacy-preserving 3D reconstruction," in *2020 IEEE/ACM 28th International Symposium on Quality of Service (IWQoS)*, Jun. 2020, pp. 1–10, doi: 10.1109/IWQoS49365.2020.9213037.

[12] S. Shirai and J. Whitehill, "Privacy-preserving annotation of face images through attribute-preserving face synthesis," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Jun. 2019, vol. 2019-June, pp. 21–29, doi: 10.1109/CVPRW.2019.00009.

[13] A. Boulemtafes, A. Derhab, and Y. Challal, "A review of privacy-preserving techniques for deep learning," *Neurocomputing*, vol. 384, pp. 21–45, Apr. 2020, doi: 10.1016/j.neucom.2019.11.041.

[14] M. Malekzadeh, R. G. Clegg, and H. Haddadi, "Replacement autoencoder: a privacy-preserving algorithm for sensory data analysis," *Proceedings - ACM/IEEE International Conference on Internet of Things Design and Implementation, IoTDI 2018*, pp. 165–176, Oct. 2017, doi: 10.1109/IoTDI.2018.00025.

[15] R. Yonetani, V. N. Boddeti, K. M. Kitani, and Y. Sato, "Privacy-preserving visual learning using doubly permuted homomorphic encryption," in *Proceedings of the IEEE International Conference on Computer Vision*, Oct. 2017, vol. 2017-Octob, pp. 2059–2069, doi: 10.1109/ICCV.2017.225.

[16] M. Der Yang, K. S. Huang, Y. F. Yang, L. Y. Lu, Z. Y. Feng, and H. P. Tsai, "Hyperspectral image classification using fast and

adaptive bidimensional empirical mode decomposition with minimum noise fraction," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 12, pp. 1950–1954, Dec. 2016, doi: 10.1109/LGRS.2016.2618930.

[17] W. Du *et al.*, "Approximate to be great: communication efficient and privacy-preserving large-scale distributed deep learning in internet of things," *IEEE Internet of Things Journal*, vol. 7, no. 12, pp. 11678–11692, Dec. 2020, doi: 10.1109/JIOT.2020.2999594.

[18] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9906, Springer International Publishing, 2016, pp. 694–711.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.

[20] H. Kuehne, H. Jhuang, E. Garrote, T. Poggio, and T. Serre, "HMDB: a large video database for human motion recognition," in *2011 International Conference on Computer Vision*, Nov. 2011, pp. 2556–2563, doi: 10.1109/ICCV.2011.6126543.

[21] K. Yun, J. Honorio, D. Chattopadhyay, T. L. Berg, and D. Samaras, "Two-person interaction detection using body-pose features and multiple instance learning," in *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 2012, pp. 28–35, doi: 10.1109/CVPRW.2012.6239234.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9908, pp. 630–645, Mar. 2016, doi: 10.1007/978-3-319-46493-0_38.

[23] C. Szegedy *et al.*, "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 1–9, doi: 10.1109/CVPR.2015.7298594.

[24] R. Leenes, R. van Brakel, S. Gutwirth, and P. De Hert, *Data Protection and Privacy: (In)visibilities and Infrastructures*, vol. 36. Cham: Springer International Publishing, 2017.

[25] J. Dai, B. Saghafi, J. Wu, J. Konrad, and P. Ishwar, "Towards privacy-preserving recognition of human activities," in *2015 IEEE International Conference on Image Processing (ICIP)*, Sep. 2015, pp. 4238–4242, doi: 10.1109/ICIP.2015.7351605.

[26] T. Orekondy, B. Schiele, and M. Fritz, "Towards a visual privacy advisor: understanding and predicting privacy risks in images," in *Proceedings of the IEEE International Conference on Computer Vision*, Oct. 2017, vol. 2017-Octob, pp. 3706–3715, doi: 10.1109/ICCV.2017.398.

[27] F. Pittaluga, S. J. Koppal, S. B. Kang, and S. N. Sinha, "Revealing scenes by inverting structure from motion reconstructions," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 145–154, Apr. 2019, doi: 10.1109/CVPR.2019.00023.

[28] J. Dai, J. Wu, B. Saghafi, J. Konrad, and P. Ishwar, "Towards privacy-preserving activity recognition using extremely low temporal and spatial resolution cameras," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 2015, pp. 68–76, doi: 10.1109/CVPRW.2015.7301356.

[29] A. Dosovitskiy and T. Brox, "Inverting visual representations with convolutional networks," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2016, pp. 4829–4837, doi: 10.1109/CVPR.2016.522.

[30] L. Lyu, X. He, Y. W. Law, and M. Palaniswami, "Privacy-preserving collaborative deep learning with application to human activity recognition," in *International Conference on Information and Knowledge Management, Proceedings*, Nov. 2017, pp. 1219–1228, doi: 10.1145/3132847.3132990.

[31] P. Weinzaepfel, H. Jégou, and P. Pérez, "Reconstructing an image from its local descriptors," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2011, pp. 337–344, doi: 10.1109/CVPR.2011.5995616.

[32] M. S. Ryoo, B. Rothrock, C. Fleming, and H. J. Yang, "Privacy-preserving human activity recognition from extreme low resolution," Apr. 2016, [Online]. Available: http://arxiv.org/abs/1604.03196.

[33] A. Mahendran and A. Vedaldi, "Visualizing deep convolutional neural networks using natural pre-images," *International Journal of Computer Vision*, vol. 120, no. 3, pp. 233–255, May 2016, doi: 10.1007/s11263-016-0911-8.

[34] Z. W. Wang, V. Vineet, F. Pittaluga, S. N. Sinha, O. Cossairt, and S. B. Kang, "Privacy-preserving action recognition using coded aperture videos," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Jun. 2019, pp. 1–10, doi: 10.1109/CVPRW.2019.00007.

[35] F. Al Machot, M. R. Elkobaisi, and K. Kyamakya, "Zero-shot human activity recognition using non-visual sensors," *Sensors*, vol. 20, no. 3, p. 825, Feb. 2020, doi: 10.3390/s20030825.

[36] F. Pittaluga and S. J. Koppal, "Privacy preserving optics for miniature vision sensors," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Jun. 2015, pp. 314–324, doi: 10.1109/CVPR.2015.7298628.

[37] F. Pittaluga and S. J. Koppal, "Pre-capture privacy for small vision sensors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 11, pp. 2215–2226, Nov. 2017, doi: 10.1109/TPAMI.2016.2637354.

[38] C. Zhang, Y. Tian, and E. Capezuti, "Privacy preserving automatic fall detection for elderly using RGBD cameras," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 7382, no. 1, Springer Berlin Heidelberg, 2012, pp. 625–633.

[39] S. N. Sur, S. Bera, S. Shome, R. Bera, and B. Maji, "Target characterization using MIMO radar," *International Journal on Smart Sensing and Intelligent Systems*, vol. 13, no. 1, pp. 1–8, 2020, doi: 10.21307/ijssis-2019-013.

[40] Miran Kim, Xiaoqian Jiang, "HEAR: Human Action Recognition via Neural Netwroks on Homomorphically Encrypted Data", arXiv:2104.0916v1 [cs.CR].

[41] B. Cheng *et al.*, "Robust emotion recognition from low quality and low bit rate video: a deep learning approach," in *2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*, Oct. 2017, pp. 65–70, doi: 10.1109/ACII.2017.8273580.

[42] D. Riboni and C. Bettini, "COSAR: hybrid reasoning for context-aware activity recognition," *Personal and Ubiquitous Computing*, vol. 15, no. 3, pp. 271–289, Mar. 2011, doi: 10.1007/s00779-010-0331-7.

[43] M. Xu, A. Sharghi, X. Chen, and D. J. Crandall, "Fully-coupled two-stream spatiotemporal networks for extremely low resolution action recognition," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Mar. 2018, pp. 1607–1615, doi: 10.1109/WACV.2018.00178.

[44] S. Zolfaghari and M. R. Keyvanpour, "SARF: smart activity recognition framework in ambient assisted living," in *Proceedings of the 2016 Federated Conference on Computer Science and Information Systems, FedCSIS 2016*, Oct. 2016, pp. 1435–1443, doi: 10.15439/2016F132.

[45] B. D. Youn *et al.*, "Statistical health reasoning of water-cooled power generator stator bars against moisture absorption," *IEEE Transactions on Energy Conversion*, vol. 30, no. 4, pp. 1376–1385, Dec. 2015, doi: 10.1109/TEC.2015.2444873.

[46]   M. Ciliberto, D. Roggen, and F. J. O. Morales, "Exploring human activity annotation using a privacy preserving 3D model," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, Sep. 2016, pp. 803–812, doi: 10.1145/2968219.2968290.
[47]   E. Cippitelli, S. Gasparrini, E. Gambi, and S. Spinsante, "A human activity recognition system using skeleton data from RGBD sensor," *Computational Intelligence and Neuroscience*, vol. 2016, pp. 1–14, 2016, doi: 10.1155/2016/4351435.
[48]   T. Wang, J. Zhao, M. Yatskar, K.-W. Chang, and V. Ordonez, "Balanced datasets are not enough: estimating and mitigating gender bias in deep image representations," in *Proceedings of the IEEE International Conference on Computer Vision*, Nov. 2018, pp. 5309–5318, doi: 10.1109/ICCV.2019.00541.
[49]   B. W. Tseng and P. Y. Wu, "Compressive privacy generative adversarial network," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2499–2513, 2020, doi: 10.1109/TIFS.2020.2968188.
[50]   X. Jin, Z. Chen, and W. Li, "AI-GAN: asynchronous interactive generative adversarial network for single image rain removal," *Pattern Recognition*, vol. 100, Apr. 2020, doi: 10.1016/j.patcog.2019.107143.
[51]   J. Ma, W. Yu, P. Liang, C. Li, and J. Jiang, "FusionGAN: a generative adversarial network for infrared and visible image fusion," *Information Fusion*, vol. 48, pp. 11–26, Aug. 2019, doi: 10.1016/j.inffus.2018.09.004.
[52]   S. Liu, A. Shrivastava, J. Du, and L. Zhong, "Better accuracy with quantified privacy: representations learned via reconstructive adversarial network," Jan. 2019, [Online]. Available: http://arxiv.org/abs/1901.08730.
[53]   Z. Wu, Z. Wang, Z. Wang, and H. Jin, "Towards privacy-preserving visual recognition via adversarial training: a pilot study," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11220 LNCS, Springer International Publishing, 2018, pp. 627–645.
[54]   P. C. Roy and V. N. Boddeti, "Mitigating information leakage in image representations: a maximum entropy approach," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 2581–2589, doi: 10.1109/CVPR.2019.00269.
[55]   B. H. Zhang, B. Lemoine, and M. Mitchell, "Mitigating unwanted biases with adversarial learning," in *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, Dec. 2018, pp. 335–340, doi: 10.1145/3278721.3278779.
[56]   Z. Gheid, Y. Challal, X. Yi, and A. Derhab, "Efficient and privacy-aware multi-party classification protocol for human activity recognition," *Journal of Network and Computer Applications*, vol. 98, pp. 84–96, Nov. 2017, doi: 10.1016/j.jnca.2017.09.005.
[57]   Z. Gheid and Y. Challal, "Novel efficient and privacy-preserving protocols for sensor-based human activity recognition," in *Proceedings - 13th IEEE International Conference on Ubiquitous Intelligence and Computing, 13th IEEE International Conference on Advanced and Trusted Computing, 16th IEEE International Conference on Scalable Computing and Communications, IEEE International*, Jul. 2017, pp. 301–308, doi: 10.1109/UIC-ATC-ScalCom-CBDCom-IoP-SmartWorld.2016.0062.
[58]   S. J. Oh, R. Benenson, M. Fritz, and B. Schiele, "Faceless person recognition: privacy implications in social media," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9907, Springer International Publishing, 2016, pp. 19–35.
[59]   D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 4489–4497, doi: 10.1109/ICCV.2015.510.
[60]   Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," *32nd International Conference on Machine Learning, ICML 2015*, vol. 2, pp. 1180–1189, 2015.
[61]   D.-Z. Du, *Minimax and its applications*. Springer US, 1995.

BIOGRAPHIES OF AUTHORS

Vijaya Kumar Kambala 🆔 [G] [SC] [P] working as Part Time Research Scholar in School of CSE VIT-AP. he is working as Assistant Professor, in PVP Siddhartha Institute of Technology, Vijayawada. His research interest includes Image Processing, Video Analysis, human activity recognition, privacy preserving human activity recognition. He can be reached at email: vijaya.18phd7017@vitap.ac.in.

Jonnadula Harikiran 🆔 [G] [SC] [P] working as Associate professor, School of CSE, Vellore Institute of Technology, VIT-AP. His reserach interest include microarray image analysis, Hyperspectral Imaging and Video Analytics. He can be reached at email: harikiran.j@vitap.ac.in.

An enhanced support vector regression model for agile projects cost estimation

Assia Najm[1], Abdelali Zakrani[2], Abdelaziz Marzak[1]
[1]Department of Mathematics and Computer Sciences, FSB M'sik, Hassan II University, Casablanca, Morocco
[2]Department of Computer Science Engineering, ENSAM, Hassan II University, Casablanca, Morocco

ABSTRACT

The appearance of agile software development techniques (ASDT) since 2001 has encouraged many organizations to move to an agile approach. ASDT presents an opportunity for researchers and professionals, but it has many challenges as well. One of the most critical challenges is agile effort prediction. Hence, many studies have investigated agile software development cost estimation (ASDCE). The objective of this study is twofold: First, to propose an improved model based on support vector regression with radial bias function kernel (SVR-RBF) enhanced by the optimized artificial immune network (Optainet). Second, to perform a detailed comparative analysis of the proposed method compared to other existing optimization techniques in the literature and applied for ASDCE. The experimental evaluation was carried out by assessing the performance of the proposed method using some trusted measures like standardized accuracy (SA), mean absolute error (MAE), prediction at level p (Pred(p)), mean balanced relative error (MBRE), mean inverted balanced relative error (MIBRE), and logarithmic standard deviation (LSD). Throughout a dataset with 21 agile projects using the leave-one-out cross-validation (LOOCV) technique. The results obtained prove that the proposed model enhances the accuracy of the SVR-RBF model, and it outperforms the majority of existing models in the literature.

This is an open access article under the [CC BY-SA](#) license.

Corresponding Author:

Assia Najm
Department of Mathematics and Computer Sciences, FSB Msik, Hassan II University
Casablanca, Morocco
Email: assia.najm-etu@etu.univh2c.ma

1. INTRODUCTION

In the past twenty years and since the announcement of twelve principles of the agile manifesto in 2001, researchers have shown an increased interest in agile software development. Agile is a generic term that encompasses different methods, approaches, practices, and techniques that meet manifesto principles and values [1]. The agile software development techniques (ASDT) have many attractive advantages like the incremental and iterative aspect of development, quick releases, and good adaptation to the volatile nature of customer requirements. Among the well-known ASDT, we found feature-driven development [2], eXtreme programming [3], and the scrum framework [4]. Nevertheless, for all these ASDT, one of the most severe challenges is to produce an initial reliable and accurate estimate of the project to be developed.

The primary element to estimate the needed development effort is the project size. Principally, story point is a highly employed measurement to deduce the size and the complexity of an agile project. Besides, the velocity factor refers to the number of story points conveyed by the team in a sprint. It significantly

impacts the development effort of agile projects. Thus, we used these two measures to predict the effort needed using ASDT.

Generally, cost estimates have an important impact on the process of software development [5]. In fact, accurate estimation leads to the success of projects, while inaccurate predictions lead to the failure of projects [6]. For example, underestimation results in less budget and fewer resources leading to a project with low quality that fails to be accomplished within estimated deadlines. While overestimation leads to expecting more resources than what is needed. Therefore, seeking an efficient technique that produces accurate estimates is useful and of high importance. Especially with the study reported by the standish-group-international (SGI) [7], where more than sixty percent of projects partially or completely fail (19% failed, 44% remains challenged, and only 37% successful).

According to Britto *et al.* [8], various algorithmic and non-algorithmic techniques have been used for agile software cost estimation (ASDCE) purposes. Unfortunately, the accuracy of effort estimations is still a significant challenge. Moreover, it did not achieve acceptable estimation accuracy. This result has been also confirmed by the systematic literature review conducted by Fernandez-Diego *et al.* [9]. In the same study, the authors reported some accuracy improvement; however, many papers present unacceptable estimations. This is why there is still a need to enhance accuracy by proposing new improved ASDCE models or optimizing the existing ones.

Furthermore, prediction techniques couldn't be assessed without applying suitable accuracy measures. According to [9], the most commonly used measures are based on the median or the mean of magnitude of relative error (MMRE, MdMRE), with 35 studies from 73 and Pred (25%). However, considering the inconvenience of MMRE and its sensitivity to outliers, further measures like the percentage of accuracy, mean absolute error (MAE), and R2 have been used in conjunction to raise the appropriateness of the accuracy. That's why our second goal from this study is to assess the performance of our proposed model using some trusted criteria as recommended in many papers in the literature like in [10]–[12].

Up to now, some studies have investigated the use of the support vector regression (SVR) model for agile see [13]–[17] thanks to its power in dealing with the agile software effort estimation concerns. However, an optimal SVR model is still needed. In this sense, Zakrani *et al.* [17] have proposed an optimized SVR model using a grid search (GS) method to tune the hyperparameters of the SVR model where every hyperparameter combination is evaluated. This exhaustive search is perceived as inefficient, the time will exponentially grow according to the dimension of evaluations. Also, the population size is user-defined and not adjustable which affects the efficiency of the algorithm and might not find the true optimum.

This present study proposes an improved SVR model with RBF (SVR-RBF) kernel for ASDCE based on the Optainet algorithm. The Optainet [18] is used to tune the SVR-RBF hyperparameters to find the best configuration model that gives accurate estimates. The Optainet explores the effectiveness of the aiNet theory, which simulates the human immune system activity (it has a robust memory and a remarkable ability to differentiate the self cells from the foreign ones). The Optainet algorithm has many advantages regarding the GS method: i) Inclusion of stopping criteria which controls the time of execution, ii) Ability to check and maintain various optimal solutions, iii) Capacity to exploit and explore the whole space of search, and iv) The capability of dynamic adjustment for the population size.

Our major objectives through this study are:
− We suggested an improved SVR-RBF model using the optimized artificial immune network to seek optimal parameters of the SVR-RBF model. This proposed hybrid model enhances the performance of the estimations.
− We applied a very known and trusted empirical evaluation protocol using the following criteria: Pred, LSD, MIBRE, MBRE, and MAE.
− We performed a robust comparative empirical study to assess the suggested model's accuracy compared to the accuracy of other techniques.

To synthesize the goal from this study, we aim during our analysis to answer the following three questions (Qus):
− Qu1: Does the Optainet enhance the performance of the predictions produced using SVR-RBF?
− Qu2: How are the estimations of the SVR-RBF using Optainet in comparison with other optimization methods?
− Qu3: Is the performance of our model more accurate than the other models?

The remainder of this study is organized in: section 2 highlights the previous ASDCE-related works. Section 3 describes the suggested model and the used optimization technique. Section 4 presents the overall experimental evaluation while the obtained results are outlined in section 5, and section 6 reports the conclusion.

2.   RELATED WORKS

Multiple papers have addressed ASDCE problems. For example, in [9], 73 relevant studies have been identified in a systematic literature review (SLR) on ASDCE papers published between 2014 and 2020. This SLR presents an update of Britto *et al*. [8] SLR study on ASDCE methods published between 2001 and November 2013 (where 25 papers were selected). According to [8], various techniques have been used for ASDCE purposes, but the most used techniques are those qualified as expert-based prediction techniques. Moreover, most methods did not achieve an acceptable estimation accuracy regarding the observed difference between the real effort value and the predicted one. Notably, the accuracy achieved by the majority of studies did not meet the suggested 25% threshold indicated in [19] for assessing the accuracy of cost prediction. For example, there are commonly accepted thresholds for MMRE and Pred measures, i.e an acceptable MMRE value should be under or equal to 25%, and an acceptable Pred (0.25) should be above or equal to 75%.

In [9], six ASDT were identified: extreme programming (XP), Scrum, feature-driven development, distributed agile software development, Kanban, and agile unified process. Overall, expert-based techniques played a significant role in cost prediction such as the planning poker method, which is intently related to story points sizing metric. Besides, the estimation accuracy challenge, some studies report a suitable value of accuracy, while many papers stated insufficient precision. Also, about 29% of studies used the accuracy measures, while nearly 18% of studies used effect size metric to compare methods. Although ASDT remains dependent on expert judgment (notably, 24.66% of studies used the planning poker method), it is revealed that there has been a significant increase in the number of papers using machine learning techniques, exactly 20.55% of papers [9]. Next, we briefly introduce some examples.

In [20], the authors suggested a model for ASDCE based on user stories, which is adapted to the agile specificity (iterative and adaptive methodology). The empirical evaluation was made based on 21 software agile projects and using MMRE and Pred accuracy measures, and the results show acceptable effort estimations values based on the two metrics. The work in [21], proposed a new formula using story point and velocity variables to estimate the cost. The combination of two optimization techniques, namely particle swarm optimization (PSO) and bee colony, was employed to determine and optimize the formula's parameters. The results revealed that the proposed technique performs better than the models proposed in [20], [22] based on MAR, R2, Pred (8%), MdMRE, and MMRE.

In [14], the authors attempted to enhance the prediction of ASDT through the use of SVR with several kernels. The model is then evaluated using 21 agile projects. The obtained results demonstrate that SVR-RBF outperforms the other models based on MMRE and Pred (25%) measures. In the same vein, Zakrani *et al*. [17] proposed an improved version of the SVR-RBF model, which uses the grid search optimization method to tune the key hyperparameters. The experimental evaluation was made based on 21 agile projects, and then the model was validated using the LOO-CV method. The results showed that the accuracy of SVR-RBF was improved, and it outperformed other studies in terms of MdMRE, MMRE, and Pred (25%).

Nonetheless, the accuracy reported in the previous studies still needs further enhancement and a robust comparative study based on reliable accuracy measures and a significant statistical test. This is exactly what motivated our present study where we optimize SVR-RBF by Optainet and conduct a comprehensive comparative analysis. The following subsections provide more details about the proposed model.

3.   THE PROPOSED METHOD: SVR-RBF-OPTAINET
3.1. SVR model: an overview

The support vector machine (SVM) is a powerful machine learning (ML) model that has offered valuable results in both classification and regression problems [23]. The SVM approach has several attractive features like the sparse way of presenting solutions, good generalization ability, and the capacity to avoid local minimums, thanks to the structural risk minimization concept. In this paper, we employ ε-SVR that introduces the ε-insensitive loss function as shown in Figure 1. It means that all errors less than a defined threshold (inside bars in the figure) are neglected, while errors induced by the points located out of the bars are calculated using ψ and ψ^* like in Figure 1. In the case of nonlinear regression, the following (1) is used where φ denotes a nonlinear function that maps the low input space to the high output space; w represents the weights vector, and c is the threshold.

$$h(x) = w^T \varphi(x) + c \tag{1}$$

We mention that w and c are selected for optimizing the next problem [24]:

Minimizing $_{\mathrm{w,b},\psi,\psi*}$

$$\frac{1}{2} < w, w > \; + B \sum_{i=1}^{l} (\Psi_i + \Psi_i^*)$$

Subjects to

$$(< w, \varphi(x_i) > +c) - y_i \leq \varepsilon + \psi_i,$$

$$y_i - (< w, \varphi(x_i) > + c) \leq \varepsilon + \Psi_i^*,$$

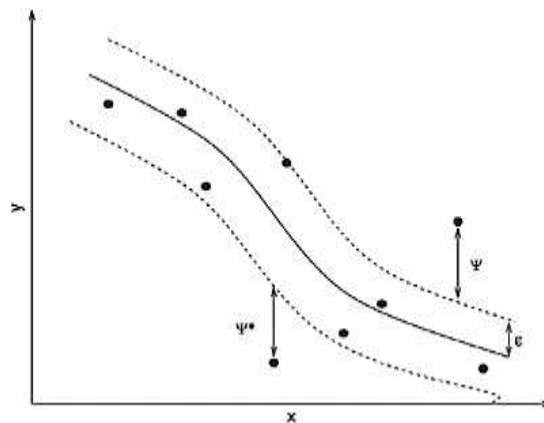$$\psi_i, \Psi_i^*, \geq 0. \tag{2}$$

Figure 1. The ε-SVR regression

The ε represents the function's deviation, the regularization parameter B reflects the trade-off between the error's tolerance above ε and the flatness of h, the ψ, ψ^* as proposed in [25] presents the slack variables that define the tolerated deviations above the error ε. The base mechanism of the SVR model is minimizing the objective function which takes into account the norm of w and the loss of ψ_i, $and\ \Psi_i^*$ as expressed in (2). We note that the Lagrangian multiplier is used in the SVR method, it is related to the dot product of φ (x). It can be realized through the kernel function determined as: $H(x_i, x_j)=<\varphi(x_i), \varphi(x_j)>$ which prevents the explicit computation of the φ (x). We refer the reader to [24] for more details. In the present study, we used the SVR-RBF towing to its ability to generate good estimates [11], [12]. The computation of the SVR-RBF kernel is done via the expression: $H(x_i, x_j)=exp(-\lambda\|x_i-x_j\|^2)$. Hence, parameter λ should be cautiously chosen beside the B and the ε parameters.

3.2. Optainet algorithm

The artificial immune network (aiNet) model was introduced by Castro and Zuben [26]. It is a graph composed of nodes (antibodies) and edges. It is essentially based on the immune system (IS) theory [27] where the IS generates many antibodies or attempts to provide the best-suited antibodies for attacking antigens. The modelization of this phenomenon corresponds to a function optimization process. Many algorithms inspired by the biological domain and especially by the IS were used to achieve global optimization like the opt-IA [28], the B-Cell algorithm [29], the opt-aiNet [18], and the Clonalg algorithm [30].

In this work, the SVR-RBF with Optainet optimization [18] is used to optimize the SVR-RBF parameters. The Optainet explores the effectiveness of the aiNet theory, which simulates the human immune system activity (it has a robust memory and a remarkable ability to differentiate the self cells from the foreign ones). The Optainet algorithm has the following advantages:

− Inclusion of stopping criteria.
− Ability to check and maintain various optimal solutions.

