Araştırma Makalesi / Research Article

Fuzzy Logic and Deep Learning Integration in Likert Type Data

Zeynep ÜNAL[1], Emre İPEKÇİ ÇETİN [2, *]

[1] *Niğde Ömer Halisdemir University, Faculty of Agricultural Sciences and Technology, Department of Biosystems Engineering, Niğde, Turkey.*

[2] *Akdeniz University, Faculty of Economics and Administrative Sciences, Department of Econometrics, Antalya, Turkey.*

*e-posta: *Corresponding author: ecetin@akdeniz.edu.tr   ORCID ID: http://orcid.org/0000-0002-8108-1919*
zeynepual@ohu.edu.tr ORCID ID: http://orcid.org/0000-0002-9954-1151

Abstract

Keywords
Deep Learning;
Logistic Regression;
Fuzzy Logic;
Likert Scale

Deep learning networks have many modern applications and demonstrate a high-performance level. As the applications of deep learning networks to real-world problems continues to spread, the reason why they are effective remains unknown. However, it is possible to make some judgments by examining the behaviour of the network in experiments. The main aim of this study is to analyse the performance of deep learning techniques in the form of a 5-point Likert-type scale by converting the artificial data sets into a fuzzy form using triangular or trapezium fuzzy numbers. To test the performance of the proposed model, which is the integration of deep learning and fuzzy logic techniques, the satisfaction estimation problem was chosen. Data sets consisting of fuzzy numbers which reach at least three or four times more parameters than normal data sets. Thus, it decreases the possibility of falling into the local optimum trap in optimization studies with big data. In the analysis conducted with deep learning, in accordance with the fuzzification examples in the literature, the defuzzification was carried out with separate results for peak, maximum, and minimum values. In contrast to the literature, the performances of the deep learning model were investigated by suggesting that fuzzy numbers produce a single result series.

Likert Tip Veride Bulanık Mantık ve Derin Öğrenme Entegrasyonu

Öz

Anahtar kelimeler
Derin Öğrenme;
Lojistik Regresyon;
Bulanık Mantık;
Likert Ölçeği

Derin öğrenme ağları birçok modern uygulamaya sahip olup yüksek performans seviyesi göstermektedir. Derin öğrenme ağlarının gerçek dünyadaki sorunlara uygulamaları yayılmaya devam ederken bunların neden etkili olduğu bilinmemektedir. Ancak deneylerde ağların davranışını inceleyerek bazı yargılarda bulunmak mümkündür. Bu çalışmanın amacı 5 noktalı Likert tipi ölçeğiyle üretilen yapay veri setlerinin üçgensel ya da yamuk bulanık sayılar kullanılarak bulanık forma dönüştürülmesi ve bu yolla verilerin çoğalması durumunda derin öğrenme tekniklerinin performansının analiz edilmesidir. Derin öğrenme ve bulanık mantık tekniklerinin entegrasyonu sonucunda önerilen modelin performansının test edilmesi için memnuniyet tahmin problemi seçilmiştir Bulanık sayılarla oluşturulan veri setleri ile normal veri setinden en az 3 ya da 4 kat daha fazla parametre sayısına ulaşılmaktadır. Böylece büyük veri ile optimizasyon çalışmalarında yerel optimuma tuzağına düşme olasılığı azalmaktadır. Derin öğrenme ile yapılan analizlerde, literatürdeki bulanıklaştırma örneklerine uygun olarak, tepe, maksimum ve minimum değerler için ayrı sonuçlarla durulaştırma gerçekleştirilmiştir. Literatürden farklı olarak bulanık sayıların tek sonuç dizisi üretmesi önerilerek derin öğrenme modelinin performansları araştırılmıştır.

1. Introduction

Deep learning is a part of the machine learning domain inspired by information processing principles of the human brain. Deep learning is based on deep neural networks that were introduced 30 years ago. Due to the difficulty of training deep neural networks, developments in this field slowed down. However, in 2006, an idea of training each layer of deep nets separately proposed by Hinton *et al*., accelerated the research in this area (Hinton *et al.* 2006). Unlike other machine learning techniques, it learns the required features independently from the training data set without

the need for the expert to define (Kappor *et al*. 2018). Deep learning, which is one of the techniques that process big data quickly and accurately, provides great benefits in many fields such as business, management, medicine, health, engineering, and scientific research (Wang *et al*. 2017). One of the areas where deep learning techniques are widely used is satisfaction estimation (Tabrizi *et al*. 2016).

Several methods are used to estimate the satisfaction examine the effects of various variables on the overall level of satisfaction (Deng and Pei 2009). Deep learning is among the suitable techniques for customer satisfaction analysis and widely used to examine the complex relationship between input variables and output variables. There are many studies in the literature that use artificial neural networks to estimate overall satisfaction. For example, Jahandideh *et al*. proposed a model based on artificial neural networks that predict how patients evaluate hospital services in general by using factors such as reliability, insurance, physical conditions, empathy, and sensitivity (Jahandideh *et al*. 2013). Najmi *et al*. identified and analyzed crucial determinants of consumer reversing behavior using partial least square- structural equation modelling and artificial neural network. This study demonstrated the advantage of artificial neural network over conventional methods in terms of capturing the non-linear relationships (Najmi *et al*. 2021).

Also, some studies have compared the logistic regression model with artificial neural networks in estimating overall satisfaction in the literature. For example, Tsaur *et al*. have applied artificial neural networks and logistic regression to measure the importance scores of services in nine international hotels. In their study, they concluded that artificial neural networks perform better than logistic regression (Tsaur *et al*. 2002). Cong *et al*. estimated the parameters such as acoustic and semantic features, emotional instability features, speech rhythm, and verbal assessments to measure the customer satisfaction score (Cong *et al*. 2016). Yau and Tang (2018) estimated the customer satisfaction level in self-service technology adopted in airports by using regression tree and Artificial Neural Networks. Artificial Neural Networks validated by 10-fold cross validation is found to be the best among the models. Kalinić *et al*. (2019) developed a predictive model of customer satisfaction related to mobile commerce. Since

conventional statistical techniques, such as multiple regression analysis, are used for the prediction of consumer satisfaction and typically examine only linear relationships among variables, they used Artificial Neural Networks for modelling complex relationships. Bekiros *et al*. (2019) proposed method for customer satisfaction prediction in the shipping industry. The study revealed the most effective optimization methods through employing artificial intelligence approaches. Wang at al. (2019) proposed an automated machine learning approach to model overall product delivery satisfaction under limited resources. Araç and Gürhanlı (2020) used artificial neural networks for customer satisfaction applications by establishing nonlinear equations. Subroto and Christianis (2021) used Classification and Regression Tree, Random Forest, Logistic Regression and Artificial Neural Network, and Multi-Layer Perceptron Models to make prediction's classification through attributes and topics from customer review.

