Transactions of the Nigerian Association of Mathematical Physics
***Volume 13, (October – December, 2020 Issue*), pp123–132**
© Trans. of NAMP

AN IMPROVED SHOPPING AGENT FOR THE PREDICTION OF PRODUCT PREFERENCES IN AN E-STORE USING ARTIFICIAL NEURAL NETWORK

Soretire T.O. and Vincent O.R.

Department of Computer Science, Federal University of Agriculture Abeokuta
Ogun State, Nigeria

Abstract

The disparity in the prices of online products from different e-store are as a result of theproliferation of e-Stores, seasonal price fluctuations, and periodic time-based deals/discounts remains a concerning issue in electronic commerce. The existing models, such as linear programming, multi-agent system, and statistical models,were time-consuming and too slow. This study presents an improvement to the shopping agent for e-stores using Multi-Layer Perceptron Neural Network. The scheme is a prediction model and a purchase-decision tool for buyers. A Resilient Propagation algorithm was employed for training the network and price prediction. The method was compared with the Back Propagation algorithm. The two models were designed with 12 neurons at the input layer, 20 neurons at the hidden layer, and 1 neuron at the output layer. Using the sliding window method, average prices for 12 consecutive weeks were inputted into the models to obtain predicted average weekly prices for the successive week. The model successfully predicted variability in the average weekly prices and could be used for purchase-decision making.

***Keywords*:** Shopping Agent, Decision Making, E-Store, E-commerce, and Artificial Neural Network.

1.0 Introduction

The need for the customers to get the best preferences of products in an e-store cannot be emphasized. Shopping agents are significantly impacting electronic commerce and have succeeded in reducing the cost and time involved in searching for goods and services on e-Stores.Comparison Shopping is the practice of comparing the prices of items from different sources in order to find the best deal. The quest for more efficient and beneficial Comparison Shopping has been the motivation of researches in the field of Information Science, Mathematics, Economics, Business Administration/Marketing, and even Psychology. Despite the increasing adoption of Shopping Agents for comparison-shopping, it has been observed that significant price dispersion still exists among e-Sellers for the same product in defiance to the "law of one price" (Baye*et al.,* 2001, DiRusso, 2011; Repiso et al., 2019); this justifies the interest in Shopping Agents development.

A common phenomenon in today's fiercely competitive and aggressively market-driven domain of e-Commerce is offering a periodic, time-based, price reduction, referred to as deals or discounts to eShopper apart from the usual seasonal price fluctuation that most goods/services are subjected to ((Niemela et al., 2019; Perera and Veloso, 2018). Amongst other reasons, this may be because e-Stores employ dynamic pricing techniques to improve their profit by dynamically adjusting their prices in response to consumer demand and competitors' pricing (Jumadinova, 2008; Redrigvez et al., 2018). Baye*et al.* (2004) also opined that merchants employ the "hit and run" sales strategy: undertaking short-term price promotions at unpredictable intervals, a method shown to be active and widely used. Consequently, e-shoppers are always a step behind the e-Sellers: the best price for a good or service today may not be the best price if a deal or discount is offered tomorrow, even in the same store. Similarly, a delay in purchase today may turn out to be a lost opportunity tomorrow (Nanavati et al., 2019; Liu et al., 2019). For a shopper to make an informed purchase-decision beyond the immediate present, there is the need for Shopping Agents with the predictive capability to forecast the price movement of a product in an e-Store of interest (Karunarathme et al., 2020; Dudarenko and Smirnov, 2020).

Correspondence Author: Vincent O.R., Email: Vincent.rebecca@gmail.com, Tel: +2347037832842

Discounts, deals, sales are terms used by e-Merchants to signify a reduction in the price of a product/service mostly for a timeframe. The cost reduction has no impact on the quality or properties of the item in any way. More often than not, casual buyers do not readily get to benefit from these offers except by coincidence. However, if buyers can have a 'foreknowledge' of future price trends, they will be able to make a more beneficial purchase-decision (Tamigneaux 2014).

Although Brynjolfsson *et al.* (2000), in his empirical study, noted that within the same price ranges, not all e-Buyers opted for the lowest price offered by a shopping agent due to their preference for a brand or consumer loyalty. Extensive empirical evidence is showing that price is the most critical factor determining where consumers decide to shop (Mansell, 2014) even more than convenience (Hajaj*et al.,* 2015; Wu and Wang, 2019). As seen with stocks and forex and other time series-based domains, e-Buyers, including brand-conscious e-Buyers, will be able to make a more informed purchase-decision if presented with forecasted price direction of a product of interest.

