IJARSCT

ISSN (Online) 2581-9429

International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)

Impact Factor: 6.252

Volume 2, Issue 1, August 2022

A Review On: Retailer Pricing Analysis using Machine Learning

Venkata Vamshi Krishna[1] and Kandakatla Mahesh[2]
Ph.D Scholar, Department of Computer Science and Engineering[1]
Research Supervisor, Department of Computer Science and Engineering[2]
Chaitanya Deemed to be University, Hanamkonda, Telangana, India

Abstract: *In this research, investigates deeply on price competition behavior between mobile commerce retailers (MCR) and traditional off-line retailers (TOR) in m-commerce era. On this basis, two related propositions are pointed out. First, it gets MCR's and TOR's perfect prices in competition, further analysis can find that in the competition between MCR and TOR when these two kinds of retailers coexist, two exterior factors and three inner factors decide the price difference. Second, as the penetrate rate of m-commerce increasing, perfect prices of two kinds of retailers will change, and the ratio of their change speed will be stable. The problem addresses price competition decisions to assist retailers in deciding just the right price in MCR and TOR while still subject to uncertainty. Efficient time series forecasting techniques, auction based market mechanism, Spot pricing, including the use of machine learning models, will help to reduce uncertainty and improve results by offering insight on future outcome-based decisions. One of the algorithms of machine learning that is used linear regression. It is easy to implement and is used for purposes such as predicting real estate prices, financialperformances and traffic. Cluster analysis is also used for the purpose of understanding the relationship between consumers and the products they mostly search for. This work proposes a hybrid model, exploring linear and nonlinear modelling. This information is used to aid the decision-maker in an optimization problem involving both the decision on the price in MCR and TOR and the best moment for the one-time per cycle retailer price replenishment in situations where the purchase price fluctuates in time. This work aims to present Regression Random Forests (RRFs) model to predict one- week-ahead and one-day-ahead spot prices. The prediction would assist retailer to plan in advance when to acquire spot instances, estimate execution costs, and also assist them in bid decision making to minimize execution costs and out-of-bid failure probability. In addition, we contribute to the literature by developing a detailed MCR and TOR channel choice model to drive the MCR and TOR operational decisions.*

Keywords: Machine Learning.

I. INTRODUCTION

Machine learning (**ML**) is a field of inquiry devoted to understanding and building methods that 'learn', that is, methods that leverage data to improve performance on some set of tasks. It is seen as a part of artificial intelligence. Machine learning algorithms build a model based on sample data, known as training data, in order to make predictions or decisions without being explicitly programmed to do so. Machine learning algorithms are used in a wide variety of applications, such as in medicine, email filtering, speech recognition, and computer vision, where it is difficult or unfeasible to develop conventional algorithms to perform the needed tasks.

A subset of machine learning is closely related to computational statistics, which focuses on making predictions using computers, but not all machine learning is statistical learning. The study of mathematical optimization delivers methods, theory and application domains to the field of machine learning. Data mining is a related field of study, focusing on exploratory data analysis through unsupervised learning.

Some implementations of machine learning use data and neural networks in a way that mimics the working of a biological brain. In its application across business problems, machine learning is also referred to as predictive analytics.

Machine learning approaches are traditionally divided into three broad categories, depending on the nature of the "signal" or "feedback" available to the learning system:

DOI: 10.48175/IJARSCT-7009

- Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.
- Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end (feature learning).
- Reinforcement learning: A computer program interacts with a dynamic environment in which it must perform a certain goal (such as driving a vehicle or playing a game against an opponent). As it navigates its problem space, the program is provided feedback that's analogous to rewards, which it tries to maximize.

1.1 Supervised Learning

A support-vector machine is a supervised learning model that divides the data into regions separated by a linear boundary. Here, the linear boundary divides the black circles from the white.

Supervised learning algorithms build a mathematical model of a set of data that contains both the inputs and the desired outputs.[35] The data is known as training data, and consists of a set of training examples. Each training example has one or more inputs and the desired output, also known as a supervisory signal. In the mathematical model, each training example is represented by an array or vector, sometimes called a feature vector, and the training data is represented by a matrix. Through iterative optimization of an objective function, supervised learning algorithms learn a function that can be used to predict the output associated with new inputs. An optimal function will allow the algorithm to correctly determine the output for inputs that were not a part of the training data. An algorithm that improves the accuracy of its outputs or predictions over time is said to have learned to perform that task.

