SGD OPTIMIZER TO REDUCE COST VALUE IN DEEP LEARNING FOR CUSTOMER CHURN PREDICTION

MUHAMMAD DIMAS ADITYA AKBAR[1], ANDRY CHOWANDA[2]

[1]Bina Nusantara University, BINUS Graduate Program-Master of Computer Sciences, Department of Computer Sciences, Jakarta, Indonesia

[2]Bina Nusantara University. Computer Science Department, School of Computer Science, Jakarta, Indonesia

E-mail: [1]muhammad.akbar028@binus.ac.id, [2]achowanda@binus.edu

ABSTRACT

The number of customers is an important indicator for companies to know the success of a product and service offered. In general, customers are grouped into two categories, loyal customers and disloyal customers. This disloyal customer refers to customers who have stopped using a product and service from a company or are often referred to as churn. For that, we need a system that can predict whether the customer has the potential to experience churn or not. The system is required to predict well with one of the indicators, namely a low-cost value. One way to reduce the cost value is to use the optimizer function. Researchers use deep learning algorithms to create predictive models. A total of 3333 rows and 20 columns of telecommunication customer public data are used as datasets. This study also compares several optimizer algorithms to find the lowest cost value. In addition to the cost value, the accuracy and f1 score results are also used as other considerations. Researchers also use cross-validation for the training process and validating the model created. To deal with the imbalance class dataset, the researcher uses the unweighted method on the cost function. The evaluation results show that SGD has the lowest cost value, which is 0.261. Meanwhile, the AdaGrad matrix classification shows the best value for accuracy and f1 score with 0.926 and 0.846, respectively

Keywords: *Customer Churn Prediciton, SGD Optimizer, AdaGrad Optimizer, Deep Learning, Telecommunications*

1 INTRODUCTION

In the era of information disclosure today, it is undeniable that we have the convenience of accessing information instantly or directly, either through social media or online news channels. This makes it easier for us to get various information related to the product or service that we need. We can also easily compare the advantages and disadvantages of the products or services offered by various companies, especially in the telecommunications business sector. This sector has a significant change in the number of customers, and this term is commonly referred to as "churn". In the world of telecommunications, the term "churn" can be interpreted as the loss of customers due to switching from one service provider to another during a certain period. A recent study concluded that the average customer churn in the telecommunications sector was 2.2% per month [1]. One of the factors that underlie the occurrence of "churn" is the value of customer satisfaction. [2] According to Kotler in his book, satisfaction is the level of feeling where a person can compare the performance of the product (service) received with what is expected. At the same time, the value of satisfaction is the benefits customers receive from the performance of a company in meeting their expectations. Customers will feel satisfied if the benefits they expect are met and happy if they get benefits that exceed expectations. Satisfied customers will tend to be loyal customers, can buy more of the products or services offered, are less sensitive to price changes, and give good comments about the company's performance. Because of the importance of maintaining the value of customer satisfaction, the company must have the right strategy in retaining its customers. One way to support this strategy is to predict whether customers will churn or not. To be able to predict well, we need a method that can understand customer patterns and habits. Approach with data mining and machine learning is one answer.

Several algorithms such as Decision Tree, Naïve Bayes, and Deep learning are machine learning algorithms that are commonly used in customer churn prediction. The decision tree algorithm has high accuracy, but the accumulated number of errors from each node level is quite large. In addition, decision trees will be difficult to implement for large and unbalanced data. Naive Bayes has the advantage that if there is a missing value, it can be ignored in the calculation. This algorithm can also be used for quantitative and qualitative data. However, nave Bayes has a drawback: its accuracy cannot be measured with only one probability. Other evidence is needed to prove it. Deep learning has many advantages, including being universal, resistant to data variations, and having a high level of generalization and scalability. In addition, deep learning algorithms have become a widely used algorithm at least in the last five years. It is proven by the number of papers discussing algorithms on the websites of the most popular paper publishers such as ScienceDirect, which amounted to 140,578 papers compared to the Decision Tree algorithm as many as 97,254 papers and Naïve Bayes as many as 15,266 papers. For this reason, researchers will use a deep learning algorithm in this study. Researchers will compare several optimizer algorithms such as SGD, Adam, AdaGrad, AdaDelta, AdamW, AdaMax and RMSprop to find the best optimizer algorithm.

2 STYLE OF PAPER

Prediction of customer churn is a classic problem that often occurs, especially in the telecommunications world. To solve these problems, there is a lot of literature that uses various techniques, including machine learning, data mining, and techniques that combine machine learning and data mining or what are called hybrid techniques.