This work presents a novel shopping agent as a predictive model using an Artificial Neural Network for the e-commerce environment. This study develops, implements, and tests a shopping agent that provides all the e-commerce functionalities. The model is customer-centric, which targets e-stores periodically with a price discount as part of its marketing strategy.

2.0 Related Work

There are several studies on shopping agent and robot technology for e-commerce especially many researches have focused on how robot agent could make a cheaper and better transaction over the network, but only a few have analyzed customers preferences.

Braz*et al.* (2003) proposed a Mobile Bargain Agent for Online Shopping (MBAOS), a comparison-shopping agent designed with mobile devices in mind. The MBAOS system involves a buyer, a Mobile Bargain Agent (MBA) for the buyer, and suppliers. Huang and Tsai (2009) designed and implemented a cross-language comparison-shopping agent (WebShopper), which is capable of performing online price comparisons of computer books across e-stores using two languages (English and Chinese). It also factored in delivery fees and exchange rates in a bid to facilitate accurate comparison. Multilingual ontology was constructed and maintained manually and restricted to computer books.

The improvements of WebShopper to partly automate the ontology construction method, fully automate the product-classification approach, and inclusion of a semantic search mechanism that considers concept similarity by Huang *et al.* (2011) gave rise to WebShopper+. Blazewiczet al. (2010) formally modeled the Internet Shopping Optimization Problem (ISOP). Specifically, he considered a problem in which a customer would like to buy a given set product {1, …, n} from a given set of Internet shops {1, …, m} at the minimum cost, taking into account also the delivery costs associated with the shops where one or more products are bought. This problem was modeled after the well-known Facility Location Problem (FLP) and solved using the greedy algorithm in 2010 and forecasting 2014.

Kanda et al. (2009) developed a guide robot for a shopping mall and conducted a field trial with it. The robot was designed to interact naturally with customers and to provide the shopping information effectively. It was also designed to interact with people to build a rapport repeatedly; since a shop- a ping mall is a place people repeatedly visit. The study provided the chance to design a robot for multiple interactions explicitly; thus, RFID tags were used for personal identification. The robot was semi-autonomous, partially controlled by a human operator, to cope with the difficulty of speech recognition in a real environment and to handle unexpected situations. A field trial was conducted at a shopping mall for 25 days to observe how the robot performed this task and how people interacted with it. The robot interacted with approximately 100 groups of customers each day.

Ghuli P. *et al.* (2014) designed a scalable, distributed Shopping Agent based on distributed computing methods known as Map-Reduce. They developed a distributed crawler for online shoppers to compare the prices of the requested products from different vendors and get the best deal in one place. Since crawling consumes a broad set of computer resources to process the vast amount of data in fat e-commerce servers in a real-world scenario, the researchers used an alternative way of map-reduce paradigm to process a large amount of data by forming Hadoop cluster of cheap commodity hardware. The study described an implementation of a shopping agent on a distributed web crawler using a map-Reduce paradigm to crawl the web pages.

Another research was the design of a shopping mall guide robot, named KeJia, which was designed for customer navigation, information providing, and entertainment in a real-time environment. The study introduced the designs of robot's hardware and software, faced challenges, and multimodal interaction methods, including using a mobile phone app. In order to adapt the current localization and navigation techniques to such large and complex shopping mall environment, a series of related improvements and new methods were compared. The robot was deployed in a large shopping mall for field test and stable operation for a reasonably long time. The result demonstrated the stability, validity, and feasibility of this robot system and showed a positive reward to the original design (Chen et al., 2015).

Lopez-Loces*et al.* (2016) proposed further optimization of internet shopping problem by also taking into account the delivery costs associated with the shops where one or more products are bought. He proposed using Integer Linear Programming (ILP) model and two heuristic solutions, the MinMin algorithm and the cellular processing algorithm.Zhang et al. (2017) introduced a new approach to controlsoft robots. The approach contributed in two ways: in designing an abstract representation of the state of soft robots and developing a reinforcement learning method to derive effective control policies. The reinforcement learning process can be trained quickly by ignoring the specific materials and structural properties of the soft robot. They applied the approach to the Honeycomb PneuNets Soft Robot and demonstrated the effectiveness of the training method and its ability to produce good control policies under different conditions.

3.0 METHODOLOGY

The predictive Internet Shopping Agent (shopper) model we are proposing would seek to fetch the price history of a product of interest via web service from the relevant website. In our model, the data will be extracted and transformed into a daily time series format, which is then mapped and time-boxed to a weekly time series pattern for forecasting. Our model uses Multilayer Perception Neural Network (MLP-NN) with a sigmoidal activation function and Resilient Propagation (RPRO) as its training algorithm.

