MECOMARE, 11 (2) (2022) pp. 78-84

Published by: TRIGIN Publisher

International Journal of Mechanical Computational and Manufacturing Research

Journal homepage: www.trigin.pelnus.ac.id

Implementation of Backpropagation ANN in Predicting Long Bean Crop Production in Sumatra Island Province

Zodi Martua Siallagan[1], Solikhun[2]
[1]STIKOM Tunas Bangsa, Indonesia
[2]AMIK & STIKOM Tunas Bangsa Pematangsiantar, Indonesia

ARTICLEINFO

Article history:

Received June 08, 2022
Revised July 05, 2022
Accepted August 28, 2022

Keywords:

ANN
Backpropagation
Long beans
Prediction

ABSTRACT

The production of long bean vegetable crops in Indonesia is very high, this is because this plant is easy to cultivate. Predicting the production of long bean vegetable crops on the island of Sumatra, where the data source comes from BPS (Central Bureau of Statistics). In predicting the use of ANN (Artificial Neural Networks) and the method used in this study is the backpropagation algorithm, this method will be used to predict or predict the production of long bean vegetable crops on the island of Sumatra. The results have been obtained using 4 models, namely the 6-5-1, 6-10-1, 6-15-1, and 6-20-1 models. Among the 4 existing models, the 6-5-1 model has the more accurate accuracy or the lowest error value with an MSE of 0.00711838.

Corresponding Author:

Solikhun
AMIK & STIKOM Tunas Bangsa Pematangsiantar
Jl. Sudirman, No.1, 2 & 3, Banjar, Pematang Siantar City, North Sumatra, 21142, Indonesia
Email: solikhun@amiktunasbangsa.ac.id

1. INTRODUCTION

Long bean plant or synthetic vigna L is a vegetable commodity that has long been known and favored by many people (Pranindar et al., 2017). In long beans there is vitamin A, vitamin B, vitamin C, and minerals, especially in young pods (Pertiwi et al., 2021). Long bean seeds contain protein, fat, and carbohydrates so that long beans are a good source of vegetable protein for humans. Production of long bean vegetable crops in Indonesia is very large and has spread to various regions on the island of Sumatra. Long bean vegetable plants on the island of Sumatra continue to produce. The demand for long beans every year on the island of Sumatra sometimes goes up and sometimes down. BPS (Central Statistics Agency) said that in 2019-2020 the production of vegetable beans on the island of Sumatra experienced an increase. In the sales sector, this will greatly impact because the demand for goods is low while production is high, so it will cause losses. In this case, a prediction is needed to avoid losses. The method that will be used in this case is the fletcher-Reever backpropagation algorithm. This algorithm is one of the ANN (Artificial Neural Network) methods (Andrijasa & Mistianingsih, 2016).

2. RESEARCH METHOD

2.1 Artificial Neural Network

The research method used is an artificial neural network with machine learning methods (Eddy et al., 2018). Machine learning is a branch of artificial intelligence or artificial intelligence that allows systems to adapt human abilities to learn (Devianto & Dwiasnati, 2020). This algorithm is also trained to make predictions in data development through the use of statistics (Saifudin & Wahono, 2015). Algorithms or sequences of statistical processes are trained to find certain patterns and features in large amounts of data (Azhari et al., 2021). It aims to make better decisions (Izzawati & Lisnawati, 2015). The better the algorithm obtained, the better the accuracy of the decisions and predictions of the system (Firzatullah, 2021). Machine learning works based on the analysis of the data embedded in it (F. A. Nugraha et al., 2020). This input and output data processing training can help predict answers and find the correct intrinsic pattern in the input data (Manurung et al., 2022).

2.2 Research Source

The data sample used is a dataset sample of long bean vegetable production in the provinces on the island of Sumatra from 2007-2020 data sourced from BPS (Central Statistics Agency).