The approach with supervised learning is one of the techniques that can be used to solve the above problems. Supervised learning is an algorithm that prioritizes input to output as desired, usually termed class. The quality of learning outcomes is highly expected in the given class. The more appropriate the input and output given, the more accurate the results obtained. Thus, the user is very instrumental in validating the input and output. This type of algorithm is usually used to solve classification and regression problems [3]. One of the supervised learning algorithms is the Artificial Neural Network (ANN).

Artificial Neural Networks have been widely used to solve problems, especially for prediction problems such as air compressor load forecasting [4], student performance prediction [5], monthly rainfall prediction [6], heat demand prediction [7], performance prediction for task migration [8] and many others.

Deep learning is one of the algorithms developed based on the ANN. The basic difference is in the number of layers used. If more than one layer is used, then the model can already be called deep learning. Deep learning was introduced in 2000 and has been widely implemented in various fields, such as health [9], marketing management [10][27], financial [11] and sports [12]. Likewise in the field of computer science such as computer vision [13][14][15], speech recognition [16][17] dan natural language processing [18][19][20].

The advantages of deep learning, among others, can be implemented in almost all fields; this makes deep learning have universal capabilities. In addition, deep learning can process various variations of data naturally according to the facts. Furthermore, deep learning does not require artificially designed features. Optimization features can be updated automatically. The next advantage is that deep learning has transfer learning capabilities. This is useful when there is insufficient data available for a problem to be solved [3].

The optimizer function is used to minimize the occurrence of cost values. Some optimizers like Adam [21], AdaDelta [22], RMS-Prop [23], AdaGrad [24], and SGD [25] has been successfully used in machine learning models, as well as proving that the optimizer function is one of the factors that can determine the success of the built model. For this reason, researchers will compare the results of several of these optimizers to determine the optimal optimizer for use in the built machine learning model.

3 TABLES AND FIGURES

3.1 DATA COLLECTION

This study uses a public dataset sourced from Kaggle on churn in telecommunications [26]. The dataset contains 3,333 rows and 21 columns. The dataset is a class imbalance with a comparison true and false ratio in 15:85.

Table 1: Dataset Information

No	Attribute Name	Description
1	State	Location of phone number's state origin

2	Account length	Length of customer account
3	International plan	Whether or not a customer has an international plan.
4	Voicemail plan	Whether or not a customer has a voice mail plan
5	Area code	Identifier for geographic region of phone number origin
6	Phone number	Customer phone number
7	Number vmail messages	Number of voice mail sent by customers
8	Total day minutes	Total minutes spent on calls during the day.
9	Total day calls	Total calls made during the day.
10	Total day charge	Total charge for calls during the day
11	Total eve minutes	Total minutes spent on calls during the evening
12	Total eve calls	Total calls made during the evening.
13	Total eve charge	Total charge for calls during the evening
14	Total night minutes	Total minutes spent on calls during the night.
15	Total night calls	Total calls made during the night
16	Total night charge	Total charge for calls during the night
17	Total intl minutes	Total minutes spent on international calls
18	Total intl calls	Total international calls made
19	Total intl charge	Total charge for international calls.
20	Customer service calls	Number of calls made to customer service.
21	Churn	Customer churn class

3.2 DATA PREPROCESSING

Data preprocessing is an exercise in techniques used on data sets to remove noise, missing values, and inconsistent data. The preprocessing data process is divided into several steps, namely data cleaning, data transformation, and data reduction.

Data cleaning is a process where we will clean up irrelevant data and lost data. Data reduction is the process of reducing the amount of data to be analyzed easily or because the data is not relevant to the output class. While the data transformation changes the shape of the data to suit the needs of researchers, in this case is following the machine learning model used.

3.3 DATA REDUCTION

Data reduction is the process of reducing the size or amount of data from the actual data set. In this case, the researcher only uses 20 columns out of a total of 21 columns. This means that there is 1 column that the researcher does not use, namely the "phone number" column. This is because the column is considered irrelevant to the output class.

3.4 DATA TRANSFORMATION

First, we will change the "account length" column into three categories: short, middle, and long. The trick is to divide three by the maximum value in that column, and the result is 243/3=81. This value is what the researcher uses as a reference in determining the interval for each group < 82, 82 – 162, and > 162 long.

Furthermore, the dataset will be changed into three data groups, namely, continuous data, category data, and class data. To find out which columns will be converted into the three data groups above, we can see them in Table 3.2. The number of each data group is data continuous 14 columns, data category five columns, and data class 1 column.

Table 2: Define Dataset

Group Name	Column Name
Continues	number vmail message, total day minutes, total day calls, total day charge, total eve minutes, total eve calls, total eve charge, total night minutes, total night calls, total night charge, total int minutes, total intl calls, total intl charge, customer service calls

Category	state, international plan, voice mail plan, area code, account length group
Class	churn

3.5 MODEL BUILDING

In this study, we will build a neural network model. The model will be given input based on the column in the continuous data group, adding the number with the number of columns in the categorical data group and the results of the embedding calculation. The results of this model can be either a single regression or a classification. This research will focus on the results in the form of a single regression. Figure 1 is an illustration of the stages in the model we built.