3.1 The Customer Agent Model for Shopping

Figure 1 presents the shopping agent model where a customer identifies all the attributes of the intended production and assign the agent to move around the network and get the best product according to the user specification.

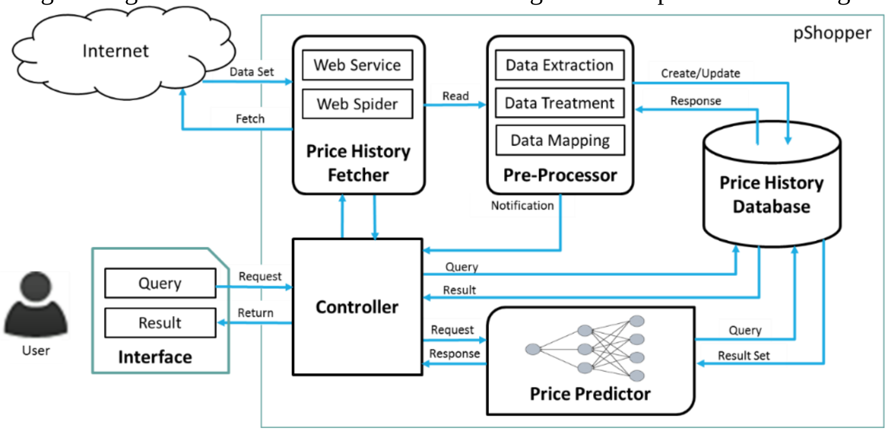

Figure 1:The Model for Shopping Agent

The following are the explanation of the model. The controller oversees the whole operation of the shopper, and initiate required processes. It also provides an interface for the user to query the shopping agent and obtain results of the internal operations. The price history Fetcher, on the other hand, seeks to fetch important price details of exciting products in targeted eCommerce stores via web service or any other channel provided by the data source. If a fetch is successful, the results are sent to the pre-processor; otherwise, a failure response is communicated to the Controller.

Pre-Processor records are extracted from the fetched data and treated and formatted for storage in a database. Price History Database serves as a data warehouse for pre-processed data. It is the first point of call by the controller on obtaining new query results to assess if some or all the required records are available on the database and the magnitude to be fetched by the Price History Fetcher. It also returns required records to the Price Predictor for prediction.

Price Predictor implements the Neural Network. It picks input (pre-processed historic price sequence) from the database, which it works with to present a predicted price returned to the controller for onward presentation by the shopping agent. Figure 1 depicts the UML sequence diagram of the shopper. The diagram shows the sequence of interactions that take place when a shopping agent sends a request to the shopper for a forecasted price in preparation for submitting a search result to a user.

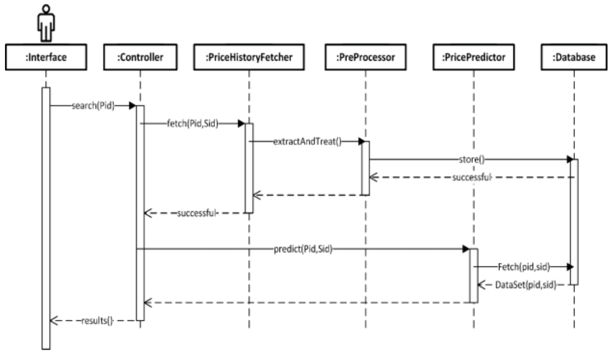

Figure 2: Sequence Diagram for shopper

3.4 Data Extraction and Treatment

Suppose there exist a finite number (n) of eStores in an oligopolistic market that is accessible to our shopping agents. The set of all the stores can be defined by equation 2

$$S_n = \{S_1, S_2, S_3, \dots S_n\} \quad (1)$$

When a buyer queries the shopping agent for a unique item (i) on the day (d), the prices that several stores (S) offer the product which meets the product descriptions of item (i) can be represented by Eq. (3) where $P \in \mathbb{R}$

$$P^d_{i,S_n} = \{P_{i,S_1}, P_{i,S_2}, \dots P_{i,S_n}\} \quad (2)$$

An ordered set of best prices can be given as Eq. (4) where $a_n \leq a_{n+1}$ for all $a \in Bp$

$$Bp_{(i,S_n)} = Asc\,\{P_{i,S_n}, \rho\} \quad (3)$$

For the best price obtained, the price history fetcher would request for the list of past prices for the product item (i) in the selected store (s).

Algorithm 1 below highlights the data extraction and treatment process that transforms the fetched data to a daily time series data for the product (i) such that $[p_1,\ p_2, p_3, \dots\ p_n]$ is strictly chronological.