Customer service perceptions generally contain uncertainty. Applying the Likert scale to represent customer perceptions based on linguistic assessments does not address this uncertainty. Human perceptions and attitudes are subjective and uncertain. In addition, differences in individual perception and personality affect this uncertainty. The traditional Likert scale assumes that distance between the consecutive scale-point is constant, but in reality, there is no crisp boundary among the scale values (Tóth *et al*. 2020). To address the information lost problem when applying the Likert method, increasing the scale points on a Likert scale or apply Likert scale in two stage was recommended. Although the recommended solutions have some advantages, difficulty in application emerged because they tired the survey participants. To overcome these problems, new alternative Likert scale based on fuzzy sets theory was proposed by Lin (2017), Bahadır (2017), Biyan and Bircan (2018). In the literature, there are studies to solve this situation by expressing the Likert scale with fuzzy numbers and thus obtaining successful results. (Tóth *et al*. 2019) introduced fuzzy number-based methodology that adds properties to Likert scales to model human judgment in more precise and reliable method. The study states that it is possible to map the non-linear relationship between quality attributes and customer satisfaction.

There are studies that shows that integrating fuzzy logic with conventional techniques improves the

model prediction. Deng and Pei achieved successful results by integrating the technique of artificial neural networks with fuzzy logic (Deng *et al*. 2009). Lin integrated multiple regression with fuzzy set qualitative comparative analysis to explore the relationship among service range, motivation to ride, ride convenience, service satisfaction, satisfaction with facilities, and intention to re-ride Lin (2017). Hendalianpour and Razmi (2017) applied Fuzzy Neural Net- work for the customer's satisfaction measurement. The proposed model was successfully implemented based on both qualitative and quantitative inputs. Wahyudi at al. (2018) integrated fuzzy and survival analysis to predict Customer Satisfaction.

Although there are many studies using various methods to predict customer satisfaction, the number of studies predicting customer satisfaction using deep learning and fuzzy logic is rather limited. The study aims to analyze the performance of fuzzy deep learning in predicting customer satisfaction, regardless of the existing data structure. To test the techniques regardless of the data structure artificial data was preferred. Testing technique with artificial data is an important technique for creating reproducible experimental findings (Kennedy, Delany, Mac Nomee 2011). In addition, the performances of the techniques were investigated by converting the 5-point Likert scale to a fuzzy scale, with defuzzification and without the defuzzification stage. The changing behavior of the researched techniques depending on the amount of data were examined. Artificial datasets containing 100, 200, 300, 400, 500, 600, 700, 800, 900, and 1000 samples were produced in the format of a 5-point Likert scale. These data sets were converted to fuzzy form using triangular or trapezoid fuzzy numbers.

2. Material and Method

The fuzzification of the Likert scale has made a great contribution to wider body of research, especially those involving surveys. The information loss due to the nature of the Likert scale and the information discrepancy caused by the closed response form were overcome with the help of fuzzy sets. Since the concept of consensus is applied on the fuzzy Likert scale, it has been observed that it can provide a more accurate measurement result than the traditional Likert scale (Li 2013).

In this study, the artificial data produced in the 5-point Likert-type scale and fuzzy Likert-type scale format were first processed by basic classification techniques such as logistic regression. In estimating customer satisfaction, the logistic regression model is widely used in the literature, examples of which were given in introduction part. In addition, studies comparing logistic regression with artificial neural networks techniques that form the basis of deep learning are frequently encountered in literature. Based on these studies, a pilot study was implemented using the logistic regression method. The dataset containing a fuzzy Likert type scale was included in the analysis as a minimum, maximum, and peak points separately as seen in the literature. Then, the logistic regression model that produced three separate results series was combined with the defuzzification method in a single result. Then, in contrast to the literature, it is aimed to express this minimum, maximum and peak points as a single data set, to reach a single logistic regression model and to produce a single result series.

2.1. Production of Artificial Data

Real world data are often used to investigate artificial intelligence techniques. However, there are some disadvantages when using real data to test techniques. For example, it is difficult to obtain real data in many areas for various reasons such as budget, technical, or ethics. In addition, the limited use of real data is another disadvantage. In other words, data sets do not contain purposeful models, or it requires particular preparation to find the pattern inside. Obtaining experimental results by producing artificial data or in other words synthetic data can overcome these disadvantages (Peng and Hanke 2016).

The use of artificial data allows for the identification and control of variability that is expected to occur but has not yet occurred in practice. The ability to control parameters enables a comprehensive investigation of the performance of classification models under different conditions. The test technique with artificial data is an important technique for generating reproducible experimental findings (Kennedy *et al*. 2011).

In the study, a function written in Python was used to create the artificial data set. The "truncnorm" function in the Python "scipy.stats" library was used to generate 5-point Likert scale data that fit the normal distribution. Using the Truncnorm function,

random numbers are generated in accordance with the normal distribution between -1 and 1. The sum and average of these numbers are produced as 0. The numbers produced were then converted into a range of 1 to 5.

The aim of first step was to conduct experimental studies with uncomplicated data sets where relations between variables are symmetrical. For the first trial independent variables have been obtained by using the Equation (1), where Y is dependent variable representing satisfaction, $X_1, X_2, \ldots X_{10}$ are independent variables. Since it is aimed to create a balanced class for the dependent variable, the "three" value on the Likert scale has been used as the threshold value.

$$Y = X_1 + X_2 + \cdots + X_{10} \tag{1}$$

The aim of second step was to conduct experimental studies with more complicated but balanced data sets and to observe the difference between simple and complex models. At this step an experimental study has been conducted with the data set created using the Equation (2).

$$Y = X_1^2 + 2 * X_2 + X_3 + 4 * X_4 + X_5 + X_6 + 5 * X_7 +$$

$$X_8 + X_9 + X_{10} \tag{2}$$

The aim of third step was to conduct experimental studies with more complicated unbalanced data sets and to observe the difference balance and unbalanced datasets. The number 3.5, which is the upper value of the fuzzy number, was used instead of the three values in the Likert scale.

In order to observe the classification performance of the models, 30 different artificial datasets were created for each step. To observe changing behaviors of the researched techniques depending on the amount data, different sub-clusters with 100, 200, 300, 400, 500, 600, 700, 800, 900, and 1000 samples were created for each dataset. The 1080 trials were performed for each observed model.
After the network is trained with the training set, a test dataset of samples never seen before by the network must be used to measure the quality of the model. However, a validation set is needed to select the correct values of the hyper parameters, such as learning rate, number of epochs etc. (Heaton, 2015). That is the reason why the network performance is evaluated with validation and test datasets. In the literature, different ratios are used while creating training, validation and test datasets, and the most used ratio is 70/30 (Islam and Raj, 2017). Using this

ration obtained datasets were divided into training, validation, and test sets as 70% training and 30% test sets. Then 30% of the training set was used to create the validation data set.