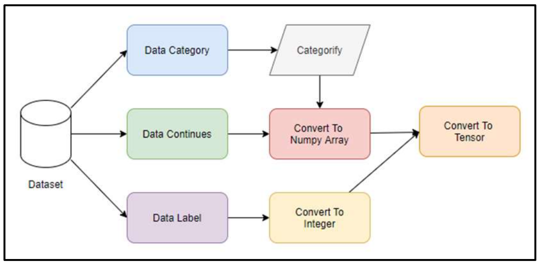


Figure 1: Model Building

3.6 MODEL TRAINING AND VALIDATING

In this study, the parameters used are 100 epochs, 0.5 cost function weight, 0.05 optimizer learning rate, and 4-fold cross-validation. The values that will be used as a reference in this research are cost value and performance matrices. To calculate the cost value, we will use Cross-entropy. Cross-entropy is a measure of the difference between two probability distributions for a given random variable or event. The researcher will use the weight parameter in Cross-entropy to handle the dataset imbalance. Researchers use the confusion matrix to calculate the performance matrix. Next, we will divide the dataset into testing and validating training data with a ratio of 20:80. The dataset with a ratio of 80 will be used as training and validating data using cross-validation. Cross-Validation is a sampling method used to evaluate the model on a limited sample of data. We will apply cross-validation with a 4-layer setup to a data set that has 3333 rows. The result is that cross-validation will divide the data into 25% validating data and 75% training data. This process will be repeated in four layers. Until all the data has become validating data and training data. See Figure 2 for illustration of cross validation.

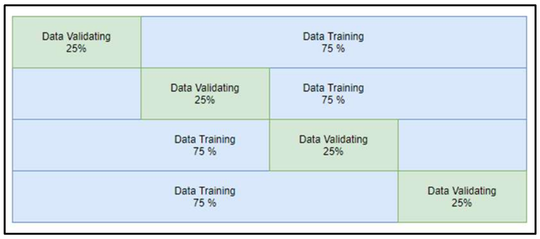


Figure 2: Illustration of Cross Validation 4 Layers

4 EQUATIONS

Researchers will compare the results of several optimizer functions to find the optimizer function that has the best results for use in this model and dataset. Researchers will also check whether the results are included in the overfitting and underfitting categories. Overfitting is a condition where the built model gives good results on training and validation data but gives poor results on data testing. While underfitting is a condition where the model built gives poor results both on training and validation data as well as on test data so that it still cannot be used.

4.1 OPTIMIZER FUNCTION

4.1.1 ADAM OPTIMIZER

From the observation of the cost value and accuracy of the model made by Adam's implementer, it was found that the model was not overfitting and underfitting for more details can be seen in figure 2.

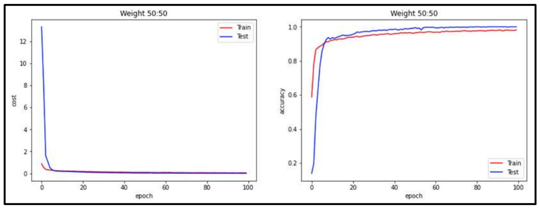


Figure 2: Adam Optimizer Overfitting and Underfitting Test

If you look at the confusion matrix in Figure 3 below, this model gets the highest value is 549 data with criteria of prediction label is churn and true label is churn. The lowest value is 9 data with criteria of prediction label is not churn and true label is churn.

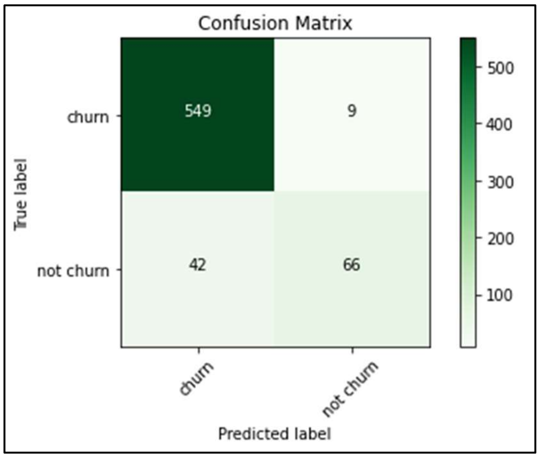
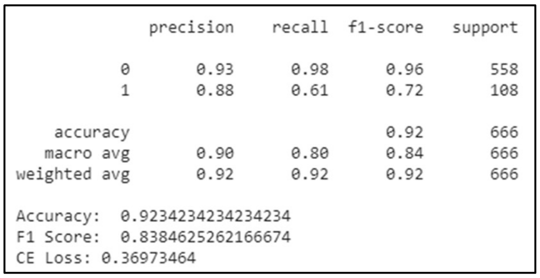


