DOI: 10.15439/2017F224
ISSN 2300-5963 ACSIS, Vol. 13

A Decision Support System for Demand Forecasting based on Classifier Ensemble

Irem Islek
Idea Teknoloji Cozumleri,
Istanbul, Turkey
Email: irem.islek@ideateknoloji.com.tr

Sule Gunduz Oguducu
Department of Computer Engineering,
Istanbul Technical University,
Istanbul, Turkey
Email: sgunduz@itu.edu.tr

Abstract—**Demand forecasting is the process of constructing forecasting models to estimate the quantities of several products that customers will purchase in the future. When the warehouse and the number of products grow, forecasting the demand becomes dramatically hard. Most of the demand forecasting models rely on a single classifier or a simple combination of these models. In order to improve demand forecasting accuracy, we investigate several different classifiers such as MLP, Bayesian Network, Linear Regression and SVM analyzing their accuracy and performance. Moreover, we also studied some classifier combination techniques by approaching from demand forecasting perspective. In this paper, we propose a methodology to combine various forecasting models using neural networks rather for supporting demand forecasting. The proposed methodology is tested against single classifiers and classifier ensemble models using a real dataset. Experiments indicate that the proposed methodology outperforms all the other single classifiers tested in this study and their simple combinations.**

I. Introduction

SUPPLY chain is defined as a set of entities directly involved in the activities associated with the upstream and downstream flows of products, services, finances, and/or information from a source to a customer [1]. Supply chains can be categorized into three groups such as Direct Supply Chain, Extended Supply Chain and Ultimate Supply Chain [2]. In our problem, we focus on Direct Supply Chain which contains some manufacturers, warehouses and customers. In this type of supply chain, products of manufacturer are transported to warehouses and customers reach these products through warehouses. Considering all given definitions above, supply chain management (SCM) can be thought as a process which deals with the total flow of materials from suppliers through end users [3]. There are various sub-processes of supply chain management which are quite complicated and challenging such as demand forecasting. Demand forecasting can be summarized as an estimation of a supply chain constituent's (such as warehouse, end sale point etc.) expected sales during a specified future period. Forecasting demand correctly for different constituents provides planning all processes of supply chain effectively. For instance, accurate demand forecasting prevents redundant shipping charges or storage costs. Thus, forecasting the demand of warehouses is an important task and it forms the motivation of our work. In this paper, we study forecasting the demand of warehouses with low error rate problem.

In our previous work, we tried to cluster warehouses according to their sale behaviors using bipartite graphs with the purpose of reducing error rate of demand forecasting. After that warehouse clusters are constructed, we set individual Bayesian Network models for every warehouse cluster. This approach provided improvement in forecasting performance. We defined using different machine learning models and combining them as our future work in an attempt to provide further improvement in forecasting performance [4].

Because of the fact that forecasting demand of warehouses is considerably hard, using a single model can be incapable to solve this problem. Thus, combining multiple models rather than using stand-alone models seems reasonable. Stacked Generalization [5] is a way of combining machine learning models. There are several studies which use Stacked Generalization for combining machine learning models in different domains such as predicting protein types [6], automatic music tagging [7], forecasting fraudulent financial statements [8], etc. In Stacked Generalization, there are two levels which are called as level-0 and level-1. The forecasting outputs of level-0 algorithms are used as input to a level-1 generalizer. In other words, the output results of level-0 are added as parameters to the level-1 generalizer. For example, if level-0 contains three different models, three input parameters must be added to level-1 generalizer. Essentially, level-1 generalizer learns how to map combined outputs to real class values. In this paper, we propose a novel method to combine our four different level-0 methods which are Multi Layer Perceptron (MLP), Bayesian Network, Linear Regression and Support Vector Machine (SVM) in level-1 generalizer. Instead of using only level-0 forecasting outputs as input parameters to level-1, we construct different instances using both level-0 model input parameters and level-0 model forecasting outputs. Then, we use these instances in level-1 generalizer. In addition to that, we apply Stacked Generalization using only binary combinations of our level-0 models in other trial. In other words, we take level-0 models' binary combinations separately to apply stacked generalization. After that, we apply same methodology using only triple combinations of level-0 models. In another trial, we apply regular stacked generalization methodology for comparison. All the results of mentioned trials can be seen in experiments section.

One of the contributions of this study is using level-0 input parameters and level-0 forecasting outputs together as input to level-1 generalizer differently from Stacked Generalization. In addition to that, there is any study that uses stacked generalization in order to forecast demand of warehouses.