2.2. Fuzzification of Datasets

Training and Test datasets were converted to triangular and trapezoidal fuzzy numbers using fuzzy number functions. Triangular fuzzy numbers are given in Sreekumar and Mahapatra (2015) studies for triangular Likert scale, trapezoidal fuzzy numbers given in Güner and Çomak (2014) studies for trapezoid Likert scale were used. The fuzzy equivalents of the Likert scale are given in Figure 1 and Figure 2. Two separate sequences have been defined for fuzzy triangular numbers and fuzzy trapezoid numbers.
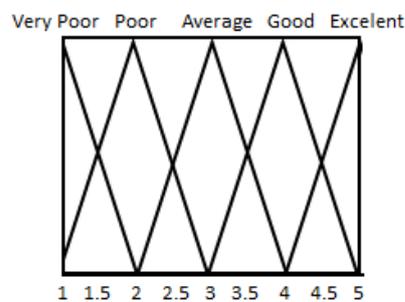
Figure 1. Triangular Fuzzy Numbers
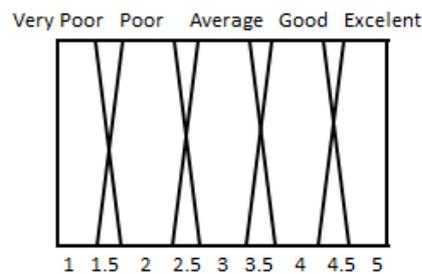
Figure 2. Trapezoid Fuzzy Numbers

2.3. Pilot Study with Logistic Regression

One of the most classical methods in classification problems is the logistic regression method. The predictive success of logistic regression largely depends on dependent variables and the structure of the data and is affected by fewer factors than the deep learning model. Different function settings such as regulation parameters are available in the Keras library to improve the success of the logistic regression model. However, since the purpose of the pilot study is to observe whether there will be an increase in performance if the dataset is

converted to a fuzzy set, the testing was done using the default settings of the function. This first model was labeled as Logistics Regression Model with Likert Type Data (LR-LTD). The flow chart of model is given in Figure 3.
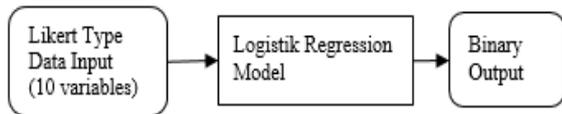
Figure 3. Logistics Regression Model with Likert Type Data (LR-LTD)

In the fuzzy logistic regression model, a separate data set was obtained for the minimum value, the peak value, and the maximum value, which represents the triangular number. Thus, estimated values were created for each data set. In other words, the output is also produced as a fuzzy triangular number. Then, defuzzification was applied using the center of gravity method and a single output value was obtained. The fuzzy logistic regression model was tested with data containing triangular numbers as well as with the data set containing trapezoid fuzzy numbers. Similarly, a separate data set was obtained for its minimum value, peak value, and maximum value. Thus, estimated values were created for each data set and defuzzification was applied. These two models were labeled as Triangular and Trapezoid Fuzzy Logistic Regression Model (FLR-1 and FLR-2). The flow chart of model is given in Figure 4.
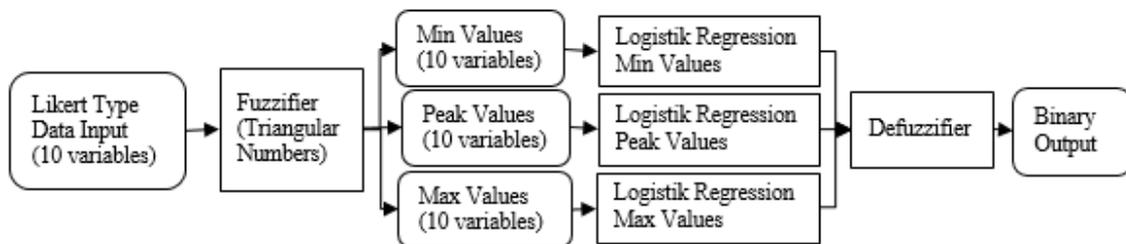
Figure 4. Triangular Fuzzy Logistic Regression Model (FLR-1)

In the literature, separate data sets are created for each element of triangular and trapezoid numbers, and separate models are analyzed and combined by the defuzzification method. In this study, the minimum value, the peak value, and the maximum value of the triangular fuzzy numbers were inserted in a single dataset. Logistic regression was applied with this fuzzy data set and binary estimates were created. Since the output is not a fuzzy number, defuzzification is not required. These two models were labeled as Logistic Regression Model with Fuzzy Likert Type Data (LR-FLTD1 and LR-FLTD2). The flow chart of model is given in Figure 5.
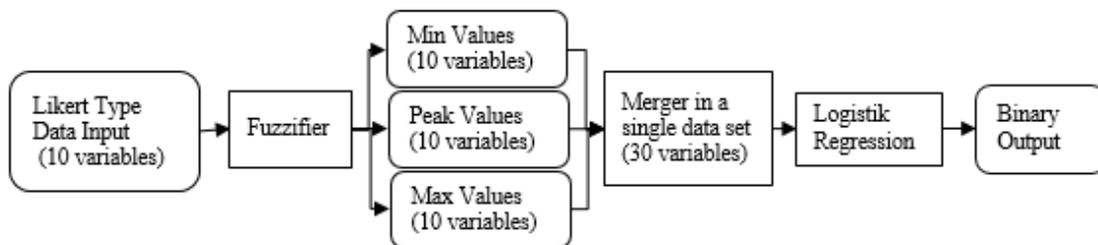
Figure 5. Logistic Regression Model with Triangular Fuzzy Likert Type Data (LR-FLTD1)

2.4. Deep Learning Models

The performance of the Deep Learning model is sensitive to the choice of net- work architecture (Güner and Çomak 2014). The combination of different layer number and node number in each layer forms different architectures of the deep network. In order to find the best architecture for datasets experiments were carried out with a combination of different layer numbers, node numbers, and activation functions. A random set of architectures were produced for these trials. There are ten independent variables in the input layer and one dependent variable in the output layer. Starting from the input layer data is analyzed and summarized appearing as a single variable in the output layer (Goodfellow *et al*. 2016). Therefore, values from one to nine values were chosen for the number of nodes in the hidden layer. "Sigmoid" and "Tanh" functions which are widely used in artificial

neural networks were selected as the activation functions. In addition, due to the advantages, it provides in deep learning architectures the "Relu" activation function has been chosen. Also, since the fuzzy numbers are used the "Softplus" activation function which is the smooth version of the relu activation function has been chosen (Patterson and Gibson 2017).

The sequential function in keras.models library is used to create this architecture. Relu activation function is used for hidden layers and the Sigmoid activation function is used to produce 1 or 0 results for the output layer. Random assignment of the initial values of the weights was made in accordance with the uniform distribution.

Number of iterations was set to value 1000, but to prevent over learning, the training will be stopped at the optimal values with the early termination technique. A validation dataset was used to determine the criteria of early termination (Albon 2018). In such a testing phase, more realistic results were obtained by testing with data that the network never saw. K-fold verification has been applied to ensure that the validation dataset has the same properties as the rest of the data (Raschka and Mirjalili 2017). However, since the artificial data produced is homogeneous, it did not make a difference. On the contrary, this has led to undesired results for the experimental study since it increases the calculation time. For this reason, k-fold validation has not been applied for the whole study. Another technique used to prevent overfitting is dropout regulation (Hahn and Choi 2020). It did not contribute to the classification performance of the architecture due to the characteristics of the artificial data produced. There are many approaches in the literature to improve the performance of deep learning architecture (Srivastava *et al*. 2014). However, the aim of the study is to observe the change in performance when all the architectures are constructed similarly, and the Likert scale is transformed into a fuzzy scale rather than finding the best classification architecture.