Table 1. Raw data

	Year						
Province	2007	2008	2009	2010	2011	2012	2013
ACEH	17032	13380	12868	18507	17021	18728	15253
NORTH SUMATRA	46815	41995	34628	41097	47612	50593	40653
WEST SUMATRA	7143	8689	9955	8775	9367	11319	12850
JAMBI	10451	7953	9974	11056	12830	11572	12447
RIAU	7715	5703	6361	7842	8894	7712	7829
SOUTH SUMATRA	11508	17121	19019	22303	12922	12544	12078
BENGKULU	13453	20067	20412	23086	15702	12108	10520
LAMPUNG	13220	16161	19096	21685	17870	17575	20578
KEP. BANGKA BELITUNG	4138	4327	5443	5962	5185	4051	5383
KEP. RIAU	4971	3851	4242	3476	3210	4658	4728

	Year						
Province	2014	2015	2016	2017	2018	2019	2020
ACEH	12577	13210	12130	11481	10942	10819	11220
NORTH SUMATRA	44306	45095	40428	43946	32107	29313	32189
WEST SUMATRA	11295	12943	12029	14108	19724	20822	19472
JAMBI	12789	8795	12531	11192	12083	9210	10045
RIAU	8561	7981	11129	7788	8697	7959	8208
SOUTH SUMATRA	9017	8968	9236	9932	12309	9755	8942
BENGKULU	10091	7512	6778	6034	5244	4919	5098
LAMPUNG	17978	16888	18089	15922	13658	12465	13500
KEEP. BANGKA BELITUNG	4436	2928	2304	2159	2136	1817	2166
KEEP. RIAU	7298	4704	4035	4461	4867	4550	3943

2.3 Research Stages

Figure 1 explains that the initial step that must be taken at the research stage is, the first to collect the data to be studied (based on table 1). The next stage is to separate the data into research data and test data. Data for 2007-2012 with a target of 2013 being the training data and data for 2014-2019 with a target of 2020 being the data to be tested. Next is to normalize the training data and test data using the equation formula (Prayudha et al., 2019)

$$x' = \frac{0,8(x-a)}{b-a} + 0,1 \quad (1)$$

X'= normalized data result
x = data to be normalized
a = highest value
b = lowest value

Training data with normalized test data is entered in MATLAP for processing, and is continued by building a multi-layer neural network (training data input) (Khairururizal, 2021). Next is to apply the Fletcher-Reeves algorithm. Creating a multi-layer neural network using the tangig function and the logsig function (Ginantra et al., 2022). The next step is to initialize the network parameters based on the training function (traincgf). To search for performance results, must Enter a training process command and view the results when performance is found (D. Nugraha & Rosa, n.d.). If the results of the training reach convergence, it will be continued by entering the normalized test data (Junaidi et al., 2021). But if the results of the training data still have not reached convergence, then return to the initialization stage of network parameters. The next stage is carried out by simulating test data based on the results of the training. When everything has been done, the final stage is to conduct an evaluation to see the best architectural model based on the lowest performance/MSE test. (Sinaga et al., 2019) (D. Nugraha & Rosa, n.d.).

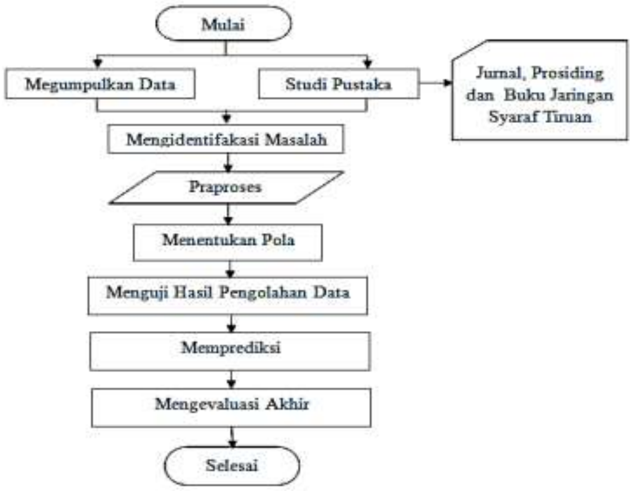

Figure 1. Research Stages

3. RESULTS AND DISCUSSIONS

3.1 Normalized Data Results

Table 2 below is the result of the normalization of training data in 2007-2012 with a target in 2013, which is sourced from table 1. The data is normalized using the siqmoid function.