The paper is organized as follows. Section II reviews the related literature. In Section III we describe the background of models which are used in this paper. Section IV presents the proposed approach. Section V describes the experiments and the results obtained. Finally, in Section VI we conclude and discuss some possible future works.

II. Related Works

Forecasting demand of warehouses is a problem of estimating the future-dated sales amounts of products for warehouses. There are some different methodologies which have been applied in the domain of warehouse demand forecasting problem. These techniques can be collected as two main types such as (1) stand-alone forecasting models and (2) hybrid forecasting models which use multiple models together. The stand-alone models can be collected two different types which are (1) statistical models and (2) machine learning models.

Moving average and Box-Jenkins are some examples of the basic and popular statistical models. There are some prior works which used these traditional models for demand forecasting [9], [10]. These methodologies were insufficient to solve demand forecasting problem which is quite complex and hard. Thus, artificial intelligence models were started to use. Especially, Neural Networks were used in a large number of works [11]–[16]. Some works compared Neural Networks with traditional statistical models and showed that Neural Networks provides better results than traditional statistical models. Because of the popularity of Neural Networks models for demand forecasting problem, some studies compared another models with Neural Networks. For instance, Efendigil et. al. compared Adaptive Neural Fuzzy Inference System (ANFIS) with Neural Networks. They claimed that ANFIS provides better results than Neural Networks in their study [17].

Some recent studies tried to combine different models in order to reduce the error rate of demand forecasting. In some studies, statistical models were combined with machine learning models. For instance, Aburto and Weber combined Autoregressive Integrated Moving Average (ARIMA) model with Neural Networks in their study [18]. On the other hand, Doganis et. al. used genetic algorithm with Neural Networks for demand forecasting [19].

Combining classifiers on the purpose of improving success rate is a popular approach in most domains. Stacked Generalization which is one of the classifier combining techniques combines more than one machine learning models using another machine learning model [5]. There are some studies which compared stacked generalization with stand-alone models [20]–[24]. According to these studies, stacked generalization provided better results than stand-alone models. In addition to that, some studies claimed that stacked generalization performs better than some other combining models. For instance, Ting and Witten compared Stacked Generalization with majority vote and provided lower error rate with Stacked Generalization [25]. This technique applied in a wide variety of domains likewise biomolecular event extraction [26], early diagnosis of Alzheimer disease [27], forecasting fraudulent financial statements [8], image classification [28], credit risk assessment [29], anti-spam filtering of e-mail [30], city traffic related geospatial data analysis [31], etc.

Conducted studies on demand forecasting handle different parts of the supply chain. Forecasting the demand of an end sale point is one of the most common type of demand forecasting studies. Moreover, most of the demand forecasting studies handle a limited number of warehouses and products. Because of the fact that our problem contains quite number of warehouses and products, one model cannot be sufficiently successful. Thus, approach of combining multiple models is used in this study.

III. Background

In this section, we briefly describe the stacked generalization and the forecasting models which are adopted in our work.

A. Stacked Generalization

Ensemble of classifiers is defined as a concept of combining classifiers to improve performance of individual classifiers. Stacked generalization is one of the ensemble classifiers methodologies which is used for minimizing error rate of one or more classifiers [5]. This methodology proposed by David H. Wolpert in 1992.

In stack generalization methodology, there are two basic steps which are sequential. The first step which is called level-0 contains independent machine learning models. These level-0 models are combined in level-1 generalization step. In the level-1 step, outputs of each individual level-0 models are used as an input parameter to generalizer model. Any machine learning model can be selected as generalizer according to suitability to the problem.

Level-0 models of stacked generalization are trained using a set of training data. Afterwards, another set of training data is created from prediction outputs of level-0 models. This dataset is used for training level-1 generalizer. The key point of this operation is that, forecast results in this dataset are estimated from the instances which are not in the training dataset of level-0 models. To evaluate the stacked generalization model, the output of every instance of a third dataset is predicted by level-0 models separately. Estimated forecasting results are used as input parameters in the level-1 generalizer. Finally, the forecasting output of level-1 generalizer is compared with the real output of every instance to obtain a final evaluation result. Scheme of stacked generalization can be seen at Fig. 1.