Types of supervised-learning algorithms include active learning, classification and regression. Classification algorithms are used when the outputs are restricted to a limited set of values, and regression algorithms are used when the outputs may have any numerical value within a range. As an example, for a classification algorithm that filters emails, the input would be an incoming email, and the output would be the name of the folder in which to file the email.

Similarity learning is an area of supervised machine learning closely related to regression and classification, but the goal is to learn from examples using a similarity function that measures how similar or related two objects are. It has applications in ranking, recommendation systems, visual identity tracking, face verification, and speaker verification.

1.2 Unsupervised Learning

Unsupervised learning algorithms take a set of data that contains only inputs, and find structure in the data, like grouping or clustering of data points. The algorithms, therefore, learn from test data that has not been labeled, classified or categorized. Instead of responding to feedback, unsupervised learning algorithms identify commonalities in the data and react based on the presence or absence of such commonalities in each new piece of data. A central application of unsupervised learning is in the field of density estimation in statistics, such as finding the probability density function. Though unsupervised learning encompasses other domains involving summarizing and explaining data features.

Cluster analysis is the assignment of a set of observations into subsets (called *clusters*) so that observations within the same cluster are similar according to one or more predesignated criteria, while observations drawn from different clusters are dissimilar. Different clustering techniques make different assumptions on the structure of the data, often defined by some *similarity metric* and evaluated, for example, by *internal compactness*, or the similarity between members of the same cluster, and *separation*, the difference between clusters. Other methods are based on *estimated density* and *graph connectivity*.

1.3 Semi-Supervised Learning

Semi-supervised learning falls between unsupervised learning (without any labeled training data) and supervised learning (with completely labeled training data). Some of the training examples are missing training labels, yet many machine-learning researchers have found that unlabeled data, when used in conjunction with a small amount of labeled data, can produce a considerable improvement in learning accuracy.

In weakly supervised learning, the training labels are noisy, limited, or imprecise; however, these labels are often cheaper to obtain, resulting in larger effective training sets.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Impact Factor: 6.252
Volume 2, Issue 1, August 2022

1.4 Reinforcement Learning

Reinforcement learning is an area of machine learning concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward. Due to its generality, the field is studied in many other disciplines, such as game theory, control theory, operations research, information theory, simulation-based optimization, multi-agent systems, swarm intelligence, statistics and genetic algorithms. In machine learning, the environment is typically represented as a Markov decision process (MDP). Many reinforcement learning algorithms use dynamic programming techniques.[39] Reinforcement learning algorithms do not assume knowledge of an exact mathematical model of the MDP, and are used when exact models are infeasible. Reinforcement learning algorithms are used in autonomous vehicles or in learning to play a game against a human opponent.

II. LITERATURE REVIEW

2.1 The Newsvendor Problem

$$\min C(y) = E[\ c_o * \max\{O, D - y\} + c_u * \max\{O, y - D\}\ ]$$

This work investigates the application of times series forecasting methods, such as the traditional seasonal autoregressive integrated moving average (SARIMA) [12] and more recent sophisticated Prophet [13], developed by Facebook, in the purchase price modeling, as well as a hybrid method using Multi-Layer Perceptrons (MLP) [14]

2.2 Time Series Forecasting

One of the most consecrated methods for time-series forecasting is the autoregressive integrated moving average (ARIMA) model [12]. It is an aggregation of both the autoregressive (AR) and the moving average (MA) models for stationary time-series, with an additional factor that considers a possible stationary behavior of the series after a certain number of differentiations. The SARIMA model is a modification of the ARIMA which accounts for the seasonal variation of the series in order to make a better prediction. A SARIMA (p,d,q)(P,D,Q)m model is described by 7 values that describe the series' behavior and a time-step value's dependency on previous values of the series: i) p: Trend autoregression order, ii) d: Trend difference order, iii) q: Trend moving average order, iv) P: Seasonal autoregressive order, v) D: Seasonal difference order, vi) Q: Seasonal moving average order, v) m: The number of time steps for a single seasonal period.