Algorithm 1: *Data extraction and treatment algorithm*

```
Input: string: productId; string: url; Object[]:fetchedProductJSON
Output: Double[]: currentDate; currentWeek; price
Initialize: currentDate, currentWeek, Price; intsameDayPriceCount :=0;
double sameDayPriceSum :=0;  double average = productDetail[i].getPrice();
string currentDate;

Begin
  fetchedProductJSON := crawl(url,productId);
  currentDate = fetchedProductJSON[startDate]
    currentWeek = getWeek(fetchedProductJSON[startDate])
foreach row in fetchedProductJSON {
      fetchedDate := getDate(fetchedProductJSON[timestamp]);
      fetchedWeek := getWeek(fetchedProductJSON[timestamp]);
      fetchedPrice := fetchedProductJSON[price];
      while (fetchedDate>= currentDate)
          if (fetchedDate == currentDate)
             sameDayPriceSum.add(fetchedPrice)
             sameDayPriceCount.add(1)
          else
             if (sameDayPriceCount> 0)
               price := sameDayPriceSum / sameDayPriceCount
             else
               Price := fetchedPrice
            end if
            productData.writeDB[] := [currentDate, currentWeek, price]
             currentDate.Add(1 day)
             currentWeek := getWeek(currentDate)
             sameDayPriceCount := 0
        end if
      end while
   end for each
 End
```

3.5 MultiLayer Perceptron for shopper

The scheme has 12 input neurons, 20 hidden neurons, and 1 at the input, hidden and output neurons, respectively. Each of the weekly prices of 12 consecutive weeks will be fed into the 12 neurons at the input layer. The output from the neurons in the input layer would be connected to 20 hidden neurons at the hidden layer. The 20 hidden layersare then connected to the neuron at the output layer. The output of the output neuron corresponds to the predicted price of our model for the week following the 12 weeks fed into the input neurons.

The function of the Multilayer Perceptron is to predict the week's average for the product. Input Layer: Inputs of the neuron's synapses receive n signals $[P_1, P, P_3, \dots P_n]$ which corresponds to successive n weekly prices for item i. Each synapse makes a linear modification of the signal using its synaptic weight (w_n) which causes the neuron's body receives the following signals $[p_1 w_1, p_2 w_2, p_3 w_3, \dots, p_n w_n]$. The node's signal equals the algebraic sum of the signals entering the node and is given as:

$$S = \sum_{i=0}^{n} w_i p_i \qquad (4)$$

An activation function is then applied to yield output for the neuron given as

$$y_k = f(S_k) \qquad (5)$$

For our model, since we are expecting only positive values as output (predicted price), we will use the sigmoid function given as

$$f(S) = \frac{1}{1+e^{-z}}, \qquad (6)$$

where$Z \in \mathbb{R}$ and thus yielding output between$0 \le y \le 1$.

4.0 IMPLEMENTATION AND DISCUSSION

The work was implemented using JDK- version 1.8.0_111and MySQL community version 5.0.11. The dataset was tested with products on the Amazon website. The historical dataset for the selected products was fetched from the tractor website (https://thetracktor.com) between the 12th and 15th of January, 2017. TheTracktoris an online price history tracking service that presents historical data based on historical price data for an Amazon product when either the product name, Amazon Standard Identification Number (ASIN) or Amazon URL is provided. **graph_options:** anested object which holds the start_timeand end_time values of the fetched product.

4.1 Description and Analysis

Ten (10) products were selected for Amazon.com, a popular e-Commerce store. The products spanned 3 main categories: Clothing, Shoes & Jewelry, Home Improvement, and Electronics. The Amazon Standard Identification Number (ASIN), which uniquely identifies each product, was retrieved. Table 1 shows a summary of the product description.

Table 1: Description of Sample Products

ASIN	Description	Category
B000EQS0WK	Citizen Men's BL5250-02L Eco-Drive Perpetual Calendar Chronograph Watch	Clothing, Shoes & Jewelry
B002SSUQFG	Seiko 5 Men's SNK809 Automatic Black Strap Black Dial Watch	Clothing, Shoes & Jewelry
B000LTAY1U	Seiko Men's SNK805K2 Automatic Green Dial Green Fabric Strap Watch	Clothing, Shoes & Jewelry
B0053EXKFK	Timex Men's T2N700DH Intelligent Quartz T Series Racing Fly Back Chrono Black Ion-Plating Case Sand Strap Watch	Clothing, Shoes & Jewelry
B009CP4FAK	Stanley STST18613 3-in-1 Rolling WorkShop	Home Improvement
B005F1PT32	iRoagent Roomba 780 Vacuum Cleaning Roagent	Home Improvement
B00G9CL83Q	DEWALT DWA2T35IR Impact Ready Screw Driving Set, 35-Piece	Home Improvement
B00HNJWTC8	Sony Alpha a5000 20.1MP SLR Camera (White)	Electronics
B00OBRE5UE	Samsung 850 EVO 2.5-Inch SATA III Internal SSD (MZ-75E500B/AM)	Electronics
B00P0EQD1Q	Dell Computer Ultrasharp U2715H TD5f1 27.00-Inch Screen LED-Lit Monitor	Electronics