B. Bayesian Network Algorithm

Bayesian Network is a simple, graphical representation for conditional independence assertions. In this graphical representation, every node of graph symbolize a random variable, where a random variable can take on possible values from a random experiment. In addition to that, every edge between

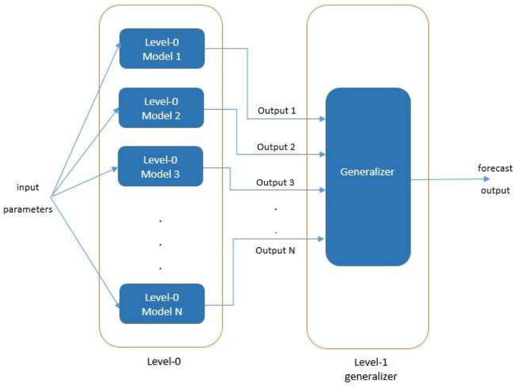

Fig. 1. Stacked Generalization

these nodes represents probabilistic dependencies among these random variables. If a Bayesian Network contains n nodes (random variables), the joint probability distribution is represented as Eq. 1.

$$p(x_1, x_2, x_3, ..., x_n) = \sum_{i=1}^{n} p(x_i|x_{P_i}) \tag{1}$$

In Eq. 1, $p(x_i|x_{P_i})$ represents the local conditional probability distribution for node i. P_i corresponds to parent nodes indices of node i. Thanks to conditional independence relationship, joint probability distribution can be represented more conveniently for large networks.

Bayesian Networks can be used for numerous applications such as classification, regression, segmentation [32].

C. Multi Layer Perceptron Algorithm

Multilayer Perceptron (MLP) which maps a set of input onto a set of appropriate outputs is one of the feed-forward artificial neural network models. MLP can be thought as multiple layers which are fully connected to next layer. A multilayer perceptron contains an input layer, an output layer and also one or more hidden layers. Each node is accepted as a neuron and has a nonlinear activation function except for the input nodes. This nonlinear function can be seen in Eq. 2.

$$y = \varphi\left(\sum_{j=1}^{n} w_j x_j - \theta\right) \tag{2}$$

In Eq. 2, x_j values are input signals and w_j values are weights associated with the jth input. θ corresponds to threshold value and $\varphi(.)$ is a sigmoid activation function. Eq. 3 shows this sigmoid activation function.

$$\varphi(x) = \frac{1}{1 + exp(-x)} \tag{3}$$

The weights are adjusted during training phase in order to obtain input-output mapping of the network. Weight updating phase continues until weights no longer change or error value reaches a threshold value. [33].

D. Linear Regression Algorithm

Linear regression is a technique which is used for modeling the relation between a scalar dependent variable y and one or more independent variables. If there is only one independent variable, it is called simple linear regression. In other case, it is called multiple linear regression [34]. In multiple linear regression, the relation between the scalar dependent variable y and independent variables is defined by Eq. 4.

$$y = \alpha + x_1\beta_1 + x_2\beta_2 + x_3\beta_3 + ... + x_n\beta_n + \epsilon \tag{4}$$

E. Support Vector Machine Algorithm

Support Vector Machine (SVM) which is a machine learning algorithm can be used for not only classification, but also regression problems. SVM regression is a nonparametric technique because it uses several kernel functions. The relation between input (x_i) and output (y_i) can be mapped by a regression function $f(x)$ which can be seen in Eq. 5.

$$f(x) = wx + b \qquad\qquad w \in X, b \in \Re \tag{5}$$

After that, following problem must be solved:

$$minimize \ \frac{1}{2}||w||^2 \tag{6}$$

$$subject\ to \begin{cases} y_i - wx_i - b \leq \epsilon \\ wx_i + b - y_i \leq \epsilon \end{cases} \tag{7}$$

There is a case where the constraints are infeasible. In this case which is called soft margin formulation, slack variables (ξ_i , ξ_i^*) are used.

$$minimize \ \frac{1}{2}||w||^2 + C\sum_{i=1}^{l} \xi_i + \xi_i^* \tag{8}$$

$$subject\ to \begin{cases} y_i - wx_i - b \leq \epsilon + \xi_i \\ wx_i + b - y_i \leq \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases} \tag{9}$$

In Eq. 8, C controls the penalty amount based on deviations which are larger than ϵ. The linear ϵ-insensitive loss function ($|\xi|_\epsilon$) ignores errors that are within ϵ distance of the observed value by treating them as equal to zero. As can be seen in Eq. 10, the loss is measured based on the distance between observed value and the ϵ boundary.

$$|\xi|_\epsilon = \begin{cases} 0, & \text{if } |\xi| \leq \epsilon \\ |\xi| - \epsilon, & \text{if } |\xi| \succ \epsilon \end{cases} \tag{10}$$

After applying Lagrangian multipliers, a model solution can be found in dual representation.

$$f(x) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) K(x_i, x) + b \tag{11}$$

$$K(x_i, x_j) = exp(-\gamma||x_i - x_j||^2) \tag{12}$$

In Eq. 11, α_i, α_i^* are nonzero Lagrangian multipliers. $K(x_i, x)$ is the kernel function. Eq. 12 shows radial basis function (RBF) kernel and γ corresponds to width parameter of RBF kernel [35].