− Capacity to exploit and explore the whole space of search.
− The capability of dynamic adjustment for the population size.

We adopt the following terminology to explain the Optainet algorithm:

− Cells: Each cell in the network is composed of population values. In the Euclidean space, it is denoted with a multivalued vector.
− Cell's: Fitness stands for the objective function value using a particular network cell.
− Cells: Affinities represent the value of distance (Euclidean) that exists between two cells.
− Cloning: Cloning is producing replication of original cells (or parents).
− Mutation: The generated copies (or offspring) will be mutated to be different from their parents (mutation).

The overall Optainet steps are described as noted below:

BEGIN

a) Randomly initialize the population.
b) If the stopping criteria are not satisfied, do:
− Calculate the fitness of each cell and evaluate it according to the used objective function. Then proceed to its normalization.
− Perform cell cloning according to the chosen Nc, which represents the number of offsprings (or clones).
− Performing the mutation operation, where each clone will be inversely proportional to its parent fitness value as expressed below:

$$cl' = cl + \sigma N(0,1) \tag{3}$$

$$\sigma = (1/\eta)exp(-h^*)$$

where $N(0,1)$ presents the random gaussian variable with standard deviation equal to 1 and a zero mean, cl is the parent cell, and the cl' is the cell resulting from the mutation of cl, also $h^* \in [0,1]$ represents the normalized fitness of a cell, while the parameter η controls the exponential function's decay. Acceptable mutations are those that are within the domain interval.

− Calculate the fitness of each cell present in the population (including cloned and mutated ones).
− Select the highest fitness cells per clone and exclude the others.
− Evaluating all cell affinities, then suppressing the cells having affinities lower than a predetermined suppression minimum σs (threshold).
− Back into the second step, after adding a p% of random cells.
c) Else, selecting the highest fitness cell and Ending the algorithm.

END

3.3. Optainet based SVR-RBF model
3.3.1. Cell design

We used the ε-SVR model with RBF kernel to implement the proposed model. So, the parameters (B, λ, ε), should be cautiously chosen and carefully optimized using our suggested Optainet-based model. For that reason, each cell in the network includes four elements: the B, λ, ε, and features bits. Figure 2 shows the cell design representation. The three first parts represent the SVR-RBF parameters.
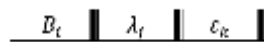

Figure 2. The cell with three parts, the B, λ, ε

3.3.2. Fitness function

Various criteria were suggested and employed in the literature to evaluate the performance of software cost prediction models. While Pred, MMRE, and MdMRE represent the frequently used ones in many studies [31]. For example, Braga *et al.* [32], and Shin and Goel [33]. Therefore, in this study, we employ a fitness function based on two criteria (4) where we excluded MMRE because it is susceptible to bias. An acceptable value for Pred (25%) is more than or equal to 75%, while for MMRE, MdMRE values must be less than or equal to 0.25. So, we look for cells with higher values of Pred and small values of MdMRE.

$$Fitness\_function = (100 - Pred(25\%)) + MdMRE \tag{4}$$

3.3.3. SVR-RBF-optainet design

We describe in this subsection the architecture of our model for parameter optimization. Especially, the design of the SVR-RBF-Optainet model. Figure 3 shows the main steps of our proposed models.
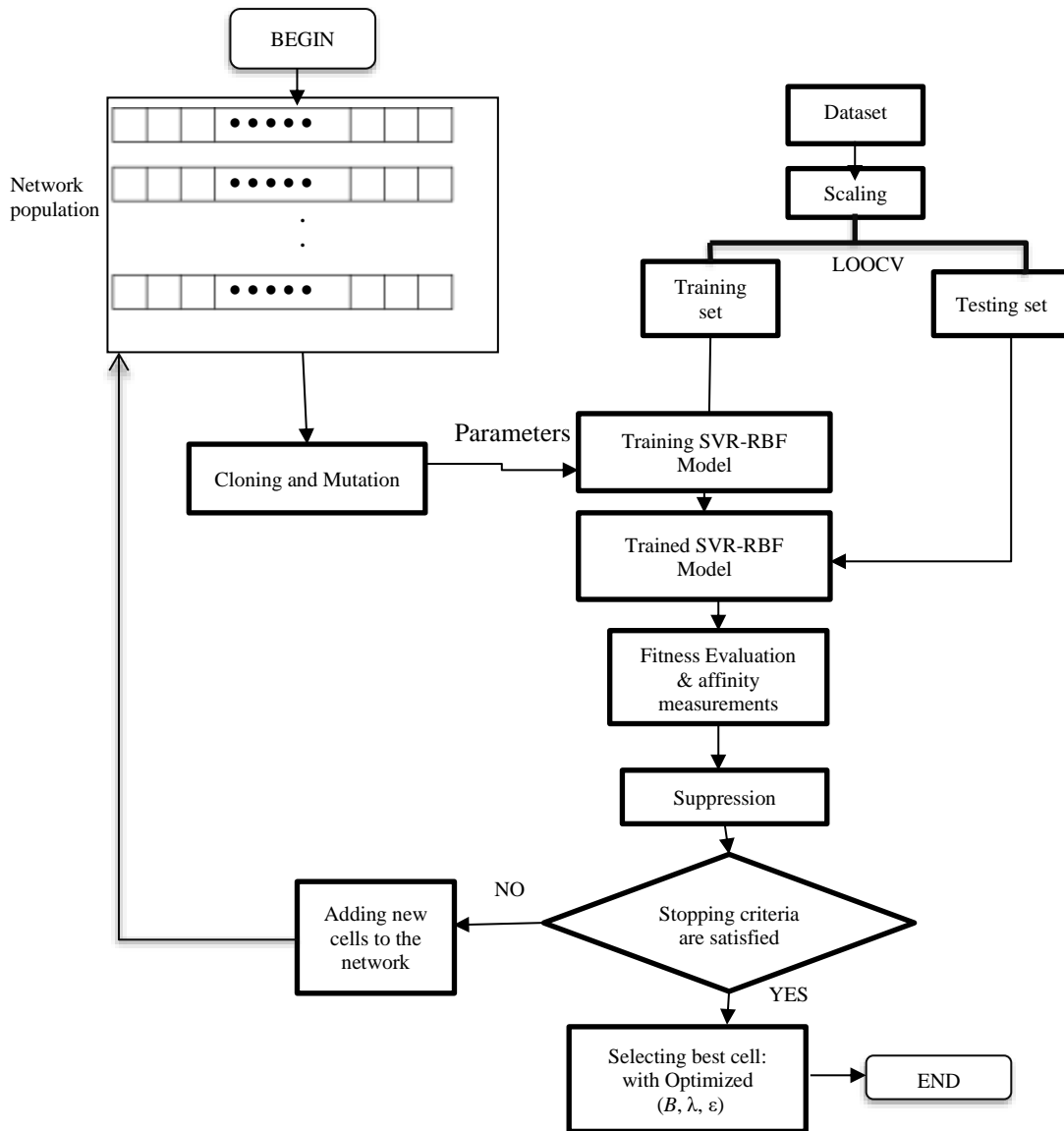

Figure 3. Design of the SVR-RBF-Optainet model

The main steps of our proposed model are described next:
− Data scaling: We firstly proceed to the data-preprocessing phase, where we normalize all the available data in the range between 0 and 1 to strengthen the accuracy of the regressor. The expression used for normalization is noted below:

$$y^* = \frac{y - y_{min}}{y_{max} - y_{min}} \tag{5}$$

where y_{max} is the maximal data value in the dataset, y_{min} is the minimal data value, y is the original value, and y^* is the value after normalization.

− After that, the main steps of the Optainet algorithm were performed: Cloning, mutation, fitness evaluation, affinity measurement, and network suppression as shown in Section 3.2. The output of this algorithm is finding the best solution with parameter optimization.

The Optainet algorithm seeks the optimal SVR-RBF configuration by exploring the parameters search space provided in Table 1. This is done by respecting the fitness function. Hence, the optimal solution that maximizes the fitness function is then chosen.

Table 1. The values of SVR-RBF model and Optainet algorithm

Methods	Hyperparameters
Optainet	Number of iteration={500}
	Size of population={50}
	Number of clones={20}
	The threshold of suppression={0.8}
	Newcomers percentage={50%}
SVR	Type={eps-regression}
	Kernel function={RBF}
	Range values of Hyparameters
	Complexity=[1000, 1500, 3000, 4000, 5000]
	Kernel parameter=[0.9,1,2]
	Epsilon=[0.01, 0.02,0.03]

4. EMPIRICAL DESIGN

4.1. Description of dataset

We employ a dataset that includes twenty-one agile projects with two features: the velocity and the story points. A summarized statistical description is presented in Table 2. It includes the number of features, the dataset size, min, max, mean, median, kurtosis, and skewness of cost. According to kurtosis and skewness values, we notice that the distribution of the cost is not normal besides the weak existence of outliers.

Table 2. Descriptive statistics of the dataset

Elements	Values					Cost	
Dataset size	21	Mini	Maxi	Mean	Median	Skewness	Kurtosis
Features' number	2	21	112	56.43	52	0.6049	-0.7679

4.2. Performance measures and validation method

Concerning the validation method, we employ the LOOCV. The LOOCV is chosen for multiple reasons: it is qualified as a deterministic method, unlike other techniques of validation [34], it produces estimations with large variance and less bias [35]. The LOOCV process is made by excluding one project (test project) from the dataset and performing the estimations with the rest (training set).

Miyazaki *et al.* [36] proposed two accuracy metrics: the mean inverted balanced relative error (MIBRE), and the mean balanced relative error (MBRE) (7) and (8) accordingly. The authors claim that these proposed measures are insensitive to asymmetry and bias contrary to the MMRE criteria (11) which is vulnerable to bias and outliers. Additionally, Shepperd and MacDonell [10] have proposed a novel accuracy criterion, which is the standardized accuracy (SA) (14) which relies on the MAE (13). The SA indicates if the estimated method is better in comparison to randomly guess (PT_0). Therefore, and as reported in [10] a value close to one means that the used method performs better than random guesses. Where MAE_{PT_0} represents the average of a large number of randomly guessing runs. In particular, we used 1000 for this work, MAE_{PT_i} represents the average of the absolute error of a specific estimation method i.

$$AbE_i = |y_i - \widehat{y_i}| \tag{6}$$

$$MIBRE = \frac{1}{N} \sum_{i=1}^{N} \frac{AbE_i}{\max(y_i, \widehat{y_i})} \tag{7}$$

$$MBRE = \frac{1}{N} \sum_{i=1}^{N} \frac{AbE_i}{\min(y_i, \widehat{y_i})} \tag{8}$$

$$MRE = \frac{AE_i}{y_i} \tag{9}$$

An enhanced support vector regression model for agile projects cost estimation (Assia Najm)

$$Pred(25\%) = \frac{1}{N}\sum_{i=1}^{N} \begin{cases} 1 \; if \; MRE_i \leq 25\% \\ \quad 0 \; otherwise \end{cases} \tag{10}$$

$$MMRE = \frac{1}{N}\sum_{i=0}^{N} MRE \tag{11}$$

$$MdMRE = median(MRE) \tag{12}$$

$$MAE = \frac{1}{N}\sum_{i=0}^{N} AbE_i \tag{13}$$

$$SA = 1 - \frac{MAE_{PT_i}}{\overline{MAE_{PT_0}}} \tag{14}$$

$$LSD = \sqrt{\frac{\sum_{i=1}^{n}(\delta_i + \frac{z^2}{2})^2}{n-1}} \tag{15}$$

$$\delta_i = \ln(y_i) - \ln(\hat{y_i}) \tag{16}$$

$$\Delta = \frac{MAR - \overline{MAR_{PT_0}}}{SPT_0} \tag{17}$$

In this present paper, we used an additional criterion (15): the logarithmic standard deviation (LSD) to assess the performance of estimation methods, where y_i presents the actual cost while $\hat{y_i}$ is the ith project's estimated cost, and z^2 is the estimation related to the δ_i variance (16). Besides, we used the effect size metric (Δ) as expressed in (17) to test if the generated estimates are produced by chance or not. The SPT_0 represents the standard deviation of randomly guessing. The interpretation of the values of (Δ) as explained in [37] is: large when the value is near to 0.8, medium if near to 0.5, and small if Δ is near to 0.2. So, we can deduce that the used technique performs better than the baseline technique if its Δ value exceeds 0.5.

4.3. Statistical test

Even though the employed accuracy measures are helpful to compare different cost prediction models (i.e. checking if one method is better than the other), it remains necessary to perform the statistical test to assess the observed difference, i.e., it is significant or not [38]. We mention that we used the Wilcoxon test; it enables us to evaluate the difference between methods using their absolute error (AbE) (6). The significance level used is 0.05.

5.　RESULTS AND DISCUSSION

We discuss in the present section all findings resulting from the assessment of the proposed method and the three compared techniques: the SVR-RBF-GS [17], the regression model of Zia's work [20], the SVR-RBF-Weka which means the SVR-RBF evaluated using default values of the weka tool. First, we tune our model using the Optainet algorithm to get the best model. Second, we evaluate our method and the three aforementioned methods (used in comparison) using Δ and SA metrics. We reported the obtained values in Table 3.

Table 3. The evaluation of each technique based on the SA and the effect size-the highest values are in bold

Approaches	SA	Δ	References
SVR-RBF-Optainet	**0.896**	**6.231**	-
SVR-RBF-GS	0.893	6.138	[17]
Zia *et al.* regression	0.865	5.755	[20]
SVR-RBF-Weka	0.195	1.261	-

A large Δ value means that the assessed technique doesn't produce prediction by chance. Also, a large value of SA means that the evaluated technique generates acceptable estimates. Therefore, in conformity with LOOCV and the values of Table 3, all techniques outperform the random guess (SA>0). Yet, The SVR-RBF-Optainet holds the highest SA value (SA=89.6%), and the SVR-RBF-Weka model holds the lowest one (SA=0.195) despite its better performance against random guessing. All techniques in Table 3

have a large Δ value (Δ >0.5) while SVR-RBF-Optainet retains the largest one (Δ=6.23). Overall, all generated estimates weren't made by chance.

We note the insufficiency of SA and Δ measures to draw any conclusions concerning the accuracy of our method in comparison to the others. Thus, we applied additional metrics to rank all used techniques throughout the Borda counting method as shown in Table 4. According to Table 4, we recognize the superiority of our method regarding others. The SVR-RBF-Optainet was ranked first, followed by SVR-RBF-GS while the SVR-RBF-Weka can be considered as the worst method for being classed last. In sum, the obtained results reveal that our proposed model, besides all compared methods, outperforms the random guess (SA>0). Also, SVR-RBF-Optainet is the best (ranked first). To make clear and justify the yielded conclusion, the statistical Wilcoxon test was performed. This test was made based on the absolute error AbE and with 0.05 as a significant level. Table 5 shows the resulting p-values. We notice that SVR-RBF-Optainet significantly outperforms the SVR-RBF-Weka. This result confirms its ranking obtained from the Borda count method. However, the difference in performance regarding the others (p-values) is not statistically significant.

Table 4. The ranks of methods based on borda count

Ranks	Techniques	References
1	SVR-RBF-Optainet	-
2	SVR-RBF-GS	[17]
3	Zia *et al.* regression	[20]
4	SVR-RBF-Weka	-

Table 5. The *p*-values of methods based on statistical test.

Methods	SVR-RBF-Weka	Zia *et al.* regression	SVR-RBF-GS
p-values	0.0000	0.1228	0.7525

To deeply analyze the performance of our proposed model, we compare it to many studies reported recently in literature which is the case for the following papers: [14], [17], [20]–[22], [39]. However, these articles applied various performance metrics and validation techniques. For instance, in [21], the authors didn't mention the technique used for validation, and they calculated the Pred (8%) rather than the frequently employed prediction which is Pred (0.25). Also, in [39], the employed measures are MdMER and MMER rather than MdMRE and MMRE. Even though we attempted to compare our model to the previously cited studies by being consistent with the experimental design which is commonly used in cost prediction as shown in Table 6.

Table 6. The comparison between methods based on the values of MdMRE, MMRE, and Pred

Techniques	Pred (25%)	MdMRE	MMRE	References
SVR-RBF-Optainet (the proposed model)	100	0.0392	0.0596	-
	Pred (0.08)=71.428	0.0397: MdMER	0.0602: MMER	
SVR-RBF-GS	100	0.0426	0.0620	[17]
	Pred (0.08)=66.667	0.0408: MdMER	0,0613: MMER	
SVR-RBF-Weka	28.571	0.4710	0.4671	-
SVR-RBF	95.9052	NA	0.0747	[14]
Zia *et al.* regression model	57.14	0.0714	0.0719	[20]
ABC-PSO	100	0.0333	0.0569	[21]
	Pred (8%)=66.667	0.0344: MdMER	0,0564: MMER	
DT*	38.09	NA	NA	[39]
SGB*	85.71	NA	NA	
RF*	66.67	NA	NA	
GRNN**	85.9182	NA	0.3581	[22]
PNN**	87.6561	NA	1.5776	
GMDH-PNN**	89.6689	NA	0.1563	
CCNN**	94.7649	NA	0.1486	

*DT: Decision Tree, SG: Stochastic Gradient Boosting, RF: Random Forest.
**GRNN: General Regression Neural Network, PNN: Probabilistic Neural Network, GMDH-PNN: GMDH PolynomialNeural Network, CCNN: Cascade-Correlation Neural Network.

We observe from Table 6 that our model performs better than all compared models based on all metrics, apart from the ABC-PSO model in terms of MMRE, and MdMRE (where it has a slightly higher MdMRE (+0.006), and MMRE (+0.002)). Even if our model produces similar Pred (25%) than that of the

ABC-PSO, we observe that it outperforms it greatly based on the value of Pred (8%) with exactly 71.42% for SVR-RBF-Optainet against 66.67% for ABC-PSO) so with a remarkable improvement of (+4.76). We notice that the ABC-PSO method can be excluded from the comparison since the author didn't mention which validation technique was employed. So, they might have used the whole dataset to train the model and test it as well. Finally, from our conducted analysis, we conclude that the Optainet-based SVR-RBF is a promising agile effort estimation model that can produce reliable and accurate estimates.

6. CONCLUSION

The present work proposed an enhanced model for ASDCE purposes. This model is based on the optimized ainet that tunes the parameters of the SVR-RBF model. We assessed our model's performance and made a detailed comparison analysis utilizing the trusted protocol that is based on: (SA, Δ, Borda counting and significance test) and by using the most used criteria, which are the Pred (25%), MdMRE, and MMRE. The findings indicate the superiority of our method regarding others. According to the statistical test results, the SVR-RBF-Optainet was ranked first based on the Borda counting method, and it significantly outperforms the SVR-RBF. Additionally, The SVR-RBF-Optainet outperforms the SVR-RBF regarding the Pred (25%) (+0.42) and MMRE (-0.1). Moreover, our model performs better than all compared models based on all metrics, apart from the ABC-PSO model in terms of MMRE, and MdMRE. In addition, the SVR-RBF-Optainet outperforms the SVR-RBF-GS, and all SVR models using different kernel optimization. Besides, the SVR-RBF-Optainet outperforms greatly Zia's regression based on the MdMRE, MMRE, and Pred. Also, it performs better than the ABC-PSO regression technique in terms of Pred (8%). Finally, we can conclude that the Optainet based SVR-RBF model is a powerful technique. However, we are unable to confirm its superiority in all situations. So, further research should be done using various datasets to generalize the results.

REFERENCES

[1] *Agile practice guide. Newtown square*, Project Ma. Pennsylvania: The Project Management Institute, 2017.
[2] S. R. Palmer and M. Felsing, *A practical guide to feature-driven development*. Pearson Education, 2002.
[3] K. Beck, *Extreme programming explained: embrace change*. 2000.
[4] K. Schwaber, "SCRUM development process," in *Business Object Design and Implementation*, London: Springer London, 1997, pp. 117–134.
[5] A. Trendowicz and R. Jeffery, *Software project effort estimation*. Cham: Springer International Publishing, 2014.
[6] A. ZAKRANI, M. HAIN, and A. IDRI, "Improving software development effort estimating using support vector regression and feature selection," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 8, no. 4, pp. 399–410, Dec. 2019, doi: 10.11591/ijai.v8.i4.pp399-410.
[7] L. M. Alves, G. Souza, P. Ribeiro, and R. J. Machado, "Longevity of risks in software development projects: a comparative analysis with an academic environment," *Procedia Computer Science*, vol. 181, pp. 827–834, 2021, doi: 10.1016/j.procs.2021.01.236.
[8] R. Britto, V. Freitas, E. Mendes, and M. Usman, "Effort estimation in global software development: a systematic literature review," in *2014 IEEE 9th International Conference on Global Software Engineering*, Aug. 2014, pp. 135–144, doi: 10.1109/ICGSE.2014.11.
[9] M. Fernandez-Diego, E. R. Mendez, F. Gonzalez-Ladron-De-Guevara, S. Abrahao, and E. Insfran, "An update on effort estimation in agile software development: a systematic literature review," *IEEE Access*, vol. 8, pp. 166768–166800, 2020, doi: 10.1109/ACCESS.2020.3021664.
[10] M. Shepperd and S. MacDonell, "Evaluating prediction systems in software project estimation," *Information and Software Technology*, vol. 54, no. 8, pp. 820–827, Aug. 2012, doi: 10.1016/j.infsof.2011.12.008.
[11] M. Hosni, A. Idri, A. Abran, and A. B. Nassif, "On the value of parameter tuning in heterogeneous ensembles effort estimation," *Soft Computing*, vol. 22, no. 18, pp. 5977–6010, Sep. 2018, doi: 10.1007/s00500-017-2945-4.
[12] A. Idri and I. Abnane, "Fuzzy analogy based effort estimation: an empirical comparative study," in *2017 IEEE International Conference on Computer and Information Technology (CIT)*, Aug. 2017, pp. 114–121, doi: 10.1109/CIT.2017.29.
[13] E. Scott and D. Pfahl, "Using developers' features to estimate story points," in *Proceedings of the 2018 International Conference on Software and System Process*, May 2018, pp. 106–110, doi: 10.1145/3202710.3203160.
[14] S. M. Satapathy, A. Panda, and S. K. Rath, "Story point approach based agile software effort estimation using various SVR kernel methods," in *The 26th International Conference on Software Engineering and Knowledge Engineering (SEKE 2014)*, 2014, pp. 304–307.
[15] S. Porru, A. Murgia, S. Demeyer, M. Marchesi, and R. Tonelli, "Estimating story points from issue reports," in *Proceedings of the The 12th International Conference on Predictive Models and Data Analytics in Software Engineering*, Sep. 2016, pp. 1–10, doi: 10.1145/2972958.2972959.
[16] O. Malgonde and K. Chari, "An ensemble-based model for predicting agile software development effort," *Empirical Software Engineering*, vol. 24, no. 2, pp. 1017–1055, Apr. 2019, doi: 10.1007/s10664-018-9647-0.
[17] A. Zakrani, A. Najm, and A. Marzak, "Support vector regression based on grid-search method for agile software effort prediction," in *2018 IEEE 5th International Congress on Information Science and Technology (CiSt)*, Oct. 2018, pp. 1–6, doi: 10.1109/CIST.2018.8596370.
[18] L. N. de Castro and J. Timmis, "An artificial immune network for multimodal function optimization," in *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, vol. 1, pp. 699–704, doi: 10.1109/CEC.2002.1007011.
[19] S. D. Conte, H. E. Dunsmore, and V. Y. Shen, *Software engineering metrics and models*. Benjamin-Cummings Publishing Co.,

Inc.Subs. of Addison-Wesley Longman Publ. Co390 Bridge Pkwy. Redwood City, CAUnited States, 1986.

[20]   Ziauddin, S. K. Tipu, and S. Zia, "An effort estimation model for agile software development," *Advances in Computer Science and its Applications (ACSA)*, vol. 2, no. 1, pp. 314–324, 2012.

[21]   T. T. Khuat and M. H. Le, "A novel hybrid ABC-PSO algorithm for effort estimation of software projects using agile methodologies," *Journal of Intelligent Systems*, vol. 27, no. 3, pp. 489–506, Jul. 2018, doi: 10.1515/jisys-2016-0294.

[22]   A. Panda, S. M. Satapathy, and S. K. Rath, "Empirical validation of neural network models for agile software effort estimation based on story points," *Procedia Computer Science*, vol. 57, pp. 772–781, 2015, doi: 10.1016/j.procs.2015.07.474.

[23]   V. N. Vapnik, *The nature of statistical learning theory*. New York, NY: Springer New York, 2000.

[24]   A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, Aug. 2004, doi: 10.1023/B:STCO.0000035301.49549.88.

[25]   C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.

[26]   L. Nunes de Castro and F. J. V. Zuben, "aiNet," in *Data Mining*, IGI Global, 2002, pp. 231–260.

[27]   N. K. Jerne, "Towards a network theory of the immune system," *Annales d'immunologie*, vol. 125C, no. 1–2, pp. 373–389, Jan. 1974, [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/4142565.

[28]   V. Cutello, G. Nicosia, and M. Pavone, "Exploring the capability of immune algorithms: a characterization of hypermutation operators," in *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2004, pp. 263–276.

[29]   J. Kelsey and J. Timmis, "Immune inspired somatic contiguous hypermutation for function optimisation," in *Genetic and Evolutionary Computation {\textemdash} {GECCO} 2003*, Springer Berlin Heidelberg, 2003, pp. 207–218.

[30]   L. N. de Castro and F. J. Von Zuben, "The clonal selection algorithm with engineering applications," 2000.

[31]   C. J. Burgess and M. Lefley, "Can genetic programming improve software effort estimation? A comparative evaluation," *Information and Software Technology*, vol. 43, no. 14, pp. 863–873, Dec. 2001, doi: 10.1016/S0950-5849(01)00192-6.

[32]   P. L. Braga, A. L. I. Oliveira, G. H. T. Ribeiro, and S. R. L. Meira, "Bagging predictors for estimation of software project effort," in *2007 International Joint Conference on Neural Networks*, Aug. 2007, pp. 1595–1600, doi: 10.1109/IJCNN.2007.4371196.