Figure 3: Adam Optimizer Confusion Matrix Result

While, for matrix evaluation results in Figure 4, we have a cost value are 0.369, an F1 Score is 0.838 and an accuracy is 0.923.



```
              precision    recall  f1-score   support

           0       0.93      0.98      0.96       558
           1       0.88      0.61      0.72       108

    accuracy                           0.92       666
   macro avg       0.90      0.80      0.84       666
weighted avg       0.92      0.92      0.92       666

Accuracy:  0.9234234234234234
F1 Score:  0.8384625262166674
CE Loss: 0.36973464
```

Figure 4: Adam Optimizer Matrix Evaluation Result

4.1.2 ADADELTA OPTIMIZER

From the observation of the cost value and accuracy of the model created by implementing AdaDelta, it was found that the model is not overfitting and underfitting for more details can be seen in Figure 5.

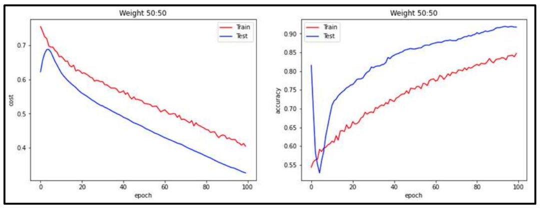


Figure 5: AdaDelta Optimizer Overfitting and Underfitting Test

If you look at the confusion matrix in Figure 6 below, this model gets the highest value is 540 data with criteria of prediction label is churn and true label is churn. The lowest value is 18 data with criteria of prediction label is not churn and true label is churn.

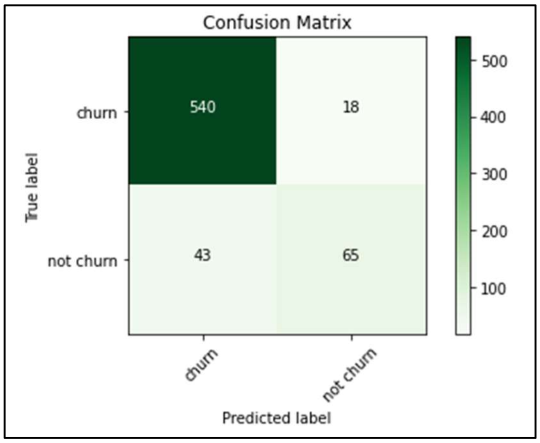
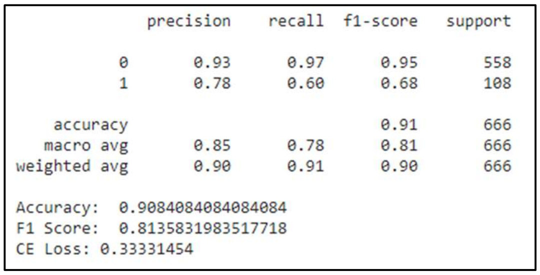


Figure 6: AdaDelta Optimizer Confusion Matrix Result

While, for matrix evaluation results in Figure 7, we have a cost value are 0.333, an F1 Score is 0.813 and an accuracy is 0.908.



```
              precision    recall  f1-score   support

           0       0.93      0.97      0.95       558
           1       0.78      0.60      0.68       108

    accuracy                           0.91       666
   macro avg       0.85      0.78      0.81       666
weighted avg       0.90      0.91      0.90       666

Accuracy:  0.9084084084084084
F1 Score:  0.8135831983517718
CE Loss: 0.33331454
```

Figure 7: AdaDelta Optimizer Matrix Evaluation Result

4.1.3 RMS-PROP OPTIMIZER

From the results of observations of the cost value and accuracy of the model made by implementing RMS-Prop, it was found that the model was not overfitting and underfitting for more details, it can be seen in Figure 8.