In order to find the best architecture, each architecture was trained using 30 datasets, and the architecture with the highest success score was chosen. According to the results obtained the architecture with the most successful results and the shortest time is selected. While the Tanh function works best in single-layer artificial neural networks it has been observed that the performance decreases as the architect deepens. Sigmoid function increased the classification success up to three layers, but it was observed to be the slowest function among the selected functions.

When this result is compared with the literature the reason is explained as the learning process slows down as the output values get closer to the values of 0 and 1 and the update amounts of the weights will decrease. Therefore, although the Sigmoid function is used effectively in artificial neural networks it is not preferred much in deep networks (Goodfellow *et al*. 2016). Although the soft plus function produces the same results as the Relu function the calculation time is higher. For this reason, the "Relu" activation function was chosen for the deep learning architecture to be applied to the Likert type dataset. Considering the relu activation function the two-layer network with five and two neurons respectively was selected. The concept of deep learning is based on artificial neural net- works and artificial neural network with more than one hidden layer is defined as a deep network (Deng and Yu 2014). For this reason, the architecture chosen in the study has been accepted as Deep learning architecture. This first deep model was labeled as Deep Learning Network with Likert Type Data (DL-LTD). The flow chart of model is given in Figure 6.
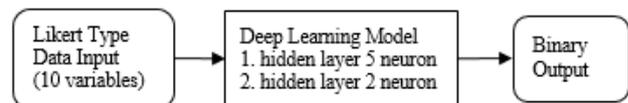
Figure 6. Deep Learning Network with Likert Type Data (DL-LTD).

The architecture designed for the Deep Learning model is also used for fuzzy deep learning. As in the Fuzzy Logistic Regression model, a separate data set was obtained for the minimum value the peak value and the maximum value which represents the triangular number. Thus, estimated values were also created for each data set. Then defuzzification was applied using the center of gravity method and a single output value was obtained. These two deep models were labeled as Triangular and Trapezoid Fuzzy Deep Learning Model (FDL-1 and FDL-2). The flow chart of model is given in Figure 7.
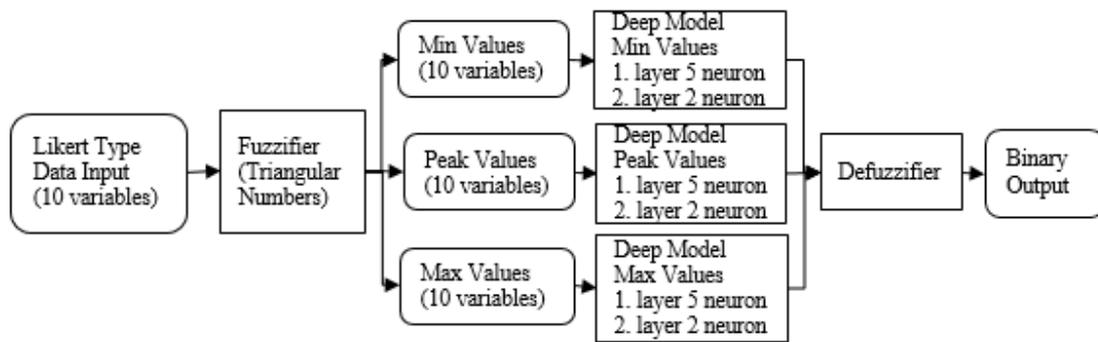
Figure 7. Triangular Fuzzy Deep Learning Model (FDL-1)

In this study unlike the literature instead of obtaining a separate data set for the minimum, peak, and maximum values of the fuzzy number these values were combined in a single data set, and the analyzes were done accordingly. Ishibuchi and Nii (1998) have shown that fuzzifying neural networks are possible by extending inputs and weights to fuzzy numbers. In this study, the effect of the network on the success of the network by converting the data into fuzzy numbers was examined on the deep learning model.

When the parameter vector is expressed in a triangular fuzzy number, the number of input neurons in the network is appeared to be tripled. Therefore, in order to find the best architecture for this dataset, experiments were made with the combination of activation functions, different layer number, and node number. The results of the experimental study for architectural selection were obtained in the same way. Looking at the results for the first layer, the success of Sigmoid and Tanh functions decreases as the architecture deepens. Therefore, only the combination of the number of layers and the number of nodes of Relu, and Softplus functions are examined for the three-layer network. When the results for the three-layer network are analyzed, since the success of the Relu function decreased and the success of the Softplus function increased, only the combination of the number of layers and the number of nodes of the

Softplus functions were examined in the next experiment. When deciding on combinations of node numbers for the four-layer architecture, only combinations with success above the average are taken into account. Since the performance decrease for the five-layer architecture is observed, no trials have been made for the deeper layer. According to these results, the most successful results are four-layered, giving 25, 20, 15, and 10 neurons respectively. When the calculation periods are examined, four-layer architecture has produced results in the shortest time. Results of selected architecture is given in Table 1.

Table 1. Architecture selection for triangular fuzzy model

Number of Layers	Number of Neurons	Average Success	Training Success	Test Success
Layer Success		0,984	0,997	0,970
4	[25,20,15,10]	0,988	0,998	0,978
4	[25,20,15,5]	0,983	0,997	0,970
4	[25,20,10,5]	0,980	0,996	0,965
4	[25,15,10,5]	0,986	0,998	0,974
4	[20,15,10,5]	0,980	0,995	0,964

This forth deep model was labeled as Deep Learning Model with Fuzzy Triangular Likert Type Data (DL-FLTD1). The flow chart of model is given in Figure 8.
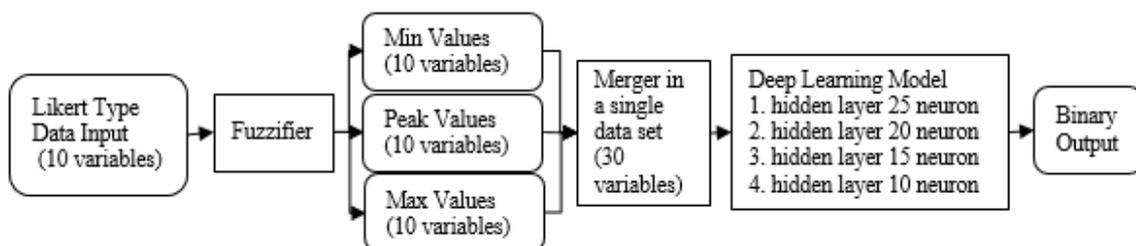
Figure 8. Deep Learning Model with Fuzzy Triangular Likert Type Data (DL-FLTD1)

When the parameter vector is expressed by trapezoid number, the number of input neurons in the network is quadrupled. Therefore, in order to find the best architecture for this dataset, experiments were made with a combination of different layer number and node number. Since the structure of the triangular fuzzy and trapezoid fuzzy data is similar, the experience obtained in the previous stage was used for model selection. Only the Softplus activation function has been attempted since it produces the best result for triangular fuzzy data. According to the results obtained, the most successful results were six-layered network with 35, 30, 25, 20, 15, and 10 neurons. When the calculation periods are examined, it was the five-layered architecture that produced results in the shortest time. Since the priority criterion was prioritized and the calculation period of the six-layer architecture was close to the five-layer architecture, it was decided to continue with the six-layer architecture. Results of selected architecture is given in Table 2.