$$P(D \leq y) = {}^{c\mathrm{u}}\ c_O{+}c_\mathrm{u}$$

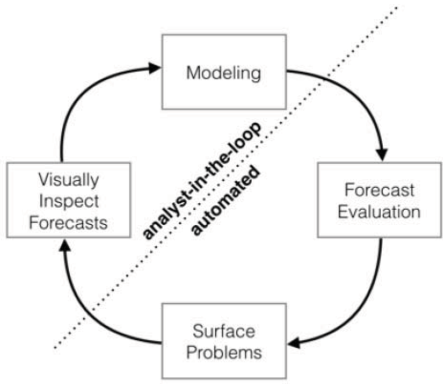
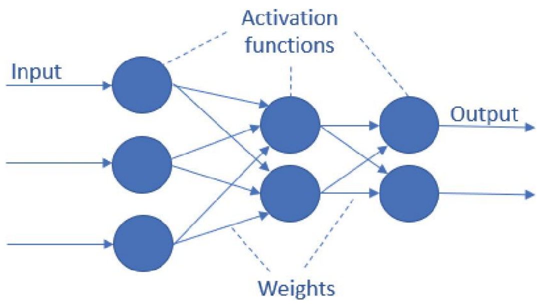


Fig. 1. Diagram of Analyst-in-the-loop user-machine collaboration.

Multi-Layer Perceptrons (MLP) [14] are widely applied Machine Learning models consisting of connected elements called neurons that combine data through a function and feed other elements of the network. With vast applicability due to their flexibility in design and architecture, these models are powerful investigators of possible combinations and frequently helpful in recognizing complex patterns.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Impact Factor: 6.252
Volume 2, Issue 1, August 2022



2.3 Optimization Model

In order to maximize his profit, the retailer must make two decisions. The decision variables are the order size y and purchase time t.

$$\min C(y, t) = E[\ c_o(t) * \max\{O, D - y\} + c_u * \max\{O, y - D\}](3)y,t$$

III. PROBLEM DESCRIPTION AND METHODOLOGY

The context of this problem is drawn around a retailer that must refresh its perishable products every cycle of s fixed time steps. At the beginning of each cycle, it must dispose of the previous inventory and begin selling the new one. It respects all three criteria for newsvendor-type problems: i) the retailer is subject to an unknown random demand. In this work, the demand is considered to follow a Normal distribution pattern;

ii) a single order is placed for each time period, with the additional factor that he must decide at which time step of the initiating cycle he will place the order for the following one; iii) underage costs are known, and overage costs are a function of the purchase price, which is fixated once the retailer has decided on the date of purchase.

The problem is based on the real case of a Brazilian supermarket chain which makes weekly purchases of perishable items. The prices of most of these items are subjectto fluctuations due to environmental and economic factors such

In order to solve for the order size y using (2), one must first solve for t. Since the objective function is monotonically increasing with respect to c_O (t) ($\max\{O, D - y\}$ is strictly non- negative), it is a secondary objective to minimize c_O (t). This overage cost is traditionally translated as the difference between the unit purchase price and the unit salvage value [15]. In the context of the supermarket, no salvage revenue was considered, due to the health hazard of expired food products. Also, disposalof scrap material was considered as a fixed cost. Therefore, the unit overage cost is solely described by the unit purchase price. In order to decide on the purchase time t that minimizes c_O(t), one must analyze the behavior or the stochastic variable.

3.1 Pure Forecasting Models

In order to analyze and decide upon the purchase prices of meat, one must hold historical data of daily behavior of the variable. Due to sparse data held by the supermarket on daily purchase price historical behavior – the only data points available are those of dates in which there was in fact a purchase – it is necessary to use an additional time series with high correlation and daily availability. Multiple series were tested until a correlation of 0.896 was found between monthly meat price fluctuations in Brazil [16] and a monthly average of the Live Cattle BOVESPA Futures prices [17]. Based on this high correlation, the Live Cattle Futures prices variations were used to estimate daily meat prices. Publicly available daily values for this index from Jan 1st, 2016 to Aug 7, 2020 were used to model the series and estimate prices (Fig. 2).

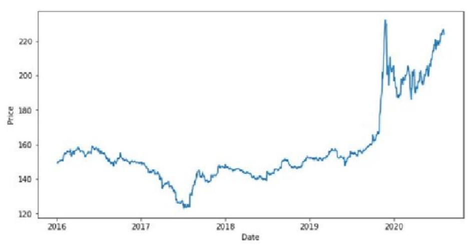


Fig. 2. Timeline of Live Cattle BOVESPA stock prices.