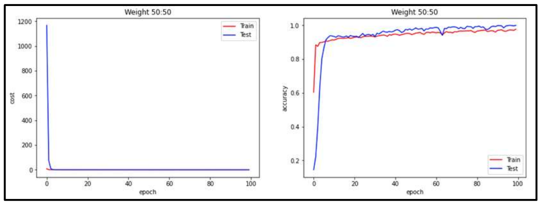


Figure 8: RMS-Prop Optimizer Overfitting and Underfitting Test

If you look at the confusion matrix in Figure 9 below, this model gets the highest value is 551 data with criteria of prediction label is churn and true label is churn. The lowest value is 7 data with criteria of prediction label is not churn and true label is churn.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

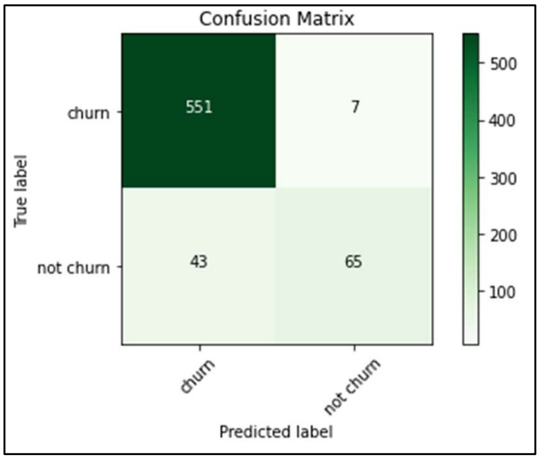
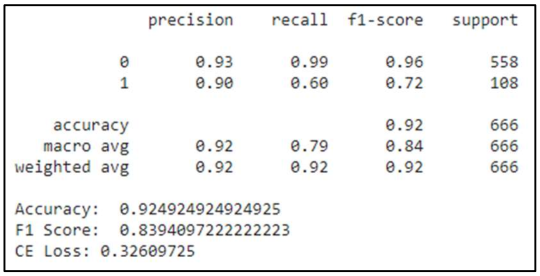


Figure 9: RMS-Prop Optimizer Confusion Matrix Result

While, for matrix evaluation results in Figure 10, we have a cost value are 0.326, an F1 Score is 0.839 and an accuracy is 0.924.



```
              precision    recall  f1-score   support

           0       0.93      0.99      0.96       558
           1       0.90      0.60      0.72       108

    accuracy                           0.92       666
   macro avg       0.92      0.79      0.84       666
weighted avg       0.92      0.92      0.92       666

Accuracy:  0.924924924924925
F1 Score:  0.8394097222222223
CE Loss: 0.32609725
```

Figure 10: RMS-Prop Optimizer Matrix Evaluation Result

4.1.4 ADAGRAD OPTIMIZER

From the results of observations of the cost value and accuracy of the model created by implementing AdaGrad, it was found that the model was not overfitting and underfitting for more details can be seen in Figure 11.

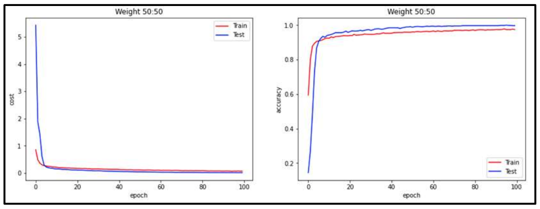


Figure 11: AdaGrad Optimizer Overfitting and Underfitting Test

If you look at the confusion matrix in Figure 12 below, this model gets the highest value is 549 data with criteria of prediction label is churn and true label is churn. The lowest value is 9 data with criteria of prediction label is not churn and true label is churn.

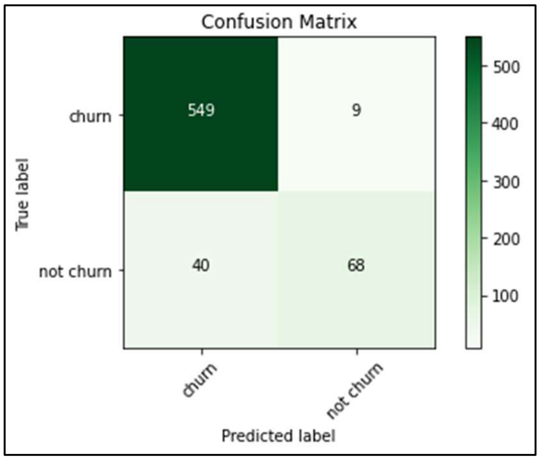
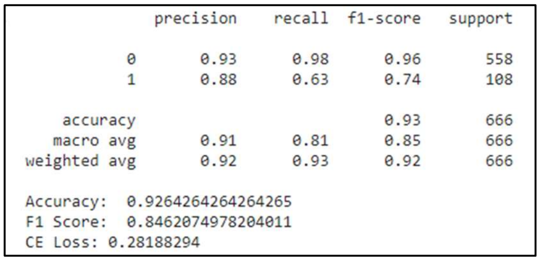


Figure 12: AdaGrad Optimizer Confusion Matrix Result

While, for matrix evaluation results in Figure 13, we have a cost value are 0.281, an F1 Score is 0.846 and an accuracy is 0.926.



```
              precision    recall  f1-score   support

           0       0.93      0.98      0.96       558
           1       0.88      0.63      0.74       108

    accuracy                           0.93       666
   macro avg       0.91      0.81      0.85       666
weighted avg       0.92      0.93      0.92       666

Accuracy:  0.9264264264264265
F1 Score:  0.8462074978204011
CE Loss: 0.28188294
```

Figure 13: AdaGrad Optimizer Matrix Evaluation Result

4.1.5 SGD OPTIMIZER

From the observation of the cost value and accuracy of the model made by implementing SGD, it was found that the model was not overfitting and underfitting for more details can be seen in Figure 14.