Table 2. Architecture selection for trapezoid fuzzy model

Number of Layers	Number of Neurons	Average Success	Training Success	Test Success
Layer Success		**0,974**	**0,994**	**0,955**
6	**[35, 30, 25,20,15,10]**	**0,989**	0,996	0,982
6	[35, 30, 25,20,15,5]	0,977	0,997	0,957
6	[35, 30, 25,20,10,5]	0,969	0,992	0,946
6	[35, 30, 25,15,10,5]	0,968	0,991	0,945
6	[35, 30, 20,15,10,5]	0,969	0,991	0,946

This fifth deep model was labeled as Deep Learning Model with Fuzzy Trapezoid Likert Type Data (DL-FLTD2).

3. The Experimental Results

3.1. Logistic Regression Results

In order to observe the classification performance of Logistic Regression, 30 different artificial datasets containing 100 samples were created. The reason for using 30 different data sets is to show that the results obtained are not accidental. Logistic regression models were tested using these datasets and performance criteria of models were recorded. The performance values of the logistic regression classification using 30 different Likert type datasets each containing 100 samples were used to obtain average performance values of a 100-sample dataset. The same procedures were performed for 200, 300, 400, 500, 600, 700, 800, 900, 1000 samples and average performance values were obtained from those 300 different datasets.

In the classification made with 200-1000 data sets, it is seen that the test success increases as the sample size increases. In the classification made with data sets containing 1000 samples it was observed that proposed method classified samples with accuracy of 97.6% for triangular fuzzy numbers and 97.9% for trapezoid fuzzy numbers, where is fuzzy logistic regression classified samples with accuracy of 92.2%.

At second step where relations between variables are more complicated the classification accuracy of proposed method is lower than at first step, but still higher that classification accuracy of fuzzy logistic regression. In the classification made with data sets containing 1000 samples it was observed that proposed method classified samples with accuracy of 94.2% for triangular fuzzy numbers and 94.6% for trapezoid fuzzy numbers, where is fuzzy logistic regression classified samples with accuracy of 86.8%. According to the results at second step where the same models were applied to complicated data the proposed method is still the best among the others. Accuracy Ratios of Logistic Regression Models for step 1 and step 2 is given in Figure 9. Here the classification success of all models increases in direct proportion to the number of data. Since logistic regression and fuzzy logistic regression models give close values to each other it looks like a single curve in the graph. Logistic regression applied with fuzzy Likert type data proposed in the study produced more successful results for each data set. It provides a great advantage especially in cases where the data set is small.
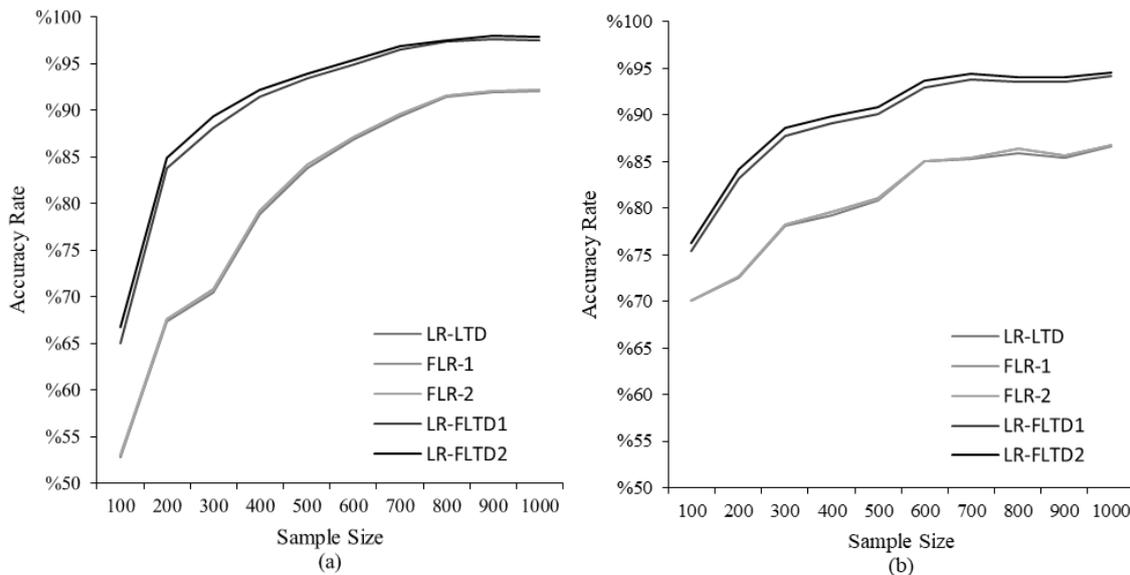
Figure 9. Accuracy Rates Graph of Logistic Regression Models (a) Step 1 and (b) Step 2

According to the results obtained the conversion of Likert type scale to the fuzzy scale in proposed way increases logistic regression classification success. Since the experimental study with logistic regression reaches the intended result, it is aimed to evaluate the test results by applying the same technique to the deep learning model.

3.2. Classification Results of Deep Learning Models

At first step in order to observe the Classification performance of Deep Learning Models, an artificial dataset containing 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000 samples used in logistic regression models test was used. The classification success of all models increases in direct proportion to the number of data. The fuzzy Likert type data proposed in the study produced successful results for each data set applied. However, the results of the study are very close to the approach in the literature and produced the same results especially for 500 samples and more. Since the complex equation is not used in the creation of data sets, there is no increase in the complexity of the relationships in the data even if the number of data increases. Therefore, the increase in the number of data makes it easier for techniques to solve these relationships.

At second step where relations between variables are more complicated the classification accuracy of proposed method is lower than at first step and it is giving almost the same results with fuzzy deep learning approach. In the classification made with data sets containing 1000 samples it was observed that proposed method classified samples with accuracy of 96.8% for triangular fuzzy numbers and 96.4% for trapezoid fuzzy numbers, where is fuzzy deep network classified samples with accuracy of 97.0%. Deep network that uses Likert Type Data without converting it to fuzzy numbers which was labelled as DL-LTD earlier classified samples with accuracy of 95.2%, which is lower than result obtained with fuzzification.