IV. Details Of The Methodology

In this paper, we study the problem of forecasting the sale amounts of products for main distribution warehouses. Because of the fact that there are a large number of warehouses and products in our case, problem is harder than regular demand forecasting problem. Thus, it is needed to use hybrid methodologies in order to solve given problem.

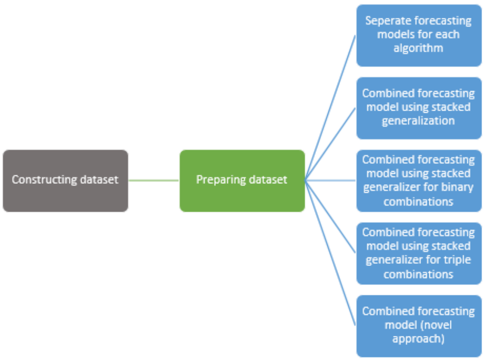
Constructing dataset
Preparing dataset
Seperate forecasting models for each algorithm
Combined forecasting model using stacked generalization
Combined forecasting model using stacked generalizer for binary combinations
Combined forecasting model using stacked generalizer for triple combinations
Combined forecasting model (novel approach)

Fig. 2. Steps of the study

First step of our methodology is constructing a dataset which contains necessary information to forecast demand of warehouses. Our dataset is constructed from real sales transaction data of a national dried fruits and nuts company from Turkey. Sales transactions of 2011, 2012 and 2013 are used in this dataset. There are ninety eight warehouses and seventy different products in the dataset. Additionally, this dataset contains warehouse related attributes such as location, number of transportation vehicles, total amount of weekly selling product, selling area in square meter, number of employees and product related attributes such as selling amount, selling time, product category.

Next step of the proposed methodology is preparing the dataset for data mining operations. In this step, data is cleaned and prepared for further operations. Moreover, moving average value of product sale amounts are calculated using past three weeks. For a specific week t, the moving average calculation equation can be seen in Eq. 13.

$$Moving\ avg.\ (t) = \frac{\sum_{i=1}^{3} sale\ amount\ (t\ \text{-}\ i)}{3} \qquad (13)$$

After these steps, forecasting model is constructed using machine learning algorithms. In this step, different methodologies are tried and compared in order to find the one with lowest error rate. Some of these methodologies use model combining techniques in order to provide better performance. In the first forecasting model construction trial of our study, four different algorithms are determined to forecast demand of warehouses. Selected algorithms are Bayesian Network, Multi Layer Perceptron, Support Vector Machine and Linear Regression. In order to make comparison with further combined forecasting models, selected algorithms are used separately for constructing a forecasting model as a baseline. Warehouse related attributes, product related attributes, time information and moving average values are used as input, selling amounts are used as output in all models. Detailed results and reviews about these four stand-alone models can be seen at Experiments and Results section.

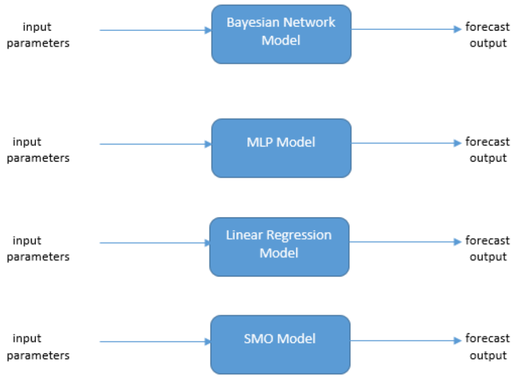

Fig. 3. Stand-alone models

Stacked Generalization methodology which is explained in Background section in detail, contains two different levels and combines selected models on the purpose of reducing the error rate. With the same purpose, stacked generalization methodology is used for combining four different machine learning models in our study. As can be seen in Figure 4, there are four separate forecasting models in level-0 step which will be used for estimating forecasting output in the next level. For every instance, forecast results are produced by individual level-0 models. After that, these forecast results are used as input parameters in level-1 generalization model. In our study, MLP machine learning algorithm is used for constructing level-1 generalizer. Level-1 generalizer maps input parameters into real sale amounts. In this approach, the output of the level-1 is the forecasting output of the total methodology.