Historic prices of the product were sourced from thetracktor.com. The price volatility ratio, which indicates how frequently the price of a product changes over time, was calculated using equation 15.

$$PriceVolatilityRatio = \frac{FrequencyofVatiations}{LengthofData} \qquad (7)$$

The effort was made to ensure the sampled products varied in terms of the length of the historical record available and the volatility of the price of the product. Normalization is done by computing

$$f(x) = \frac{(x-d_L)(n_H-n_L)}{(d_H-d_L)} + n_L \qquad (8)$$

whiledenormalization is given as

$$f(x) = \frac{(d_L-d_H)x-(n_H.d_L)+d_H.n_L)}{(n_L-n_H)}, \qquad (9)$$

where X is the value to be normalized, d_L is the lowest value of data, d_H is the highest value of data, n_L is the lowest normalization range in our instance *0* and h_L = highest normalization range in our instance *1*. Algorithm 1 highlights the normalization process for converting the input stream to a well-defined numeric range between *0* and *1*.

Algorithm 1: Normalization Algorithm

```
Input: Double[]:weeklyAverage
Output: Double[]: weeklyAverageNormalize
Initialize: weeklyAverageNormalize; double i := 0; double x;
double normHigh; double normLow; double factor;
double xH := MAX(weeklyAverage) + factor;
double xL := MIN(weeklyAverage)  - factor;

Begin
denominator = xH - xL
  for i := 0 to length(weeklyAverage)
      x = weeklyAverage[i]
      numerator := (x – xL) * (normHigh – normLow)
     weeklyAverageNormalize[i] := (numerator / denominator) + normLow
  end for
End
```

Table 4 shows the description ofthe Historic Price Dataset of the sampled Products. It would be observed that the volatility ratio varied from 1.99% (sparsely volatile – the price rarely changed during the period) to 67.28% (relatively volatile – on the average, varying more than every other day) for ProductID 5 and 9 respectively. These values are significantly low compared to most other financial time series problems (forex, exchange rate, stock market…), which are relatively more volatile, that is, varying at least daily or multiple times in a day.

Table 2: Historic Price Description of Sample Products

ProductID	Start date	Length of data (days)	Frequency of Price Variations	Price Volatility Ratio
1	25-07-2010	2365	169	7.15%
2	02-08-2010	2387	582	24.38%
3	18-10-2010	2280	538	23.60%
4	12-08-2011	1981	293	14.79%
5	29-11-2012	1510	30	1.99%
6	23-02-2012	1786	628	35.16%
7	02-12-2013	1138	39	3.43%
8	30-12-2013	1114	38	3.41%
9	12-12-2014	764	514	67.28%
10	27-10-2014	812	526	64.78%