Table 2. Training Data Normalization Results

Province	Year						
	2007	2008	2009	2010	2011	2012	2013(Target)
ACEH	0.3334	0.2717	0.2631	0.3583	0.3332	0.3620	0.3033
NORTH SUMATRA	0.8362	0.7548	0.6305	0.7397	0.8497	0.9000	0.7322
WEST SUMATRA	0.1664	0.1925	0.2139	0.140	0.2040	0.2369	0.2628
JAMBI	0.2223	0.1801	0.2142	0.2325	0.2624	0.2412	0.2560
RIAU	0.1761	0.1963	0.2074	0.2324	0.2502	0.2302	0.2322
SOUTH SUMATRA	0.2401	0.3349	0.3669	0.4224	0.2640	0.2576	0.2497
BENGKULU	0.2729	0.3846	0.3904	0.4356	0.3109	0.2502	0.2234
LAMPUNG	0.2690	0.3187	0.3682	0.4119	0.3475	0.3425	0.3932
KEEP. BANGKA BELITUNG	0.1157	0.1189	0.1377	0.1465	0.1333	0.1142	0.1367
KEEP. RIAU	0.1297	0.1108	0.1174	0.1045	0.1000	0.1244	0.1256

Table 3 below is the Result of Normalization of Testing Data for 2014-2019 with a target of 2020, which is sourced from table 1. The data is normalized using the siqmoid function.

Table 3. Testing Data Normalization Results

Province	Year						
	2014	2015	2016	2017	2018	2019	2020,(Target)
ACEH	0.2899	0.3106	0.2906	0.2786	0.2687	0.2664	0.2738
NORTH SUMATRA	0.8854	0.9000	0.8137	0.8788	0.6599	0.6083	0.6614
WEST SUMATRA	0.2752	0.3057	0.2888	0.3272	0.4310	0.4513	0.4264
JAMBI	0.3028	0.2290	0.2980	0.2733	0.2898	0.2367	0.2521
RIAU	0.2247	0.2139	0.2721	0.2104	0.2272	0.2135	0.2181
SOUTH SUMATRA	0.2331	0.2322	0.2371	0.2500	0.2939	0.2467	0.2317
BENGKULU	0.2529	0.2053	0.1917	0.1780	0.1633	0.1573	0.1606
LAMPUNG	0.3987	0.3786	0.4008	0.3607	0.3189	0.2968	0.3160
KEEP. BANGKA BELITUNG	0.1484	0.1205	0.1090	0.1063	0.1059	0.1000	0.1065
KEEP. RIAU	0,2013	0.1534	0.1410	0.1489	0.1564	0.1505	0.1393

3.2 Training and Testing

Data processing is carried out using tools in Matlap which aims to determine the best architectural model. The method used in the architectural model is Fletcher-Reeves. The architecture used is 4 models, namely 6-5-1, 6-10-1, 6-15-1 and 6-20-1. For the first data structure is called input, the second data is called hidden and the third data is called output. Parameters used in the Fletcher-Reevester algorithm.

3.2.1 Training and Testing 6-5-1

The results of the 6-5-1 architectural model get 170 iterations of epochs. The results of the training are in table 4 and the test results are in table 5.

Table 4. Training Data Results

Training Data										
No	X1	X2	X3	X4	X5	X6	Target (Y1)	Epoch 170		
								actual	Error	Perf / MSE
1	0.3334	0.2717	0.2631	0.3583	0.3332	0.3620	0.3033	0.3068	-0.0035	0.000016903
2	0.8362	0.7548	0.6305	0.7397	0.8497	0.9000	0.7322	0.7322	0.0000	
3	0.1664	0.1925	0.2139	0.140	0.2040	0.2369	0.2628	0.2646	-0.0018	
4	0.2223	0.1801	0.2142	0.2325	0.2624	0.2412	0.2560	0.2474	0.0086	
5	0.1761	0.1963	0.2074	0.2324	0.2502	0.2302	0.2322	0.2388	-0.0066	
6	0.2401	0.3349	0.3669	0.4224	0.2640	0.2576	0.2497	0.2464	0.0033	
7	0.2729	0.3846	0.3904	0.4356	0.3109	0.2502	0.2234	0.2270	-0.0036	
8	0.2690	0.3187	0.3682	0.4119	0.3475	0.3425	0.3932	0.3914	0.0018	
9	0.1157	0.1189	0.1377	0.1465	0.1333	0.1142	0.1367	0.1340	0.0027	
10	0.1297	0.1108	0.1174	0.1045	0.1000	0.1244	0.1256	0.1271	-0.0015	