With the aim of determining whether or not some of the selected machine learning algorithms are dominant to other ones, various combinations of the selected four machine learning algorithms are used in level-0, differently from the previous trial. Firstly, 2-combinations of the four algorithms are determined: MLP and Bayesian Network, MLP and Linear Regression, MLP and SVM, Bayesian Network and SVM, Bayesian Network and Linear Regression, SVM and Linear Regression. These six different combinations are used respectively as level-0 models. Instead of using four different models, two models which are from one of the 2-combination set are used as level-0 models in this trial. Thus, two forecasting outputs are added as input parameters to the level-1 generalizer in this trial. Constructing stacked generalization is repeated six

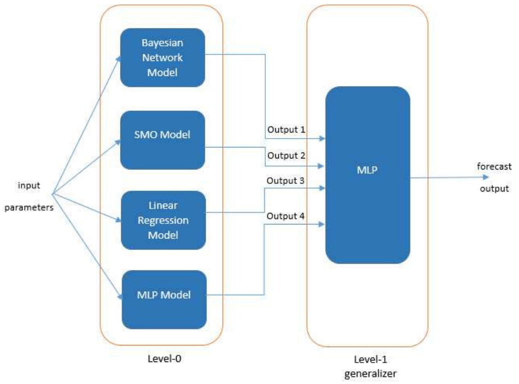

Fig. 4. Stacked Generalization for our problem

times because of that 2-combination set contains six element. Detailed figure of this trial can be seen at Figure 5.

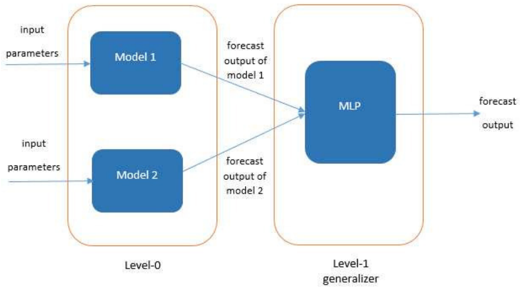

Fig. 5. Stacked Generalization using binary combinations

Same procedure is applied using 3-combinations in the next trial. 3-combination set contains MLP, Bayesian Network, SVM, MLP, Bayesian Network, Linear Regression, MLP, SVM, Linear Regression and Bayesian Network, SVM, Linear Regression. There are three different models in level-0 and three forecasting outputs are added to level-1 generalizer model as input parameters. As might be expected, stacked generalizer methodology is constructed four times in this trial. Relevant figure can be seen at Figure 6.

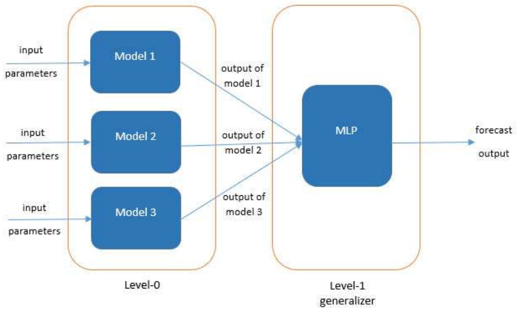

Fig. 6. Stacked Generalization using triple combinations

In the last trial, a novel approach which is similar to Stacked Generalization from some aspects is applied. This approach is also contains two levels. Especially, level-0 is completely similar to level-0 of Stacked Generalization. This methodology differs from Stacked Generalization in respect to construction input parameters of level-1. In Stacked Generalization, forecasting outputs of the level-0 models are added to input parameters of level-1 generalizer. For instance, if there are four different models in level-0, four forecasting outputs will be added as new input parameters to the level-1 for every instance. Instead of using only forecasting outputs of level-0 in level-1, input parameters of level-0 and forecasting outputs of level-0 are used together as input parameters of level-1.

As can be seen in Figure 7, level-0 models estimate forecasting outputs using x number of input parameters. Because of that there are four different models in level-0 of our trial, four different forecasting outputs are estimated. There are four forecasting outputs after level-0. In the level-1, these forecasting outputs and input parameters of level-0 are used as inputs of level-1 to estimate the output of the total methodology. Using this new instance, a new forecasting output is estimated by level-1 neural network.