[33]   M. Shin and A. L. Goel, "Empirical data modeling in software engineering using radial basis functions," *IEEE Transactions on Software Engineering*, vol. 26, no. 6, pp. 567–576, Jun. 2000, doi: 10.1109/32.852743.

[34]   E. Kocaguneli and T. Menzies, "Software effort models should be assessed via leave-one-out validation," *Journal of Systems and Software*, vol. 86, no. 7, pp. 1879–1890, Jul. 2013, doi: 10.1016/j.jss.2013.02.053.

[35]   E. Kocaguneli, T. Menzies, and J. W. Keung, "On the value of ensemble effort estimation," *IEEE Transactions on Software Engineering*, vol. 38, no. 6, pp. 1403–1416, Nov. 2012, doi: 10.1109/TSE.2011.111.

[36]   Y. Miyazaki, A. Takanou, H. Nozaki, N. Nakagawa, and K. Okada, "Method to estimate parameter values in software prediction models," *Information and Software Technology*, vol. 33, no. 3, pp. 239–243, Apr. 1991, doi: 10.1016/0950-5849(91)90139-3.

[37]   J. Cohen, "A power primer," *Psychological Bulletin*, vol. 112, no. 1, pp. 155–159, 1992, doi: 10.1037/0033-2909.112.1.155.

[38]   J. Demsar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

[39]   S. M. Satapathy and S. K. Rath, "Empirical assessment of machine learning models for agile software development effort estimation using story points," *Innovations in Systems and Software Engineering*, vol. 13, no. 2–3, pp. 191–200, Sep. 2017, doi: 10.1007/s11334-017-0288-z.

BIOGRAPHIES OF AUTHORS

Assia Najm ⓘ 🔗 SC Ⓟ is a Ph.D. student at Hassan II university at Casablanca, department of Mathematics and Computer Sciences. She received an engineer's degree in telecommunication and technology of information from INPT, Rabat, Morocco, in 2014. Her research interests include software cost estimation, software metrics, fuzzy logic, decision trees, and optimization. She can be contacted at email: assia.najm-etu@etu.univh2c.ma.

Dr. Abdelali Zakrani ⓘ 🔗 SC Ⓟ is an associate professor at Hassan II university at Casablanca, He received the B.Sc. degree in Computer Science from Hassan II University, Casablanca, Morocco, in 2003, and his DESA degree (M.Sc.) and Ph.D. in the same major from University Mohammed V, Rabat, in 2005 and 2012 respectively. His research interests include software cost estimation, software metrics, fuzzy logic, neural networks, decision trees. He can be contacted at email: zakrani@gmail.com

Dr. Abdelaziz Marzak ⓘ 🔗 SC Ⓟ is currently a professor of computer science at the Faculty of Science Ben M'sik Casablanca, where he is the head of technologies of information and communication. He has several publications on the Web and databases. He received a Ph.D. in computer science in 2000 and a Ph.D. in computer science in 1997 both from Mohammed V University at Rabat, Morocco. He can be contacted at email: marzak@hotmail.com

❏   276

Model optimisation of class imbalanced learning using ensemble classifier on over-sampling data

Yulia Ery Kurniawati, Yulius Denny Prabowo
Department of Informatics, Faculty of Computers Science and Design, Institut Teknologi dan Bisnis Kalbis, Jakarta, Indonesia

Article Info

ABSTRACT

Data imbalance is one of the problems in the application of machine learning and data mining. Often this data imbalance occurs in the most essential and needed case entities. Two approaches to overcome this problem are the data level approach and the algorithm approach. This study aims to get the best model using the pap smear dataset that combined data levels with an algorithmic approach to solve data imbalanced. The laboratory data mostly have few data and imbalance. Almost in every case, the minor entities are the most important and needed. Over-sampling as a data level approach used in this study is the synthetic minority oversampling technique-nominal (SMOTE-N) and adaptive synthetic-nominal (ADASYN-N) algorithms. The algorithm approach used in this study is the ensemble classifier using AdaBoost and bagging with the classification and regression tree (CART) as learner-based. The best model obtained from the experimental results in accuracy, precision, recall, and f-measure using ADASYN-N and AdaBoost-CART.

Corresponding Author:

Yulia Ery Kurniawati
Department of Informatics, Faculty of Computer Science and Design, Institut Teknologi dan Bisnis Kalbis
Jalan Pulomas Selatan Kav 22, Kayu Putih, Pulogadung, Jakarta Timur 13210, Indonesia
Email: yulia.kurniawati@kalbis.ac.id

1.   INTRODUCTION

One of the problems of machine learning and data mining is imbalanced data. Imbalanced occurs when there is disproportion among the number of examples of each class in the dataset [1] and usually in the most essential and needed entities. It will be a complicated issue when dealing with the multiclass problem. It will be hard to acknowledge a priori of the multi-majority and multi-minority classes that should be stressed during the learning stage. For example, machine learning in data mining has difficulty classifying minority classes or classes with the smallest number of instances because the algorithm assumes that the class distribution is balanced. So that in some cases, there are errors in classifying the results for each class. The result is errors in the classification of minority classes due to the class imbalance that tends to focus on the majority class and ignore the minority class at the time of classification. The imbalanced data can be found in many areas such as medical [2], [3], abnormal electricity consumption [4], price forecasting [5], credit evaluation [6], and cyanobacteria bloom [7].

There are two approaches to solving this problem in dealing with class imbalance: the data level approach, the algorithmic approach, and hybrid-based approaches [8], [9]. The data-level approach can use the sampling method. This data sampling method is divided into two: the sampling method in the minority class (over-sampling) [10], [11], and the majority class sampling method (under-sampling) [12], [13]. Meanwhile, the algorithm approach is an approach by designing new algorithms or refining existing algorithms, and it uses the ensemble method. Ensemble methods use one set of classifiers to make a

prediction. The generalisation ability of the ensemble is generally much stronger than the individual ensemble members [8]. There is two ensemble categorisation, parallel and sequential ensemble. The parallel ensemble obtains base learners in parallel, for example, bagging [6], [14]. In comparison, the sequential ensemble produces base learners sequentially, where the previous base learner influences the next generation of learners, for example, by using adaptive boosting (AdaBoost).

Kurniawati *et al.* on adaptive synthetic-nominal (ADASYN-N) and adaptive synthetic-KNN (ADASYN-KNN) for multiclass imbalance learning on laboratory test dat proposed ADASYN-N in their study in 2018 [2]. It can handle nominal data types that ADASYN proposed by He *et al.* [11] cannot. This study used an over-sampling method to solve cases of class imbalance in the pap-smear result dataset [15]. The over-sampling methods used are synthetic minority oversampling technique-nominal (SMOTE-N), ADASYN-N, and ADASYN-KNN. The result is that ADASYN-N performed better than SMOTE-N on all performance matrices for NBC.

Fithrasari *et al.* on handling imbalance data in classification model with nominal predictors in 2020, studied handling imbalanced data in classification models with nominal predictors [16]. They used Survei Kinerja dan Akuntabilitas Kependudukan Keluarga Berencana dan Pembangunan Keluarga (SKAP KKBPK) data Jawa Timur Province in 2018. ADASYN-N, SMOTE-N, and SMOTE-N-ENN were used for imbalanced dataset handling then tested using classification and regression trees (CART). ADASYN-N gave the best average area under the curve (AUC) compared with SMOTE or synthetic minority oversampling technique-nominal edited nearest neighbor (SMOTE-N-ENN). It could increase accuracy from 0.737 to 0.963.

Rachburee and Punlumjeak on oversampling technique in student performance classification from engineering course, conducted a study to combine oversampling methods with several classifier models [17]. The oversampling methods that were used were SMOTE, Borderline-SMOTE, SVMSOTE, and ADASYN. The classifier models were applied using MLP, gradient boosting, AdaBoost, and random forest. The result was Borderline-SMOTE gave the best result among other models.

The absence of further research to find the best model based on the pap-smear result dataset [15], so this study will combine the over-sampling methods, which is a data-level approach with an algorithm approach. This algorithm approach used an ensemble classifier, AdaBoost.M1 and bagging, and based learner used CART. This study chose CART because CART and decision tree are unstable learning algorithms, and the ensemble method can improve the generalisation performance and accuracy of unstable learning algorithms [18].

2. METHOD

This study was how to optimise the model of imbalanced class learning using ensemble learning on over-sample data. Figure 1 shows the research flow in this research. The imbalance dataset was the pap smear results dataset [15]. Data level and algorithm approach used to optimise class imbalance handling that was. The first approach was data level one using oversampling. It uses SMOTE-N [10] and ADASYN-N [2] to handle nominal features. SMOTE-N can solve the overfitting problem in random oversampling [10]. While ADASYN, the algorithm was proposed by He *et al.* improve SMOTE to generate the synthesis instances based on the idea of adaptively generating minority data samples based on their distribution [11]. But, ADASYN can only compute the numerical data, so for this study, ADASYN-N [2] were used because the dataset is nominal. The second one is the algorithm approach. It used an ensemble classifier using CART combined with AdaBoost and Bagging. Thirty stages of 10-fold cross-validation were used to validate the model. It divided the dataset into tenfold, then one-fold will be data test, and the rest will be data training. It will repeat until all folds become testing data. The evaluation matrices were accuracy, precision, recall, f-score.

2.1. Dataset

The dataset used is a dataset of pap smear results conducted by Kurniawati *et al.* [15]. It was used because it has a huge difference between the minority and majority classes. There is no further research to improve the classifier's performance on this dataset using over-sampling and ensemble classifiers. The dataset contains 38 features: microscopic features of the anatomical pathology results of the Pap smear test and 75 instances divided into seven classes. Table 1 is the list of the seven classes in the dataset.

Figure 2 shows the number of instances from each class in the dataset. It showed that Chronic Inflammation had the most amount with 31 instances, and the Ca Cervix Suspect had the lowest amount with two ones. Thus, the ratio of the most amount and the lowest one was 31:2. The impact of the imbalance ratio for each class is poor performance when classifying the minority class.
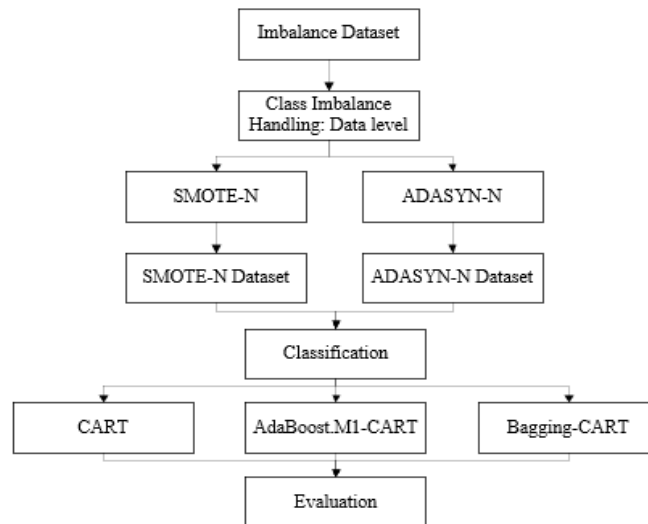
Figure 1. Research flow

Table 1. Classes

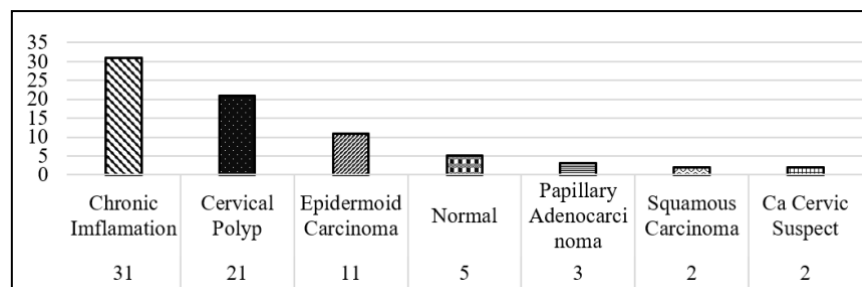| No	Class
1	Chronic Imflamation
2	Cervical Polyp
3	Epidermoid Carcinoma
4	Normal
5	Papillary Adenocarcinoma
6	Squamous Carcinoma
7	Ca Cervic Suspect

Figure 2. The instance number of each class

2.2. Class imbalanced handling

Class imbalance learning, also known as CIL, is learning with a class imbalance. The dataset has class imbalanced if it disproportionates the number of instances from each class in the dataset [19] or one class instance is higher than the other [20]. Datasets, where the most common class is less than twice the most minor class, will only be slightly unbalanced. In contrast, the dataset with an imbalance ratio of 10: 1 will be imbalanced, and the dataset with an imbalance ratio of 1000:1 will be very unbalanced [8].

The impact of imbalance is the learning and ability in rare classes. There are two aspects of the approach in dealing with imbalanced datasets, namely data level and algorithms [8], [21], [22]. The first approach to overcoming class imbalance is sampling the minority class (over-sampling). Over-sampling is a method of balancing class distribution by randomly replicating instances of minority classes. However, over-sampling increases the likelihood of overfitting occurring because it duplicates the instances exactly. In 2002, Chawla *et al.* [10] proposed a solution to deal with overfitting in the over-sampling method, namely SMOTE. SMOTE makes use of the nearest neighbours and the desired amount of over-sampling. Meanwhile, under-sampling is a method to balance the class by reducing instances in the majority class randomly. However, the

under-sampling method has a disadvantage, namely the loss of information and data that is considered necessary for the decision-making process by machine learning. The second approach is an algorithm. One of the algorithm approaches is the ensemble method. The ensemble method uses a set of classifiers to make predictions. An ensemble's generalizability is generally more robust than individual ensemble members [8]. There are two categories of ensembles, namely parallel ensembles and sequential ensembles. Parallel ensembles produce parallel base learners, for example, bagging. Consecutive ensembles make base learners sequentially, whereas previous base learners have influenced subsequent learner generations, for instance, AdaBoost. In this study, data level and algorithm will use to handle the class imbalance problem: the data level and algorithm approaches. The data level approach will use an over-sampling method with SMOTE-N and ADASYN-N. The algorithm approach will use ensemble methods.

2.2.1. SMOTE-N

SMOTE-N is a development of SMOTE used for nominal features with nominal features proposed by Chawla as the development of SMOTE [10]. At SMOTE-N, the modified version value difference metric (VDM) was proposed by Cost and Salzberg. It was used to calculate the nearest neighbour. New set feature values can be created by taking the majority vote of the feature vector considering its k nearest neighbour to generate new minority class feature vectors.

2.2.2. ADASYN-N

ADASYN is a method for oversampling approach to learning with an unbalanced dataset proposed by He *et al.* [11]. The main idea of ADASYN is to use distribution weights for data on minority classes based on the level of learning difficulty. Synthesised data are generated from minority classes that are difficult to learn compared to minority data that are easier to learn. ADASYN enhances learning in two ways. First, it reduces the bias caused by class imbalances, and the second adaptively shifts the boundaries of classification decisions towards data difficulty. ADASYN-N is a development of ADASYN with a nominal type data approach called ADASYN-N developed by Kurniawati *et al.* [2]. The nearest neighbour in ADASYN-N was calculated using a modified version of VDM as in SMOTE.

2.2.3. Ensemble methods

The ensemble method trains base learners from the training data to make predictions and then combine them to make the final decision. In contrast, the standard machine learning method only produces one learner [8]. An ensemble can increase the learner with better performance than random guess into the learner with strong generalizability and very successful in many machine learning challenges for real-life applications [23].

The base learner is often referred to as the weak learner. It indicates that the base learner can have weak generalizability in the ensemble method. However, most learning algorithms, such as decision trees, neural networks, or other machine learning methods, can be called to train the base learner, and ensemble methods can improve performance [8].

The ensemble method can be categorised into parallel and sequentially based on how the base learner is generated. The parallel one produces the base learner in parallel, for example, Bagging. The sequentially one produces the base learner sequentially, where the base learner influences the next generation, for instance, AdaBoost.

Bagging is a method for generating multiple versions of a predictor and using predictors to get a combined predictor [24]. Bagging is a representation of a parallel ensemble method. Bagging should be used with an unstable learner, for example, a decision tree, because the more unstable the base learner is, the better its performance will be. However, it turned out that bagging resulted in a combined model that performed better than a single model built from the original training data and never substantially worse off [25]. Here is the Bagging pseudocode [26]:

Algorithm bagging for classification
```
Input: S: Training set; T: number of iterations; n: number of bootstrap; I: weak learner
Output: Bagged classifier: H(x) = sign (∑ᵀₜ₌₁ hₜ(x)) where hₜ ∈ [−1,1] is induced classifier
for t = 1 to T do
Sₜ ←RandomSampleReplacement (n, S)
hₜ ← I(Sₜ)
end for
```

AdaBoost is a machine learning technique to improve the performance of the weak learner. The method called the weak learner iteratively, the training data used is taken from several subsets of the entire database. A single robust classifier is then constructed by combining the resulting weak learner with the resampling training set [27]. There are many boosting variations, one of which is AdaBoost.M1, specially

designed for classification. Adaboost.M1 is a simple generalisation of AdaBoost for more than two classes or multiclass [1], [27], which has the same algorithm as AdaBoost for the multiclass base instead of the binary learner. Here is the AdaBoost.M1 pseudocode [26]:

Algorithm Adaboost.M1

```
Input: Training set S = {xᵢ,yᵢ}, i = 1,…,N; dan yᵢ ∈ ℂ,ℂ = {c₁,…,cₘ}; T: number of iterations; I: weak
learner
Output: Boosted classifier:
```
$H(x) = \arg\max_{y \in \mathbb{C}} \sum_{t=1}^{T} ln\left(\frac{1}{\beta_t}\right)[h_t(x) = y]$ where h_t, β_t is the induced classifier (with $h_t(x) \in \mathbb{C}$) and give
```
weight to each
```
$D_1(i) \leftarrow \frac{1}{N}$ for $i = 1,…,N$
```
for t = 1 to T do
```
$h_t \leftarrow I(S, D_t)$

$\varepsilon_t \leftarrow \sum_{i=1}^{N} D_t(i)[h_t(x_i) \neq y_i]$
```
if εₜ > 0,5 then
T ← t − 1
return
end if
```
$\beta_t = \frac{\varepsilon_t}{1 - \varepsilon_t}$

$D_{t+1}(i) = D_t(i).\beta^{1-[h_t(x_i) \neq y_i]}$ for $i = 1,…,N$
```
Normalise Dₜ₊₁ for the proper distribution
end for
```

2.2.4. CART

Classification and regression tree algorithm or CART is a regression tree and classification tree method that will produce a classification tree if it consists of categorical attributes and create a regression tree if it consists of continuous attributes [28]. CART will select several attributes and interactions between the most dominant attributes in determining the attributes' results depending on the binary sorting procedure. In choosing the best splitter, CART strives to maximise the average purity of the two child nodes. The way to measure purity can be selected freely, and it can be by the criteria of splitting or the splitting function. The most common splitting function is the Gini index. Gini index calculation as shown in (1):

$$Gini(t) = 1 - \sum_{i=0}^{c-1}[p(i|t)]^2 \tag{1}$$

where $P(i|t)$ is the relative frequency of class i at node t, and c is the number of classes. Thus, the calculation will get the highest if the distribution is from a uniform class and the smallest if it contains identical class records.

2.2.5. Evaluation

The confusion matrix is used to calculate the evaluation matrices such as accuracy, recall, precision, f-score, and receiver operating character (ROC) area as evaluation matrices. Table 2 shows the confusion matrix. TP is the condition where the classifier correctly classifies the positive result. Otherwise, TN is a condition where the classifier correctly classifies a negative result. Meanwhile, FP is a condition in which the classifier identifies a positive result as negative, and FN is a condition where the classifier identifies a negative result as positive.

Table 2. Confusion matrix

Actual class	Predicted class	
	Positive	Negative
Positive	True positive (TP)	False negative (FN)
Negative	False positive (FP)	True negative (TN)

The calculation of accuracy, recall, precision, and f-score using (2)-(5):

$$Accuracy = \frac{TP+TN}{TP+FP+TN+FN} \tag{2}$$

$$Precision = \frac{TP}{TP+FP} \tag{3}$$

$$Recall = \frac{TP}{TP+FN} \tag{4}$$

$$F - score = (1 + \beta^2) \times \frac{presisi \times recall}{(\beta^2 \times presisi)+recall} \tag{5}$$

ROC area, commonly known as the AUC, is a technical standard for classifier evaluation. The wider the AUC area, the better the model and the excellent interpretation of the probability that the classifier ranks randomly selected positive instances over randomly selected negative instances [26]. Every curve on the ROC curve represents the performance of different classifiers in the dataset. The X-axis represents the false positive rates (FPR), and Y-axis represents the true positive rates (TPR). FPR and TPR calculate using (6) and (7):

$$FPR = \frac{FP}{(TN+FP)} \tag{6}$$

$$TPR = \frac{TP}{(TP+FN)} \tag{7}$$

3.   RESULTS AND DISCUSSION

New datasets were generated using SMOTE-N and ADASYN-N then tested using CART, AdaBoost-CART, and Bagging-CART. Implementation with the classifier using 30-Stages 10-Cross Fold Validation. The evaluation matrices used were accuracy, recall, precision, f-score, and ROC area. Table 3 shows the algorithm configuration.

Table 3. Algorithm configuration

Algorithm	Configuration	
	Variable	Value
SMOTE-N	KNN	5
	%N	Adjusted to the number of instances
ADASYN-N	KNN	5
	d_{th}	0.75
	β	1

Both SMOTE-N and ADASYN-N used the five nearest neighbours. ADASYN-N used 0.75 for d_{th} or maximum tolerance level of imbalance class ratio and 1 for β or level balance. As for SMOTE-N, the value of the %N adjusted depends on the number of instances generated by ADASYN-N. Figure 3 shows each optimisation's performance from both datasets and all the classifications used in this study. The optimisation using data level or data level and algorithm could improve the accuracy from 89.34% to 96.39%. The imbalance dataset using CART had the lowest mean accuracy, equal to 77.87%. The best accuracy used ADASYN-N with AdaBoost-CART as the classifier obtained 96.39%. The mean accuracy from SMOTE-N and ADASYN-N datasets are better than the imbalanced dataset.
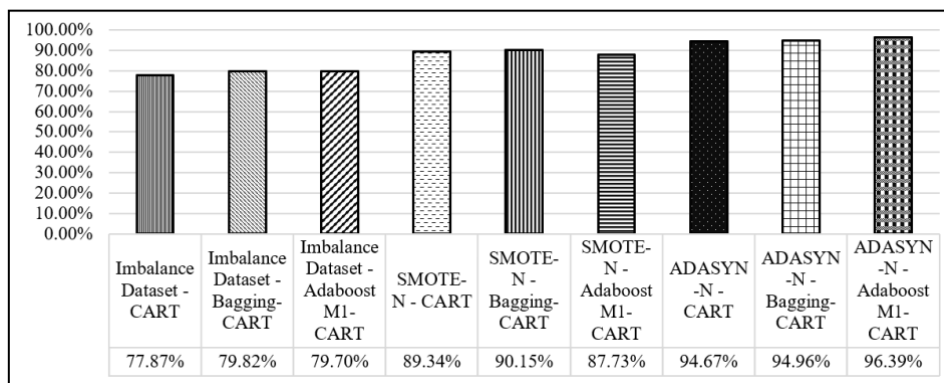
Figure 3. Mean accuracy

Table 4 shows all the performance matrices used in this study, such as precision, recall, f-measure, and ROC Area. ADASYN-N with AdaBoost-CART had the best performance matrices in precision, recall, and f-measure. However, the best ROC area was obtained by ADASYN-N with Bagging-CART. It was reasonable because the model obtained using Boosting is iterative. Therefore, the new model was affected by the previous model's performance. It encourages new models to become experts for instances that the previous model correctly handles by assigning a greater weight to their instances.

Table 4. Evaluation

Model	Precision	Recall	F-Measure	ROC Area
Imbalance Dataset - CART	67.16%	77.87%	71.88%	84.25%
Imbalance Dataset - Bagging-CART	72.29%	79.82%	75.45%	90.50%
Imbalance Dataset - AdaboostM1-CART	80.29%	79.70%	79.47%	91.67%
SMOTE-N - CART	90.05%	89.34%	89.19%	95.11%
SMOTE-N - Bagging-CART	90.71%	90.15%	90.01%	95.74%
SMOTE-N - AdaboostM1-CART	87.82%	87.73%	87.67%	95.83%
ADASYN-N - CART	95.13%	94.67%	94.42%	98.29%
ADASYN-N - Bagging-CART	95.45%	94.96%	94.72%	99.22%
ADASYN-N - AdaboostM1-CART	96.59%	96.39%	96.27%	98.56%

4. CONCLUSION

Laboratory test data, such as a dataset of pap smear results, most have little data and imbalance. Almost in every case, the least entities are the most important and needed. Based on this study's results, the optimisation of Class Imbalanced Learning using both data level and algorithm using ensemble classifier on over-sampling data could increase all the evaluation matrices performance on laboratory test data. ADASYN-N is better than SMOTE-N for over-sampling the dataset used in this study. The best model was obtained using ADASYN-N with AdaBoost-CART. Moreover, this study will use another based-learner besides CART to get the best model for imbalance laboratory test data.