Table 5. Test Data Results

Test Data										
							Target	Epoch 170		
No	X7	X8	X9	X10	X11	X12	(Y1)	actual	Error	Perf / MSE
1	0.2899	0.3106	0.2906	0.2786	0.2687	0.2664	0.2738	0.1918	0.0820	0.007118378
2	0.8854	0.9000	0.8137	0.8788	0.6599	0.6083	0.6614	0.6598	0.0016	
3	0.2752	0.3057	0.2888	0.3272	0.4310	0.4513	0.4264	0.6391	-0.2127	
4	0.3028	0.2290	0.2980	0.2733	0.2898	0.2367	0.2521	0.2094	0.0427	
5	0.2247	0.2139	0.2721	0.2104	0.2272	0.2135	0.2181	0.2265	-0.0084	
6	0.2331	0.2322	0.2371	0.2500	0.2939	0.2467	0.2317	0.2203	0.0114	
7	0.2529	0.2053	0.1917	0.1780	0.1633	0.1573	0.1606	0.1322	0.0284	
8	0.3987	0.3786	0.4008	0.3607	0.3189	0.2968	0.3160	0.1901	0.1259	
9	0.1484	0.1205	0.1090	0.1063	0.1059	0.1000	0.1065	0.0898	0.0167	
10	0,2013	0.1534	0.1410	0.1489	0.1564	0.1505	0.1393	0.1237	0.0156	

3.2.2 Training and Testing 6-10-1

The results of the 6-10-1 architectural model get 166 iterations of epochs. The results of the training are in table 6 and the test results are in table 7.

Table 6. Training Data Results

Training Data										
No	X1	X2	X3	X4	X5	X6	Target (Y1)	Epoch 166		
								actual	Error	Perf / MSE
1	0.3334	0.2717	0.2631	0.3583	0.3332	0.3620	0.3033	0.3031	0.0002	0.000000017
2	0.8362	0.7548	0.6305	0.7397	0.8497	0.9000	0.7322	0.7322	0.0000	
3	0.1664	0.1925	0.2139	0.140	0.2040	0.2369	0.2628	0.2628	0.0000	
4	0.2223	0.1801	0.2142	0.2325	0.2624	0.2412	0.2560	0.2562	-0.0002	
5	0.1761	0.1963	0.2074	0.2324	0.2502	0.2302	0.2322	0.2322	0.0000	
6	0.2401	0.3349	0.3669	0.4224	0.2640	0.2576	0.2497	0.2499	-0.0002	
7	0.2729	0.3846	0.3904	0.4356	0.3109	0.2502	0.2234	0.2233	0.0001	
8	0.2690	0.3187	0.3682	0.4119	0.3475	0.3425	0.3932	0.3933	-0.0001	
9	0.1157	0.1189	0.1377	0.1465	0.1333	0.1142	0.1367	0.1366	0.0001	
10	0.1297	0.1108	0.1174	0.1045	0.1000	0.1244	0.1256	0.1256	0.0000	

Table 7. Test Data Results

No	X7	X8	X9	X10	X11	X12	Target (Y1)	Epoch 166 actual	Error	Perf / MSE
1	0.2899	0.3106	0.2906	0.2786	0.2687	0.2664	0.2738	0.3772	-0.1034	0.015991919
2	0.8854	0.9000	0.8137	0.8788	0.6599	0.6083	0.6614	0.5830	0.0784	
3	0.2752	0.3057	0.2888	0.3272	0.4310	0.4513	0.4264	0.3579	0.0685	
4	0.3028	0.2290	0.2980	0.2733	0.2898	0.2367	0.2521	0.4743	-0.2222	
5	0.2247	0.2139	0.2721	0.2104	0.2272	0.2135	0.2181	0.4222	-0.2041	
6	0.2331	0.2322	0.2371	0.2500	0.2939	0.2467	0.2317	0.2991	-0.0674	
7	0.2529	0.2053	0.1917	0.1780	0.1633	0.1573	0.1606	0.2104	-0.0498	
8	0.3987	0.3786	0.4008	0.3607	0.3189	0.2968	0.3160	0.5167	-0.2007	
9	0.1484	0.1205	0.1090	0.1063	0.1059	0.1000	0.1065	0.1144	-0.0079	
10	0,2013	0.1534	0.1410	0.1489	0.1564	0.1505	0.1393	0.1380	0.0013	

3.2.3 Training and Testing 6-15-1

The results of the 6-15-1 architectural model get the epoch results of 118 iterations. The results of the training can be seen in table 8 and the test in table 9.

