Müşteri Bölütlemesinde Yapay Sinir Ağları
Artificial Neural Networks in Customer Segmentation

Şükrü Ozan* and Leonardo O. Iheme*
*AdresGezgini Inc. Research & Development Center,
Izmir, Turkey
sukruozan@adresgezgini.com, leonardoiheme@adresgezgini.com

***Özetçe*—Müşteri bölütlemesi müşteri ilişkileri yönetimi literatüründe ve yazılımlarında, doğrudan müşteri memnuniyeti ile ilgili olduğu için önemli bir yöntemdir. Müşterileri iki ayrık gruba bölmek için kullanılan en yaygın yöntem bir grup müşteriyi özel bir etiketle etiketlemektir. Bu çalşmada şirketimizin benzeri bir şekilde bölütlenmiş müşteri verisi ve ilgili istatistiksel veriler çok katmanlı algılayıcı (MLP) olarak nitelendirilen yapay sinir ağı tabanlı bir makine öğrenmesi modelinin eğitim ve test aşamalarında kullanılmıştır. İlgili öznitelikler belirlenip yapay sinir ağı eğitimi için uyarlandığında ve sisteme ait hiper parametreler kapsamlı bir ızgara araması ile uyumlandırıldığında, sistem müşteri bölütleme yaklaşımımızı çok iyi bir şekilde genellemekte ve kısa sürede iyi doğruluk oranına erişmektedir. Önerilen sistem şirketimizin bilgi sistemine, müşteriler ile ilgili veri tablolarını sıklıkla analiz edip, bir müşterinin özel müşteri olarak etiketlenip etiketlenmemesi gerektiğine anlık olarak karar verebilecek şekilde entegre edilebilir. Bu otomatik karar verme mekanizması şirketimizin müşteri memnuniyetini önemli ölçüde arttırabilecektir.**

***Anahtar Kelimeler*—Yapay Sinir Ağları, Müşteri Bölütleme, Makine Öğrenmesi, Çok Katmanlı Algılayıcı, Müşteri Memnuniyeti.**

***Abstract*—Customer segmentation is an important method both in customer relationship management literature and software since it directly relates with customer satisfaction of the companies. The most common way to separate customers into two distinct groups is to tag a group of customers with a special label. In this study, our company's likely segmented customer data and related statistical data are used to train and test a neural network based machine learning model, namely Multi layer Perceptron (MLP) . Once the related features are tailored for artificial neural network training and the hyper parameters are tuned accordingly by deploying an extensive grid search algorithm, the system achieved a good generalization of our customer segmentation strategy and hence a good overall accuracy within a few epochs. The proposed system can be integrated to our company's data framework such that it can frequently analyze the customer related data tables and can decide whether a customer is to be promoted or is to remain unchanged. This automatic decision mechanism can improve our company's customer satisfaction.**

***Keywords*—Artificial Neural Networks, Customer Segmentation, Machine Learning, Multi layer Perceptron, Customer Satisfaction.**

I. Introduction

Since it is directly related to customer satisfaction, customer segmentation is an important management method in CRM literature. It is also an active research area especially in industrial management literature [1],[2].

It is common to perform a binary classification and label customers with two different labels such as "standard" and "premium". In this work, our company's likely segmented customer data is analyzed. The segmentation has been performed manually through the years, since the company's foundation. This study aims to find a machine learning model which can successfully generalize the company's data segmentation intuition.

Previously the same problem was addressed in a recent work [3], where the proposed method compares three introductory level machine learning methods, namely Normal Equation, Multivariate Linear Regression and Logistic Regression. In all three methods, logistic regression was proven to have a significantly superior efficiency. In this study, a state-of-art artificial neural network model, namely multi layer perceptron (MLP), is used to solve the same problem. MLP is an artificial neural network class which utilizes back propagation method [4] for its parameters to be optimized.

In the next section, detailed information about the data and implementation details are given. The third section explains the application of the MLP model to the specially tailored data. In the fourth section the application results of these methods are shown by using plots which are conventionally used in machine learning literature. The last section includes interpretation of the achieved results and concluding comments.

II. The Data

The data used in this study is from the customer database of our company which serves as a Google Ads Premier Agency. The customers are mostly small and medium-sized enterprises (SMEs) from a very wide range of industries, which results in a high variance in customers online advertising investment amounts.