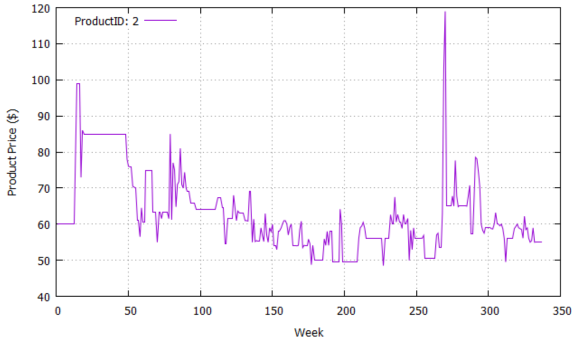

Figure 4:Weekly Time Series Graph of the Products 2

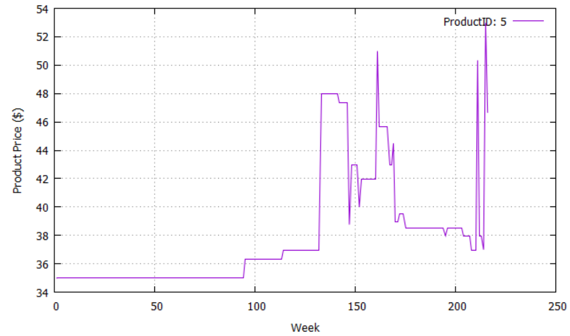

Figure 5: Weekly Time Series Graph of the Products 5

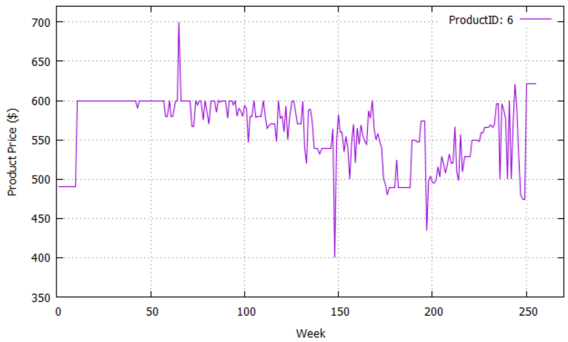

Figure 6: Weekly Time Series Graph of the Products 6

4.2 Multi-Layer Perceptron Neural Network Design

Our proposed model is univariate as the only variable taken into account by the ANN is the price time series. The network configuration implemented for our proposed model as 12 x 20 x 1. That is 12 neurons in the input layer, 20 neurons in a single hidden layer, and 1 neuron in the output layer. This structure was arrived at by experimentations as it gave the optimal result. One hidden layer was adopted because increasing the number of hidden layers also increases computation time and the danger of overfitting.

Table 2 shows a tabulation of the structure of the Neural Network. The 12 neurons in the input layer were fed with the weekly average time-series data points of 12 consecutive weeks using the sliding window technique. The result of the computations of the ANN is the neuron in the output layer holds the forecasted average week's price of the subsequent week.The sigmoid activation function was implemented,considering that our datasets are nonlinear and continuously differentiable.

Table 3: Neural Network Structure and Parameters

Predictive Model	Multi-Layer Perceptron Neural Network
Number of input Neurons	12
Hidden Layer(s)	1
Number of Hidden Neurons	20
Activation Function	Sigmoid activation function
Output Neuron	1

The network design was implemented using Encog core 3.3 Machine Learning Framework.

4.3 Discussion

The standard practice is to divide the dataset into two distinct sets: training dataset and testing datasets. Table 5 shows the training parameters implemented for shoppers. As shown, 70% of the dataset is devoted to training, while the remaining 30% would be used for testing.Training a neural network to learn patterns in the data involves iteratively presenting it with examples to the correct known answers. Our model proposed the use of the Resilient Propagation Training (RPROP) algorithm. To benchmark, a Backpropagation training algorithm was also implemented. In both instances, a maximum error of 0.005 was set as the stopping criteria.

Accessing the efficiency of a training algorithm involves observing the number of iterations (epochs) required for the output of the neural network to match the anticipated output. Table 6 shows the result obtained during the training of our Neural Network model using Resilient Propagation (RPROP) and Backpropagation (BP) training algorithm. It would be observed that for all the sample products, both training algorithms succeed in reaching the training error threshold of 0.005. However, in all instances, RPROP performed better as it required a lower number of epoch to reach the error threshold.

5.0 Conclusion

This study proposed a shopping agent to enhance the purchase of products in an e-store. The model expresses a decision-making feature of shopping agents by suggesting to an eShopper if it is the best time to purchase by comparing the offered price with the week’s forecasted average price. A signal is generated when the offered price is higher than the week’s predicted average price while a buysignal is generated when the price falls below it.

The prediction is handled by the Multi-Layer Perceptron Neural Network (MLP NN) component, which minesthe historical record of the store. The proposed neural network structure has 12 neurons at the input layer, which is fed with the normalized weekly moving average price of the product using the sliding window technique. The input neurons are connected to 20 neurons in a single hidden layer, which in turn is connected to 1 neuron at the output layer. The network was trained using the Resilient Propagation Training (RPROP) algorithm. The results obtained were compared using a separate training algorithm – Back Propagationalgorithm.

The performance indicates that based on historical price data, the average weekly prices of products in e-Commerce sites can be forecasted at a statistically acceptable level. Consequently, the forecast results can be integrated into a shopping agent to serve as a purchase decision-making tool.This can assist an e-Shopper to decide when best to buy.

References

[1] Bakos, J. Y. 1997. Reducing Buyer Search Costs: Implications for Electronic Marketplaces. *Management Science* 43(12):1613-1630.

[2] Blazewicz, J., Bouvry, P., Kovalyov, M.Y. and Musial, J. 2014. Internet Shopping with Price Sensitive Discounts, *4OR—A Quarterly Journal of Operations Research* 12(1): 35–48.

[3] Blazewicz, J., Kovalyov, M. Y., Musial, J., Urbanski, A. P., and Wojciechowski A. 2010. Internet Shopping Optimization Problem. *International Journal of Applied Mathematics and Computer Science* 20(2):385–390.

[4] Braz C and Ncho A, 2003. MBAOS: A Mobile Bargain Agent for Online Shopping. IFT6862 Artificial Intelligence.