Table 8. Training Data Results

No	X1	X2	X3	X4	X5	X6	Target (Y1)	Epoch 118 actual	Error	Perf / MSE
1	0.3334	0.2717	0.2631	0.3583	0.3332	0.3620	0.3033	0.3033	0.0000	0.000000006
2	0.8362	0.7548	0.6305	0.7397	0.8497	0.9000	0.7322	0.7322	0.0000	
3	0.1664	0.1925	0.2139	0.140	0.2040	0.2369	0.2628	0.2628	0.0000	
4	0.2223	0.1801	0.2142	0.2325	0.2624	0.2412	0.2560	0.2560	0.0000	
5	0.1761	0.1963	0.2074	0.2324	0.2502	0.2302	0.2322	0.2321	0.0001	
6	0.2401	0.3349	0.3669	0.4224	0.2640	0.2576	0.2497	0.2497	0.0000	
7	0.2729	0.3846	0.3904	0.4356	0.3109	0.2502	0.2234	0.2234	0.0000	
8	0.2690	0.3187	0.3682	0.4119	0.3475	0.3425	0.3932	0.3931	0.0001	
9	0.1157	0.1189	0.1377	0.1465	0.1333	0.1142	0.1367	0.1368	-0.0001	
10	0.1297	0.1108	0.1174	0.1045	0.1000	0.1244	0.1256	0.1255	0.0001	

Table 9. Test Data Results

No	X7	X8	X9	X10	X11	X12	Target (Y1)	Epoch 118 actual	Error	Perf / MSE
1	0.2899	0.3106	0.2906	0.2786	0.2687	0.2664	0.2738	0.3538	-0.0800	0.012449055
2	0.8854	0.9000	0.8137	0.8788	0.6599	0.6083	0.6614	0.5267	0.1347	
3	0.2752	0.3057	0.2888	0.3272	0.4310	0.4513	0.4264	0.6736	-0.2472	
4	0.3028	0.2290	0.2980	0.2733	0.2898	0.2367	0.2521	0.3657	-0.1136	
5	0.2247	0.2139	0.2721	0.2104	0.2272	0.2135	0.2181	0.3608	-0.1427	
6	0.2331	0.2322	0.2371	0.2500	0.2939	0.2467	0.2317	0.2967	-0.0650	
7	0.2529	0.2053	0.1917	0.1780	0.1633	0.1573	0.1606	0.1700	-0.0094	
8	0.3987	0.3786	0.4008	0.3607	0.3189	0.2968	0.3160	0.2818	0.0342	
9	0.1484	0.1205	0.1090	0.1063	0.1059	0.1000	0.1065	0.1123	-0.0058	
10	0,2013	0.1534	0.1410	0.1489	0.1564	0.1505	0.1393	0.1330	0.0063	

3.3 Training and Testing 6-20-1

The results of the 6-5-1 architectural model get an epoch of 162 iterations. The results of the training can be seen in table 10 and the test in table 11.

Table 10. Training Data Results

No	X1	X2	X3	X4	X5	X6	Target (Y1)	Epoch 162 actual	Error	Perf / MSE
1	0.3334	0.2717	0.2631	0.3583	0.3332	0.3620	0.3033	0.3034	-0.0001	0.000000038

2	0.8362	0.7548	0.6305	0.7397	0.8497	0.9000	0.7322	0.7322	0.0000
3	0.1664	0.1925	0.2139	0.140	0.2040	0.2369	0.2628	0.2626	0.0002
4	0.2223	0.1801	0.2142	0.2325	0.2624	0.2412	0.2560	0.2560	0.0000
5	0.1761	0.1963	0.2074	0.2324	0.2502	0.2302	0.2322	0.2323	-0.0001
6	0.2401	0.3349	0.3669	0.4224	0.2640	0.2576	0.2497	0.2501	-0.0004
7	0.2729	0.3846	0.3904	0.4356	0.3109	0.2502	0.2234	0.2231	0.0003
8	0.2690	0.3187	0.3682	0.4119	0.3475	0.3425	0.3932	0.3930	0.0002
9	0.1157	0.1189	0.1377	0.1465	0.1333	0.1142	0.1367	0.1365	0.0002
10	0.1297	0.1108	0.1174	0.1045	0.1000	0.1244	0.1256	0.1256	0.0000