REFERENCES

[1] A. Fernández, S. García, M. Galar, R. C. Prati, B. Krawczyk, and F. Herrera, *Learning from imbalanced data sets*. Cham: Springer International Publishing, 2018.
[2] Y. E. Kurniawati, A. E. Permanasari, and S. Fauziati, "Adaptive synthetic-nominal (ADASYN-N) and adaptive synthetic-KNN (ADASYN-KNN) for multiclass imbalance learning on laboratory test data," in *2018 4th International Conference on Science and Technology (ICST)*, Aug. 2018, pp. 1–6, doi: 10.1109/ICSTC.2018.8528679.
[3] L. Wang *et al.*, "Classifying 2-year recurrence in patients with dlbcl using clinical variables with imbalanced data and machine learning methods," *Computer Methods and Programs in Biomedicine*, vol. 196, Nov. 2020, doi: 10.1016/j.cmpb.2020.105567.
[4] H. Qin, H. Zhou, and J. Cao, "Imbalanced learning algorithm based intelligent abnormal electricity consumption detection," *Neurocomputing*, vol. 402, pp. 112–123, Aug. 2020, doi: 10.1016/j.neucom.2020.03.085.
[5] S. Hasmita, F. Nhita, D. Saepudin, and A. Aditsania, "Chili commodity price forecasting in bandung regency using the adaptive synthetic sampling (ADASYN) and k-nearest neighbor (KNN) algorithms," in *2019 International Conference on Information and Communications Technology (ICOIACT)*, Jul. 2019, pp. 434–438, doi: 10.1109/ICOIACT46704.2019.8938525.
[6] J. Sun, J. Lang, H. Fujita, and H. Li, "Imbalanced enterprise credit evaluation with DTE-SBD: Decision tree ensemble based on SMOTE and bagging with differentiated sampling rates," *Information Sciences*, vol. 425, pp. 76–91, Jan. 2018, doi: 10.1016/j.ins.2017.10.017.
[7] J. Shin, S. Yoon, Y. W. Kim, T. Kim, B. G. Go, and Y. K. Cha, "Effects of class imbalance on resampling and ensemble learning for improved prediction of cyanobacteria blooms," *Ecological Informatics*, vol. 61, Mar. 2021, doi: 10.1016/j.ecoinf.2020.101202.
[8] H. Ye and Y. Ma, *Imbalanced learning: foundations, algorithms, and applications*, 1st ed. Wiley-IEEE Press, 2013.
[9] B. S. Raghuwanshi and S. Shukla, "Class imbalance learning using UnderBagging based kernelised extreme learning machine," *Neurocomputing*, vol. 329, pp. 172–187, Feb. 2019, doi: 10.1016/j.neucom.2018.10.056.
[10] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.
[11] H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, Jun. 2008, pp. 1322–1328, doi: 10.1109/IJCNN.2008.4633969.
[12] W. A. Luqyana, B. L. Ahmadie, and A. A. Supianto, "K-nearest neighbors undersampling as balancing data for cyber troll detection," in *2019 International Conference on Sustainable Information Engineering and Technology (SIET)*, Sep. 2019, pp. 322–325, doi: 10.1109/SIET48054.2019.8986079.
[13] Y.-S. Jeon and D.-J. Lim, "PSU: particle stacking undersampling method for highly imbalanced big data," *IEEE Access*, vol. 8, pp. 131920–131927, 2020, doi: 10.1109/ACCESS.2020.3009753.
[14] I. Fakhruzi, "An artificial neural network with bagging to address imbalance datasets on clinical prediction," in *2018 International Conference on Information and Communications Technology (ICOIACT)*, Mar. 2018, pp. 895–898, doi: 10.1109/ICOIACT.2018.8350824.
[15] Y. E. Kurniawati, A. E. Permanasari, and S. Fauziati, "Comparative study on data mining classification methods for cervical cancer prediction using pap smear results," in *2016 1st International Conference on Biomedical Engineering (IBIOMED)*, Oct.

2016, pp. 1–5, doi: 10.1109/IBIOMED.2016.7869827.

[16] K. Fithriasari, I. Hariastuti, and K. S. Wening, "Handling imbalance data in classification model with nominal predictors," *International Journal of Computing Science and Applied Mathematics*, vol. 6, no. 1, Feb. 2020, doi: 10.12962/j24775401.v6i1.6643.

[17] N. Rachburee and W. Punlumjeak, "Oversampling technique in student performance classification from engineering course," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 4, pp. 3567–3574, Aug. 2021, doi: 10.11591/ijece.v11i4.pp3567-3574.

[18] T. Bi, P. Li, J. Huang, and K. Zhang, "Imbalance data classification method based on cluster boundary sampling RF-bagging," in *International Conference on Software Intelligence Technologies and Applications & International Conference on Frontiers of Internet of Things 2014*, 2014, vol. 2014, no. CP660, pp. 305–311, doi: 10.1049/cp.2014.1580.

[19] V. H. Barella, L. P. F. Garcia, M. C. P. de Souto, A. C. Lorena, and A. C. P. L. F. de Carvalho, "Assessing the data complexity of imbalanced datasets," *Information Sciences*, vol. 553, pp. 83–109, Apr. 2021, doi: 10.1016/j.ins.2020.12.006.

[20] A. Mahmoud, A. El-Kilany, F. Ali, and S. Mazen, "TGT: a novel adversarial guided oversampling technique for handling imbalanced datasets," *Egyptian Informatics Journal*, vol. 22, no. 4, pp. 433–438, Dec. 2021, doi: 10.1016/j.eij.2021.01.002.

[21] K. Li, W. Zhang, Q. Lu, and X. Fang, "An improved SMOTE imbalanced data classification method based on support degree," in *2014 International Conference on Identification, Information and Knowledge in the Internet of Things*, Oct. 2014, pp. 34–38, doi: 10.1109/IIKI.2014.14.

[22] F. Koto, "SMOTE-out, SMOTE-cosine, and selected-SMOTE: an enhancement strategy to handle imbalance in data level," in *2014 International Conference on Advanced Computer Science and Information System*, Oct. 2014, pp. 280–284, doi: 10.1109/ICACSIS.2014.7065849.

[23] S. González, S. García, J. Del Ser, L. Rokach, and F. Herrera, "A practical tutorial on bagging and boosting based ensembles for machine learning: Algorithms, software tools, performance study, practical perspectives and opportunities," *Information Fusion*, vol. 64, pp. 205–237, Dec. 2020, doi: 10.1016/j.inffus.2020.07.007.

[24] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, Aug. 1996, doi: 10.1007/BF00058655.

[25] I. H. Witten, E. Frank, and M. A. Hall, *Data mining: practical machine learning tools and techniques*. Elsevier, 2011.

[26] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463–484, Jul. 2012, doi: 10.1109/TSMCC.2011.2161285.

[27] O. Herouane, L. Moumoun, and T. Gadi, "Using bagging and boosting algorithms for 3D object labeling," in *2016 7th International Conference on Information and Communication Systems (ICICS)*, Apr. 2016, pp. 310–315, doi: 10.1109/IACS.2016.7476070.

[28] M. T. M. K. Sabariah, S. T. A. Hanifa, and M. T. S. Sa'adah, "Early detection of type II diabetes mellitus with random forest and classification and regression tree (CART)," in *2014 International Conference of Advanced Informatics: Concept, Theory and Application (ICAICTA)*, Aug. 2014, pp. 238–242, doi: 10.1109/ICAICTA.2014.7005947.

BIOGRAPHIES OF AUTHORS

Yulia Ery Kurniawati (ID) (SC) (P) is currently a lecturer at Informatics Department at Institut Teknologi dan Bisnis Kalbis (Kalbis Institute) Jakarta. She obtained her bachelor degree in informatics at Universitas Sebelas Maret and master degree in information technology from Universitas Gadjah Mada. Her research interest in artificial intelligence focuses on machine learning and class imbalance learning. She can be contacted at email: yulia.kurniawati@kalbis.ac.id.

Yulius Denny Prabowo (ID) (SC) (P) is a lecturer at the Informatics study program at Institut Teknologi dan Bisnis Kalbis. He obtained a bachelor's degree from Atmajaya University Yogyakarta, a master's degree from the University of Indonesia and is currently pursuing doctoral education. His research focuses on a cross-section of artificial intelligence, machine learning, cognitive intelligence, and the Indonesian language. He can be contacted at email: yulius.prabowo@kalbis.ac.id

☐ 284

Prediction of diabetes disease using machine learning algorithms

Monalisa Panda[1], Debani Prashad Mishra[1], Sopa Mousumi Patro[1], Surender Reddy Salkuti[2]
[1]Department of Electrical Engineering, International Institute of Information Technology Bhubaneswar, Bhubaneswar 751029, India
[2]Department of Railroad and Electrical Engineering, Woosong University, Daejeon 34606, Republic of Korea

Article Info	ABSTRACT
	Diabetes mellitus is a powerful chronic disease, which is recognized by lack of capability of our body for metabolization of glucose. Diabetes is one of the most dangerous diseases and a threat to human society, many are becoming its victims and, regardless of the fact that they are trying to keep it from rising more, are unable to come out of it. There are several conventional diabetes disease health monitoring strategies. This disease was examined by machine learning (ML) algorithms in this paper. The goal behind this research is to create an effective model with high precision to predict diabetes. In order to reduce the processing time, K-nearest neighbor algorithm is used. In addition, support vector machine is also introduced to allocate its respective class to each and every sample of data. In building any sort of ML model, feature selection plays a vital role, it is the process where we select the features automatically or manually and it contributes most to our desired performance. Overall, four algorithms are used in this paper to understand which can easily evaluate the total effectiveness and accuracy of predicting whether or not a person will suffer from diabetes.

Corresponding Author:

Surender Reddy Salkuti
Department of Railroad and Electrical Engineering, Woosong University
Jayang-dong, Dong-gu, Daejeon-34606, Republic of Korea
Email: surender@wsu.ac.kr

1. INTRODUCTION

Diabetes mellitus is a condition characterized by a metabolic process disorder and an excessive rise in blood sugar concentration due to lack of insulin, a peptide hormone secreted by pancreatic islet beta cells [1], [2]. If we step around the dangerous impact of diabetes, we will undoubtedly conclude that it can lead to significant complications, or even premature death. So, to decrease the mortality rate and improve a patient's health status. Therefore, machine learning algorithms are now used to identify and diagnose diseases in order to minimize the death risk and improve a patient's health status, as machine learning (ML) contributes to specific decisions. There are essentially two main clinical forms, type 1 diabetes and type 2 diabetes[3], which are indicated as (T1D) and (T2D) [4] respectively, according to the origin and progression of the condition, which is nothing but the disorder's etiopathology. Nearly 90% of all diabetic patients have T2D, which is predominantly characterized by insulin hormone resistance. Lifestyle, physical activity, food or dietary patterns and inheritance are the real causes of T2D, T1D is believed to be due to the autoimmunological degradation of pancreatic-β cells in the langerhans islets.

Different researchers are designing a multiple diabetes prediction method based on a variety of algorithms. In [5] previously suggested a method for classifying diabetes disease via the use of the support vector machine (SVM). For diagnosis, the Pima Indian Diabetes (PID) dataset is used. Using the radial basis function (RBF) SVM kernel as the classifier, 78% of the accuracy was achieved. Orabi*et al.* [6] designed a

method for the prediction of diabetes. Pradhan and Bamnote [7] presents several genetic programmingrelated algorithms and several tests are performed on this dataset. As a classifier for diabetes disease prediction, in [8] uses J48 decision tree (DT) (74.8% accuracy) and naïve bayes (79.5% accuracy). A prediction model with two sub-modules was developed in [9] to predict diabetes-chronic disease.

In [10] estimates the 250 million individuals are currently affected by diabetes and will cross 500 million by 2025. DT is used to find ways of extracting attributes and features from a fixed dataset [11]. Until testing, the dataset is trained to predict and store the results for each and every new instantiated object in a separate class. In [12] presents an algorithm that classifies the risk of diabetes mellitus[13]. This paper illustrates diabetes disease prediction based on the characteristics of the datasets. Gradient boost is adequate to equate logistic regression (LR), SVM, and k-nearest neighbor (KNN) to the rest of the classifier. Like dataset selection, extraction of attributes, implementation of algorithms after breaking the full dataset into a training and test dataset. Finally, the outcome shown in this paper demonstrates the proposed model's ability to predict diabetes with less time in the earlier process.

Section 1 is all about the introduction regarding Diabetes mellitus i.e., its cause, symptoms, types. Section 2 is about the research method, the model diagram and a brief description about the test and training dataset, the algorithms used for predicting the disease like SVM, KNN, LR, and gradient boost. Section 3 is describing the dataset in a clear way and the figure of outcome indicates the percentage patients with and without the disease. The section 4 depicts the results where we get gradient boost classifier give 81.25%. In the last section, the final conclusion of our processes is described based on our model.

2. RESEARCH METHOD

The required information and necessary steps in order to build the model to predict diabetes by using the classifiers are described as separate sections. The research approach mentioned in this paper explicitly defines the entire model's working criteria. Figure 1 depicts the procedure of proposed approach. The brief discussion of steps involved in the proposed approach are presented next:

−   The first step is collection of data Kaggle [14]. This dataset contains all total of 768 instances and 9 attributes [14]. The dataset is briefly discussed in the dataset section.

−   The second step is data-preprocessing, in this step the null values are checked and removed also the categorical values converted into numerical ones.

−   In the third step exploratory data analysis [15] is performed where each columns correlation matrix was formed and also some visualizations like box plots [15] were done to check the outlier values. Some other visualization [16] was also done to check how [16] the features are related to the label column. Feature selection [17] which plays a major role is also done in this step where the important feature has been selected for the model and after the selection the data has been fit into the model for the prediction [17].

−   In this step the dataset was divided into training and testing test [18]. For this work, the dataset has been divided into training and testing part with test size of 0.25. i.e., the training data [19] consists around 75% of the whole data whereas testing data contains 25% of the whole dataset. In the dataset there is a total of 768 instances and 9 attributes, so the training data will contain 576 instances (75% of 768) and testing data will contain 192 instances (25% of 768).

−   Algorithms used: There are 4 algorithms used in this paper. Those are LR, KNN, SVM, and gradient boost. These 4 algorithms are discussed next.
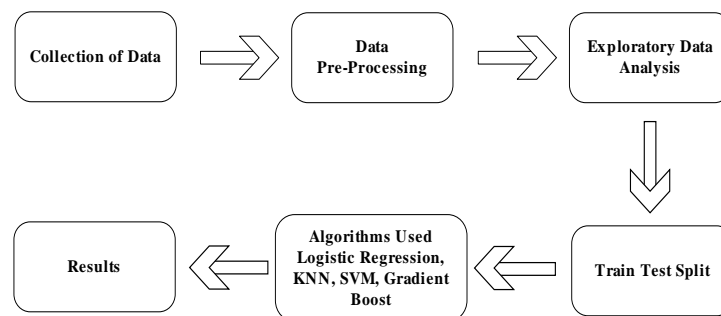

Figure 1. Proposed approach model diagram

2.1. Logistic regression (LR)

Various algorithms are used to decide which is the best match for this dataset and will provide better results. Logistic regression, KNN, SVM, and gradient boost are such algorithms. LR helps us in solving classification problems, it uses S-curve instead of a straight line for fitting the points [20]. Logistic is taken from the function logit that is used in this method of classification[20].

2.2. K-nearest neighbor (KNN)

It has two properties: i) lazy learning algorithm: since there is no separate step of preparation. It utilizes all the data during classification for training and ii) algorithm for non-parametric learning: about the underlying data, it does not assume anything. First of all, the data set is fed as input, including the dataset for testing and training. The loading of the dataset and data preprocessing takes place in the next step. After that decision, to get the desired results, the KNN algorithm is implemented. Steps involved in this algorithm are:

− Using euclidean, manhattan or hamming distances, the distance between the test data and each row of training data is measured.
− Now, arrange them in an ascending order based on the distance values.
− Next the top k rows [21] are picked from the sorted list.
− The class is allocated to the test point based on the frequent classes of these rows [21].

2.3. Support vector machine (SVM)

Support vectors are the most important data points of the training dataset. If we remove the data points then the position of dividing hyper plane will change rather than being 2 non-overlapping classes [22]. And in constant visualization nonlinear separation works well as compare to linear one. Steps involved in this algorithm are: i) import the dataset; ii) explore the data to figure out what they look like [23]; iii) then data is split into attributes and labels [23]; iv) the data is divided into training and testing datasets; and v) SVM algorithm is trained for our desired output or results.

The last step involved in this algorithm is the comparing all the precision [24] of the algorithms to get the results. This performance evaluation is carried out for all algorithms in this phase performance evaluation, the performance of the models has been evaluated using the confusion matrix and classification report where the accuracy, recall, f1-score for each algorithm has been calculated, the comparison between all algorithms is discussed in the results section. The output of a classification model is represented using a confusion matrix. The uncertainty/confusion matrix can represented by,

$$Confusion\ Matrix = \begin{bmatrix} TRUE^+ & FALSE^+ \\ FALSE^- & TRUE^- \end{bmatrix} \tag{1}$$

True positive (TP): cases in which the classifier predicted TRUE (they have the disease) and TRUE was the correct class (patient has disease). Real negatives (TN): cases where FALSE (no illness) was predicted by the model and FALSE was the right class (patient do not have disease). False positives (FP) (type I error): the classifier predicted TRUE, but FALSE was the correct class (patient did not have disease). False negatives (FN) (type II error): instances where FALSE (patients have no disease) has been predicted by the machine learning model, but they actually have the disease.

Key Performance Indicator (KPI) calculation is as follows:

− Accuracy of classification=(TP+TN)/(TP+TN+FP+FN)
− Misclassification rate=(FP+FN)/(TP+TN+FP+FN)=(error rate)
− Precision=TP/Total TRUE Predictions=TP/(TP+FP) (how much was it accurate when the model predicted the TRUE class?)
− Recall=TP/Real TRUE=TP/(TP+FN) (how much did the classifier get it right when the class was actually TRUE?)

3. DATASET DESCRIPTION

This dataset has been taken from Kaggle [14]. The datasets consist of 768 rows and 9 columns, with 8 rows being instances, and the target variable being 1 row (output). The target variable is outcome, while Predictor variables include the patient's number of births, Triceps skin fold thickness measurement (mm), their body mass index (BMI) (weight in kg/(height in m)^2), blood pressure diastolic (mm Hg), insulin level, era, and so on. Figure 2 depicts the instances of outcome column. The outcome column is the target variable and contains zeros and ones where zeros represent that patient don't have diabetes disease whereas one represents patient have diabetes disease. From Figure 2 we can say that there are 500 instances present in the

outcome column are 0 and 268 are 1 that is around 66.10% don't have diabetes and around 34.90% have disease in the dataset.
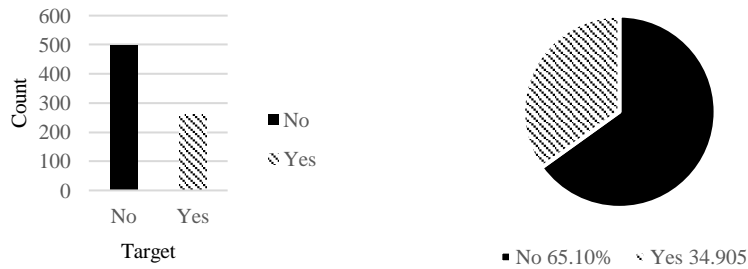

Figure 2. Instances of outcome

4.   RESULTS AND DISCUSSION

The final result has been derived successfully using the mentioned four machine learning algorithms. Table 1 shows the accuracy of all the models used in this paper, their precision, recall and f1-score is also shown in Table 1. By comparing all the algorithms, it can be observed that the best algorithm based on accuracy is gradient boost with an accuracy of 81.25%.

Figure 3 depicts the accuracy comparison among the models used for this work. From this figure it can concluded that Gradient Boost algorithm gives the best accuracy of around 81.25% and KNN gives the lowest accuracy of 78% among these four algorithms, in addition to this logistic regression gives 81% whereas Support vector classifier gives 80% accuracy.

Table 1. Comparison among the models

Name	Accuracy	Precision	Recall	F1-score
Gradient Boost	0.8125	0.7600	0.6290	0.6964
Logistic Regression	0.8073	0.7660	0.5806	0.6605
SVM	0.8021	0.7400	0.5968	0.6607
KNN	0.7813	0.7273	0.5161	0.6038

Model	Accuracy
GradientBoostingClassifier	0.81
LogisticRegression	0.81
SVC	0.8
kNeighborsClassifier	0.78
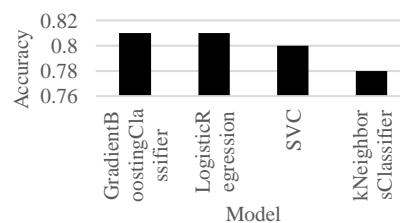
Figure 3. Model accuracy comparison

Figure 4 depicts the precision comparison among the models used for this work. From this figure, it can be concluded that gradient boost algorithm gives the best precision score of around 0.76 and k-neighbors classifiers gives the lowest precision score of 0.73 among these four algorithms, in addition to this LR [25] gives 0.77 whereas support vector classifier gives 0.74 precision score. Figure 5 depicts the precision comparison among the models used for this paper. From this figure, it can be concluded that gradient boost algorithm gives the best recall score of around 0.63 and K-Neighbors classifiers [26] gives the lowest recall score of 0.52 among these four algorithms, in addition to this logistic regression gives 0.58 whereas Support vector classifier gives 0.6 recall score.

Figure 6 depicts the F1-score comparison among the models used for this work. From this figure, it can be concluded that gradient boost algorithm gives the best F1-score [27] of around 0.7 and k-neighbors classifiers gives the lowest F1-score of 0.6 among these four algorithms, in addition to this logistic regression and support vector classifier gives same F1-score of 0.66. From the simulation results, it can be concluded that in this paper there are total 4 algorithms are used LR [28], KNN [29], gradient descent [30], and the best accuracy achieved is 81.25% which has given by gradient descent classifier.

Model	Precision
GradientBoostingClassifier	0.76
LogisticRegression	0.77
SVC	0.74
kNeighborsClassifier	0.73
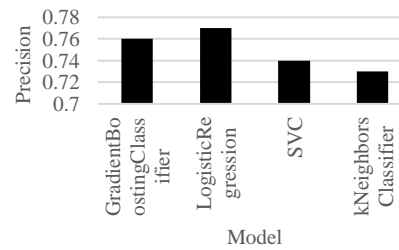
Figure 4. Model precision comparison

Model	Recall
GradientBoostingClassifier	0.61
LogisticRegression	0.58
SVC	0.6
kNeighborsClassifier	0.52
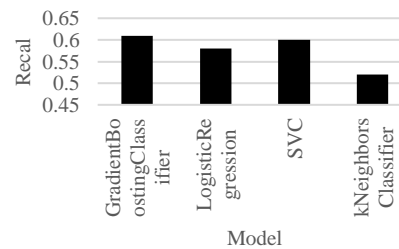
Figure 5. Model recall comparison

Model	F1-Score
GradientBoostingClassifier	0.68
LogisticRegression	0.66
SVC	0.66
kNeighborsClassifier	0.6
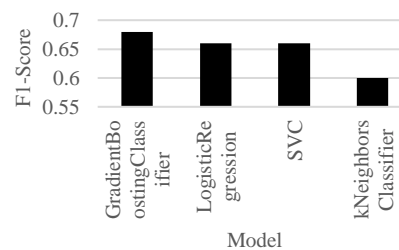
Figure 6. Model F1-score comparison

5. CONCLUSION

Using sophisticated statistical techniques and the availability of a large number of epidemiological and genetic diabetes risk datasets, ML has the considerable potential to restructure or shake up the risk of diabetes prediction. It is clearly seen from this paper that gradient boosting classifier works well for this type of dataset, which is also confirmed by model accuracy and recall. And KNN works well for the dataset includes a large number of datasets that it is easier to minimize processing time. And SVM deals with a wide number of functions for the dataset in a better way. This model can be used for future work, this application can be used by taking patients' past health records and showing whether or not the person has diabetes.

REFERENCES

[1] H. Wu, S. Yang, Z. Huang, J. He, and X. Wang, "Type 2 diabetes mellitus prediction model based on data mining," *Informatics in Medicine Unlocked*, vol. 10, pp. 100–107, 2018, doi: 10.1016/j.imu.2017.12.006.

[2] J. S. Kaddis, B. J. Olack, J. Sowinski, J. Cravens, J. L. Contreras, and J. C. Niland, "Human pancreatic islets and diabetes research," *Journal of the American Medical Association (JAMA)*, vol. 301, no. 15, pp. 1580–1587, Apr. 2009, doi: 10.1001/jama.2009.482.

[3] A. B. Olokoba, O. A. Obateru, and L. B. Olokoba, "Type 2 diabetes mellitus: a review of current trends," *Oman Medical Journal*, vol. 27, no. 4, pp. 269–273, Jul. 2012, doi: 10.5001/omj.2012.68.

[4] T. Zheng *et al.*, "A machine learning-based framework to identify type 2 diabetes through electronic health records," *International Journal of Medical Informatics*, vol. 97, pp. 120–127, Jan. 2017, doi: 10.1016/j.ijmedinf.2016.09.014.

[5] V. A. Kumari and R. Chitra, "Classification of diabetes disease using support vector machine," *International Journal of Engineering Research and Applications (IJERA)*, vol. 3, no. 2, pp. 1797–1801, 2018.

[6] K. M. Orabi, Y. M. Kamal, and T. M. Rabah, "Early predictive system for diabetes mellitus disease," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 9728, Springer International Publishing, 2016, pp. 420–427.

[7] M. Pradhan and G. R. Bamnote, "Design of classifier for detection of diabetes mellitus using genetic programming," in *Advances in Intelligent Systems and Computing*, vol. 327, Springer International Publishing, 2015, pp. 763–770.

[8] A. Iyer, J. S, and R. Sumbaly, "Diagnosis of diabetes using classification mining techniques," *International Journal of Data Mining & Knowledge Management Process*, vol. 5, no. 1, pp. 01–14, Jan. 2015, doi: 10.5121/ijdkp.2015.5101.

[9] T. A. Rashid, S. M. Abdullah, and R. M. Abdullah, "An intelligent approach for diabetes classification, prediction and description," in *Advances in Intelligent Systems and Computing*, vol. 424, Springer International Publishing, 2016, pp. 323–335.

[10] P. Samant and R. Agarwal, "Machine learning techniques for medical diagnosis of diabetes using iris images," *Computer Methods and Programs in Biomedicine*, vol. 157, pp. 121–128, Apr. 2018, doi: 10.1016/j.cmpb.2018.01.004.

[11] N. Yilmaz, O. Inan, and M. S. Uzer, "A new data preparation method based on clustering algorithms for diagnosis systems of heart and diabetes diseases," *Journal of Medical Systems*, vol. 38, no. 5, May 2014, Art. no. 48, doi: 10.1007/s10916-014-0048-7.