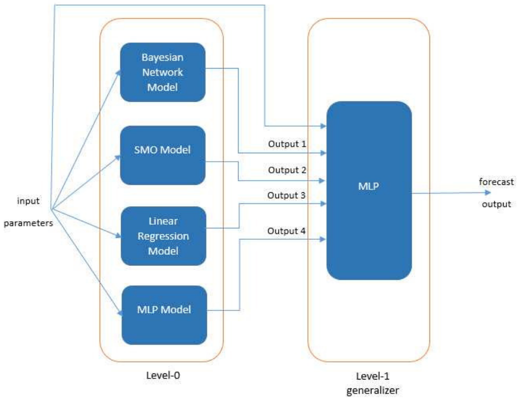

Fig. 7. Our approach for combining models

Comparison of all methodologies can be seen in the next section, Experiments and Results.

V. Experiments and Results

In this section, we describe the sales data used in our experiments and present the results obtained by our approach for demand forecasting compared to other applied methodologies.

A. *Dataset*

Dataset is composed from the real sales data of a national dried fruits and nuts company from Turkey. This company gives service to all of the cities in Turkey. The warehouse and product counts are 98 and 70, respectively. Dataset contains 15317141 instances which are taken from real sales transaction data of 2011, 2012 and 2013 years. These instances are obtained from 93987868 sales transactions detail lines.

Attributes in the dataset can be collected two main groups: warehouse attributes and item attributes. Warehouse related attributes are location, number of sub-warehouses it has, number of employees, number of customers, number of vehicles, selling area in square meter and total amount of weekly selling products. Item related attributes are item category, selling time and moving average of previous 3 weeks. Selling amount of the item is selected as output of the model.

B. Experimental Results

First of all, the dataset is divided into 3 partitions in order to train and test the methodologies. The first, second and third datasets are composed from real sales transaction data of 2011, 2012 and 2013, respectively. Cause of dividing dataset into 3 partitions is that combined methodologies contain two levels of machine learning models and these sequential levels cannot be trained using the same training dataset. Thus, first dataset is used for training level-0 of combined methodology while second dataset is used for training level-1. Third dataset is used for testing the combined methodologies. Differently from the combined methodologies, methodologies which contain only one machine learning algorithm are trained using both first and second datasets. The third dataset is used for testing the stand-alone methodologies likewise the combined methodologies.

The error rate of the methodologies are estimated using Mean Average Percentage Error (MAPE). The calculation formula of MAPE can be seen in Eq. 14. In the given formula, A_t means actual sales amount and F_t means forecasting sales amount, for this study.

$$MAPE = \frac{100}{n} \sum_{t=1}^{n} \frac{|A_t - F_t|}{|A_t|} \tag{14}$$

Table 1 shows the error rate of stand-alone methodologies. As can be seen in this table, Bayesian Network gives better results than the other stand alone models for our problem.

TABLE I
ERROR RATES OF STAND-ALONE METHODOLOGIES

Model	MAPE Value
Bayesian Network	17.5%
MLP	21.2%
Linear Regression	19.8%
SVM	23%

With the motivation of reducing error rate, combining more than one model approach is applied. For combining models, stacked generalization is used at the beginning. Table 2 shows the result of stacked generalization error rate.

TABLE II
ERROR RATE OF STACKED GENERALIZATION

Model	MAPE Value
Stacked Generalization	21.9%

After that, instead of combining all of the selected models (Bayesian Network, SVM, Linear Regression and MLP), binary combinations of these models are used as level-0 models. Reason of this trial is determining whether some models are dominant or not. Comparisons of these 2-combinations can be seen in Table 3.

TABLE III
ERROR RATE OF STACKED GENERALIZATION WITH 2-COMBINATIONS

Model	MAPE Value
Stacked Generalization with Bayesian Network and SVM	22%
Stacked Generalization with Bayesian Network and MLP	20.7%
Stacked Generalization with Bayesian Network and Linear Regression	20.5%
Stacked Generalization with SVM and MLP	23.8%
Stacked Generalization with SVM and Linear Regression	23%
Stacked Generalization with MLP and Linear Regression	22.8%

Table 4 shows the results of stacked generalization with 3-combinations.

TABLE IV
ERROR RATE OF STACKED GENERALIZATION WITH 3-COMBINATIONS

Model	MAPE Value
Stacked Generalization with Bayesian Network, SVM and MLP	22.6%
Stacked Generalization with Bayesian Network, SVM and Linear Regression	21.4%
Stacked Generalization with Bayesian Network, MLP and Linear Regression	20.3%
Stacked Generalization with SVM, MLP and Linear Regression	23.4%

Finally, we try our methodology for the same dataset. In this methodology, we use both input parameters of level-0 and forecasting outputs of level-0 as input of level-1.