Table 11. Test Data Results

Test Data										
No	X7	X8	X9	X10	X11	X12	Target (Y1)	Epoch 162		
								actual	Error	Perf / MSE
1	0.2899	0.3106	0.2906	0.2786	0.2687	0.2664	0.2738	0.3972	-0.1234	
2	0.8854	0.9000	0.8137	0.8788	0.6599	0.6083	0.6614	0.6057	0.0557	
3	0.2752	0.3057	0.2888	0.3272	0.4310	0.4513	0.4264	0.4672	-0.0408	
4	0.3028	0.2290	0.2980	0.2733	0.2898	0.2367	0.2521	0.3756	-0.1235	
5	0.2247	0.2139	0.2721	0.2104	0.2272	0.2135	0.2181	0.3900	-0.1719	0.008245659
6	0.2331	0.2322	0.2371	0.2500	0.2939	0.2467	0.2317	0.3001	-0.0684	
7	0.2529	0.2053	0.1917	0.1780	0.1633	0.1573	0.1606	0.2359	-0.0753	
8	0.3987	0.3786	0.4008	0.3607	0.3189	0.2968	0.3160	0.3970	-0.0810	
9	0.1484	0.1205	0.1090	0.1063	0.1059	0.1000	0.1065	0.1220	-0.0155	
10	0,2013	0.1534	0.1410	0.1489	0.1564	0.1505	0.1393	0.1621	-0.0228	

3.4 Determination of the Best Architectural Model

After conducting training and testing data on the 6-5-1, 6-10-1, 6-15-1 and 6-20-1 models using the Matlab application and Microsoft excel, the best model architecture was obtained, namely 6-5- 1. Model 6-5-1 with epoch 170 is the best model among other models because it has high accuracy and has a low MSE/performance value of 0.00711838. We can see the results from table 12 and the graph.

Table 12. Comparison of all Architectural Models

Algorithm	Architecture	Training Function	Epoch (Iteration)	MSE Training	MSE Testing/Performance
fletcher-Reeves	6-5-1	Traincgf	170	0.00001690	0.00711838
	6-10-1	Traincgf	166	0.000000002	0.01599192
	6-15-1	Traincgf	118	0.00000001	0.01244905
	6-20-1	Traincgf	162	0.000000004	0.00824566

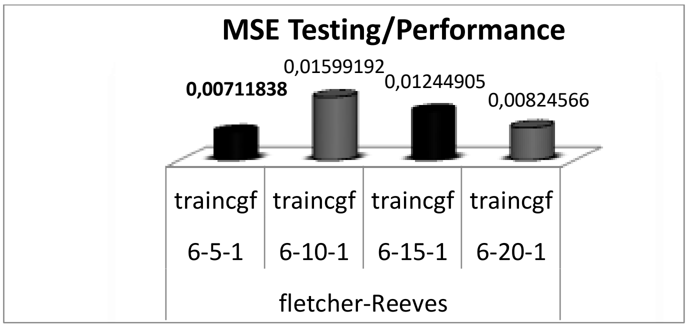

Figure 2. Best Architectural Model

4. CONCLUSION

The Artificial Neural Network method makes it easier to do research, where the machine learning method can help to find performance values and also determine the best value from the sample data to be studied. The application that participates in this research is the matlab application, because matlab itself has a feature to calculate performance and to find the best value with the help of the Fletcher-Reeves algorithm. Conducted testing with 4 samples including 6-5-1, 6-10-1, 6-15-1, and 6-20-1. Of the five samples, there are samples that get the best results from other samples, namely sample 6-5-1 with an MSE/Performance value of 0.00711838. This research has obtained the highest accuracy or has the lowest performance value by using the help of matlab tools.

ACKNOWLEDGEMENTS

We would like to thank all those who have contributed to this research, so that the research can be carried out properly.