[12] N. Nai-arun and R. Moungmai, "Comparison of classifiers for the risk of diabetes prediction," *Procedia Computer Science*, vol. 69, pp. 132–142, 2015, doi: 10.1016/j.procs.2015.10.014.

[13] K. G. M. M. Alberti and P. Z. Zimmet, "Definition, diagnosis and classification of diabetes mellitus and its complications. Part 1: diagnosis and classification of diabetes mellitus. Provisional report of a WHO Consultation," *Diabetic Medicine*, vol. 15, no. 7, pp. 539–553, Jul. 1998, doi: 10.1002/(SICI)1096-9136(199807)15:7<539::AID-DIA668>3.0.CO;2-S.

[14] Kaggle, "Pima Indians diabetes database." https://www.kaggle.com/uciml/pima-indians-diabetes-database.

[15] C. H. Yu, "Exploratory data analysis in the context of data mining and resampling," *International Journal of Psychological Research*, vol. 3, no. 1, pp. 9–22, Jun. 2010, doi: 10.21500/20112084.819.

[16] M. N. O. Sadiku, A. E. Shadare, S. M. Musa, and C. M. Akujuobi, "Data visualization," *International Journal of Engineering Research And Advanced Technology(IJERAT)*, vol. 2, no. 12, pp. 11–16, 2016, doi: 10.1007/978-1-4020-4409-0_56.

[17] J. Miao and L. Niu, "A survey on feature selection," *Procedia Computer Science*, vol. 91, pp. 919–926, 2016, doi: 10.1016/j.procs.2016.07.111.

[18] M. A. Shafique and E. Hato, "Formation of training and testing datasets, for transportation mode identification," *Journal of Traffic and Logistics Engineering*, vol. 3, no. 1, 2015, doi: 10.12720/jtle.3.1.77-80.

[19] R. Medar, V. S. Rajpurohit, and B. Rashmi, "Impact of training and testing data splits on accuracy of time series forecasting in machine learning," in *2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, Aug. 2017, pp. 1–6, doi: 10.1109/ICCUBEA.2017.8463779.

[20] J. L. Alzen, L. S. Langdon, and V. K. Otero, "A logistic regression investigation of the relationship between the learning assistant model and failure rates in introductory STEM courses," *International Journal of STEM Education*, vol. 5, no. 1, Dec. 2018, Art. no. 56,doi: 10.1186/s40594-018-0152-1.

[21] P. Ray and D. P. Mishra, "Support vector machine based fault classification and location of a long transmission line," *Engineering Science and Technology, an International Journal*, vol. 19, no. 3, pp. 1368–1380, Sep. 2016, doi: 10.1016/j.jestch.2016.04.001.

[22] D. P. Mishra and P. Ray, "Fault detection, location and classification of a transmission line," *Neural Computing and Applications*, vol. 30, no. 5, pp. 1377–1424, Sep. 2018, doi: 10.1007/s00521-017-3295-y.

[23] P. Ray, D. P. Mishra, and G. K. Budumuru, "Location of the fault in TCSC-based transmission line using SVR," in *2016 International Conference on Information Technology (ICIT)*, Dec. 2016, pp. 270–274, doi: 10.1109/ICIT.2016.061.

[24] M. Arora, U. Kanjilal, and D. Varshney, "Evaluation of information retrieval: precision and recall," *International Journal of Indian Culture and Business Management*, vol. 12, no. 2, pp. 224–236, 2016, doi: 10.1504/ijicbm.2016.074482.

[25] A. Mujumdar and V. Vaidehi, "Diabetes prediction using machine learning algorithms," *Procedia Computer Science*, vol. 165, pp. 292–299, 2019, doi: 10.1016/j.procs.2020.01.047.

[26] K. Vizhi and A. Dash, "Diabetes prediction using machine learning," *International Journal of Advanced Science and Technology*, vol. 29, no. 6, pp. 2842–2852, May 2020, doi: 10.32628/cseit2173107.

[27] J. Liu *et al.*, "Artificial intelligence in the 21st century," *IEEE Access*, vol. 6, pp. 34403–34421, 2018, doi: 10.1109/ACCESS.2018.2819688.

[28] C. Y. J. Peng, T. S. H. So, F. K. Stage, and E. P. St. John, "The use and interpretation of logistic regression in higher education journals," *Research in Higher Education*, vol. 43, no. 3, pp. 259–293, 2002, doi: 10.1023/A:1014858517172.

[29] Y. Cai, D. Ji, and D. Cai, "A KNN research paper classification method based on shared nearest neighbor," in *Proceedings of NTCIR-8 Workshop Meeting*, 2010, pp. 336–340.

[30] D. Yi, S. Ji, and S. Bu, "An enhanced optimization scheme based on gradient descent methods for machine learning," *Symmetry*, vol. 11, no. 7, Jul. 2019, Art. no. 942, doi: 10.3390/sym11070942.

BIOGRAPHIES OF AUTHORS

Monalisa Panda [ID] [SC] [P] received the Bachelor of Technology (B.Tech.) degree in Electrical and Electronics Engineering from International Institute of Information Technology, Bhubaneswar, Odisha, India 2021. She has worked many Machine Learning Companies she has published many blogs in Medium. She is currently working as a Software Developer in Mindtree Ltd, Bangalore in Dotnet Core Technology. Her research areas of interest include Machine Learning, Data Analytics Big Data, Database Systems, Neural Networks, MongoDB, Hadoop Framework, Spark. She can be contacted at email: monalisapanda94@gmail.com.

Debani Prasad Mishra 🆔 🔍 SC Ⓟ received the B.Tech. in electrical engineering from the Biju Patnaik University of Technology, Odisha, India, in 2006 and the M.Tech in power systems from IIT, Delhi, India in 2010. He has been awarded the Ph.D. degree in power systems from Veer Surendra Sai University of Technology, Odisha, India, in 2019. He is currently serving as Assistant Professor in the Dept of Electrical Engg, International Institute of Information Technology Bhubaneswar, Odisha. He has 11 years of teaching experience and 2 years of industry experience in the thermal power plant. He is the author of more than 80 research articles. His research interests include soft Computing techniques application in power systems, signal processing and power quality. 3 students have been awarded Ph.D. under his guidance and currently 4 Ph.D. Scholars are continuing under him. He can be contacted at email: debani@iiit-bh.ac.in.

Sopa Mousumi Patro 🆔 🔍 SC Ⓟ has received the Bachelor of Technology (B.Tech.) degree in Electrical and Electronics Engineering from International Institute of Information Technology, Bhubaneswar, Odisha, India in 2021. She is currently working as a System Engineer at Infosys Limited, Mysore. Her research areas of interest include Machine Learning, Data Analytics Big Data, Database Systems, Neural Networks, MongoDB, Hadoop Framework. She can be contacted at email: smousumipatro330@gmail.com.

Surender Reddy Salkuti 🆔 🔍 SC Ⓟ received the Ph.D. degree in electrical engineering from the Indian Institute of Technology, New Delhi, India, in 2013. He was a Postdoctoral Researcher with Howard University, Washington, DC, USA, from 2013 to 2014. He is currently an Associate Professor with the Department of Railroad and Electrical Engineering, Woosong University, Daejeon, South Korea. His current research interests include power system restructuring issues, ancillary service pricing, real and reactive power pricing, congestion management, and market clearing, including renewable energy sources, demand response, smart grid development with integration of wind and solar photovoltaic energy sources, artificial intelligence applications in power systems, and power system analysis and optimization. He can be contacted at email: surender@wsu.ac.kr.

Green building factor in machine learning based condominium price prediction

Suraya Masrom[1], Thuraiya Mohd[2], Abdullah Sani Abd Rahman[3]
[1]Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Perak Branch, Malaysia
[2]Faculty of Architecture, Planning and Surveying, Universiti Teknologi MARA, Perak Branch, Malaysia
[3]Faculty of Sciences and Information Technology, Universiti Teknologi PETRONAS, Perak, Malaysia

Article Info	ABSTRACT
	The negative impact of massive urban development promotes the inclusion of green building aspects in the real estate and property industries. Green building is generally defined as an environmentally friendly building, which rapidly emerged as a national priority in many countries. Acknowledging the benefits of green building, Green Certificate and Green Building Index (GBI) has been used as one of the factors in housing prices valuation. To predict a housing price, a robust approach is crucial, which can be effectively gained from the machine learning technique. As research on green building with machine learning techniques is rarely reported in the literature, this paper presents the fundamental design and the comparison results of three machine learning algorithms namely deep learning (DL), decision tree (DT), and random forest (RF). Besides the performance comparisons, this paper presents the specific weight correlation in each of the machine learning models to describe the importance of the green building to the model. The results indicated that RF has been outperformed others while Green Certificate and GBI have only been slightly important in the DL model. *This is an open access article under the [CC BY-SA](#) license.*

Corresponding Author:

Thuraiya Mohd
Faculty of Architecture, Planning and Surveying, Universiti Teknologi MARA
Perak Branch, Malaysia
Email: thura231@uitm.edu.my

1. INTRODUCTION

The increasing demand for condominiums in urban areas has created environmental pollution issues. As a result, green building has been given wide attention by properties stakeholders such as investors, developers, occupiers, and consumers in combating the issue of environmental impact. In Malaysia, the real estate market has reached a high level of maturity, where house buyers are becoming more selective and demanding [1]. Besides prime location, green building elements such as attractive landscape, indoor air quality, non-toxic and sustainable materials, are regarded as higher preferences by the house owners [2]. Therefore, predicting the condominium prices with green buildings is becoming important for the developer as well as to the potential buyers so that they can acquire information on the condominium price trends.

Recently, machine learning [3], [4] has become a vital predictive approach in a variety of domains [5]–[7], including in real estate valuation. Several studies have proven the capability of machine learning in generating higher accurate results in housing price prediction [8], [9]. Despite the widespread use of machine learning in real estate valuation as well as in the building price prediction, the inclusion of the green building aspect is difficult to be found in the literature. Recent studies on green building mostly relied on conventional

approaches such as multiple linear regression. Therefore, the objective of this study is to extend the state-of-the-art of green building by focusing on machine learning price prediction models.

The existence of advanced techniques of machine learning can be deployed and to start with a few of them is highly crucial. Besides identifying the machine learning performances through prediction accuracy and time efficiency, another important question that needs to be answered is how the green building aspect contributed to the performances of different machine learning models. This issue has not been broadly discussed in the existing research reports.

One of the important steps in machine learning deployment is to select the machine learning algorithms [10]. In the real estate industry, tree-based machine learning [11], [12] and deep learning (DL) [13] have been utilized wisely. Two common types of tree-based machine learning are random forest (RF) and decision tree (DT). These two algorithms have been reported as the best outperforming machine learning when tested on different cases of housing price prediction such as in [14], where the researchers have looked into the contribution of economic variables to RF modeling of the real estate market. By viewing different multi-featured aspects, researchers in [15], studied the performances of the RF algorithm in the housing price prediction. Different in [16] and [17], the researchers focused on the DT algorithm for implementing housing price estimation. As in [18], DL was used for the application of predicting housing prices of the real estate market in Taiwan while researchers in [19] reported the advantages of DL over the autoregressive integrated moving average (ARIMA) model in the forecasting of housing prices. Smart real estate assessment [20], DL with XGBoost [21] and DL based on textual information [22] are among the advancements of DL research in the housing industries.

2. RESEARCH METHOD
2.1. Data collection and datasets

In this research, the scope of the building area is within the Federal Territory of Kuala Lumpur district. This is because most development of the condominium green building is located in this area [23]. The related data on the green condominium building of the Kuala Lumpur district were collected from the valuation and property service department [24], which after data cleaning, 240 records with 14 columns were used for the models. The dependent variable is the transaction price per square feet in Ringgit Malaysia (RM). The indicator of green building status was named as Green Certificate, which consists of the certification and the Green Building Index (GBI) code developed by the Malaysian Institute of Architects and the Association of Consulting Engineers Malaysia (ACEM) [25]. The green building status that has been given to the condominium, can be either certified or index (silver, gold and platinum). In the collected dataset, 81.7% are green building certified while the rest 18.3% are given by GBI index value. From the GBI index building, 5.4% are silver indexed, 12.5% are gold indexed and 0.4% with platinum indexed. By using Pearson correlation, the correlation of 13 independent variables to the condominium Transaction Price is listed in Table 1.

The correlation weights in Table 1 are a global weight, which indicates how important the variables are to the Transaction Price, beyond the specific prediction model. However, one interesting feature in RapidMiner software is specific dependency weight that calculates the important contribution of independent variables in a particular machine learning model. Therefore, although the Green Certificate variable contributed a very low correlation to the transaction price in general as shown in Table 1, it would be interesting to look at how important it is on the specific machine learning models.

Table 1. Selected independent variables with the Pearson correlation weight

Independent variable	Correlation weight
Green Certificate	0.032
Level Property Unit	0.062
Building Floor	0.070
Date of Transaction	0.135
Distance	0.190
Age of Building	0.227
Type of Property	0.282
No of Bedroom	0.242
Security of Building	0.350
Mukim	0.371
Population Density	0.458
Lot Area	0.714
Main Floor Area	0.714

2.2. Machine learning

The dataset was divided with a split training and testing approach with ratio 60:40 percentages. Thus, from the 240 records, 144 of the datasets were used for the machine learning training and 90 records were used for testing. For the DL algorithm, the configuration is given in Table 2.

For tree-based machine learning, preliminary experiments have been conducted to identify the optimal parameters. As for DT, the relevant parameter is the tree maximal depth. As listed in Table 3, six values of maximal depth have been observed. It has been identified that the lowest error rate was generated with 7 maximal depths.

RF is an extension of the DT algorithm, which has one additional parameter besides the maximal depth. As shown in Table 4, the maximal depth is dedicated to the internal sub-trees. Based on 12 configurations of sub-trees and depths, the most optimal setting has been presented by 20 numbers of trees and 7 maximal depths. This configuration generated 13.6% error rate.

All the experiments were implemented with the RapidMiner software tool in a notebook computer with 16 GB RAM. Figure 1 shows some of the processes in RapidMiner, which is to set the training and testing to 60:40 percentages. Therefore, from the 240 records, 144 were used for training and the rest 96 records for testing.

Table 2. DL configuration

Parameters	Configuration
Number of epochs	10
Output function	Linear
Inner function	Rectifier
Number of layers	4
Number of neurons per layer	Layer 1: 13 neurons
	Layer 2: 50 neurons
	Layer 3: 50 neurons
	Layer 4: 1 neuron

Table 3. DT optimal parameter

Maximal depth	Error rate
2	35
4	18
7	**16.7**
10	16.8
15	16.9
25	16.9

Table 4. RF optimal parameter

Number of trees	Maximal depth	Error rate (%)
20	2	31.3
60	2	30.4
100	2	31.8
140	2	30.8
20	4	17.2
60	4	17.8
100	4	18.0
140	4	18.0
20	**7**	**13.6**
60	7	15.9
100	7	15.1
140	7	15.5

3.   RESULTS AND DISCUSSION

In this section, the results of the machine learning models are presented in two divisions. Firstly, the performance results of each machine learning algorithm are given regarding the prediction accuracy and processing time. Secondly, the weight of contributions of each independent variable to the building transaction price is analyzed to identify the importance of the GBI in each of the machine learning models.

3.1. The machine learning performances

Table 5 lists the performance results of each machine learning model. The $R^\wedge$ indicates the correlation between predicted values and actual values [26] which, the more it nearer to 1, the fitter it is. Root

mean square error (RMSE) [27] explains the average distances/differences between the prediction and the actual values. Additionally, relative error describes how large the error is relative to the actual values in percentage ratio. It can be seen that all machine learning models performed very well to predict the transaction price with a lower error value.
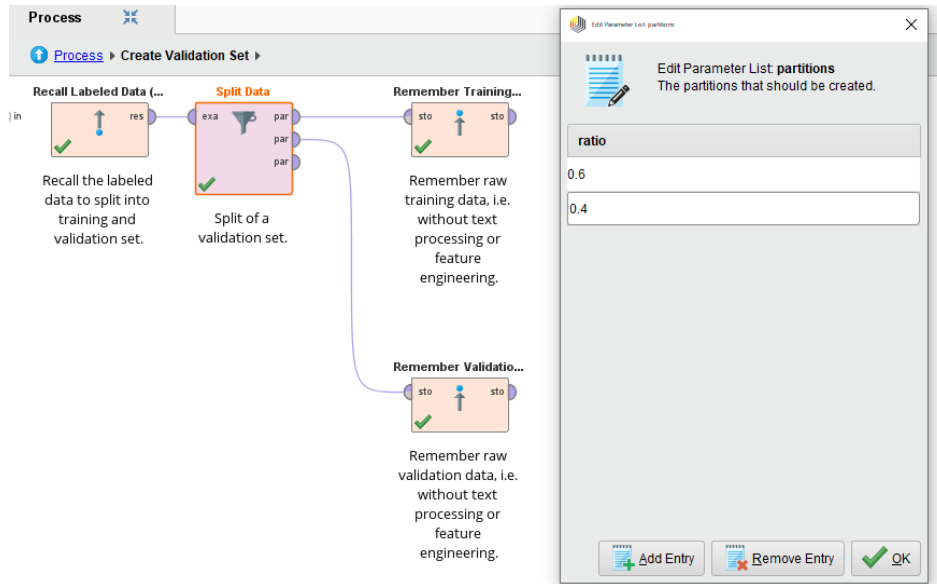

Figure 1. The ratio of training and testing for all the machine learning models

Table 5. The R^ and root mean square error (RMSE) results

Machine learning algorithm	R^ (+/-Std.Dev)	RMSE (+/-Std.Dev)	Relative Error (+/-Std.Dev)
Deep learning (DL)	0.932 (0.069)	684036.166(288080.644)	13.8% (2.8%)
Decision tree (DT)	0.848(0.278)	826146.506(295228.403)	14.3% (1.4%)
Random forest (RF)	0.936(0.116)	636316.217(371553.444)	12.6% (1.2%)

The most outperformed model from the three machine learning algorithms was RF with 94% fitness and 12.6% relative error. A very small difference has been generated by the DL algorithm with 0.93% fitness and 13.8% relative error. Although DT has presented the lowest performances compared to the two algorithms, the results were still within a good range, which is above 70% of R^ and below 20% of relative error. Additionally, in terms of efficiency in completing the prediction, DT was the fastest algorithm as presented in Figure 2. It only took 106 seconds of total time to complete the training and prediction testing. Otherwise, both DL and RF have taken up to above 400 seconds of total times.
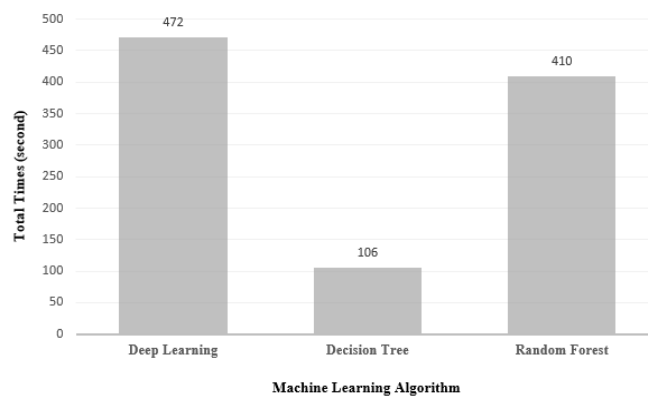

Figure 2. The efficiency of each machine learning model measured from the total times

Furthermore, the following Figures 3 to 5 can illustrate the accuracy of each model. The prediction charts show the prediction versus the actual values of the transaction price. The more predicted values (gray dot) that closer to the diagonal dotted line means the better is the machine learning.
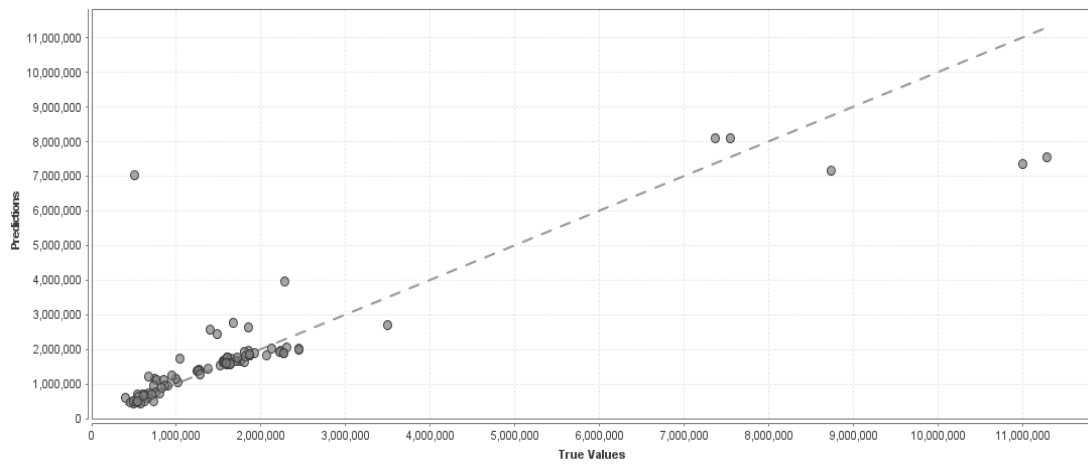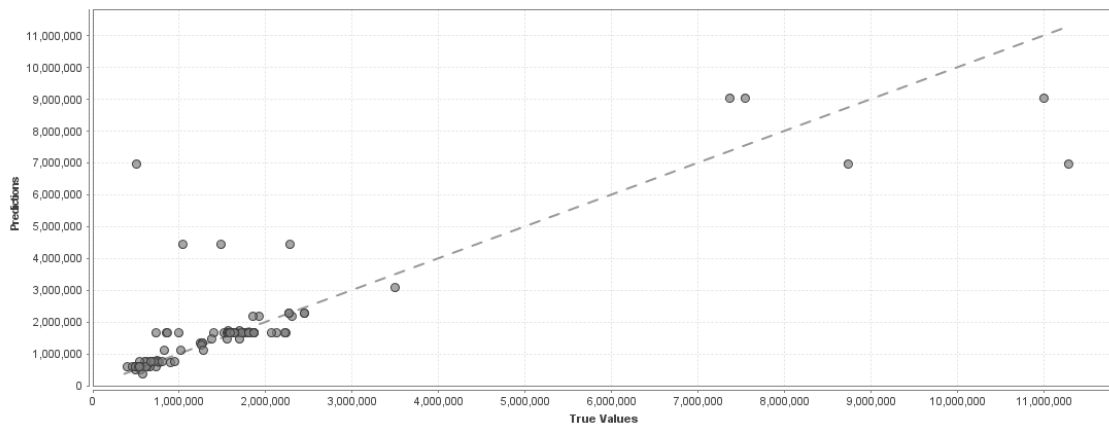

Figure 3. The prediction chart of DL
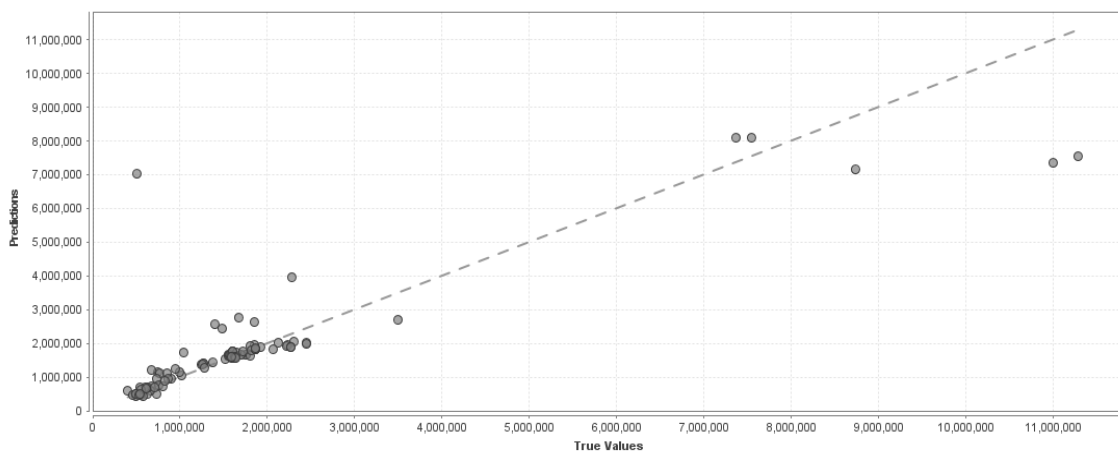

Figure 4. The prediction chart of DT

Figure 5. The prediction chart of RF

DT in Figure 4 has more gray dots that are far from the dotted line compared to DL and RF. Therefore, it shows that DT produced lower accuracy results compared to the other two algorithms. The results plotted in these prediction charts were consistent with the results listed in Table 5.