TABLE V
COMPARISON OF BEST RESULTS OF ALL TRIALS

Model	MAPE Value
Our methodology	12.7%
Stacked Generalization	21.9%
Bayesian Network	17.5%
Stacked Generalization with Bayesian Network and Linear Regression	20.5%
Stacked Generalization with Bayesian Network, MLP and Linear Regression	20.3%

The results of warehouse demand forecasting indicate that our methodology which uses input parameters and forecasting outputs of level-0 together in level-1 has the overall best performance.

VI. CONCLUSION

In this paper, we proposed an approach which combines multiple machine learning models in order to forecast demands of warehouses. Our methodology consists of two levels likewise Stacked Generalization. Differently from Stacked Generalization methodology that constructs level-1 generalizer with only the forecast outputs of level-0 models, our methodology takes input parameters of level-0 and forecast outputs of level-0 models together as input parameters into level-1 generalizer. In addition to that, we apply Stacked Generalization which has never been used before in warehouse demand forecasting, in order to compare results with our methodology. Experiments are performed on three-year real sales data of a national dried fruits and nuts company from Turkey.

The experimental results show that our approach achieves better results for forecasting demands of warehouses. In terms of MAPE, the proposed method provides nearly 5% less error than best result of stand-alone models. Moreover, it drops error rate nearly 9% compared to Stacked Generalization.

Acknowledgment

This research was supported by Ministry of Science, Industry and Technology of Turkey SANTEZ project 0484.STZ.2013-2.

References

[1] M. L. Christopher, *Logistics and Supply Chain Management,* Pitman Publishing, 1992.

[2] J. T. Mentzer, W. DeWitt, J.S. Keebler, S. Min, N. W. Nix, C. D. Smith, Z. G. Zacharia, "Defining Supply Chain Management," *Journal of Business Logistics,* vol. 22, no. 2, 2001.

[3] T. C. Jones, D. W. Riley, "Using Inventory for Competitive Advantage through Supply Chain Management,“ *International Journal of Physical Distribution and Materials Management,* vol. 15, no. 5, pp. 16-26, 1985.

[4] I. Islek, S. G. Oguducu, "A retail demand forecasting model based on data mining techniques," in *24th International Symposium on Industrial Electronics (ISIE),* Buzios, Brasil, 2015, pp. 55-60.

[5] D. H. Wolpert, "Stacked Generalization," *Neural Networks,* vol. 5, pp. 241-259, 1992.

[6] S. Q. Wang, J. Yang, K. C. Chou, "Using stacked generalization to predict membrane protein types based on pseudo-amino acid composition," *Journal of Theoretical Biology,* vol. 242, no. 4, pp. 941-946, 2006.

[7] S. R. Ness, A. Theocharis, G. Tzanetakis, "Improving automatic music tag annotation using stacked generalization of probabilistic svm outputs," in *17th ACM international conference on Multimedia,* 2009, pp. 705-708.

[8] S. Kotsiantis, E. Koumanakos, D. Tezlepis, V. Tampakas, "Forecasting Fradulent Financial Statements Using Data Mining," *International Journal of Computational Science,* vol. 3, no. 2, pp. 104-110, 2006.

[9] L. M. Liu, S. Bhattacharyya, S. L. Sclove, R. Chen, W.J. Lattyak, "Data Mining on Time Series: An Illustration Using Fast-Food Restaurant Franchise Data," *Computational Statistics & Data Analysis,* vol. 37, pp. 455-476, 2001.

[10] K. Gilbert, "An ARIMA supply chain model," *Management Science,* vol. 51, no. 2, pp. 305-310, 2005.

[11] P. C. Chang, Y. W. Wang, C. H. Liu, "The Development of a Weighted Evolving Fuzzy Neural Network for PCB Sales Forecasting," *Expert Systems with Applications,* vol. 32, pp. 86-96, 2007.

[12] Z. L. Sun, T. M. Choi, Y. Yu, "Sales Forecasting Using Extreme Learning Machine With Applications In Fashion Retailing," *Decision Support Systems,* vol. 46, pp. 411-419, 2008.

[13] Y. Yu, T. Choi, C. Hui, "An Intelligent Fast Sales Forecasting Model for Fashion Products," *Expert Systems with Applications,* vol. 38, pp. 7373-7379, 2011.