According to the results at second step where the same models were applied to complicated data the proposed method classifies data with the almost the same accuracy rate as the other observed fuzzy methods, but higher than deep network without fuzzification. The average values of accuracy rates for training and testing obtained as a result of the classification made for all data sets is given in Figure 10.
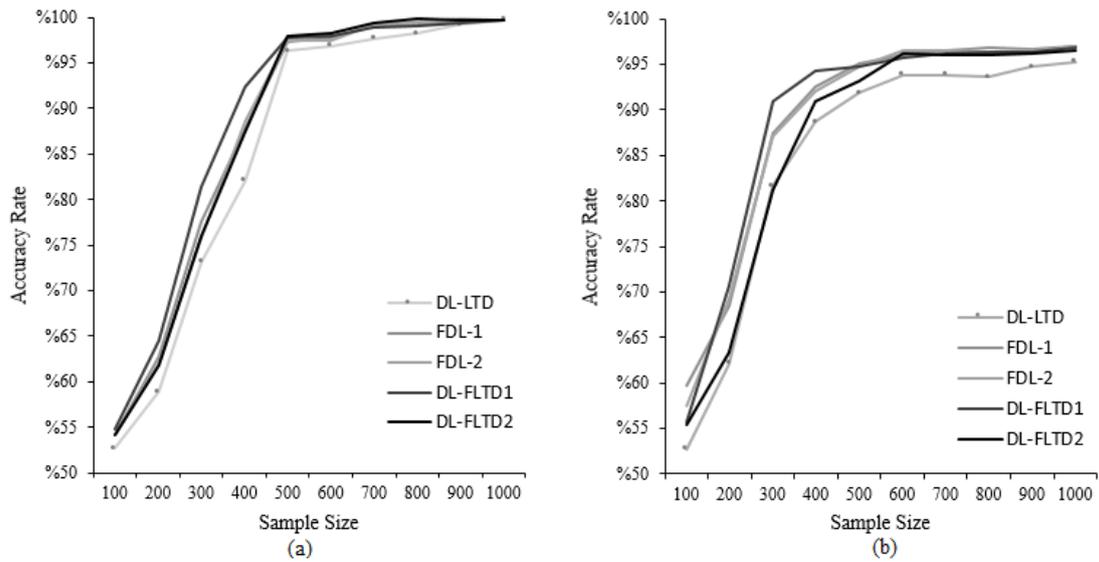
Figure 10. Accuracy Rates Graph of Deep Learning Models (a) Step 1 and (b) Step 2
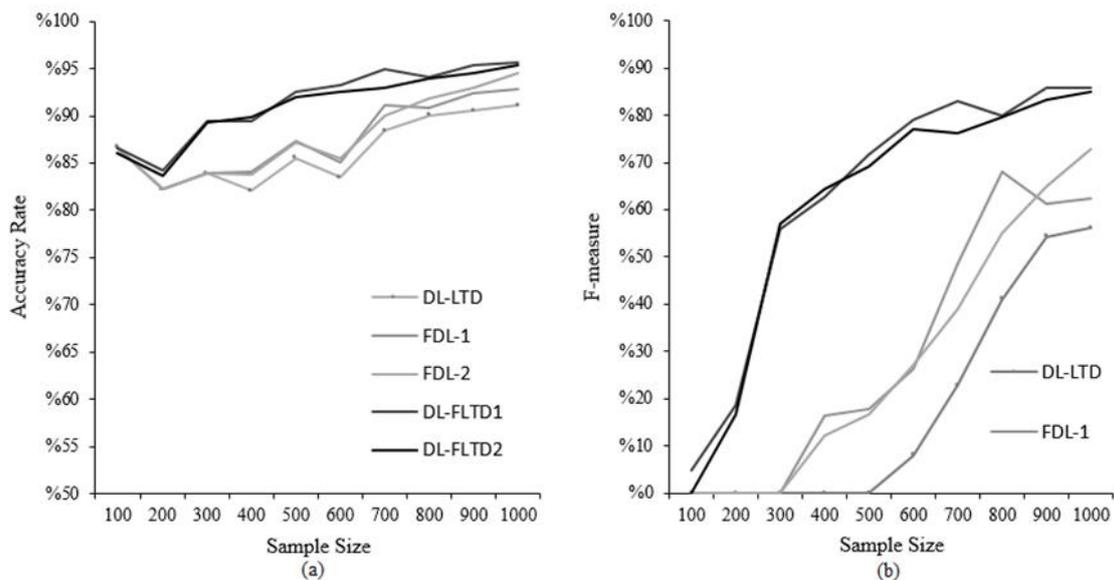
Figure 11. Accuracy Rates (Step 3) and (b) F-measure Graph of Deep Learning Model (Step 3)

At third step experiments were performed with more complicated unbalanced data sets and to observe the difference in behavior of models when applied to balance and unbalanced datasets. In the classification made with data sets containing 500 samples it was observed that proposed method classified samples with accuracy of 92.5% for triangular fuzzy numbers and 92.0% for trapezoid fuzzy numbers, where is fuzzy deep network classified samples with accuracy of 87.3%. Deep network that uses Likert Type Data without converting it to fuzzy numbers which was labelled as DL-LTD earlier classified samples with accuracy of 85.5%, which is lower than result obtained with fuzzification. According to the results at third step fuzzification affect positively classification accuracy rate, but the proposed methods is observer to be the best in unbalanced dataset. Due to increase of

variable numbers the proposed method uses larger dataset, which gives an opportunity to solve relation in data set easier. When evaluating the classification success of methods in the unbalanced data set F-measure should be checked (Mahani and Baba Ali 2020). The average values of accuracy rates and F-measure for training and testing obtained as a result of the classification made for all data sets is given in Figure 11. According to the results the performance of proposed method is better than other compared methods.

The proposed not only improves the predictive success of the model but also contributes to the calculation speed. In the approach in the literature, a model is established for the minimum value, the peak value and the maximum value, which represent the triangular number, and separate

estimated values are created, and are converted to the crisp number using the defuzzification technique. The technique proposed in this study produces results in a shorter period of time since calculations are made for a single model. The graphic of obtained results is given in Figure 12.
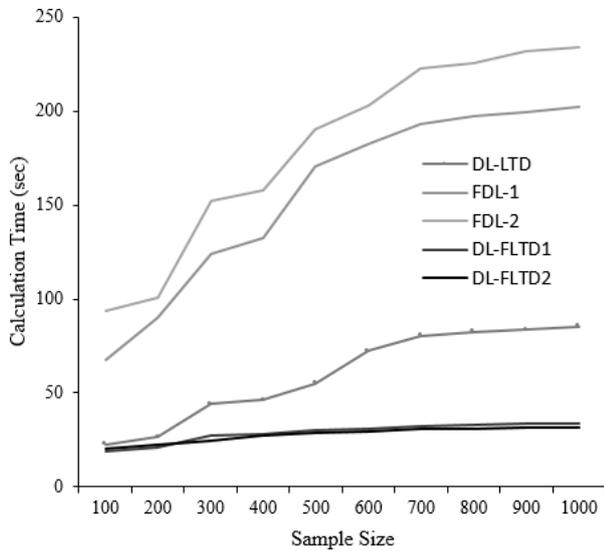
Figure 12. Calculation Time Graph of Deep Learning Models (sec)

According to the results obtained, the conversion of Likert type scale to a fuzzy scale affects the classification speed of the deep learning model positively. While the calculation time for the cluster containing 100 samples was 22 seconds on average with the Likert type data, the average reached 85 seconds for the cluster containing 1000 samples. While the calculation time for the cluster containing 100 samples of fuzzy deep learning models in the literature was 67 and 93 seconds on average, it reached 202 and 233 seconds for the cluster containing 1000 samples. When using fuzzy Likert type data for the training of the deep learning model, the average time of calculation for the cluster containing 100 samples was 19 and 20 seconds, while the average for the cluster containing 1000 samples reached 34 and 32 seconds. It is concluded that the proposed method is the most accurate and fast classification method.