3.2. The correlations of variables in the machine learning models

This section explains the importance of each independent variable in each of the machine learning models. Figure 6 presents the weights of correlation of each independent variable/attribute in the DL model. It can be seen that only 11 out of the 13 attributes were contributed to the price prediction in DL.
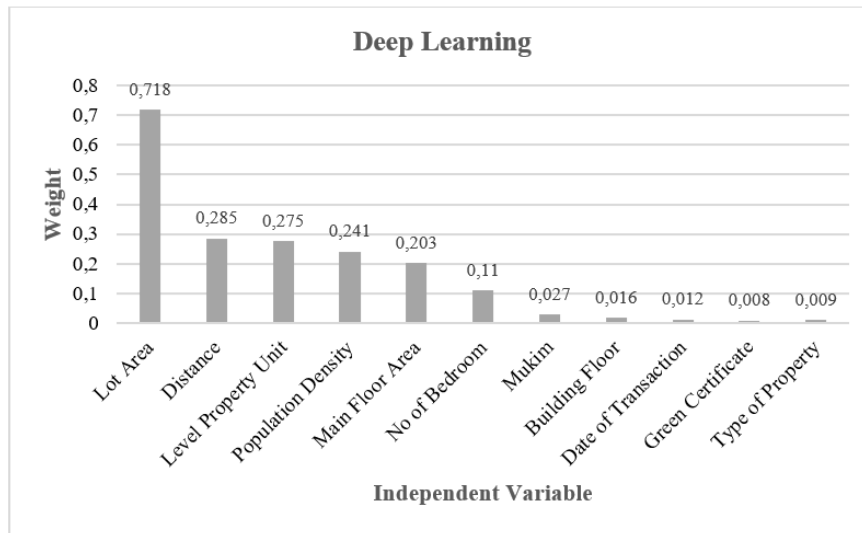

Figure 6. Weight of each independent variable in DL

It can be seen in Figure 6 that the Green Certificate has a very low weight (0.0078) to the prediction of transaction price in the DL model. Main floor area, which generally has a strong correlation with transaction price as given in Table 1 before, inversely has lower weight in DL (0.2). The only variable that has a very strong weight in DL is lot area (0.7 weight value). In contrast, main floor area has become the most important variable in the DT prediction model as presented in Figure 7.
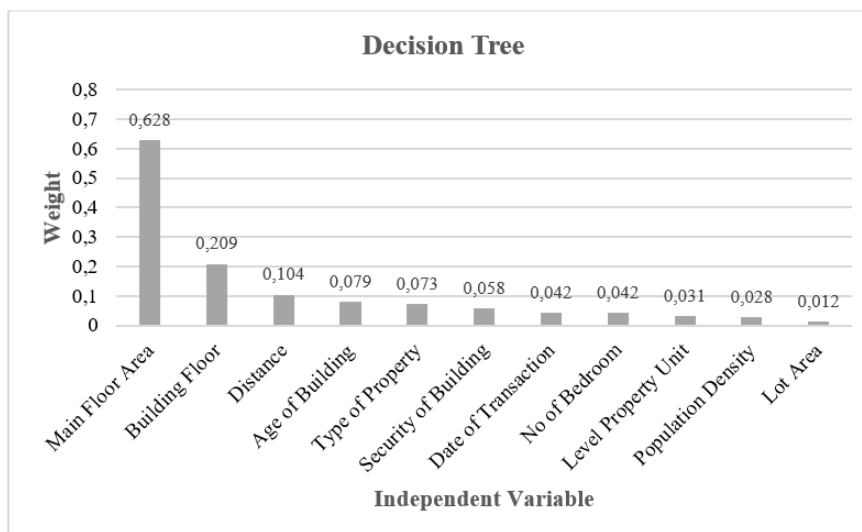

Figure 7. Weight of each independent variable in DT

In Figure 7, the green certificate seems to be no more significant in DT. Different from DL, main floor area is the most important to DT (0.6) but the weight of the lot area is only 0.012, which is very low compared to 0.7 in DL. Lastly, Figure 8 presents the correlation weights of the independent variables in RF.
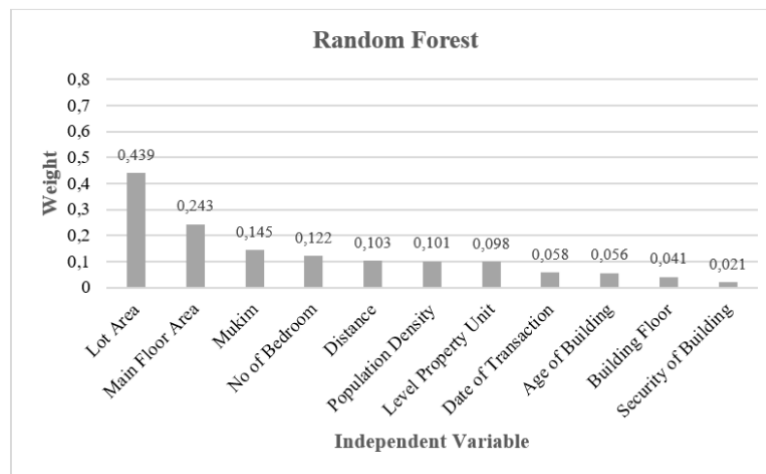

Figure 8. Weight of each independent variable in RF

Similar to DT, the Green Certificate was disappeared to have any weight in RF. Nevertheless, similar to DL, lot area played the most important role in RF. However, Lot Area in RF has a very low weight (0.439) compared to the weight in DL (0.718).

Regardless of each machine learning model, it can be generally viewed that Green Certificate and GBI have not yet been an important factor in the condominium price prediction for the area of the Federal Territory of Kuala Lumpur. This is maybe caused by the imbalanced GBI data distribution in the dataset. Additionally, the insufficient green building dataset can indicate that the green building prospect in Malaysia is still in its infancy.

4. CONCLUSION

This paper presents the report of research that seeks to address the role of green building through the Green Certificate and GBI in predicting the condominium price. Focused on DL, DT and RF algorithms, interpretation of the finding was described with some limitations on the methodology as well as on the tested dataset. The main challenge of this study is the limited size of green building datasets from the metropolitan area of Kuala Lumpur. Therefore, more extensive works are needed shortly for the green building as well as in the price prediction model either with the established conventional methods or with the robust machine learning approaches.

ACKNOWLEDGEMENTS

REFERENCES

[1] M. Solla, L. H. Ismail, A. sharainon M. Shaarani, and A. Milad, "Measuring the feasibility of using of BIM application to facilitate GBI assessment process," *Journal of Building Engineering*, vol. 25, Art. no. 100821, Sep. 2019, doi: 10.1016/j.jobe.2019.100821.

[2] P. A. M. Khan, A. Azmi, N. H. Juhari, N. Khair, and S. Z. Daud, "Housing preference for first time home buyer in Malaysia," *International Journal of Real Estate Studies*, vol. 11, no. 2, pp. 1–6, 2017.

[3] V. S. Padala, K. Gandhi, and P. Dasari, "Machine learning: the new language for applications," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 8, no. 4, pp. 411–421, Dec. 2019, doi: 10.11591/ijai.v8.i4.pp411-421.

[4] B. Mahesh, "Machine learning algorithms-a review," *International Journal of Science and Research (IJSR)*, vol. 9, no. 1, pp. 381–386, 2020.

[5] N. Razali, S. Ismail, and A. Mustapha, "Machine learning approach for flood risks prediction," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 9, no. 1, pp. 73–80, Mar. 2020, doi: 10.11591/ijai.v9.i1.pp73-80.

[6] M. E. Amran *et al.*, "Optimal distributed generation in green building assessment towards line loss reduction for Malaysian public hospital," *Bulletin of Electrical Engineering and Informatics*, vol. 8, no. 4, pp. 1180–1188, 2019.

[7]   N. S. Ahmad Yasmin, N. A. Wahab, and A. N. Anuar, "Improved support vector machine using optimization techniques for an aerobic granular sludge," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 5, pp. 1835–1843, Oct. 2020, doi: 10.11591/eei.v9i5.2264.

[8]   T. D. Phan, "Housing price prediction using machine learning algorithms: the case of Melbourne City, Australia," in *2018 International Conference on Machine Learning and Data Engineering (iCMLDE)*, Dec. 2018, pp. 35–42, doi: 10.1109/iCMLDE.2018.00017.

[9]   P.-Y. Wang, C.-T. Chen, J.-W. Su, T.-Y. Wang, and S.-H. Huang, "Deep learning model for house price prediction using heterogeneous data analysis along with joint self-attention mechanism," *IEEE Access*, vol. 9, pp. 55244–55259, 2021, doi: 10.1109/ACCESS.2021.3071306.

[10]   M. Praveena and V. Jaiganesh, "A literature review on supervised machine learning algorithms and boosting process," *International Journal of Computer Applications*, vol. 169, no. 8, pp. 32–35, Jul. 2017, doi: 10.5120/ijca2017914816.

[11]   R. S. Olson, N. Bartley, R. J. Urbanowicz, and J. H. Moore, "Evaluation of a tree-based pipeline optimization tool for automating data science," in *Proceedings of the Genetic and Evolutionary Computation Conference 2016*, Jul. 2016, pp. 485–492, doi: 10.1145/2908812.2908918.

[12]   H. Shamsudin, M. Sabudin, and U. K. Yusof, "Hybridisation of RF(Xgb) to improve the tree-based algorithms in learning style prediction," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 8, no. 4, pp. 422–428, Dec. 2019, doi: 10.11591/ijai.v8.i4.pp422-428.

[13]   I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[14]   S. Levantesi and G. Piscopo, "The importance of economic variables on london real estate market: a random forest approach," *Risks*, vol. 8, no. 4, Art. no. 112, Oct. 2020, doi: 10.3390/risks8040112.

[15]   R. Sawant, Y. Jangid, T. Tiwari, S. Jain, and A. Gupta, "Comprehensive analysis of housing price prediction in pune using multi-featured random forest approach," in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, Aug. 2018, pp. 1–5, doi: 10.1109/ICCUBEA.2018.8697402.

[16]   H. Wu and C. Wang, "A new machine learning approach to house price estimation," *New Trends in Mathematical Science*, vol. 4, no. 6, pp. 165–171, Dec. 2018, doi: 10.20852/ntmsci.2018.327.

[17]   G.-Z. Fan, S. E. Ong, and H. C. Koh, "Determinants of house price: a decision tree approach," *Urban Studies*, vol. 43, no. 12, pp. 2301–2315, Nov. 2006, doi: 10.1080/00420980600990928.

[18]   C. Zhan, Z. Wu, Y. Liu, Z. Xie, and W. Chen, "Housing prices prediction with deep learning: an application for the real estate market in Taiwan," in *2020 IEEE 18th International Conference on Industrial Informatics (INDIN)*, Jul. 2020, vol. 2020-July, pp. 719–724, doi: 10.1109/INDIN45582.2020.9442244.

[19]   F. Wang, Y. Zou, H. Zhang, and H. Shi, "House price prediction approach based on deep learning and ARIMA model," in *2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT)*, Oct. 2019, pp. 303–307, doi: 10.1109/ICCSNT47585.2019.8962443.

[20]   H. Xu and A. Gade, "Smart real estate assessments using structured deep neural networks," in *2017 IEEE SmartWorld, Ubiquitous Intelligence and Computing, Advanced and Trusted Computed, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, Aug. 2017, pp. 1–7, doi: 10.1109/UIC-ATC.2017.8397560.

[21]   Y. Zhao, G. Chetty, and D. Tran, "Deep learning with XGBoost for real estate appraisal," in *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, Dec. 2019, pp. 1396–1401, doi: 10.1109/SSCI44817.2019.9002790.

[22]   X. Zhou, W. Tong, and D. Li, "Modeling housing rent in the atlanta metropolitan area using textual information and deep learning," *ISPRS International Journal of Geo-Information*, vol. 8, no. 8, Art. no. 349, Aug. 2019, doi: 10.3390/ijgi8080349.

[23]   S. M. Algburi, A. A. Faieza, and B. T. H. T. Baharudin, "Review of green building index in Malaysia; existing work and challenges," *International Journal of Applied Engineering Research*, vol. 11, no. 5, pp. 3160–3167, 2016.

[24]   K. J. Kam, S. Y. Chuah, T. S. Lim, and F. Lin Ang, "Modelling of property market: the structural and locational attributes towards Malaysian properties," *Pacific Rim Property Research Journal*, vol. 22, no. 3, pp. 203–216, Sep. 2016, doi: 10.1080/14445921.2016.1234361.

[25]   T. L. Mun, "The development of GBI Malaysia (GBI)," *Pam/Acem*, no. April 2008, pp. 1–8, 2009.

[26]   E. Kasuya, "On the use of r and r squared in correlation and regression," *Ecological Research*, vol. 34, no. 1, pp. 235–236, Jan. 2019, doi: 10.1111/1440-1703.1011.

[27]   W. Wang and Y. Lu, "Analysis of the mean absolute error (MAE) and the root mean square error (RMSE) in assessing rounding model," *IOP Conference Series: Materials Science and Engineering*, vol. 324, no. 1, Art. no. 012049, Mar. 2018, doi: 10.1088/1757-899X/324/1/012049.

BIOGRAPHIES OF AUTHORS

Associate Professor Ts. Dr Suraya Masrom ⓘ 🔍 SC Ⓟ is the head of Machine Learning and Interactive Visualization (MaLIV) Research Group at Universiti Teknologi MARA (UiTM) Perak Branch. She received her Ph.D. in Information Technology and Quantitative Science from UiTM in 2015. She started her career in the information technology industry as an Associate Network Engineer at Ramgate Systems Sdn. Bhd (a subsidiary of DRB-HICOM) in June 1996 after receiving her bachelor's degree in computer science from Universiti Teknologi Malaysia (UTM) in Mac 1996. She started her career as a lecturer at UTM after receiving her master's degree in computer science from Universiti Putra Malaysia in 2001. She transferred to the Universiti Teknologi MARA (UiTM), Seri Iskandar, Perak, Malaysia, in 2004. She is an active researcher in the meta-heuristics search approach, machine learning, and educational technology. She can be contacted through email at suray078@uitm.edu.my.

Thuraiya Mohd is an Associate Professor in Universiti Teknologi MARA. Graduated with Ph.D. in Real Estate in 2012 and MSc in Land Development and Administration in 2003 from Universiti Teknologi Malaysia. Received Bachelor's degree in Estate Management from Universiti Teknologi MARA in 2001. Before joining Universiti Teknologi MARA, she served with Ismail & Co as a Valuation Executive for 2 years. Lecturing experience in UiTM, Perak Branch for 20 years in the Department of Estate Management. Lecturing experience includes teaching core courses of Valuation and Property Development to undergraduate students, Built Environment Theory to PhD students and Economics of Green Architecture to Masters students. She was awarded an Excellent University Community Transformation Centre (UCTC) Award 2015 by the Ministry of Education (MOE), MALAYSIA. Received a few grants (as leader) from the MOE and Ministry of Finance, MALAYSIA in the areas of property development, machine learning, and disaster management. Her current research interests focus on areas of green development, sustainable real estate, housing, and disaster management. She also has been involved in machine learning research that applied to real estate problems. She is currently one of the research members of the Machine Learning and Interactive Visualization Research Group at UiTM Perak Branch. She achieved Professional Qualifications as Surveyor (Sr) and is a professional member of the Board of Valuers, Appraisers, Estate Agents and Property Managers (BOVEAP). She can be contacted at email: thura231@uitm.edu.my.

Ts. Abdullah Sani Abd Rahman obtained his first degree in Informatique majoring in Industrial Systems from the University of La Rochelle, France in 1995. He received a master's degree from Universiti Putra Malaysia in Computer Science, with specialization in Distributed Computing. Currently, he is a lecturer at the Universiti Teknologi PETRONAS, Malaysia and a member of the Institute of Autonomous System at the same university. His research interests are cybersecurity, data analytics and machine learning. He is also a registered Professional Technologist. He can be contacted at email: sani.arahman@utp.edu.my.

Machine learning algorithms for electrical appliances monitoring system using open-source systems

Viet Hoang Duong, Nam Hoang Nguyen
Department of Instrumentation and Industrial Information, Hanoi University of Science and Technology, Vietnam

Article Info	ABSTRACT
	Two main methods to minimize the impact of electricity generation on the environment are to exploit clean fuel resources and use electricity more effectively. In this paper, we aim to change the user's electricity usage by providing feedback about the electrical energy consumed by each device. The authors introduced two devices, load monitoring device (LMD) and activity monitoring device (AMD). The function of the LMD is to provide feedback on the operating status and energy consumption of electrical appliances in a home, which will help people consume electrical energy more efficiently. The parameters of LMD are used to predict the on/off state of each electrical appliance thanks to machine learning algorithms. AMD with audio sensors can assist LMD to distinguish electrical devices with the same or varying power over time. The system was tested for three weeks and achieved a state prediction accuracy of 93.60%. *This is an open access article under the [CC BY-SA](#) license.*

Corresponding Author:

Nam Hoang Nguyen
School of Electrical Engineering, Hanoi University of Science and Technology
No. 1 Dai Co Viet, Hai Ba Trung District, Hanoi, Vietnam
Email: nam.nguyenhoang@hust.edu.vn

1. INTRODUCTION

This paper is an extension of work originally presented in the 2019 International Conference on Advanced Computing and Applications (ACOMP) [1]. Electricity generation plays a significant role in increasing greenhouse gas emissions, and this situation is getting worse. In the US, electricity consumption in 2018 was 16 times higher than that in 1950 [2]. Also, in this year, electric energy primarily came from three sources: residential (39%), commercial (36%), and industry (25%). It is clear that the highest proportion is in the residential section, so it's needed to take measures to reduce electricity consumption from this source.

According to [3], Sarah Darby estimated that could save up to 15% electrical energy if aware of the feedback on the power consumption of electrical appliances. Similarly, Sébastien Houde researched the effects of power consumption feedback on users on a large scale (1065 apartments in eight months) [4]. The results show that total power consumption reduced by about 5.7%. Furthermore, Carrie Armela pointed out that if users were given feedback on the energy consumption of each electrical appliance, they could consume 12% less electrical energy [5].

Instead of changing users' habits, Tsai *et al.* built a system to automatically monitor and adjust indoor electricity usage using machine learning algorithms [6]. Research shows that electrical appliances in idle or standby mode account for 3-11% of total indoor energy consumption. Therefore, the team trained the system to completely turn off electrical devices when not in use instead of operating in idle or standby mode. To accomplish this, many smart plugs with electrical metering and on/off functions are installed in the house.

Since it is time-consuming and costly to integrate each smart plug into every device, in this paper, the authors move towards changing users' habits by providing feedback on power consumption levels. To

accomplish this purpose, the authors introduced a load monitoring device (LMD) to monitor the on/off status and power consumption of each electrical device in the house. Besides, an activity monitoring device (AMD) with an audio sensor is also used for the purpose of supporting LMD to identify electrical devices with the same power or continuously varying power over time.

2. SURVEY ON MONITORING ELECTRICAL APPLIANCES

Hart introduced the concept of non-intrusive appliance load monitoring (NALM) to indicate a system that identifies electrical appliances using only a single electronic meter [7]. Each electrical device has different active power (P) and reactive power (Q). Therefore, these two parameters are used as "Signature" for each electrical appliance.

Applying hart's device identification method, Weiss *et al.* used a commercial digital meter and software on the phone to perform the device recognition algorithms [8]. The algorithms allow us to add a new device through the software on the phone and then save the parameters of this device in a database. The system has a recognition accuracy of up to 87%. Laughman *et al.* used harmonics as an identifier for each device [9]. The authors show that a computer and an incandescent bulb and have similar P and Q. However, only the computer can produce a third harmonic, and the incandescent lamp does not. Therefore, harmonics can be used to differentiate these two devices.

Another method developed by Norford and Leeb is analyzing the transient state [10]. Each device with a different structure will have a different switching power at start-up. Srinivasan *et al.* used harmonic parameters as inputs to various machine learning algorithms for electrical device identification [11]. The results showed that the machine learning method multilayer perceptron (MLP) and radial basis function (RBF) gave similar results and were better than the support vector machine (SVM) method. Patel *et al.* observed noise occurring on the line every time a device was turned on/off or operating [12]. The authors point out that the resistive load does not cause noise in operation but does cause transient noise when turned on or off. Inductive and solid-state loads cause additional noise during operation.

Lam *et al.* proposed a solution to use the voltage-current trajectory (V-I trajectory) as identifiers for each electrical appliance [13]. This method plots the graphs of voltage and current over one cycle and then relies on the shapes of graphs to analyze and classify electrical devices. Applying the V-I trajectory method in practice, Baets *et al.* used convolutional neural networks to analyze V-I trajectory images [14].

Because the application of machine learning algorithms in NALM is very potential, some papers evaluated the performance of some machine learning models in the NALM application [11], [15]. Kolter and Matthew have built a massive database for the development of identification algorithms [16]. The database includes information about the energy consumption of many devices in 10 homes for 19 days, with a total data capacity of up to 1 terabyte of raw data. In contrast, to reduce manual labeling data, Khaled Chahine developed a system that can extract the signatures and label them automatically [17].

Due to the hardware and software complexity in the NALM application, Semwal and Prasa focused on optimization algorithms using minimum features from smart meters [18]. Iksan *et al.* proposed a smoothing method for filtering out peak signals [19]. The system achieved better accuracy with this method.

Instead of using electric meters, Laput *et al.* used only a printed circuit board (PCB) with multiple sensors [20]. This board can recognize not only electrical devices but also indoor activities such as opening/closing doors, removing tissue paper, and draining the faucet. In particular, most events have a significant impact on the microphone and the accelerometer. The test was conducted in many places and achieved an accuracy of 96%.

A lot of research about device state recognition already published. Those researches mainly focus on new identifiers to solve problems that exist when using old identifiers. In this paper, common electrical parameters and the MLP network model are used to identify the states of electrical appliances. Besides, the AMD with a microphone is also used for the purpose of supporting the identification of electrical devices with the same power or continuous varying power over time.

3. THE SCOPE AND NOVELTY OF THE PAPER

The authors conducted the experiment in a private house instead of a large building. We do not perform the identification of electrical appliances whose powers change continuously over time. The AMD with a microphone is capable of analyzing acoustic noises from running appliances to differentiate them. This device has been built to support LMD to realize time-varying power as well as similar power appliances. However, in this paper, we have not developed algorithms to combine data of LMD with AMD.

With the purpose of developing an accessible, easy-to-use, and low-cost measurement device, we have built the LMD based on open-source platforms (both hardware and software). The hardware schematics are relatively simple, with only voltage and current measurement channels. The Arduino software library is

extremely accessible to newcomers. Therefore, it is much less time-consuming for newcomers to be familiar with the device. Also, they can easily customize the hardware and develop new features for the device.

As opposed to other mentioned papers in section 2, when the authors made many efforts to figure out new identifying characteristics from appliances, we use only some basic electrical parameters (e.g., voltage, current, and power) that can be obtained with the simple hardware to recognize household appliances. As demonstrated in [20], most events have an impact significantly on sound and vibration sensors. Thus, instead of using advanced electrical parameters (e.g., harmonics, noises on the line) that require complicated hardware, we only add the AMD with the microphone to support LMD to recognize appliances.

4. PROPOSED SYSTEM DESIGN

4.1. Open-source design concepts

The concept of open-source hardware was first released by Bruce Perens in 1997. Open-source hardware makes it easy for everyone to design and develop new features for their own projects. In addition, open-source software is said to be highly reliable because many communities are involved in debugging and testing. As a result, open-source will make it faster and easier for people to conduct their projects. Thanks to these reasons, open-source boards are also of great use for education [21], [22].

Arduino is a company that specializes in providing boards with open source in both hardware and software. Arduino provides an easy-to-use integrated development environment (IDE) that contains many simple functions and libraries in C++ languages. In this paper, in order to have an easy-to-customize measuring device, we developed a digital meter, LMD, based on the Arduino Due board. Similarly, a popular open hardware device is Shenzhen Xunlong Software's Orange Pi Zero board. This board uses a high-performance ARM® Cortex™ -A7 microprocessor, so AMD uses this board to analyze sound and differentiate home electrical appliances.

4.2. System architecture

As shown in Figure 1, there are two types of monitoring devices: LMD and AMD. The LMD is programmed to measure electrical parameters with a cycle of one second. LMD passes these data to a computer to run machine learning algorithms that predict the state of electrical appliances. In the following sections of the paper, the authors will analyze problems in using LMD to identify electrical appliances. The first problem is to distinguish devices with similar power consumption, and the second is to distinguish devices with varying power over time. Therefore, AMD equipment with the audio sensor is used to overcome these problems in the future.
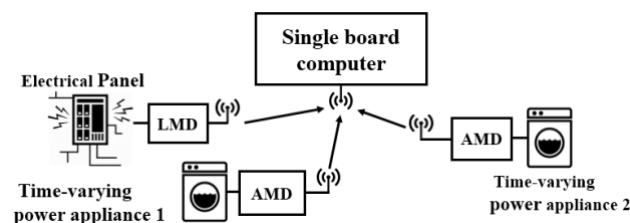

Figure 1. The proposed system model

5. LOAD MONITORING DEVICE (LMD)

5.1. Hardware structure and measurement program

Figure 2 shows the block diagram of the LMD. Because the LMD is integrated into the electrical panel, LMD are powered by the 220 VAC grid. The LMD measurement circuit includes the voltage and current measuring channel. The schematics of these measuring channels are built based on the reference design of microchip technology [23], [24]. Similarly, as mentioned above, the Arduino Due board is used as the central processing unit of the LMD. This block will process the signals from the voltage and current channels, display the calculated parameters on the linear complementary dual (LCD), send them to the computer via Wi-Fi, and store the energy consumed on the electrically erasable programmable read-only memory (EEPROM). LMD can measure six electrical parameters, which are voltage (Urms), current (Irms), active power (P), reactive power (Q), power factor (cosφ), and energy consumption (E). Urms and Irms are calculated by (1), as shown in Figure 2.
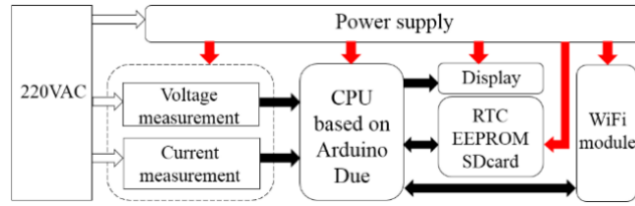
Figure 2. The hardware block diagram of LMD

$$U_{rms} = \sqrt{\frac{\sum_{n=0}^{N-1} u^2(n)}{N}}; I_{rms} = \sqrt{\frac{\sum_{n=0}^{N-1} i^2(n)}{N}} \tag{1}$$

Active power and reactive power are calculated from $u(n)$ and $i(n)$, as shown in (2). Where $u(n)$, $i(n)$, and $i_{90\text{-}degree\text{-}shift}(n)$ are instantaneous voltage, instantaneous current, and instantaneous current shifted by 90 degrees. N is the number of samples. The sampling frequency is 2000Hz, the measurement data update rate is 1Hz, so N=2000.

$$P = \frac{\sum_{n=1}^{N}[u(n) \times i(n)]}{N}; Q = \frac{\sum_{n=1}^{N}[u(n) \times i_{90-degree-shift}(n)]}{N} \tag{2}$$

5.2. Appliance states detecting algorithm

In recent years, artificial intelligent (AI) has become a phenomenon in the world. Machine learning (ML) is a subset of AI. Machine learning is traditionally divided into three main groups: supervised learning, unsupervised learning, and reinforcement learning. In this system, we combine the supervised learning method with a three-layer MLP to train the system in realizing the state of electrical appliances. As shown in Figure 3 and (3), the input, hidden, and output layers are represented by vectors x, a, and y, respectively. Matrix weight W and bias b represent connections between the two layers.
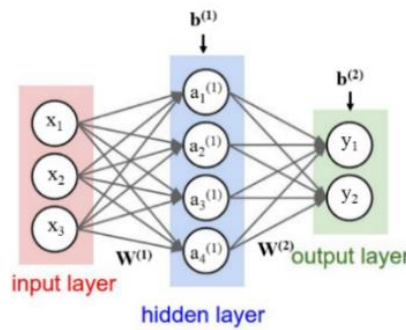

Figure 3. Three layers multilayer perceptron (MLP)

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \dots \end{bmatrix}; a^{(l)} = \begin{bmatrix} a_1^{(l)} \\ a_2^{(l)} \\ \dots \end{bmatrix}; y = \begin{bmatrix} y_1 \\ y_2 \\ \dots \end{bmatrix}; W = \begin{bmatrix} w_{11}^{(l)} & w_{12}^{(l)} & \dots \\ w_{21}^{(l)} & w_{22}^{(l)} & \dots \\ \dots & \dots & w_{ij}^{(l)} \end{bmatrix}; b = \begin{bmatrix} b_1^{(l)} \\ b_2^{(l)} \\ \dots \end{bmatrix} \tag{3}$$

As demonstrated in (4) describe the relationship between the input and output layer. Where l is layer number, l =1, 2, …, L (L is the last layer, $a^{(0)}$ is input vector, $a^{(L)}$ is output vector), $g(z)$ is activation function which is sigmoid function shown in (5).

$$z^{(l)} = W^{(l)}a^{(l-1)} + b^{(l)}; a^{(l)} = g(z^{(l)}) \tag{4}$$

$$g(z) = \frac{1}{1 + e^{-z}} \tag{5}$$

First, we must determine the components of the input vector x and the output vector y. In this case, the input vector is the electrical parameters obtained from the LMD. The output vector is the on/off states of each device. The next step is to determine the optimum W and b matrices to show the relationship between the input and output vectors. With the supervised training method, a training data set is used to find these two matrices. The test data set is then used to test the predictive accuracy of the system.