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 1, August 2022
Impact Factor: 6.252

3.2 Hybrid Forecasting Model

An alternative hybrid model designed to retrieve complex patterns from the time series is an MLP, in which the inputs are predictions outputted by the SARIMA and Prophet models, as well as previous values from the time series itself. Fig. 3 illustrates the single hidden layer model in which each data sample would have as features in the machine learning model the prediction of each pure model for the time step to be predicted by the MLP ($\hat{T}_t$). Besides the initial hybrid model, other variations using up to 5 previous predictions by the pure models were tested ($\hat{T}_{t-i}$, $i \in [O,5]$). In each case, the number of previous predictions used as input was the same as the number of previous values from the time series used in the model.

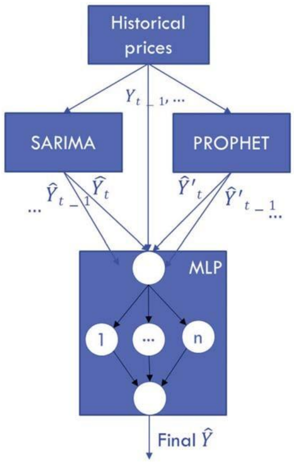


Fig. 3. SARIMA, Prophet and MLP hybrid model.

Long Short-Term Memory (LSTM) is a special kind ofRecurrent Neural Network (RNN) capable of learning long-term dependencies. It's also capable of catching data frompast stages and using it for future predictions. LSTMs are explicitly designed to avoid the long-term dependency prob- lem. Remembering information for long periods is practically their default behavior, not something they struggle to learn. A common LSTM unit is composed of a cell, an input gate. A different number of stocks with different risk levels are used. For low-risk stocks [2] we used A10 Networks (ATEN), Core-Mark Holding Company (CORE), Eli Lilly and Company (LLY), and Chemed Corpo-ration (CHE). For medium-risk [3], we used iShares 20+ Year Treasury Bond ETF (TLT), Vanguard Extended Duration Trea- sury ETF (EDV), iShares Barclays 10-20 Year Treasury Bond Fund (TLH), and Vanguard Long-Term Government Bond ETF (VGLT). For high-risk stocks [1], we used Household robotics company (IRBT), GameStop (GME), Jumia (JMIA), and New Age Beverages (NBEV).

3.3 LSTM Model Hyper-Parameters

Our LSTM model is composed of 2 layers, one hidden and one output. The hidden layer is with timesteps of 120 days, 50 neurons, *batchsize* = 32, and tanh is used as activation function. The input weights of the hidden layer are initialized randomly. The output layer contains one neuron to predictthe closing price of the next day. Closing price values were normalized in the range of 0–1, before using them as an input for the neural network. This normalization prevents high-value inputs from dominating the final output. We applied different values of training periods (5 years and 15 years). Also, different Epochs for training data are applied (25 epochs, 50 epochs, and 100 epochs). Each financial time series datasetwas split into 80-20 subsets where 80% of the data was used for training and the remaining 20% was used for testing. 'Adam' optimizer is applied to minimize the loss function witha learning rate of 0.001. A dense layer connection has been used with a dropout rate of 20 to avoid overfitting.

3.4 Evaluation Methodology

The root-mean-square error (RMSE) is a frequently used measure of the differences between values (sample or popu-lation values) predicted by a model or an estimator and the values observed. The RMSD represents the square root of the second sample moment of the differences between predicted values and observed values or the quadratic mean of

IJARSCT
ISSN (Online) 2581-9429
International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)
Volume 2, Issue 1, August 2022
Impact Factor: 6.252

these differences. RMSE is given in Equation 1,where y_i is the observed value and x_i is the prediction for the corresponding observed value.