[5] Brin, S., and Page, L. 1997. The Anatomy of a Large Scale Hypertextual Web Search Engine, Technical Report, Stanford University, Stanford, CA.

[6] Bynjolfsson, E., and Smith, M. D., and Montgomery, A. L. 2007. `The Great Equalizer? Consumer Choice Behavior at Internet Shopagents'. Working paper. Draft available on World Wide Web at http://e-Commerce .mit.edu/Erik.

[7] Chen, Y., Wu, F., Shuai, W., Wang, N., Chen, R., & Chen, X. (2015, October). Kejia robot–an attractive shopping mall guider. *International Conference on Social Robotics* (pp. 145-154). Springer, Cham.

[8] DiRusso D. J., Mudambi, S. M., and Schuff, D. 2011. Determinants of prices in an Online Marketplace. *Journal of Product & Brand Management.* 20 (5): 420–428.

[9] Doorenbos, R. B., Etzioni, O., and Weld, D. S., 1996. A Scalable Comparison-Shopping Agent for the World-Wide Web.

[10] Dudarenko, D.M., & Smirnov, P.A. (2020). Configuring hyperparameters of an artificial neural network for navigating a mobile platform. *Proceedings of the Southwestern State University , 23* (6), 115-132.

[11] Eddahibi, M., Nejeoui, A., and Cherkaoui, C., 2012. Towards an Intelligent and Deeply Automated Shopping Agent. *International Journal of Computer Applications* 45(24): pp

[12] Ghuli P. and Shettar R. 2014. A Novel Approach to Implement a Shopping Agent on a Scalable Web Crawler. International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, Issue 1, pp 5326 – 5326.

[13] Graupe, D., 2007. Principles of Artificial Neural Networks, 2nd Edition. World Scientific Publishing Co. Pte. Ltd

[14] Gurau, C., 2009. How Good is Your Shopping Agent? Users’ Perception Regarding Shopping Agents’ Service Quality.

[15] Hajaj, C., Hazon, N., and Sarne, D. 2015. Improving Comparison Shopping Agents’ Competence Through Selective Price Disclosure. *Electronic Commerce Research and Applications* 14:563–581

[16] Heaton, J., 2011. Programming Neural Networks with Encog3 in Java. Heaton Research, Inc., Missouri, USA. https://www.statista.com/outlook/243/100/e-commerce/worldwide#

[17] Huang S. L., Tsai Y. H. 2011. Designing a Cross-Language Comparison-Shopping Agent, Decision Support Systems 50, Pages 428–438

[18] Huang, S. L., and Tsai, Y. H. 2009. Developing a Shopagent with Multilingual Ontology for Global E-Commerce. *Journal of Internet Technology*. 10 (2): 111-118.

[19] Huang, S. L., Tsai, Y. H. 2011. Designing a Cross-Language Comparison-Shopping Agent. *Decision Support Systems* 50: 428–438.

[20] Janssen, M. C. W., Moraga, J. L., Wildenbeest, M. R., Brousseau, E., Curien, N., 2007. Consumer Search and Pricing Behavior. In: Brousseau, E. and Curien, N. (eds) Internet Markets, Internet and Digital Economics. Cambridge University Press, Cambridge, UK.

[21] Jumadinova, J., and Dasgupta, P., 2008. Firefly-Inspired Synchronization for Improved Dynamic Pricing in Online Markets In Proceedings of the Second IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO), pp. 403–412.

[22] Kanda, T., Shiomi, M., Miyashita, Z., Ishiguro, H., &Hagita, N. (2009, March). An affective guide robot in a shopping mall. In *Proceedings of the 4th ACM/IEEE international conference on Human-robot interaction* (pp. 173-180).

[23] Karthikeyan, P., and Sathiyamoorthy, E., 2012. A Survey of Applications of Mobile Agents in E-Business.

[24] Karunarathne, D., Morales, Y., Kanda, T., & Ishiguro, H. (2020). Understanding a public environment via continuous robot observations. *Robotics and Autonomous Systems*, *126*, 103443.

[25] Kephart, J. O., and Greenwald, A. R. 2002. Shopagent Economics. *Autonomous Agents and Multi-Agent Systems* 5 (3): 255-287.

[26] Kotler P. and Armstrong G. 1991. Principles of Marketing (5th ed.), Prentice-Hall, Englewood Cliffs, New Jersey.

[27] Lange, D. B., and Oshima, M., 1998. Programming and Deploying Java mobile agents with Aglets, Addison-Wesley.

[28] Liang C. C. and Liang W., 2014. Efficient communication architecture for the C2C agent. *Computer Standards and Interfaces* 36 (3): 641-647.