At the time of writing this paper, the total amount of unique customers with payment history was 15138, 13889 of which are labeled as standard customer and the remaining

1249 are labeled as premium customer. Since the amount of premium customers are nearly one tenth of the pipenv standard customers, this makes it difficult for a regression based learning algorithm to achieve high accuracy because of the unbalanced nature of the sample distribution. However, using neural networks makes it possible to generalize a successful model despite the irregularities in the data set.

III. Application of MLP Model

In this section, implementation details of the system are given. Data prepation is the first important step. When the data is ready, the necessary system parameters, i.e. hyper parameters, are optimized to achieve the best result.

A. Data Preparation

The database comprises not only of numerical information such as number of payments and total amount of payments but also various categorical information of an individual customer such as the city, service category etc. Each of these information plays a significant role while deciding to promote a customer as "premium customer". Hence it is important to use them in the proposed framework. Few generic features like maximum amount of payments and average of payments are also valuable information and they are added as feature columns in the training data. The data is prepared by joining information gathered from different database tables by making optimized MySQL queries into a single data frame.

While preparing data to be used in training and to construct the MLP model, Keras [5], which is a high level API specially designed to run on top of frameworks like Tensorflow [6], is used. This API is capable of handling both categorical and numerical data together. It also makes it easy to bucketize some numerical columns such as date of subscription and treat this feature as a categorical feature. Date of subscription is such an example of a numerical feature which can be treated as a categorical feature.

The gradient descent algorithm which is implicitly implemented in neural network training for optimization, requires the normalization of the numerical features for faster convergence. Hence, as another preprocessing step, the numerical columns are normalized accordingly. In Table I, 10 randomly selected data from the training data set can be seen. The first column is the id of the sample, the last column is the label where 1 means premium customer and the middle columns represent major features inserted in the training data.

B. Hyper Parameter Tuning

Once the data preparation is completed, the system can be trained. By intuition, we expected the MLP model to produce good results. A visualization of a sample sequential MLP model can be seen in Figure 1.

As is customary with machine learning pipelines, once the data is ready for training, the next step is hyper parameter tuning/optimization. This step is crucial since it significantly affects the accuracy of the system. Various components of a machine learning model can be treated as hyper parameters and be optimized. In some applications even the activation function type is treated as a hyper parameter and different activation functions can be tested in the hyper parameter tuning phase. In this study, significant components such as learning rate, batch size, number of layers, number of nodes in layers, number of epochs, validation set size and dropout ratio are selected as hyper parameters.

Hyper parameter search is traditionally performed by simply imposing a grid search where the best combination of hyper parameter values is found by testing combinations of predefined discrete values of each. In Table II selected hyper parameters with their reasonable value sets and their optimized values, in the last column, can be seen.

According to the optimized hyper parameters, a 4 densely connected layers, including the input and output layers, followed by a dropout layer, with 64 nodes and dropout ratio 0.5 has given the best accuracy of 94.53%. Moreover, the best learning rate is found as 0.01. Together with the optimized batch size value, ReLU is the activation function used in the input and hidden network layers. Since this is a binary classification, the sigmoid function is applied at the output node of the network.

20% of the data is reserved as test set and the remaining 80% is used as training set. The validation set is chosen from the training set according to the validation ratio hyper parameter which is also given in Table II. The number of epochs is another important hyper parameter since in practical applications long epoch times may result in over training which reduces the modal accuracy and increases the model loss. Hence in most practical applications, short epochs are a better consideration. In order to keep track of the optimal epoch amount, it is also introduced as a hyper parameter in the grid search. 25 is found as the optimal epoch for this study. It is observed that for this problem the convergence is achieved at around 25 epochs and longer training causes over training and increases the model loss.

The grid search used for hyper parameter tuning is an extensive search hence it requires training different model structures multiple times. Doing this task by using conventional programming, i.e. simply running the whole algorithm by using a single CPU, requires significant amount of time. The multiprocessing approach proposed in [7] is inherited and all cores of the workstation were used in parallel to compute model accuracy and model loss for each possible combination of hyper parameters listed in Table II.

C. Optimization

The model uses binary cross entropy to calculate model loss. Moreover, RMSprop is the chosen optimizer. Learning rate is the most important parameter of this optimizer. The hyper parameter search in the previous section has given 0.01 as the optimal learning rate within the given list of learning rates.