Active power P and reactive power Q are used as "Signature" for each device. Table 1 shows the power of some indoor appliances to be predicted in the experiment section. We can see from the table that the devices have different P and Q.

Observing the load graph in Figure 4, each time a device is turned on or off, a rising or falling edge appears on the graph. The amplitude of the edge is the power consumption of the device that has been turned on/off. We developed an algorithm called "edge detection" to identify the potential edges. Even when no appliances are turned on/off, the total power is constantly changing. Thus, it is necessary to define a threshold value large enough for the algorithm to eliminate these fluctuations. In this paper, the oscillation threshold for P and Q are 15W and 8VAr, respectively.

Table 1. Specifications of some electrical devices used for the experiment

| No	Appliance	Real Power (W)	Reactive Power (VAr)	Power Factor (cosφ)
1	Hairdryer (mode 1)	455	13	0.99
2	Hairdryer (mode 2)	893	31	0.99
3	Kettle 1	1374	5	0.99
4	Kettle 2	1958	50	0.99
5	LED lamp	22	11	0.89
6	Compact lamp	65	-8	0.99
7	Fan (with electronic circuit)	45	-12	0.96
8	Incandescent lamp	60	0	1.0
9	Chandelier	202	0	1.0
10	Heating lamp (mode 1)	260	0	1.0
11	Heating lamp (mode 2)	526	0	1.0
12	Fluorescent lamp	30	69	0.4
13	Heating bag	730	0	1.0
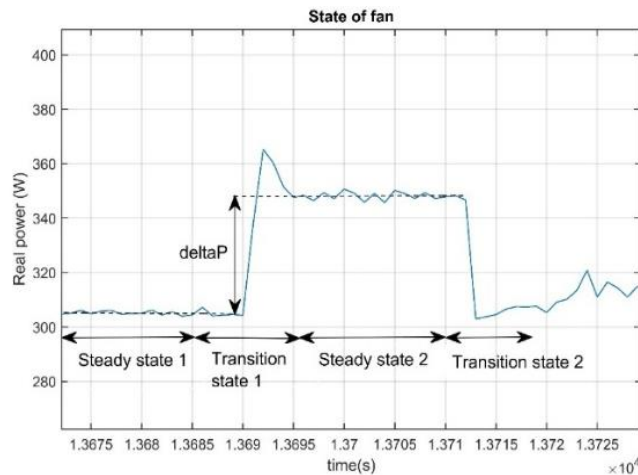
Figure 4. Rising and falling edge when turning on/off the fan

We have the following input vector of the MLP network:

$$x = \begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} \tag{6}$$

A total of 12 devices are used during the test, in which hair-dryer and heating lamp devices have multiple operating modes. Thus, the 32 outcomes are represented by vectors, as shown in Tables 2 and 3. As shown in (7) specifically describes a pair of input vectors $x^{(1)}$ and output vectors $y^{(1)}$. Where m is the number of data points collected, the matrices X and Y are the product of combining all m input and output vectors

($X \in \mathbb{R}^{2 \times m}$, $Y \in \mathbb{R}^{32 \times m}$). These data are divided into two groups as the training set and test set. Training set is used to train the MLP model. The Test Set is used to evaluate the prediction accuracy of the MLP model after it has been trained. The goal of the training process is to find two optimal matrices W and b, so that the predicted outputs approximate the actual outputs. The difference between these two outputs is evaluated using (8) (the error function).

Table 2. Output vector y

Output	1	2	3	…	32
y (Output vector)	$\begin{bmatrix}1\\0\\0\\…\\0\end{bmatrix}$	$\begin{bmatrix}0\\1\\0\\…\\0\end{bmatrix}$	$\begin{bmatrix}0\\0\\1\\…\\0\end{bmatrix}$	…	$\begin{bmatrix}0\\0\\0\\…\\1\end{bmatrix}$

Table 3. Output vector and device states

Output	State
1	Hairdryer is on (mode 1)
2	Hairdryer is off (mode 1)
…	…
32	Heating bag is off

$$x^{(1)} = \begin{bmatrix} 455.36 \\ 12.19 \end{bmatrix}; y^{(1)} = \begin{bmatrix} 1 \\ 0 \\ … \\ 0 \end{bmatrix}; X = \begin{bmatrix} | & | & … \\ x^{(1)} & x^{(2)} & … \\ | & | & … \end{bmatrix}; Y = \begin{bmatrix} | & | & … \\ y^{(1)} & y^{(2)} & … \\ | & | & … \end{bmatrix} \tag{7}$$

$$J = \frac{1}{m} \sum_{i=1}^{m} \sum_{k=1}^{K} -y_k^{(i)} \log_e a_k^{L(i)} - (1 - y_k^{(i)}) \log_e (1 - a_k^{L(i)}) \tag{8}$$

Where a_k^L is predicted output k of the output layer, y_k is the corresponding label k in Training Set, K is the number of units of the output layer ($K=32$), m is the total number of collected data in Training Set. With cost function J, the value J is small when $a_k^L \approx y_k$. The problem to be solved now is to find the minimum value of the function J; from there, we have the matrices W and b, respectively. Finding the minimum value of J by solving as shown in (8) is a complicated task, and thus Gradient Descent and Backward propagation are two useful algorithms to solve this problem [25].

Overall, the process of recognizing the on/off states of electrical devices is described in Figure 5. The computer will always collect the electrical parameters from the meter over Wi-Fi. Once all the parameters have been received, the computer runs the "edge detection" algorithm to detect the changes of P and Q. If it detects an edge ΔP and ΔQ, the computer runs the machine learning algorithm to identify which appliance has just been turned on or off. After that, the computer will return to collect new data from the meter.
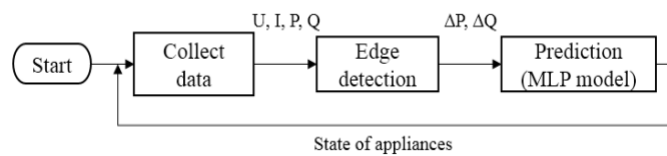

Figure 5. Process of identifying electrical appliance state

6. ACTIVITY MONITORING DEVICE (AMD)

The disadvantage when identifying devices from power P and Q is that it is challenging to identify devices with the same or continuously varying powers over time. AMD equipment is developed for small-scale installations solely to address the two problems mentioned above. According to [20], the two sensors with the most data change every time an event occurs: the sound sensor and the accelerometer. Therefore, the authors use a microphone for AMD to implement system support for device recognition. In this paper, the authors only test AMD functionality but have not combined LMD to identify appliances.

The hardware structure of AMD is illustrated in Figure 6. Direct current (DC) power is converted from a 220 V alternating current (AC) power line to supply the Orange Pi Zero board, a microphone, and a display. Orange Pi Zero runs Ubuntu Ambian operating system and supports Python language. We deploy Python's libraries related to the fast fourier transform (FFT) algorithm for analyzing sounds from electrical appliances.
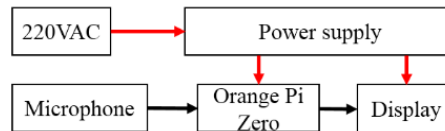
Figure 6. The hardware block diagram of AMD

We performed an acoustic analysis from three typical electrical appliances, the hairdryer, the fan, and the microwave oven. Fan noise mainly comes from ball bearings and propellers when rotating. The sound from the hairdryer is generated by the heater, front and rear grids, and the air filter. The microwave noise comes from the cooling fan to make sure the magnetron does not heat up. Figure 7 comparing measurement results in the spectrum of Figure 7(a) fan (57-58 Hz) and Figure 7(b) hairdryer (900-940 Hz) after ten times of recording and analyzing the sound of three devices at a distance of 1 meter. Figure 8 comparing measurement results in the spectrum of Figure 8(a) microwave oven (195-210 Hz) and Figure 8(b) all three appliances at the same time. It is clear that each device's feature still appears clearly when all of the three appliances operate simultaneously. Therefore, this feature can be used as the "Signature" for each device.
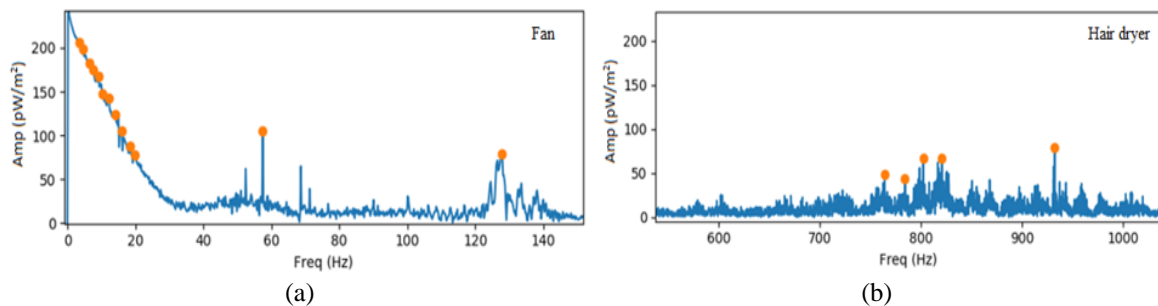

| (a) | (b) |

Figure 7. Spectrums of (a) fan VINAWIND QB 300Đ and (b) hairdryer PHILIPS HP 4840
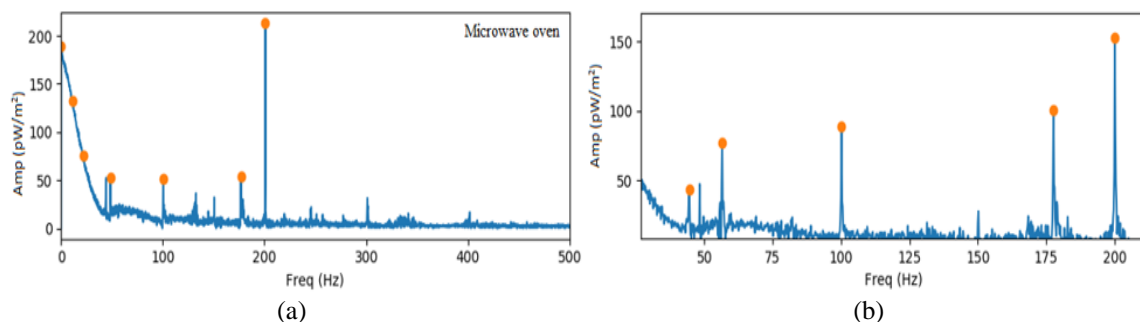

| (a) | (b) |

Figure 8. Spectrums of (a) microwave oven DAEWOO KOG-1A4H and (b) three appliances

7.   EXPERIMENTS

Figure 9 comparing device PCBs of Figure 9(a) the LMD board and Figure 9(b) AMD board. The two boards are based on open-source designs from Microchip/Arduino and the Orange Pi Zero. LMD can measure Urms, Irms, active power P, reactive power Q, power factor, and energy consumption E. The computer will receive these parameters from the LMD via Wi-Fi to perform the algorithm in Figure 5. The accuracy of both voltage and current channels is under 1% after being calibrated.

The authors tested the monitoring system at the private house, including the bathroom, laundry room, and bedroom. The current transformer of the electronic meter was installed to measure power simultaneously in three rooms. The number of devices is listed in Table 1; some of them have multiple operating modes such as hairdryer and heating lamp, so a total of 32 on/off cases needs to be recognized. The

experiment was performed on weekday evenings (monday to friday). At the end of the week, the authors ran the test all day. The total duration of the experiment was three weeks. The MLP model was trained to recognize 32 on/off cases before conducting the experiment. A total of 215 data points were collected. The 70% points for the Training set and 30% points for the Test set. After the end of 30000 iterations, the value of the error function J is 0.295. The prediction result is 93.65% accurate on the test set. After three weeks, the system predicted a total of 766 events, of which 49 were wrongly predicted. As a result, the system achieved a prediction accuracy of 93.60%. Figure 10 shows the active and reactive power graph during the 4th test day. After testing, the authors found some disadvantages of the current algorithm. Figure 11 comparing measurement power of (a) warmer bag and (b) washing machine. The algorithms based on P and Q perform poorly for devices with continuously varying power. For example, observe Figure 11(a), the power of the heating bag gradually increases by more than 240 W over time. Likewise, the power of the washing machine changes continuously between 1958 W and 2150 W in soaking mode as shown in Figure 11(b). Therefore, the algorithm is unable to obtain the correct ΔP value. Second, the system confuses devices with nearly the same power, such as incandescent and compact lamps (almost identical active power P). In some cases, the reactive power Q of the compact lamp is less than the detection threshold of the edge detection algorithm of 8 VAr. The algorithm ignores this power edge, thus incorrectly predicting the incandescent lamp. We built AMD with the microphone to tackle the above two problems in the future. The first version of AMD using the Orange Pi Zero board. The three devices used in the recognition experiment are the hairdryer, the fan, and the microwave oven. The test was conducted at a distance from 0.5 to 2 m. At each distance, AMD predicts the device state multiple times to evaluate prediction accuracy. Table 4 shows the test results; we can see that the system is most accurate at a distance of less than 1.5 m.

(a)   (b)

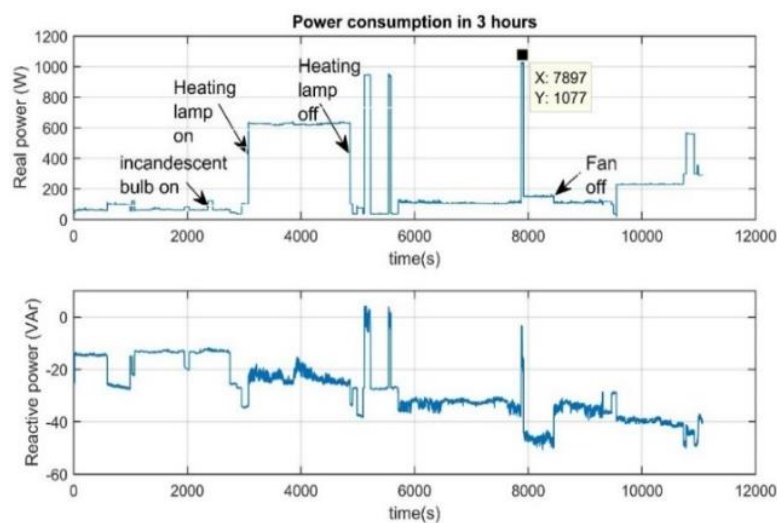Figure 9. Designing of the (a) LMD board and (b) AMD board
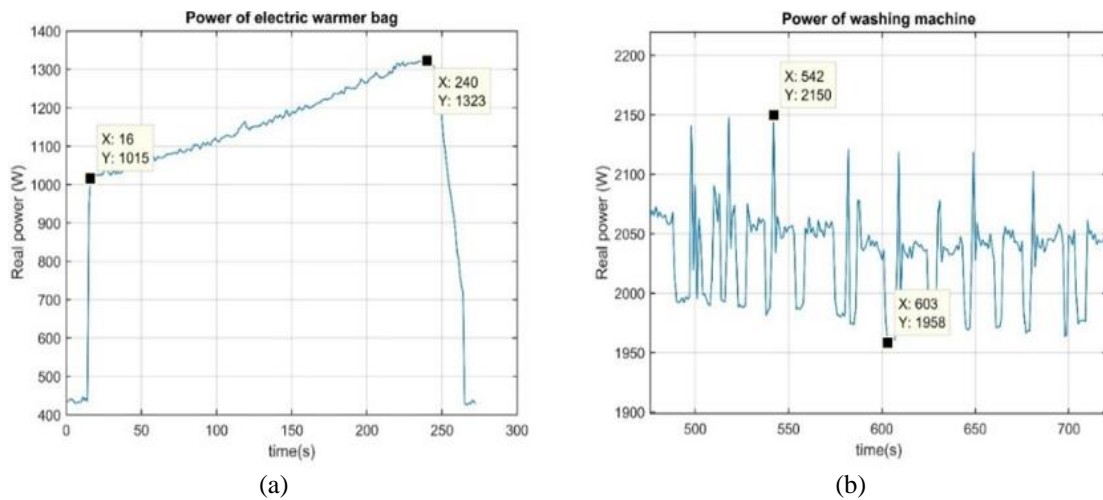

Figure 10. Power consumption graph in 4th test day

Figure 11. Power of (a) warmer bag and (b) washing machine

Table 4. AMD test results

	0.5 m	1 m	1.5 m	2 m
Fan	40/40 (100%)	40/40 (100%)	35/40 (88%)	19/40 (48%)
Hair Dryer	40/40 (100%)	39/40 (98%)	37/40 (93%)	25/40 (62.5%)
Microwave Oven	20/20 (100%)	20/20 (100%)	19/20 (95%)	13/20 (65%)

8. CONCLUSION

This paper presents two devices call LMD and AMD. LMD is the electronic meter that is installed at the electrical panel of a room or a house. LMD provides information about power consumption in order to detect which electrical appliances are running. This device provides electricity consumption information of each electrical appliance to homeowners so that they can adjust their usage plans more efficiently. Meanwhile, AMD is equipped with the microphone; the function of this device is to provide more information on appliances with time-varying power or similar power. The authors have built the first version of the electrical equipment state recognition system using an open-source platform. The system applies the supervised learning method and the MLP model. Active power P and reactive power Q are used as Signatures for each device. The experiment was conducted over three weeks in three different rooms. The authors trained the system to identify 12 devices in which two devices have multiple modes. The accuracy of the system is 93.60%. However, during testing, the authors found that the system works inefficiently for devices with constantly changing power. It is also difficult for the system to differentiate devices with similar power. Therefore, the authors intend to combine data from AMD to solve the two analyzed problems in the future.

REFERENCES

[1] N. H. Nguyen and V. H. Duong, "A system for monitoring the electric usage of home appliances using machine learning algorithms," in *Proceedings - 2019 International Conference on Advanced Computing and Applications, ACOMP 2019*, Nov. 2019, pp. 158–164, doi: 10.1109/ACOMP.2019.00032.
[2] "Electricity explained-use of electricity," *U.S. Energy Information Administration*, 2020. https://www.eia.gov/energyexplained/electricity/use-of-electricity.php (accessed Feb. 2020).
[3] S. Darby, "The Effectiveness of feedback on energy consumption," *Environmental Change Institute University of Oxford*, pp. 1–21, 2006, [Online]. Available: https://www.eci.ox.ac.uk/research/energy/downloads/smart-metering-report.pdf.
[4] S. Houde, A. Todd, A. Sudarshan, J. A. Flora, and K. C. Armel, "Real-time feedback and electricity consumption: A field experiment assessing the potential for savings and persistence," *Energy Journal*, vol. 34, no. 1, pp. 87–102, Jan. 2013, doi: 10.5547/01956574.34.1.4.
[5] K. Carrie Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? The case of electricity," *Energy Policy*, vol. 52, pp. 213–234, Jan. 2013, doi: 10.1016/j.enpol.2012.08.062.
[6] K. L. Tsai, F. Y. Leu, and I. You, "Residence energy control system based on wireless smart socket and IoT," *IEEE Access*, vol. 4, pp. 2885–2894, 2016, doi: 10.1109/ACCESS.2016.2574199.
[7] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992, doi: 10.1109/5.192069.
[8] M. Weiss, A. Helfenstein, F. Mattern, and T. Staake, "Leveraging smart meter data to recognize home appliances," in *2012 IEEE International Conference on Pervasive Computing and Communications, PerCom 2012*, Mar. 2012, pp. 190–197, doi: 10.1109/PerCom.2012.6199866.
[9] C. Laughman *et al.*, "Power signature analysis," *IEEE Power and Energy Magazine*, vol. 1, no. 2, pp. 56–63, Mar. 2003, doi:

10.1109/MPAE.2003.1192027.

[10] L. K. Norford and S. B. Leeb, "Non-intrusive electrical load monitoring in commercial buildings based on steady-state and transient load-detection algorithms," *Energy and Buildings*, vol. 24, no. 1, pp. 51–64, Jan. 1996, doi: 10.1016/0378-7788(95)00958-2.

[11] D. Srinivasan, W. S. Ng, and A. C. Liew, "Neural-network-based signature recognition for harmonic source identification," *IEEE Transactions on Power Delivery*, vol. 21, no. 1, pp. 398–405, Jan. 2006, doi: 10.1109/TPWRD.2005.852370.

[12] S. N. Patel, T. Robertson, J. A. Kientz, M. S. Reynolds, and G. D. Abowd, "At the flick of a switch: detecting and classifying unique electrical events on the residential power line," in *UbiComp 2007: Ubiquitous Computing*, Springer Berlin Heidelberg, 2007, pp. 271–288.

[13] H. Y. Lam, G. S. K. Fung, and W. K. Lee, "A novel method to construct taxonomy electrical appliances based on load signatures," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 2, pp. 653–660, 2007, doi: 10.1109/TCE.2007.381742.

[14] L. De Baets, J. Ruyssinck, C. Develder, T. Dhaene, and D. Deschrijver, "Appliance classification using VI trajectories and convolutional neural networks," *Energy and Buildings*, vol. 158, pp. 32–36, Jan. 2018, doi: 10.1016/j.enbuild.2017.09.087.

[15] K. Khalid, A. Mohamed, R. Mohamed, and H. Shareef, "Performance comparison of artificial intelligence techniques for non-intrusive electrical load monitoring," *Bulletin of Electrical Engineering and Informatics*, vol. 7, no. 2, pp. 143–152, Jun. 2018, doi: 10.11591/eei.v7i2.1190.

[16] J. Z. Kolter and M. J. Johnson, "REDD : a public data set for energy disaggregation research," in *SustKDD workshop*, 2011, no. 1, pp. 1–6.

[17] K. Chahine, "Towards automatic setup of non intrusive appliance load monitoring – Feature extraction and clustering," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 2, pp. 1002–1011, Apr. 2019, doi: 10.11591/ijece.v9i2.pp1002-1011.

[18] S. Semwal, R. S. Prasad, and P. Juneja, "Identifying appliances using NIALM with minimum features," *International Journal of Electrical and Computer Engineering*, vol. 4, no. 6, pp. 909–922, Dec. 2014, doi: 10.11591/ijece.v4i6.6715.

[19] N. Iksan, J. Sembiring, N. Hariyanto, and S. H. Supangkat, "Residential load event detection in NILM using robust cepstrum smoothing based method," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 2, pp. 742–752, Apr. 2019, doi: 10.11591/ijece.v9i2.pp742-752.

[20] G. Laput, Y. Zhang, and C. Harrison, "Synthetic sensors: towards general-purpose sensing," in *Conference on Human Factors in Computing Systems - Proceedings*, May 2017, vol. 2017-May, pp. 3986–3999, doi: 10.1145/3025453.3025773.

[21] J. Jalden, X. C. Moreno, and I. Skog, "Using the arduino due for teaching digital signal processing," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, Apr. 2018, vol. 2018-April, pp. 6468–6472, doi: 10.1109/ICASSP.2018.8461781.

[22] C. Hochgraf, "Using arduino to teach digital signal processing," 2013, [Online]. Available: https://pdfs.semanticscholar.org/290a/8c9aab485d3b8f4ebe8e08584bd300d7666c.pdf.

[23] "Atmel AVR465 : Single-Phase Power/Energy Meter with Tamper Detection," 2013. [Online]. Available: http://ww1.microchip.com/downloads/en/Appnotes/Atmel-2566-Single-Phase-Power-Energy-Meter-with-Tamper-Detection_Ap-Notes_AVR465.pdf.

[24] "Atmel AVR1631: Single Phase Energy Meter using XMEGA A," 2012. [Online]. Available: http://ww1.microchip.com/downloads/en/Appnotes/doc42039.pdf.

[25] T. H. Vu, *The fundamentals of Machine Learning*. 2018.

BIOGRAPHIES OF AUTHORS

Viet Hoang Duong received his bachelor's degree in 2016 and Master's degree in 2020 from Hanoi University of Science and Technology (HUST), Vietnam. His research interests include smart grids, machine learning, and instrumentation. He can be contacted at email: viet.dhca180160@sis.hust.edu.vn.