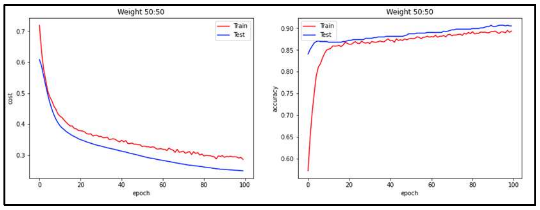


Figure 14: SGD Optimizer Overfitting and Underfitting Test

If you look at the confusion matrix in Figure 15 below, this model gets the highest value is 555 data with criteria of prediction label is churn and true label is churn. The lowest value is 3 data with criteria of prediction label is not churn and true label is churn.

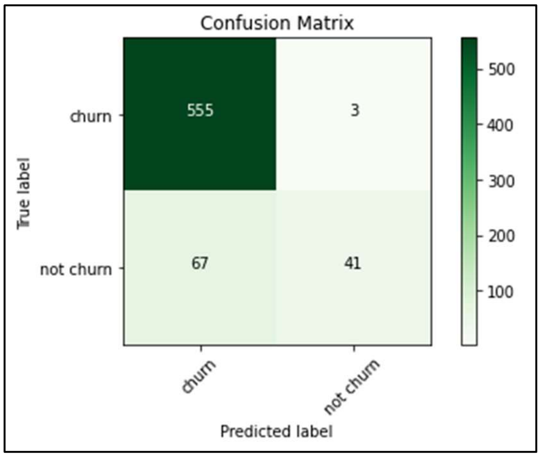
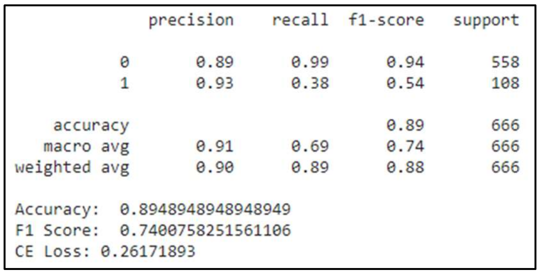


Figure 15: SGD Optimizer Confusion Matrix Result

While, for matrix evaluation results in Figure 16, we have a cost value are 0.261, an F1 Score is 0.740 and an accuracy is 0.894.



```
              precision    recall  f1-score   support

           0       0.89      0.99      0.94       558
           1       0.93      0.38      0.54       108

    accuracy                           0.89       666
   macro avg       0.91      0.69      0.74       666
weighted avg       0.90      0.89      0.88       666

Accuracy:  0.8948948948948949
F1 Score:  0.7400758251561106
CE Loss: 0.26171893
```

Figure 16: AdaGrad Optimizer Matrix Evaluation Result

4.2 COMPARISON RESULT

When compared to previous research with the same data source, in this paper we add one additional column, namely the long group of accounts that are included in the deep learning process. This is because the length of the account is one of the habits of the customer.

From the experimental results, Figure 7 shows that the optimizer function SGD shows the best performance with the lowest cost value, which is 0.261. Followed by AdaGrad and RMS-Prop with values of 0.281 and 0.326, respectively. While the worst performance is shown by Adam with a cost value of 0.369.

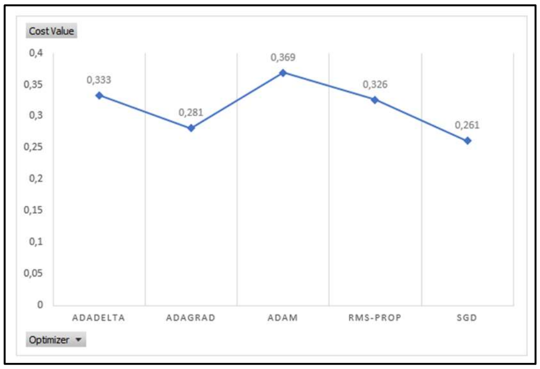


Figure 7: Chart Optimizer Function vs Cost Value

The comparison uses the accuracy value in Figure 8. The best value was scored by AdaGrad with a value of 0.926, followed by RMS-Prop and Adam with a value of 0.924 and 0.923, respectively. While the worst performance faced by SGD with an accuracy value of 0.894.