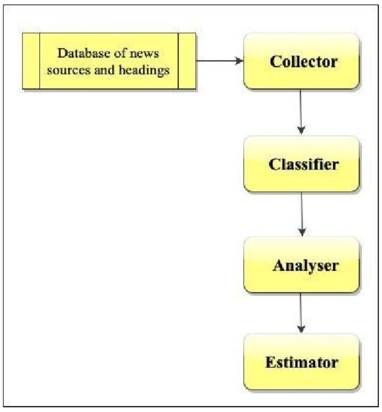


Fig. 1: Flowchart of our methodology

IV. PREDICTION

A sequential troupe of daily trading details of stocks over a set time period of N days is defined in the LSTM model. These daily details in sequence describe the trends of the stock with the attributes like the day high/low, open price, closing price and trade volume on a specific day within the N days. Comparing in sequence the closing prices of 3 consecutive trading sessions with that of the last day, the earning rates were calculated. The model comprises of two layers namely, an input layer which consists of number of cells equal to the sequence learning attributes that one sequence may hold, the LSTM layers, a compact layer, and an output layer consisting of similar number of cells. There are 4 types of learning features that could be given to the LSTM models. They are:

1. The historical trade details.
2. The technical analysis derived from thesehistorical trade details.
3. The movement of the market indices.
4. The economic fundamentals.

First and the third type of data is essential for theforecasting of the prices of different stocks.

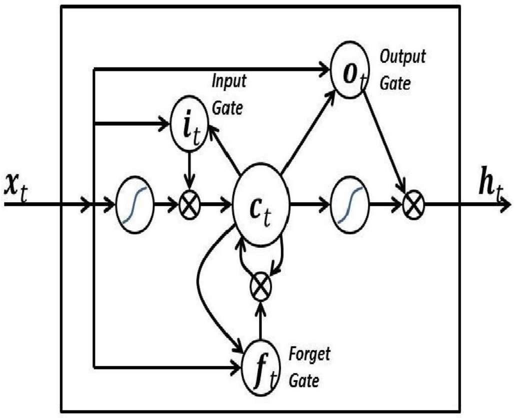


Fig. 2: Long Short-Term Memory Architecture[9].

V. CONCLUSION

The proposed hybrid model proved to outperform both benchmark time series forecasting techniques in providing the best decision-making for a newsvendor-type procurementsituation.

All 6 hybrid MLP models showed the best performance with the largest number of neurons in their hidden layer, 100 of them, although the number of variables was very small for such a large model. This may suggest that there is more complexity to the dataset than the models have been able to address. It is still underfitting. Adding more lags can significantly improve results. Interestingly, although the hybrid models showed the worst performance in predicting the value of the series, they have succeeded in offering better insight into the decision as to which time step to place the order (with the exception of the simplest case of the hybrid model, which showed to be the least reliant). This shows that the best predictors are not necessarily the best policy advisors.

Another interesting observation is that, although the use of these models shows improvement over arbitrary policies, the best hybrid model tested still accounts for a purchase policy that is 1% more expensive than it could be. One percent saved on regular, high volume product procurement can save companies millions of dollars. This shows there is still plenty of opportunity for improvement, testing increasingly sophisticated techniques. Future work involving Recurrent Neural Networks and other successful time series forecasting methods, including hybrid models, should show promising results

REFERENCES

[1]. H. Chen, R. H. Chiang and V. C. Storey, "Business intelligence and analytics: From big data to big impact," *MIS Quarterly*, vol. 36, no. 4, pp. 1165–1188, 2012.

[2]. R. Jaffari, M. Memon, T. Hafiz and R. Iftikhar, "Framework of business intelligence systems for infrastructure design and management," in *Int. Conf. on Latest Trends in Electrical Engineering and Computing Technologies (INTELLECT)*, Karachi, Pakistan: IEEE, pp. 1–8, 2017.

[3]. D. Arnott, F. Lizama and Y. Song, "Patterns of business intelligence systems use in organizations," *Decision Support Systems*, vol. 97, pp. 58–68, 2017.

[4]. Z. Jourdan, R. K. Rainer and T. E. Marshall, "Business intelligence: An analysis of the literature," *Information Systems Management*, vol. 25, no. 2, pp. 121–131, 2008.

[5]. P. Wlodarczak, M. Ally and J. Soar, "Data mining in IOT: Data analysis for a new paradigm on the internet," in

[6]. *Proc. of the Int. Conf. on Web Intelligence*, Leipzig, Germany, ACM, pp. 1100–1103, 2017.

[7]. D. D. Phan and D. R. Vogel, "A model of customer relationship management and business intelligence systems for catalogue and online retailers," *Information & Management*, vol. 47, no. 2, pp. 69–77, 2010.