REFERENCES

Andrijasa, M. F., & Mistianingsih, M. (2016). Penerapan Jaringan Syaraf Tiruan Untuk Memprediksi Jumlah Pengangguran di Provinsi Kalimantan Timur Dengan Menggunakan Algoritma Pembelajaran Backpropagation. *Informatika Mulawarman: Jurnal Ilmiah Ilmu Komputer*, *5*(1), 50–54.

Azhari, M., Situmorang, Z., & Rosnelly, R. (2021). Perbandingan Akurasi, Recall, dan Presisi Klasifikasi pada Algoritma C4. 5, Random Forest, SVM dan Naive Bayes. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, *5*(2), 640–651.

Devianto, Y., & Dwiasnati, S. (2020). Kerangka kerja sistem kecerdasan buatan dalam meningkatkan kompetensi sumber daya manusia Indonesia. *J. Telekomun. Dan Komput*, *10*(1), 19.

Eddy, T., Irfandy, T., & Muhammad, N. Y. U. (2018). Machine Learning Penentuan Penerima Beasiswa Peningkatan Prestasi Akademik (PPA) Menggunakan Metode Jaringan Saraf Tiruan (JST). *Seminar Nasional Teknik Elektro Dan Informatika 2018*, 391–396.

Firzatullah, R. M. (2021). Menggunakan Sistem Pendukung Keputusan Penentuan Uang Kuliah Tunggal Universitas XYZ Menggunakan Algoritma Backpropagation. *PETIR*, *14*(2), 170–180.

Ginantra, N. L. W. S. R., GS, A. D., Andini, S., & Wanto, A. (2022). Pemanfaatan Algoritma Fletcher-Reeves untuk Penentuan Model Prediksi Harga Nilai Ekspor Menurut Golongan SITC. *Building of Informatics, Technology and Science (BITS)*, *3*(4), 679–685.

Izzawati, N., & Lisnawati, L. (2015). Efektivitas pelatihan perencanaan karir terhadap peningkatan efikasi diri dalam pengambilan keputusan karir. *Jurnal Psikologi Tabularasa*, *10*(1).

Junaidi, G. R., Septian, Y., Wanto, A., Windarto, A. P., & Hartama, D. (2021). Analisis Performance Algoritma Powell-Beale untuk Prediksi Impor Alumunium. *Seminar Nasional Ilmu Sosial Dan Teknologi-SANISTEK*, 290–298.

Khairururizal, F. (2021). PREDIKSI PENYAKIT THT (TELINGA, HIDUNG, TENNGOROKAN) DENGAN METODE JARINGAN SYARAF TIRUAN PERCEPTRON. *Prosiding Seminar Nasional Fortei7 (SinarFe7)*, *4*(1), 610–614.

Manurung, S. F., Andriansya, A., Permana, J., & Pangestu, R. (2022). Pemanfaatan Algoritma JST untuk Menentukan Model Prediksi Umur Harapan Hidup Saat Lahir. *Hello World Jurnal Ilmu Komputer*, *1*(1), 19–35.

Nugraha, D., & Rosa, E. M. (n.d.). *Pengaruh Pembelajaran Berbasis Caring dalam Meningkatan Performance Mahasiswa Keperawatan di Skill Laboratory Universitas Muhammadiyah Yogyakarta*.

Nugraha, F. A., Harani, N. H., & Habibi, R. (2020). *Analisis Sentimen Terhadap Pembatasan Sosial Menggunakan Deep Learning*. Kreatif.

Pertiwi, S. K., Rizal, K., & Triyanto, Y. (2021). Pengaruh pupuk organik cair urin kambing dan pestisida alami terhadap pertumbuhan tanaman kacang panjang beda varietas di desa Gunung Selamat. *Indonesian Journal of Community Services*, *3*(1), 19–30.

Pranindar, A. B., Rusmarini, U. K., & Astuti, Y. T. M. (2017). PENGARUH DOSIS KOMPOS ISI RUMEN SAPI DAN PUPUK NPK TERHADAP PERTUMBUHAN DAN HASIL TANAMAN KACANG PANJANG (Vigna sinensis). *JURNAL AGROMAST*, *2*(2).

Saifudin, A., & Wahono, R. S. (2015). Penerapan teknik ensemble untuk menangani ketidakseimbangan kelas pada prediksi cacat software. *IlmuKomputer. Com Journal of Software Engineering*, *1*(1), 28–37.