[14] S. H. Ling, *Genetic Algorithm and Variable Neural Networks: Theory and Application,* Lambert Academy Publishing, 2010.

[15] K. F. Au, T. M. Choi, Y. Yu, "Fashion Retail Forecasting by Evolutionary Neural Networks," *International Journal of Production Economics,* vol. 114, pp. 615-630, 2008.

[16] R. S. Gutierrez, A. Solis, S. Mukhopadhyay, "Lumpy Demand Forecasting Using Neural Networks," *International Journal of Production Economics,* vol. 111, pp. 409-420, 2008.

[17] T. Efendigil, S. Onut, C. Kahraman, "A Decision Support System for Demand Forecasting with Artificial Neural Networks and NeuroFuzzy Models: A Comparative Analysis," *Expert Systems with Applications,* vol. 36, no. 3, pp. 6697-6707, 2009.

[18] L. Aburto, R. Weber, "Improved Supply Chain Management Based On Hybrid Demand Forecasts," *Applied Soft Computing,* vol. 7, no. 1, pp. 136-144, 2007.

[19] P. Doganis, A. Alexandridis, P. Patrinos, H. Sarimveis, "Time Series Sales Forecasting For Short Shelf-Life Food Products Based On Artificial Neural Networks And Evolutionary Computing," *Journal Of Food Engineering,* vol. 75, pp. 196-204, 2006.

[20] A. Ghorbani, K. Owrangh, "Stacked Generalization In Neural Networks: Generalization On Statistically Neutral Problems," in *IEEE International Joint Conference on Neural Networks,* 2001, pp. 1715- 1720.

[21] D. W. Fan, P. K. Chan, S. J. Stolfo, "A comparative evaluation of combiner and stacked generalization," in *AAAI-96 workshop on Integrating Multiple Learned Models,* 1996, pp. 40-46.

[22] G. Sigletos, G. Paliouras, C. D. Spyropoulos, M. Hatzopoulos, "Combining information extraction systems using voting and stacked generalization," *Journal of Machine Learning Research,* vol. 6, pp. 1751-1782, 2005.

[23] L. Breiman, "Stacked regressions," *Machine learning,* vol. 24, no. 1, pp. 49-64, 1996.

[24] I. Guler, E. D. Ubeyli "ECG beat classifier designed by combined neural network model," *Pattern recognition,* vol. 38, no. 2, pp. 199-208, 2005.

[25] K. M. Ting, I. A. Witten "Issues in Stacked Generalization," *Journal Of Artificial Intelligence Research,* vol. 10, pp. 271-289, 1999.

[26] A. Majumder, A. Ekbal, S. K. Naskar, "Biomolecular Event Extraction using a Stacked Generalization based Classifier," in *13th International Conference on Natural Language Processing,* 2016.

[27] H. Gandhi, D. Green, J. Kounios, C. M. Clark, R. Polikar, "Stacked Generalization for Early Diagnosis of AlzheimerâĂŹs Disease," in *28th IEEE EMBS Annual International Conference,* 2006.

[28] C. F. Tsai, "Training Support Vector Machines Based On Stacked Generalization For Image Classification," *Neurocomputing,* vol. 64, pp. 479-503, 2005.

[29] M. Doumpos, C. Zopounidis, "Model Combination For Credit Risk Assessment: A Stacked Generalization Approach," *Annals of Operations Research,* vol. 151, no. 1, pp. 289-306, 2007.

[30] G. Sakkis, I. Androutsopoulos, G. Paliouras, V. Karkaletsis, C. Spyropoulos, P. Stamatopoulos, "Stacking Classifiers for Anti-Spam Filtering of E-mail," in *Empirical Methods in Natural Language Processing,* pp. 44-50, 2001.

[31] X. Liu, L. Yu, P. Peng, F. Lu, "A Stacked Generalization Framework for City Traffic Related Geospatial Data Analysis," in *Asia-Pacific Web Conference,* 2016, pp. 265-276.

[32] N. Friedman, D. Geiger, M. Goldszmidt, "Bayesian Network Classifiers," *Machine Learning,* vol. 29, no. 2-3, pp. 131-163, 1997.

[33] S. Haykin, *Neural Networks: A Comprehensive Foundation,* Prentice Hall, 1998.

[34] D. A. Freedman, *Statistical Models: Theory and Practice,* Cambridge University Press, 2009.

[35] D. Basak, S. Pal, D. C. Patranabis, "Support vector regression," *Neural Information Processing-Letters and Reviews,* vol. 11(10), pp. 203-224, 2007.