[29] Liu, W., Jing, Z., D'Eleuterio, G., Chen, W., Yang, T., & Pan, H. (2019, February). Shape Memory Alloy Driven Soft Robot Design and Position Control Using Continuous Reinforcement Learning. In *2019 2nd International Conference on Intelligent Autonomous Systems (ICoIAS)* (pp. 124-130). IEEE.

[30] Lopez-Loces, M. C., Musial, J., Pecero, J. E., Fraire-Huacuja, H. J., Blazewicz, J., and Bouvry P. 2016. Exact and Heuristic Approaches to Solve the Internet Shopping Optimization Problem with Delivery Costs. *International Journal of Applied Mathematics and Computer Science* 26(2): 391–406.

[31] Maes, P., Guttman, R. H., and Moukas A. G. 1999. Agents that Buy and Sell – Transforming Commerce As We Know It. *Communications of the ACM* 42: 81–91.

[32] Mansell G. 2014: Consumers' Attitudes Towards Sustainability, Ethics & Shopping TradeExtensions Research Bulletin.

[33] Meher K. N., Jadhav S., and Lokhande P. S. 2013. Implementation of Mobile Agent Architecture for E-Commerce Application for Mobile Devices. *International Journal of Advanced Research in Computer Engineering & Technology* 2 (12): 3033-3041.

[34] Musial, J., 2012. Applications of Combinatorial Optimization for Online Shopping, NAKOM, Pozna´n.

[35] Nanavati, A., Tan, X. Z., Connolly, J., &Steinfeld, A. (2019, November). Follow The Robot: Modeling Coupled Human-Robot Dyads During Navigation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 3836-3843). IEEE.

[36] Nath, R., Bal, S., and Singh, M., 2007. Load Reducing Techniques the Websites and other Resources: A comparative Study and Future Research Directions. *Computer Journal of Advanced Research in Computer Engineering* 1 (1): 39-49.

[37] Niemelä, M., Heikkilä, P., Lammi, H., &Oksman, V. (2019). A social robot in a shopping mall: studies on acceptance and stakeholder expectations. In *Social Robots: Technological, Societal, and Ethical Aspects of Human-Robot Interaction* (pp. 119-144). Springer, Cham.

[38] Perera, V., & Veloso, M. (2018, September). Interpretability of a Service Robot: Enabling User Questions and Checkable Answers. In *GCAI* (pp. 176-187).

[39] Repiso, E., Garrell, A., &Sanfeliu, A. (2019). Adaptive side-by-side social robot navigation to approach and interact with people. *International Journal of Social Robotics*, 1-22.

[40] Riedmiller, M., and Braun, H., 1993. A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm.

[41] Rodriguez, I., Astigarraga, A., Lazkano, E., Martínez-Otzeta, J. M., &Mendialdua, I. (2018). Robots on stage: A cognitive framework for socially interacting robots. *Biologically inspired cognitive architectures*, *25*, 17-25.

[42] Sadeddin, K. W., Serenko A., and Hayes J., 2007. Online Shopping Agents for Electronic Commerce: The Comparison of functionality and performance. *International Journal of Electronic Business*. 5(6): 576–589.

[43] Sapna, S., Tamilarasi, A., and Kumar M. P. (2012). Backpropagation Learning Algorithm Based on the Levenberg Marquardt Algorithm. *The Fourth International Workshop on Computer Networks & Communications.* 393–398.

[44] Serenko A and Hayes J, 2010. Investigating the functionality and performance of online shopping agents for electronic commerce: a follow-up study. *International Journal of Electronic Business* 8(1): 1–15.

[45] Singh, M., 2013. Important Pricing Methods in 4P's. *International Journal of Management and Business Studies* 3(2).

[46] Smith, M. D., 2002. The Impact of Shopagents on Electronic Markets. *Journal of the Academy of Marketing Science* 30(4): 442-450.

[47] Stigler, G., 1961. The Economics of Information. *Journal of Political Economy* 69 (Jan-Feb): 213-225.

[48] Vaidyanathan, R., and Aggarwal, P., 2002. The Impact of Shopping Agents on Small Business E-Commerce Strategy. *Journal of Small Business Strategy*13 (1): 63-79.

[49] Wu, Y., & Wang, Y. (2019). Asymptotic tracking control of uncertain nonholonomic wheeled mobile robot with actuator saturation and external disturbances. *Neural Computing and Applications*, 1-11.

[50] Zhang, H., Cao, R., Zilberstein, S., Wu, F., & Chen, X. (2017, August). Toward effective soft robot control via reinforcement learning. In *International Conference on Intelligent Robotics and Applications* (pp. 173-184). Springer, Cham.