[8]. F. Wang and M. Head, "How can the web help build customer relationships? An empirical study on e-tailing,"

[9]. *Information & Management*, vol. 44, no. 2, pp. 115–129, 2007.

[10]. R. Amit and C. Zott, "Value creation in e-business," *Strategic Management Journal*, vol. 22, no. 6–7, pp. 493– 520, 2001.

[11]. G. Linden, B. Smith and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, 2003.

[12]. E. Brynjolfsson, Y. J. Hu and M. D. Smith, "From niches to riches: Anatomy of the long tail," *Sloan Management Review*, vol. 47, no. 4, pp. 67–71, 2006.

[13]. D. S. Soper, "A framework for automated web business intelligence systems," in *Annual Hawaii Int. Conf. on System Sciences (HICSS)*, Big Island, HI, USA: IEEE, pp. 217a, 2005.

[14]. H. W. Ahmad, S. Zilles, H. J. Hamilton and R. Dosselmann, "Prediction of retail prices of products using local competitors," *International Journal of Business Intelligence and Data Mining*, vol. 11, no. 1, pp. 19–30, 2016.

[15]. S. Kisilevich, D. Keim and L. Rokach, "A GIS-based decision support system for hotel room rate estimation and temporal price prediction: The hotel brokers' context," *Decision Support Systems*, vol. 54, no. 2, pp. 1119–1133, 2013.

[16]. H. Zhao, "Futures price prediction of agricultural products based on machine learning," *Neural Computing andApplications*, vol. 33, no. 3, pp. 837–850, 2021.

[17]. L. Xuefeng, L. Lu, W. Lihua and Z. Zhao, "Predicting the final prices of online auction items," *Expert Systems with Applications*, vol. 31, no. 3, pp. 542–550, 2006.

[18]. N. Archak, A. Ghose and P. G. Ipeirotis, "Show me the money! Deriving the pricing power of product features by mining consumer reviews," in *Proc. of the 13th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data*

Mining, San Jose, California, USA: ACM, pp. 56–65, 2007.

[19]. N. B. Lassen, R. Madsen and R. Vatrapu, "Predicting iPhone sales from iPhone tweets," in *18th Int. Enterprise Distributed Object Computing Conf.*, Ulm, Germany: IEEE, pp. 81–90, 2014.

[20]. K. K. Tseng, R. F. Y. Lin, H. Zhou, K. J. Kurniajaya and Q. Li, "Price prediction of e-commerce products through Internet sentiment analysis," *Electronic Commerce Research*, vol. 18, no. 1, pp. 65–88, 2018.

[21]. S. Rajagopal, K. S. Hareesha and P. P. Kundapur, "Performance analysis of binary and multiclass models using azure machine learning," *International Journal of Electrical & Computer Engineering*, vol. 10, no. 1, pp. 2088– 8708, 2020.

[22]. F. Abramovich, V. Grinshtein and T. Levy, "Multiclass classification by sparse multinomial logistic regression,"arXiv preprint arXiv, vol. 1, e print 2003.01951, 2020.

[23]. B. Siswoyo, "MultiClass decision forest machine learning artificial intelligence," *Journal of Applied Informatics and Computing (JAIC)*, vol. 4, no. 1, pp. 1–7, 2020.

[24]. J. Yan, Z. Zhang, K. Lin, F. Yang and X. Luo, "A hybrid scheme-based one-vs-all decision trees for multi-class classification tasks," *Knowledge-Based Systems*, vol. 198, no. 4, pp. 105922, 2020.

[25]. I. D. E PyCharm Python, "For Professional Developers," (Accessed on 12-04-2021), 2000. [Online]. Available:https://www.jetbrains.com/pycharm/.

[26]. Microsoft, "SQL Server Management Studio 2018," (Accessed on 12-04-2021), 2019. [Online]. Available: https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms.

[27]. Microsoft Azure, "Microsoft Azure Machine Learning Studio," (Accessed on 12-04-2021), 2021. [Online]. Available: https://azure.microsoft.com/en-us/services/machine-learning/.

[28]. Sammut and G. I. Webb, "Encyclopedia of machine learning," *Springer Science & Business Media*, Springer, New York, 2011, 978-0-387-30768-8.