IV. Results and Discussions

In this study customer segmentation strategy of our company is modelled by using one of the state of art machine

Traning Sample	category	city	login_date	max_payment	nof_payments	total_payments	mean_payment	label
5328	19	35	(1388871000.0, 1415323500.0]	0.000089	0.037500	0.001284	0.070858	0
216	70	34	(1335966000.0, 1362418500.0]	0.003125	0.030682	0.011689	0.081064	1
15004	19	6	(1468228500.0, 1494681000.0]	0.000446	0.006818	0.001226	0.073992	0
14783	87	34	(1256608500.0, 1283061000.0]	0.000134	0.003409	0.000456	0.070885	0
5280	93	34	(1388871000.0, 1415323500.0]	0.000134	0.004545	0.000428	0.070512	0
12212	62	34	(1388871000.0, 1415323500.0]	0.000046	0.007955	0.000544	0.070773	0
1332	48	35	(1309513500.0, 1335966000.0]	0.000045	0.009091	0.000540	0.070423	0
2389	49	34	(1229838570.0, 1256608500.0]	0.000045	0.000000	0.000372	0.070870	0
1704	66	35	(1309513500.0, 1335966000.0]	0.000134	0.001136	0.000428	0.070199	0
5078	63	34	(1362418500.0, 1388871000.0]	0.000223	0.064773	0.002293	0.071018	0

TABLE I: 10 randomly selected samples and their corresponding feature values can be seen in this table. These are the vast majority of the features. Features named category, city and login_date are categorical values. Login date is normally a numerical value which shows time in unix time stamp format. But for convenience it is discretized and treated as a categorical value as well. The remaining columns, from max_payment up to mean_payment, are numerical columns which are normalized for achieving faster convergence rate. The last column represents the binary label indicating standard and premium customers.

Hyper Parameter	Search Values	Optimized Value
batch_size	[10,25,50]	25
learning_rate	[0.01,0.025,0.5]	0.01
nof_layers	[3,4,5]	4
nof_epoch	[10,25,50]	25
dropout_param.	[0.1,0.25,0.5]	0.5
nof_nodes	[16,48,64]	64
validation_ratio	[0.1,0.2,0.3]	0.2

TABLE II: Selected hyper parameters, first column, with their reasonable value sets, middle column, and their optimized values, the last column. The hyper parameter combination in the last column gives the best accuracy for the data set.

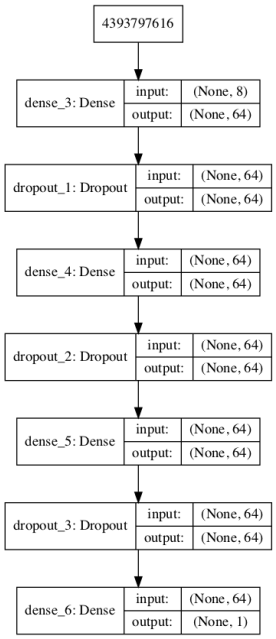


Fig. 1: Visualization of a sample MLP model used in this study

learning methods called MLP which is a feed forward neural network structure optimized by using back propagation.

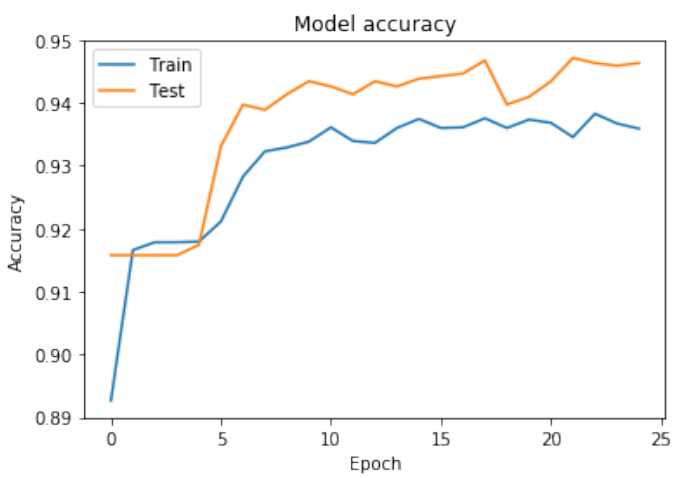
The number of layers, number of nodes in the layers, dropout ratio, batch size, learning rate and size of the validation set size are considered as hyper parameters and they are optimized by using grid search algorithm. The network structure built by using the optimized parameters can be seen in Figure 1. Since the grid search algorithm requires an extensive search within all possible combinations of the hyper parameters, the multiprocessing approach inherited from [7] has improved the processing time for the overall procedure significantly. The workstation used in this study is a DELL Workstation with a CPU of 8 cores and 16 Logical Processors.