4. Discussion

When classified using the Likert type data logistic regression model with fuzzy Likert type data, it has been observed that the process of turning the data into a fuzzy dataset in proposed way increases success the model. Also, it was observed that the

test success increased with an increasing number of samples in dataset. Logistic regression applied with the fuzzy Likert type data suggested in the study, each data set produced more successful results. It provides a great advantage especially in cases where the data set is small. High accuracy rate of classification model in small dataset is valuable result, because in some cases find large dataset is quite difficult (Feng, Zhou, & Dong, 2019).

In the approach in the literature, a separate logistic regression model was established for the minimum value, the peak value, and the maximum value, which represent the triangular number, and separate estimated values were created. So the output is also produced as a fuzzy triangle. It is then converted to a precise number using the defuzzification technique. The reason why fuzzy learning is much more effective than traditional learning prevents the problem of inability to exit the lost function from the saddle point, thus increasing the convergence speed and minimizing the error (El Hatri and Baumhidi 2018).

After the fuzzy Likert scale was found to improve logistic regression performance, the deep learning architecture was tested using the fuzzy Likert scale. When there is a difference in the number of parameters of Likert type data and fuzzy Likert type data, the architectural selection procedure has been followed for each architecture. In this architecture, functions such as Sigmoid, Tanh, Relu, and Soft Plus have been tried for activation functions of hidden layers and Soft Plus function has been selected. In literature there several studies state that the softplus function outperforms Sigmoid, Relu and Tanh functions (Zheng *et al*. 2015). In order to choose better architecture, many combinations of hyper parameters should be tried besides activation, neuron, and layer numbers (Wright, Manic 2010). In this way, hyper parameter adjustment and the choice of the ideal architecture is a job that requires a long time that requires experience and ability. If the person who designs the architecture worked in the same type of application and architecture, it can reach a faster result by determining the right strategy (Goodfellow, 2016). However, since the purpose of the study is not to find the best architecture to predict the result, the similarity of the experimental environment for the models was considered by looking at the most basic hyper parameters for the choice of architecture. In the study, it is aimed to examine whether there are

improvements if the models use different scales under similar conditions.

At first step of experiment, it was observed that the classification success of all models increased in direct proportion to the amount of data. Compared to Logistic Regression Deep Network give worse results when sample size less than 500 for this experiment. As states in literature Deep Networks with small datasets commonly shows worse performance than shallow architectures (Feng *et al*. 2019). At second step where relations between variables are more complicated the classification accuracy of proposed method is lower than at first step and it gives almost the same results with fuzzy deep learning approach. According to the results at second step where the same models were applied to complicated data the proposed method classifies data with the almost the same accuracy rate as the other observed fuzzy methods, but higher than deep network without fuzzification.

At third step experiments were performed with more complicated unbalanced data sets and to observe the difference in behavior of models when applied to balance and unbalanced datasets. According to the results at third step fuzzification affect positively classification accuracy rate, but the proposed methods is observer to be the best in unbalanced dataset. Due to increase of variable numbers the proposed method uses larger dataset, which gives an opportunity to solve relation in data set easier.

The proposed not only improves the predictive success of the model but also contributes to the calculation speed. In the approach in the literature, a model is established for the minimum value, the peak value and the maximum value, which represent the triangular number, and separate estimated values are created, and are converted to the crisp number using the defuzzification technique. The technique proposed in this study produces results in a shorter period of time since calculations are made for a single model. It is concluded that the proposed method is the most accurate and fast classification method.

5. Conclusion

As a result of the integration of deep learning and fuzzy logic techniques, the performance of the models was tested on the satisfaction estimation problem. Likert scale, which is widely used in satisfaction estimation, has been converted to a fuzzy Likert scale using fuzzy numbers and used in the analysis. In the experimental study, artificial data was produced to comprehensively investigate the performance of classification models under different conditions. Since success of the deep learning architecture depends on many parameters, artificial data produced was tested with the logistic regression technique, which is the traditional technique. After concluding that transforming Likert type data into fuzzy data increases the success of the model and the data is suitable for testing the deep learning model, the design phase of the deep learning architecture has been started.

At first step of experiment, it was observed that the classification success of all models increased in direct proportion to the amount of data. However, the results of the study are very close to the approach in the literature and produced the same results especially for 500 samples and more.

At second step where relations between variables are more complicated the classification accuracy of proposed method is lower than at first step and it gives almost the same results with fuzzy deep learning approach. At third step experiments were performed with more complicated unbalanced data sets and to observe the difference in behavior of models when applied to balance and unbalanced datasets. According to the results at third step fuzzification affect positively classification accuracy rate, but the proposed methods is observer to be the best in unbalanced dataset. Due to increase of variable numbers the proposed method uses larger dataset, which gives an opportunity to solve relation in data set easier. To evaluate the classification success of methods in the unbalanced data set F-measure values were analyzed. According to F-measure values the results performance of proposed method is better than other com- pared methods.

The proposed not only improves the predictive success of the model but also contributes to the calculation speed. The technique proposed in this study produces results in a shorter period of time since calculations are made for a single model. It is concluded that the proposed method is the most accurate and fast classification method. Calculation time is not very important for this problem. However, there are situations in which computing time is very important among deep learning

applications. In such cases, it is thought that the proposed technique will increase in importance.

In future studies, comparisons can be made by using a 7-point Likert-Type scale instead of a 5-point Likert-Type Scale. After selecting different training and testing rates, experiments should be made in the space of possibilities consisting of different layers and number of neurons, to find the best architecture.

Conflicts Of Interest

No conflict of interest was declared by the authors. We would like to thank Alea Laidlaw for proofreading the manuscript.

References

Albon, C., 2018. Machine Learning with Python Cookbook, USA: O'Reilly Media, Inc, 180-186.

Araç, Y. E., Gürhanlı, A., 2020. Yapay sinir ağını kullanarak müşteri memnuniyeti analizi. *Dicle Üniversitesi Mühendislik Fakültesi Mühendislik Dergisi*, 11(**1**), 39-55.

Bahadır, E. (2017). Bulanık Mantık Yaklaşımının Eğitim Çalışmalarında Kullanılmasının Alan Yazın Işığında Değerlendirilmesi. *Uluslararası Sosyal ve Eğitim Bilimleri Dergisi*, 4(**7**), 28-42.

Bekiros, S., Loukeris, N., Matsatsinis, N., Bezzina, F., 2019. Customer satisfaction prediction in the shipping industry with hybrid meta-heuristic approaches. *Computational Economics,* 54(**2**), 647-667.

Biyan, M., & Bircan, H. (2018). Daha önce geliştirilmiş likert tipi bir ölçek ile tip-1 ve tip-2 bulanık likert ölçeğinin sonuçlarının karşılaştırılması. *Uluslararası İktisadi ve İdari İncelemeler Dergisi*, 369-382.