Nam Hoang Nguyen received his engineer degree in 2002 from Hanoi University of Technology (HUST), his Master's degree in 2004 from Hendri Poincaré University, France, and his Ph.D. in 2009 from Grenoble Polytechnic University, France. He is currently a Lecturer in the Department of Industrial Metrology and Informatics (3I), School of Electrical Engineering, Hanoi University of Technology (HUST). His main research directions are intelligent metering systems, IoT and embedded systems, IIoT, and renewable energy systems. He can be contacted at email: nam.nguyenhoang@hust.edu.vn.

Paper's title should be the fewest possible words that accurately describe the content of the paper (Center, Bold, 16pt)

Abdel-Rahman Hedar[1,2], Patricia Melin[3], Kennedy Okokpujie[4] (10 pt)
[1]Department of Computer Science, Faculty of Computers & Information, Assiut University, Assiut, Egypt (8 pt)
[2]Department of Computer Science in Jamoum, Umm Al-Qura University, Makkah, Saudi Arabia
[3]Division of Graduate Studies, Tijuana Institute of Technology, Tijuana, Mexico
[4]Department of Electrical and Information Engineering, College of Engineering, Covenant University, Ogun State, Nigeria

Article Info	ABSTRACT (10 PT)
	An abstract is often presented separate from the article, so it must be able to stand alone. A well-prepared abstract enables the reader to identify the basic content of a document quickly and accurately, to determine its relevance to their interests, and thus to decide whether to read the document in its entirety. The abstract should be informative and completely self-explanatory, provide a clear statement of the problem, the proposed approach or solution, and point out major findings and conclusions. **The Abstract should be 100 to 200 words in length.** References should be avoided, but if essential, then cite the author(s) and year(s). Standard nomenclature should be used, and non-standard or uncommon abbreviations should be avoided, but if essential they must be defined at their first mention in the abstract itself. No literature should be cited. The keyword list provides the opportunity to add 5 to 7 keywords, used by the indexing and abstracting services, in addition to those already present in the title (9 pt). *This is an open access article under the CC BY-SA license.*

Corresponding Author:

Kennedy Okokpujie
Department of Electrical and Information Engineering, College of Engineering, Covenant University
Km. 10 Idiroko Road, Canaan Land, Ota, Ogun State, Nigeria
Email: kennedy.okokpujie@covenantuniversity.edu.nga

1. INTRODUCTION (10 PT)

The main text format consists of a flat left-right columns on A4 paper (quarto). The margin text from the left and top are 2.5 cm, right and bottom are 2 cm. The manuscript is written in Microsoft Word, single space, Time New Roman 10 pt, and maximum 12 pages for original research article, or maximum 16 pages for review/survey paper, which can be downloaded at the website: http://ijai.iaescore.com.

A title of article should be the fewest possible words that accurately describe the content of the paper. The title should be succinct and informative and no more than about 12 words in length. Do not use acronyms or abbreviations in your title and do not mention the method you used, unless your paper reports on the development of a new method. Titles are often used in information-retrieval systems. Avoid writing long formulas with subscripts in the title. Omit all waste words such as "*A study of ...*", "*Investigations of ...*", "*Implementation of ...*", "*Observations on ...*", "*Effect of.....*", "*Analysis of ...*", "Design of…", etc.

A concise and factual abstract is required. The abstract should state briefly the purpose of the research, the principal results and major conclusions. An abstract is often presented separately from the article, so it must be able to stand alone. For this reason, References should be avoided, but if essential, then cite the author(s) and year(s). Also, non-standard or uncommon abbreviations should be avoided, but if essential they must be defined at their first mention in the abstract itself. Immediately after the abstract, provide a maximum of 7 keywords, using American spelling and avoiding general and plural terms and

multiple concepts (avoid, for example, 'and', 'of'). Be sparing with abbreviations: only abbreviations firmly established in the field may be eligible. These keywords will be used for indexing purposes.

Indexing and abstracting services depend on the accuracy of the title, extracting from it keywords useful in cross-referencing and computer searching. An improperly titled paper may never reach the audience for which it was intended, so be specific.

The Introduction section should provide: i) a clear background, ii) a clear statement of the problem, iii) the relevant literature on the subject, iv) the proposed approach or solution, and v) the new value of research which it is innovation (within 3-6 paragraphs). It should be understandable to colleagues from a broad range of scientific disciplines. Organization and citation of the bibliography are made in Institute of Electrical and Electronics Engineers (IEEE) style in sign [1], [2] and so on. The terms in foreign languages are written italic (*italic*). The text should be divided into sections, each with a separate heading and numbered consecutively [3]. The section or subsection headings should be typed on a separate line, e.g., 1. INTRODUCTION. A full article usually follows a standard structure: **1. Introduction, 2. The Comprehensive Theoretical Basis and/or the Proposed Method/Algorithm** *(optional)***, 3. Method, 4. Results and Discussion, and 5. Conclusion.** The structure is well-known as **IMRaD** style.

Literature review that has been done author used in the section "INTRODUCTION" to explain the difference of the manuscript with other papers, that it is innovative, it are used in the section "METHOD" to describe the step of research and used in the section "RESULTS AND DISCUSSION" to support the analysis of the results [2]. If the manuscript was written really have high originality, which proposed a new method or algorithm, the additional section after the "INTRODUCTION" section and before the "METHOD" section can be added to explain briefly the theory and/or the proposed method/algorithm [4].

2.   METHOD (10 PT)

Explaining research chronological, including research design, research procedure (in the form of algorithms, Pseudocode or other), how to test and data acquisition [5]–[7]. The description of the course of research should be supported references, so the explanation can be accepted scientifically [2], [4]. Figures 1-2 and Table 1 are presented center, as shown below and cited in the manuscript [5], [8]–[13]. The settlement curves produced at SG1 has been illustrated in Figure 2(a) and SG2 has been illustrated Figure 2(b).
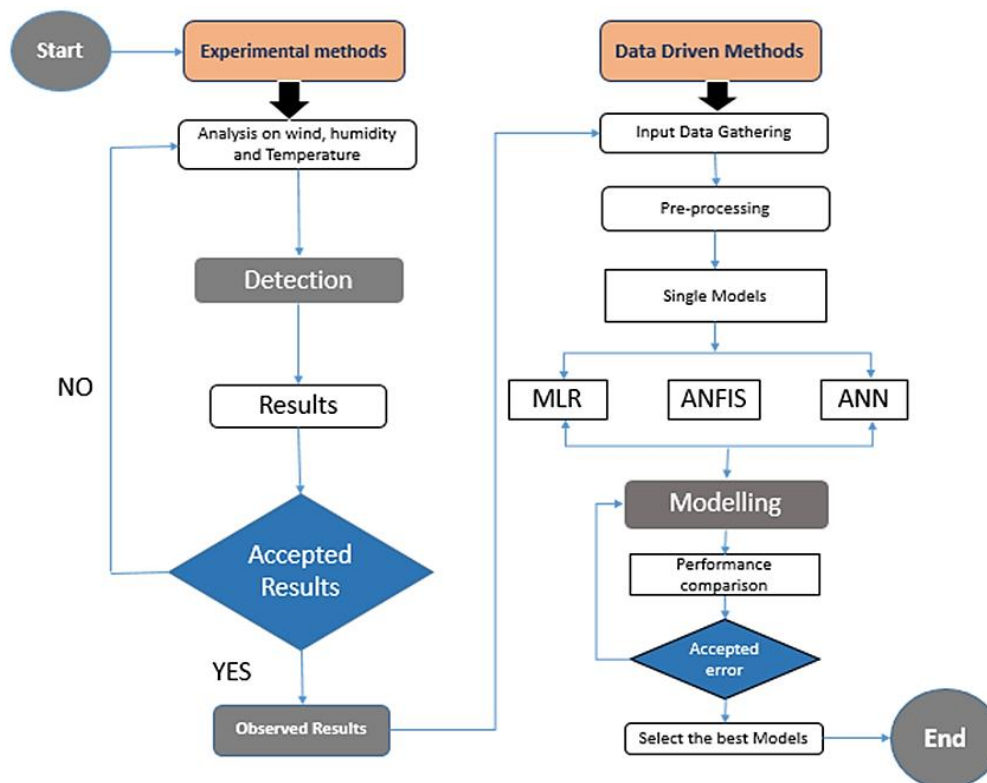

Figure 1. Shows the flowchart of the AI-based models and experimental methods applied
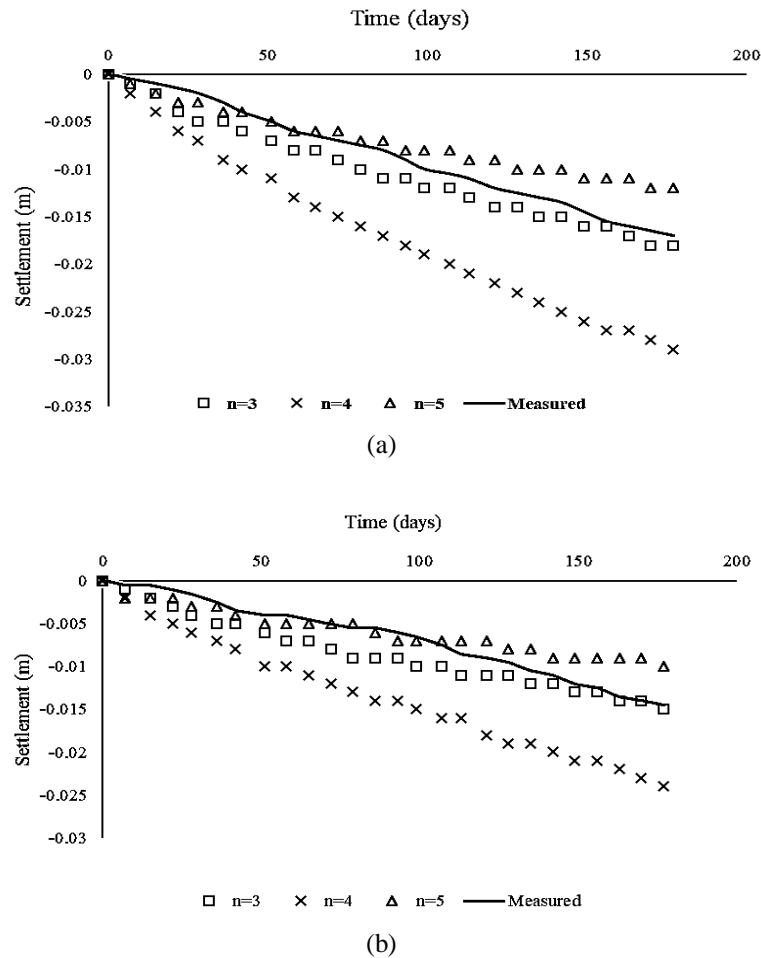
(a)

(b)

Figure 2. The relationship of soil settlement and time, (a) SG1 and (b) SG2

Table 1. The performance of ...

Variable	Speed (rpm)	Power (kW)
x	10	8.6
y	15	12.4
z	20	15.3

3.   RESULTS AND DISCUSSION (10 PT)

In this section, it is explained the results of research and at the same time is given the comprehensive discussion. Results can be presented in figures, graphs, tables and others that make the reader understand easily [14], [15]. The discussion can be made in several sub-sections.

3.1. Sub section 1

Equations should be placed at the center of the line and provided consecutively with equation numbers in parentheses flushed to the right margin, as in (1). The use of Microsoft Equation Editor or MathType is preferred.

$$E_v - E = \frac{h}{2.m} \left(k_x^2 + k_y^2 \right)$$ (1)

All symbols that have been used in the equations should be defined in the following text.

3.2. Sub section 2

Proper citation of other works should be made to avoid plagiarism. When referring to a reference item, please use the reference number as in [16] or [17] for multiple references. The use of "Ref [18]..."

should be employed for any reference citation at the beginning of sentence. For any reference with more than 3 or more authors, only the first author is to be written followed by *et al.* (e.g. in [19]). Examples of reference items of different categories shown in the References section. Each item in the references section should be typed using 9 pt font size [20]–[25].

3.2.1. Subsub section 1
yy

3.2.2. Subsub section 2
zz

4. CONCLUSION (10 PT)
Provide a statement that what is expected, as stated in the "INTRODUCTION" section can ultimately result in "RESULTS AND DISCUSSION" section, so there is compatibility. Moreover, it can also be added the prospect of the development of research results and application prospects of further studies into the next (based on result and discussion).

ACKNOWLEDGEMENTS (10 PT)

REFERENCES (10 PT)
The main references are international journals and proceedings. All references should be to the most pertinent, up-to-date sources **and the minimum of references** are **25 entries** (for original research paper) and **50 entries** (for review/survey paper). References are written in **IEEE style**. For more complete guide can be accessed at (http://ipmuonline.com/guide/refstyle.pdf). Use of a tool such as **EndNote**, **Mendeley**, or **Zotero** for reference management and formatting, and choose **IEEE style**. Please use a consistent format for references-see examples (8 pt):

[1] Journal/Periodicals
Basic Format:
J. K. Author, "Title of paper," *Abbrev. Title of Journal/Periodical*, vol. *x,* no. *x,* pp. *xxx-xxx,* Abbrev. Month, year, doi: *xxx.*
Examples:
- M. M. Chiampi and L. L. Zilberti, "Induction of electric field in human bodies moving near MRI: An efficient BEM computational procedure," *IEEE Trans. Biomed. Eng.*, vol. 58, pp. 2787–2793, Oct. 2011, doi: 10.1109/TBME.2011.2158315.
- R. Fardel, M. Nagel, F. Nuesch, T. Lippert, and A. Wokaun, "Fabrication of organic light emitting diode pixels by laser-assisted forward transfer," *Appl. Phys. Lett.*, vol. 91, no. 6, Aug. 2007, Art. no. 061103, doi: 10.1063/1.2759475.

[2] Conference Proceedings
Basic Format:
J. K. Author, "Title of paper," in *Abbreviated Name of Conf.*, (location of conference is optional), year, pp. *xxx–xxx*, doi: *xxx.*
Examples:
- G. Veruggio, "The EURON roboethics roadmap," in *Proc. Humanoids '06: 6th IEEE-RAS Int. Conf. Humanoid Robots*, 2006, pp. 612–617, doi: 10.1109/ICHR.2006.321337.
- J. Zhao, G. Sun, G. H. Loh, and Y. Xie, "Energy-efficient GPU design with reconfigurable in-package graphics memory," in *Proc. ACM/IEEE Int. Symp. Low Power Electron. Design (ISLPED)*, Jul. 2012, pp. 403–408, doi: 10.1145/2333660.2333752.

[3] Book
Basic Format:
J. K. Author, "Title of chapter in the book," in *Title of His Published Book*, X. Editor, Ed., *x*th ed. City of Publisher, State (only U.S.), Country: Abbrev. of Publisher, year, ch. *x*, sec. *x*, pp. *xxx–xxx.*
Examples:
- A. Taflove, *Computational Electrodynamics: The Finite-Difference Time-Domain Method* in Computational Electrodynamics II, vol. 3, 2nd ed. Norwood, MA, USA: Artech House, 1996.
- R. L. Myer, "Parametric oscillators and nonlinear materials," in *Nonlinear Optics*, vol. 4, P. G. Harper and B. S. Wherret, Eds., San Francisco, CA, USA: Academic, 1977, pp. 47–160.

[4] M. Theses (B.S., M.S.) and Dissertations (Ph.D.)
Basic Format:
J. K. Author, "Title of thesis," M.S. thesis, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.
J. K. Author, "Title of dissertation," Ph.D. dissertation, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.
Examples:
- J. O. Williams, "Narrow-band analyzer," Ph.D. dissertation, Dept. Elect. Eng., Harvard Univ., Cambridge, MA, USA, 1993.
- N. Kawasaki, "Parametric study of thermal and chemical nonequilibrium nozzle flow," M.S. thesis, Dept. Electron. Eng., Osaka Univ., Osaka, Japan, 1993.

*In the reference list, however, list all the authors for up to six authors. Use *et al.* only if: 1) The names are not given and 2) List of authors more than 6. *Example*: J. D. Bellamy *et al.*, Computer Telephony Integration, New York: Wiley, 2010.

See the examples:

REFERENCES

[1] T. S. Ustun, C. Ozansoy, and A. Zayegh, "Recent developments in microgrids and example cases around the world—A review," *Renew. Sustain. Energy Rev.*, vol. 15, no. 8, pp. 4030–4041, Oct. 2011, doi: 10.1016/j.rser.2011.07.033.

[2] D. Salomonsson, L. Soder, and A. Sannino, "Protection of Low-Voltage DC Microgrids," *IEEE Trans. Power Deliv.*, vol. 24, no. 3, pp. 1045–1053, Jul. 2009, doi: 10.1109/TPWRD.2009.2016622.

[3] S. Chakraborty and M. G. Simoes, "Experimental Evaluation of Active Filtering in a Single-Phase High-Frequency AC Microgrid," *IEEE Trans. Energy Convers.*, vol. 24, no. 3, pp. 673–682, Sep. 2009, doi: 10.1109/TEC.2009.2015998.

[4] S. A. Hosseini, H. A. Abyaneh, S. H. H. Sadeghi, F. Razavi, and A. Nasiri, "An overview of microgrid protection methods and the factors involved," *Renew. Sustain. Energy Rev.*, vol. 64, pp. 174–186, Oct. 2016, doi: 10.1016/j.rser.2016.05.089.

[5] S. Chen, N. Tai, C. Fan, J. Liu, and S. Hong, "Sequence-component-based current differential protection for transmission lines connected with IIGs," *IET Gener. Transm. Distrib.*, vol. 12, no. 12, pp. 3086–3096, Jul. 2018, doi: 10.1049/iet-gtd.2017.1507.

[6] S. Parhizi, H. Lotfi, A. Khodaei, and S. Bahramirad, "State of the Art in Research on Microgrids: A Review," *IEEE Access*, vol. 3, pp. 890–925, 2015, doi: 10.1109/ACCESS.2015.2443119.

[7] S. Chowdhury, S. P. Chowdhury, and P. Crossley, *Microgrids and Active Distribution Networks*. Institution of Engineering and Technology, 2009.

[8] R. Ndou, J. I. Fadiran, S. Chowdhury, and S. P. Chowdhury, "Performance comparison of voltage and frequency based loss of grid protection schemes for microgrids," in *2013 IEEE Power & Energy Society General Meeting*, 2013, pp. 1–5, doi: 10.1109/PESMG.2013.6672788.

[9] S. Liu, T. Bi, A. Xue, and Q. Yang, "Fault analysis of different kinds of distributed generators," in *2011 IEEE Power and Energy Society General Meeting*, Jul. 2011, pp. 1–6, doi: 10.1109/PES.2011.6039596.

[10] K. Jennett, F. Coffele, and C. Booth, "Comprehensive and quantitative analysis of protection problems associated with increasing penetration of inverter-interfaced DG," in *11th IET International Conference on Developments in Power Systems Protection (DPSP 2012)*, 2012, pp. P31–P31, doi: 10.1049/cp.2012.0091.

[11] P. T. Manditereza and R. Bansal, "Renewable distributed generation: The hidden challenges – A review from the protection perspective," *Renew. Sustain. Energy Rev.*, vol. 58, pp. 1457–1465, May 2016, doi: 10.1016/j.rser.2015.12.276.

[12] D. M. Bui, S.-L. Chen, K.-Y. Lien, Y.-R. Chang, Y.-D. Lee, and J.-L. Jiang, "Investigation on transient behaviours of a uni-grounded low-voltage AC microgrid and evaluation on its available fault protection methods: Review and proposals," *Renew. Sustain. Energy Rev.*, vol. 75, pp. 1417–1452, Aug. 2017, doi: 10.1016/j.rser.2016.11.134.

[13] T. N. Boutsika and S. A. Papathanassiou, "Short-circuit calculations in networks with distributed generation," *Electr. Power Syst. Res.*, vol. 78, no. 7, pp. 1181–1191, Jul. 2008, doi: 10.1016/j.epsr.2007.10.003.

[14] H. Margossian, G. Deconinck, and J. Sachau, "Distribution network protection considering grid code requirements for distributed generation," *IET Gener. Transm. Distrib.*, vol. 9, no. 12, pp. 1377–1381, Sep. 2015, doi: 10.1049/iet-gtd.2014.0987.

[15] O. Núñez-Mata, R. Palma-Behnke, F. Valencia, A. Urrutia-Molina, P. Mendoza-Araya, and G. Jiménez-Estévez, "Coupling an adaptive protection system with an energy management system for microgrids," *Electr. J.*, vol. 32, no. 10, p. 106675, Dec. 2019, doi: 10.1016/j.tej.2019.106675.

[16] M. Brucoli and T. C. Green, "Fault behaviour in islanded microgrids," in *Proceedings of the 19th international conference on electricity distribution, CIRED*, 2007, pp. 0548-(1-4).

[17] I. K. Tarsi, A. Sheikholeslami, T. Barforoushi, and S. M. B. Sadati, "Investigating impacts of distributed generation on distribution networks reliability: A mathematical model," in *Proceedings of the 2010 Electric Power Quality and Supply Reliability Conference*, Jun. 2010, pp. 117–124, doi: 10.1109/PQ.2010.5550010.

[18] L. K. Kumpulainen and K. T. Kauhaniemi, "Analysis of the impact of distributed generation on automatic reclosing," in *IEEE PES Power Systems Conference and Exposition, 2004.*, pp. 1152–1157, doi: 10.1109/PSCE.2004.1397623.

[19] A. A. Memon and K. Kauhaniemi, "A critical review of AC Microgrid protection issues and available solutions," *Electr. Power Syst. Res.*, vol. 129, pp. 23–31, Dec. 2015, doi: 10.1016/j.epsr.2015.07.006.

[20] H. A. Abdel-Ghany, A. M. Azmy, N. I. Elkalashy, and E. M. Rashad, "Optimizing DG penetration in distribution networks concerning protection schemes and technical impact," *Electr. Power Syst. Res.*, vol. 128, pp. 113–122, Nov. 2015, doi: 10.1016/j.epsr.2015.07.005.

[21] S. Chaitusaney and A. Yokoyama, "An Appropriate Distributed Generation Sizing Considering Recloser-Fuse Coordination," in *2005 IEEE/PES Transmission & Distribution Conference & Exposition: Asia and Pacific*, pp. 1–6, doi: 10.1109/TDC.2005.1546838.

[22] H. H. Zeineldin, Y. A.-R. I. Mohamed, V. Khadkikar, and V. R. Pandi, "A Protection Coordination Index for Evaluating Distributed Generation Impacts on Protection for Meshed Distribution Systems," *IEEE Trans. Smart Grid*, vol. 4, no. 3, pp. 1523–1532, Sep. 2013, doi: 10.1109/TSG.2013.2263745.

[23] D. Eltigani and S. Masri, "Challenges of integrating renewable energy sources to smart grids: A review," *Renew. Sustain. Energy Rev.*, vol. 52, pp. 770–780, Dec. 2015, doi: 10.1016/j.rser.2015.07.140.

[24] M. M. Eissa (SIEEE), "Protection techniques with renewable resources and smart grids—A survey," *Renew. Sustain. Energy Rev.*, vol. 52, pp. 1645–1667, Dec. 2015, doi: 10.1016/j.rser.2015.08.031.

[25] A. Oudalov *et al.*, "Novel Protection Systems for Microgrids," 2009. [Online]. Available: http://www.microgrids.eu/documents/688.pdf.

BIOGRAPHIES OF AUTHORS (10 PT)
The recommended number of authors is at least 2. One of them as a corresponding author.
Please attach clear photo (3x4 cm) and vita. Example of biographies of authors:

	Abdel-Rahman Hedar 🆔 📇 SC Ⓟ holds a Doctor of Informatics degree from Kyoto University, Japan in 2004. He also received his B.Sc. and M.Sc. (Mathematics) from Assiut University, Egypt in 1993 and 1997, respectively. He is currently an associate professor at Computer Science Department in Jamoum, Umm Al-Qura University, Makkah, Saudi Arabia. He is also an associate professor of artifcial intelligence in Assiut University since January 2012. His research includes meta-heuristics, global optimization, machine learning, data mining, bioinformatics, graph theory and parallel programming. He has published over 70 papers in international journals and conferences. From July 2005 to July 2007, he was a JSPS research fellow in Kyoto University, Japan. He can be contacted at email: ahahmed@uqu.edu.sa or hedar@aun.edu.eg.
	Patricia Melin 🆔 📇 SC Ⓟ received the D.Sc. degree (Doctor Habilitatus D.Sc.) in computer science from the Polish Academy of Sciences, Warsaw, Poland, with the Dissertation "Hybrid Intelligent Systems for Pattern Recognition using Soft Computing". She is a Professor of Computer Science in the Graduate Division, Tijuana Institute of Technology, Tijuana, Mexico since 1998. In addition, she is serving as Director of Graduate Studies in computer science and Head of the research group on Computational Intelligence (2000–present). Her research interests are in Type-2 Fuzzy Logic, Modular Neural Networks, Pattern Recognition, Neuro-Fuzzy and Genetic-Fuzzy hybrid approaches., She is currently the President of Hispanic American Fuzzy Systems Association (HAFSA) and is the founding Chair of the Mexican Chapter of the IEEE Computational Intelligence Society. She can be contacted at email: pmelin@tectijuana.mx.
	Dr. Kennedy Okokpujie 🆔 📇 SC Ⓟ holds a Bachelor of Engineering (B.Eng.) in Electrical and Electronics Engineering, Master of Science (M.Sc.) in Electrical and Electronics Engineering, Master of Engineering (M.Eng.) in Electronics and Telecommunication Engineering and Master of Business Administration (MBA), Ph.D in Information and Communication Engineering, besides several professional certificates and skills. He is currently lecturing with the department of Electrical and Information Engineering at Covenant University, Ota, Ogun State, Nigeria. He is a member of the Nigeria Society of Engineers and the Institute of Electrical and Electronics Engineers (IEEE). His research areas of interest include Biometrics, Artificial Intelligent, and Digital signal Processing. He can be contacted at email: kennedy.okokpujie@covenantuniversity.edu.ng.