The success of the model can also be verified by plotting model accuracy vs epoch and model loss vs epoch graphs. In Figure 2 the worst and best case model accuracy and model loss graphs are given respectively. The worst and best cases are selected from within the hyper parameter search results.

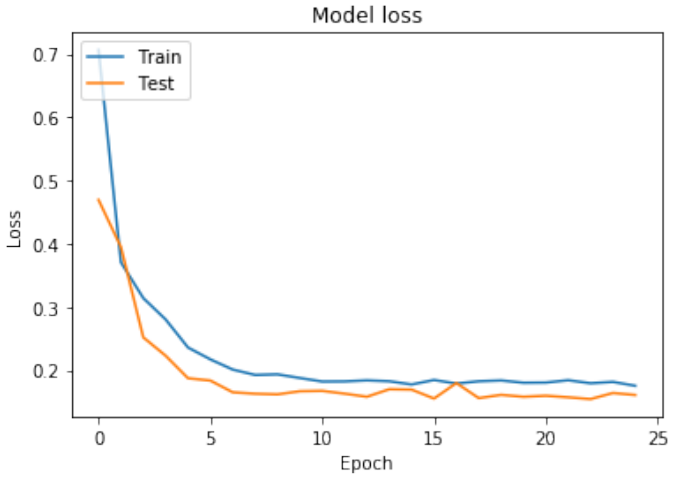
The results show that even with relatively few number of training samples it is possible to construct a generalizing model which can imitate the intuition behind our company's customer classification strategy. This model can be incorporated within our work flow and it can analyze the customers regularly and automatically decide to promote a customer and give a notice to the managers.

The performance of method can further be improved by making data augmentation and trying different ANN structures. Even performance of different activation functions can also be measured together with the hyper parameters. It is also possible to try different search methods other than grid search method, e.g. random search.

As a future work, customer behavioral analysis by using machine learning techniques will be considered. This includes not only the statistical information, like the payment information of a customer, given in this study but also the sentiment analysis of communication between customers and customer representatives. Incorporating these information with machine learning can predict the probability of losing a customer at any time and enables the company to take some preemptive actions when necessary.



(a) Best Case - Model Accuracy vs Epoch



(b) Best Case - Model Loss vs Epoch

Fig. 2: (a) The figure depicts the model accuracy over the training and the test sets. (b) The figure depicts the model loss over the training and the test sets. Optimal hyper parameter values in Table II are used.

REFERENCES

[1] K. Windler, U. Jüttner, S. Michel, S. Maklan, and E. K. Macdonald, "Identifying the right solution customers: A managerial methodology," *Industrial Marketing Management*, vol. 60, pp. 173 – 186, 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S001985011630027X

[2] R. Thakur and L. Workman, "Customer portfolio management (cpm) for improved customer relationship management (crm): Are your customers platinum, gold, silver, or bronze?" *Journal of Business Research*, vol. 69, no. 10, pp. 4095 – 4102, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0148296316300625

[3] Ş. Ozan, "A case study on customer segmentation by using machine learning methods," in *2018 International Artificial Intelligence and Data Processing Symposium (IDAP)*. IEEE, Sep 2018.

[4] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Parallel distributed processing: Explorations in the microstructure of cognition, vol. 1," D. E. Rumelhart, J. L. McClelland, and C. PDP Research Group, Eds. Cambridge, MA, USA: MIT Press, 1986, ch. Learning Internal Representations by Error Propagation, pp. 318–362. [Online]. Available: http://dl.acm.org/citation.cfm?id=104279.104293

[5] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[6] M. A. et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[7] S. Ozan, "Increasing system performance in machine learning by using multiprocessing," in *2018 26th Signal Processing and Communications Applications Conference (SIU)*, May 2018, pp. 1–4.