Cong, P., Wang, C., Ren, Z., Wang, H., Wang, Y., Feng, J.,N. 2016. Unsatisfied customer call detection with deep learning, *In 2016 10th International Symposium on Chinese Spoken Language Processing (ISCSLP)*, Tianjin, China.

Deng, W. J., Pei, W., 2009. Fuzzy neural based importance-performance analysis for determining critical service attributes. *Expert Systems with Applications*, 36(**2**), 3774-3784.

Deng, L., Yu, D., 2014. Deep learning: methods and applications. *Foundations and Trends in Signal Processing*, 7(**3-4**), 197-387.

El Hatri, C., Boumhidi, J., 2018. Fuzzy deep learning based urban traffic incident detection. *Cognitive Systems Research*, **50**, 206-213.

Feng, S., Zhou, H., Dong, H., 2019. Using deep neural network with small dataset to predict material defects. *Materials & Design*, **162**,300-310.

Güner, N., Çomak, E., 2014. Lise öğrencilerinin matematik dersine yönelik tutumlarının bulanık mantık yöntemi ile incelenmesi. *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 20(**5**), 189-196.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep learning, USA: MIT Press, 130-135

Hahn, S., Choi, H., 2020. Understanding dropout as an optimization trick. *Neurocomputing*, **398**, 64-70.

Hendalianpour, A., Razmi, J., 2017. Customer satisfaction measurement using fuzzy neural network. *Decision Science Letters*, 6(**2**), 193-206.

Heaton, J. (2015). Artificial Intelligence for Humans, Volume 3: Neural Networks and Deep Learning. Heaton Research Inc, Chesterfield, ABD, 30-55.

Hinton, G., Osindero, S., Teh, Y. W., 2006. A fast-learning algorithm for deep belief nets. *Neural computation*, 18(**7**), 1527-1554.

Ishibuchi, H., Nii, M., 1998. Fuzzification of input vectors for improving the generalization ability of neural networks. *In 1998 IEEE International Conference on Fuzzy Systems Proceedings. IEEE World Congress on Computational Intelligence*, Anchorage, AK, USA, 1153-1158.

Islam, K. K., and R. G. Raj, 2017. "Real-time (vision-based) road sign recognition using an artificial neural network." *Sensors*, 17(**4**), 853.

Jahandideh, S., Asefzadeh, S., Jahandideh, M., Asadabadi, E. B., Jafari, A., 2013. The comparison of methods for measuring quality of hospital services by using neural networks: A case study in Iran. International *Journal of Healthcare Management*, 6(**1**), 45-50.

Kappor, R., Walters, S. P., Al-Aswad, L. A., 2018. The current state of artificial intelligence in ophthalmology. *Survey of Ophthalmology*, 64(**2**), 233-240.

Kalinić, Z., Marinković, V., Djordjevic, A., Liebana-Cabanillas, F., 2019. What drives customer satisfaction and word of mouth in mobile commerce services? A UTAUT2-based analytical approach. *Journal of Enterprise Information Management*, 33(**1**), 71-94.

Kennedy, K., Delany, S. J., Mac Namee, B., 2011. A Framework for Generating Data to Simulate

Application Scoring. *In Credit Scoring and Credit Control XII*, Conference Proceedings, Edinburg.

Li, Q, 2013. A novel Likert scale based on fuzzy sets theory. *Expert Systems with Applications*, 40(**5**), 1609-1618.

Lin, Y. S., 2017. Causal complexity for passengers' intentions to re-ride. *Quality & Quantity*, 51(**5**), 1925-1937.

Mahani, A., Baba Ali, A. R., 2020. Classification problem in imbalanced datasets*.* In Recent Trends in Computational Intelligence, eds A. Sadollah, and T. Sinha, London: IntechOpen Press, 1-23.

Najmi, A., Kanapathy, K., Aziz, A. A., 2021. Understanding consumer participation in managing ICT waste: Findings from two-staged Structural Equation Modeling-Artificial Neural Network approach. *Environmental Science and Pollution Research*, 28(**12**), 14782–14796.

Subroto, A., Christianis, M., 2021. Rating prediction of peer-to-peer accommodation through attributes and topics from customer review. *Journal of Big Data*, 8(**1**), 1-29.

Tabrizi, T. S., Khoie, M. R., Sahebkar, E., Rahimi, S., Marhamati, N., 2016. Towards a patient satisfaction-based hospital recommendation system", *In 2016 International Joint Conference on Neural Networks*, Vancouver, Canada, 131-138.

Tóth, Z. E., Árva, G., Dénes, R. V., 2020. Are the 'Illnesses' of Traditional Likert Scales Treatable? *Quality Innovation Prosperity*, 24(**2**), 120-136.

Tsaur, S. H., Chiu, Y. C., Huang, C. H., 2002. Determinants of guest loyalty to international tourist hotels - a neural network approach". *Tourism Management*, 23(**4**), 397-405.

Tóth, Z. E., Jónás, T., Dénes, R. V., 2019. Applying flexible fuzzy numbers for evaluating service features in healthcare–patients and employees in the focus. *Total Quality Management & Business Excellence*, 30(**sup1**), 240-254.

Wahyudi, R. D., Hadiyat, M. A., Hartono, M., 2018. Predicting Service Reliability-Using Survival Analysis

of Customer Fuzzy Satisfaction. *The Asian Journal of Technology Management*, 11(**2**), 79-93.

Patterson, J., Gibson, A., 2017. Deep Learning: A Practitioner's Approach", USA: O'Reilly Media, Inc, 70-77.

Peng, T., Hanke, F., 2016. Towards a Synthetic Data Generator for Matching Decision Trees, *In Proceedings of the 18th International Conference on Enterprise Information Systems* (ICEIS), 135-141.

Raschka, S., Mirjalili, V., 2017. Machine Learning and Deep Learning with Python, scikit-learn and TensorFlow, UK: Packt Publishing, 189-195.

Sreekumar, S., Mahapatra, S., 2015. Service quality of Indian banks: A fuzzy inference system approach. *Asian Academy of Management Journal*, 20(**2**), 9-80.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(**1**), 1929-1958.

Wang, H., Xu, Z., Pedrycz, W., 2017. An overview on the roles of fuzzy set techniques in big data processing: Trends, challenges and opportunities. *Knowledge-Based Systems*, **118**, 15-30.

Wang , W. M., Wang, J. W., Barenji, A. V., Li, Z., & Tsui, E., 2019. Modeling of individual customer delivery satisfaction: an AutoML and multi-agent system approach. *Industrial Management & Data Systems*, 19(**4**), 840-866.

Wright, J. L., Manic, M., 2010. Neural network architecture selection analysis with application to cryptography location, *In The 2010 International Joint Conference on Neural Networks (IJCNN)* IEEE, Barcelona, Spain.

Yau, H. K., Tang, H. Y. H., 2018. Analyzing customer satisfaction in self-service technology adopted in airports. *Journal of Marketing Analytics*, 6(**1**), 6-18.

Zheng, H., Yang, Z., Liu, W., Liang, J., Li, Y., 2015. Improving deep neural networks using softplus units, *In 2015 International Joint Conference on Neural Networks (IJCNN) IEEE*, Killarney, Ireland.