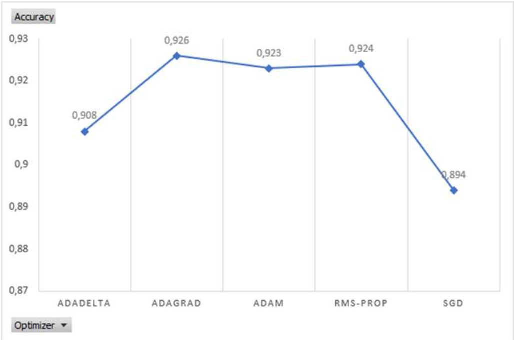


Figure 8: Chart Optimizer Function vs Accuracy

As for the comparison using the F1 score, it is shown in Figure 9. The best value was shown by AdaGrad with a value of 0.846, followed by Adam and RMS-Prop with a value of 0.838 and 0.834, respectively. While the worst results are shown by SGD with a value of 0.740.

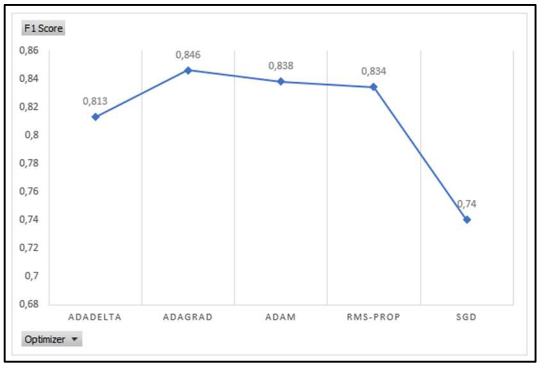


Figure 9: Chart Optimizer vs F1 Score

All experimental results of matrix evaluation can be seen in the table below:

Table 3: Optimizer Matrix Evaluation Results

No	Optimizer Function	Cost Value	Accuracy	F1 Score
1	AdaDelta	0.333	0.908	0.813
2	AdaGrad	0.281	0.926	0.846
3	Adam	0.369	0.923	0.838
4	RMSprop	0.326	0.924	0.834
5	SGD	0.261	0.894	0.740

From the results of the comparison above, we do not find any models experiencing overfitting or underfitting. As for the results of the performance matrix, it can be concluded that optimizer function with the lowest cost value is in SGD with AdaGrad and RMS-Prop. Meanwhile, the AdaGrad classification has the best value, followed by RMS-Prom and Adam. The best calculation results for cost value, accuracy, and f1 score are 0.261, 0.926, and 0.846 respectively.

This research uses a simple deep learning model and although the results of this study are satisfactory, the model with deep learning can be further improved. one of them is by enriching the data sources used by combining them with other data sources such as social media.

REFERENCES:

[1] A. Berson, S. Smith, and K. Thearling, Building Data Mining Applications for CRM. McGraw-Hill, 1999.

[2] R. Daga, Buku 1, Citra, Kualitas Produk dan Kepuasan Pelanggan, no. November. 2019.

[3] S.Suyanto; Ramadhani, Kurniawan Nur; Mandala, Deep Learning - Modernisasi Machine Learning untuk Big Data. Bandung: INFORMATIKA, 2019.

[4] D. C. Wu, B. Bahrami Asl, A. Razban, and J. Chen, "Air compressor load forecasting using artificial neural network," Expert Syst. Appl., vol. 168, no. November, p. 114209, 2021, doi: 10.1016/j.eswa.2020.114209.

[5] F. Okubo, A. Shimada, T. Yamashita, and H. Ogata, "A neural network approach for students' performance prediction," ACM Int. Conf. Proceeding Ser., pp. 598–599, 2017, doi: 10.1145/3027385.3029479.

[6] Mislan, Haviluddin, S. Hardwinarto, Sumaryono, and M. Aipassa, "Rainfall Monthly Prediction Based on Artificial Neural Network: A Case Study in Tenggarong Station, East Kalimantan - Indonesia," Procedia Comput. Sci., vol. 59, no. Iccsci, pp. 142–151, 2015, doi: 10.1016/j.procs.2015.07.528.

[7] Z. Ma et al., "Deep Neural Network-Based Impacts Analysis of Multimodal Factors on Heat Demand Prediction," IEEE Trans. Big Data, vol. 6, no. 3, pp. 594–605, 2019, doi: 10.1109/tbdata.2019.2907127.

[8] M. Rapp, A. Pathania, T. Mitra, and J. Henkel, "Neural Network-Based Performance Prediction for Task Migration on S-NUCA Many-Cores," IEEE Trans. Comput., vol. 70, no. 10, pp. 1691–1704, 2021, doi: 10.1109/TC.2020.3023022.

[9] M. Nazir, S. Shakil, and K. Khurshid, "Role of deep learning in brain tumor detection and classification (2015 to 2020): A review," Comput. Med. Imaging Graph., vol. 91, p. 101940, Jul. 2021, doi: 10.1016/j.compmedimag.2021.101940.

[10] D. Ayvaz, R. Aydoğan, M. T. Akçura, and M. Şensoy, "Campaign participation prediction with deep learning," Electron. Commer. Res. Appl., vol. 48, p. 101058, Jul. 2021, doi: 10.1016/j.elerap.2021.101058.

[11] E. Kim et al., "Champion-challenger analysis for credit card fraud detection: Hybrid ensemble and deep learning," Expert Syst. Appl., vol. 128, pp. 214–224, Aug. 2019, doi: 10.1016/j.eswa.2019.03.042.

[12] K. Rangasamy, M. A. As'ari, N. A. Rahmad, and N. F. Ghazali, "Hockey activity recognition using pretrained deep learning model," ICT Express, vol. 6, no. 3, pp. 170–174, Sep. 2020, doi: 10.1016/j.icte.2020.04.013.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[14] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit., vol. 2016-Decem, pp. 2818–2826, 2016, doi: 10.1109/CVPR.2016.308.

[15] J. A. Gliner, G. A. Morgan, N. L. Leech, J. A. Gliner, and G. A. Morgan, "Measurement Reliability and Validity," Res. Methods Appl. Settings, pp. 319–338, 2021, doi: 10.4324/9781410605337-29.

[16] Y. Dokuz and Z. Tufekci, "Mini-batch sample selection strategies for deep learning based speech recognition," Appl. Acoust., vol. 171, p.

107573, Jan. 2021, doi: 10.1016/j.apacoust.2020.107573.

[17] H. Aouani and Y. Ben Ayed, “Speech Emotion Recognition with deep learning,” in Procedia Computer Science, Jan. 2020, vol. 176, pp. 251–260, doi: 10.1016/j.procs.2020.08.027.

[18] A. Borjali, M. Magnéli, D. Shin, H. Malchau, O. K. Muratoglu, and K. M. Varadarajan, “Natural language processing with deep learning for medical adverse event detection from free-text medical narratives: A case study of detecting total hip replacement dislocation,” Comput. Biol. Med., vol. 129, p. 104140, Feb. 2021, doi: 10.1016/j.compbiomed.2020.104140.

[19] N. Shahi, A. K. Shahi, R. Phillips, G. Shirek, D. M. Lindberg, and S. L. Moulton, “Using deep learning and natural language processing models to detect child physical abuse,” J. Pediatr. Surg., Mar. 2021, doi: 10.1016/j.jpedsurg.2021.03.007.

[20] V. Sorin, Y. Barash, E. Konen, and E. Klang, “Deep-learning natural language processing for oncological applications,” The Lancet Oncology, vol. 21, no. 12. Lancet Publishing Group, pp. 1553–1556, Dec. 01, 2020, doi: 10.1016/S1470-2045(20)30615-X.

[21] X. Jiang, B. Hu, S. Chandra Satapathy, S. H. Wang, and Y. D. Zhang, “Fingerspelling Identification for Chinese Sign Language via AlexNet-Based Transfer Learning and Adam Optimizer,” Sci. Program., vol. 2020, 2020, doi: 10.1155/2020/3291426.

[22] V. N. Sewdien, R. Preece, J. L. R. Torres, E. Rakhshani, and M. van der Meijden, “Assessment of critical parameters for artificial neural networks based short-term wind generation forecasting,” Renew. Energy, vol. 161, pp. 878–892, 2020, doi: 10.1016/j.renene.2020.07.117.

[23] A. Kumar, S. Sarkar, and C. Pradhan, “Malaria Disease Detection Using CNN Technique with SGD, RMSprop and ADAM Optimizers,” in Deep Learning Techniques for Biomedical and Health Informatics, 2020, pp. 211–230.

[24] A. A. Lydia and F. S. Francis, “Adagrad-An Optimizer for Stochastic Gradient Descent,” Int. J. Inf. Comput. Sci., vol. 6, no. 5, pp. 566–568, 2019, [Online]. Available: http://ijics.com.

[25] J. Yang and G. Yang, “Modified convolutional neural network based on dropout and the stochastic gradient descent optimizer,” Algorithms, vol. 11, no. 3, pp. 1–15, 2018, doi: 10.3390/a11030028.

[26] David Becks, “Churn in Telecom’s dataset.” https://www.kaggle.com/becksddf/churn-in-telecoms-dataset.

[27] T. Zhang, S. Moro, and R. F. Ramos, “A Data-Driven Approach to Improve Customer Churn Prediction Based on Telecom Customer Segmentation,” *Futur. Internet*, vol. 14, no. 3, p. 94, 2022, doi: 10.3390/fi14030094