E-Commerce Price Forecasting Using LSTM Neural Networks

Houda Bakir, Ghassen Chniti, and Hédi Zaher

***Abstract*—In this paper, we provide a robust forecasting model to predict phone prices in European markets using Long Short-Term Memory (LSTM) neural network and Support Vector Regression (SVR). We propose a comparison study of time series forecasting models for these two techniques. LSTM, due to its architecture, is considered as the perfect solution to problems not resolvable by classic Recurrent Neural Networks (RNNs). On the other hand, Support Vector Machines (SVMs) are a very powerful machine learning method for both classification and regression. After studying and comparing several univariate models, SVR and LSTM neural networks appear to be the most accurate ones. In addition, we compared multivariate models for both these techniques. Considering the multivariate approach, by introducing more variables, we obtain better prediction performance. In fact, the SVR model is able to predict the next day price with an root mean squared error (RMSE) of 33.43 euros with the univariate model. However, using multivariate models, LSTM RNN gives the most accurate prediction for the next day's price with an RMSE of 23.640 euros.**

***Index Terms*—Time series forecasting, LSTM neural network, support vector regression, e-commerce data, machine learning, deep learning.**

I. Introduction

In the last few decades, e-commerce markets have grown almost exponentially worldwide. According to a 2016 global B2C e-commerce report made by the e-Commerce Foundation [1], the global B2C turnover is about 2,272.7 billion dollars in 2015, which represents an increase of 19.9% over the turnover for the previous year. This increase can be explained by different acting indicators such as the rate of penetration into the market, Internet accessibility, economic indicators, etc. Many retailers and market places, like Amazon, invest in research and development to the tune of 1.86 billion dollars in order to further increase their turnover. The possibility to adjust the price by forecasting competitors' product prices is similar to predicting the financial stock prices when you have adequate competitor's time series data, which essentially is the collection of the entire displayed data whether with time stamps or implicitly indicating that the data is collected from a process that varies with time [2].

Studying a time series assumed an important role in human science since the time trajectory observations of planets were made by the Babylonians [3] and Mayans. They could predict astronomical events with notable accuracy. These observations formed the base for Kepler's laws for planetary motion. Time series analyses allow the detection of cyclic or similar patterns in an observed variable. In the 19th century, this methodology was applied to economic problems by William Stanley Jevons [4]. Decomposing a time series into independent unobserved components was developed by Warren M. Persons in 1919 [5]. There are four components: a trend, a cyclical component, a seasonal cycle containing ups and downs, and a residual component containing all the other observed movements. This approach forms the basis of statistical methodologies for time series analyses and forecasting.

The main purpose of time series modeling is to carefully collate and rigorously study the past observations of a time series in order to develop an appropriate model that describes the inherent structure of the series. A forecast is then generated using this model. In other terms, time series forecasting can be expressed as the ability of predicting the future values by understanding the past. Predicting the evolution of some variables in various fields such as finance [6], insurance [7], and oceanography [8] is very valuable and priceless; this method makes scientists and researchers more rigorous in building and fitting the most adequate model for their time series.

Over many years, researchers have been working on different linear statistical models, such as ARIMA [9], to improve forecasting accuracies. However, in the late 1970s and early 1980s, it became increasingly clear that linear models are not a good fit for many real applications. During that period, researchers used several nonlinear time series models, such as the bilinear model, the threshold autoregressive model, and the autoregressive conditional heteroscedastic (ARCH) [10], [11] model. Until recently, certain robust machine learning models such as Long Short-Term Memory (LSTM) and Support Vector Regression **(**SVR) have established themselves as serious contenders to classical statistical models in the forecasting community. These techniques were freshly implemented in the interest of forecasting in many fields [12], [13]. Such models, also called data-driven models, are non-parametric, nonlinear models that only use historical data to learn the stochastic dependency between the past and the future.

In Section II, we cite some related works and we will introduce the two forecast methods used, namely, the SVR and LSTM neural networks. In Section III, we present our

Manuscript received January 12, 2018; revised March 3, 2018.

Houda Bakir is with the Laboratory CEREP, Centre de Recherche en Productique Ecole Nationale Tunisia. Avenue Taha Hussein Montfleury, 1008 Tunis, Tunisia and now she is with Datavora Research and Development unit. 45 Avenue du Japon Tunis 1073, Tunisia (e-mail: houda.bakir@datavora.com).

Ghassen Chniti was with the National Engineering School of Tunis, Tunisia 45 Avenue du Japon Tunis 1073, Tunisia (e-mail: ghasen.chniti@datavora.com).

Hédi Zaher was with Laboratoire Tech-CICO (U.T.T), France. Now he is with Datavora. 45 Avenue du Japon Tunis 1073, Tunisia (e-mail: hedi.zaher@datavora.com).

doi: 10.18178/ijmlc.2018.8.2.682

contribution by detailing our methodology and the experimental results. Finally, the conclusion and perspectives are given in Section IV.

II. Background

A. Related Works

The field of artificial intelligence (AI) includes the techniques and methods of machine learning and deep learning. Machine learning techniques can be classified into four categories: Supervised Learning, Unsupervised Learning, Reinforcement Learning, and Competitive Learning. Supervised Learning can be used for classification and regression. Unsupervised Learning is used for clustering, anomalies detection, and dimension reduction. Reinforcement Learning is used for behavior simulation, e.g., robotics, game AI, and self-driving cars.

Many works in the literature have used machine learning and deep learning techniques for time series predictions, especially in financial time series. Martin *et al.* [12] presented a Recurrent Neural Network (RNN) to predict future time series values. Mukherjee *et al.* [14] used a Support Vector Machine (SVM) for nonlinear time series prediction. Lee Giles *et al.* [15] used RNNs and Grammatical Inference to predict changes in foreign exchange rates for the next day. Cao and Tay [16] demonstrated that a SVM with a regularized radial basis function can yield a better prediction than a Back Propagation Neural Network for financial forecasting (e.g., EUREX-BUND and MATIF-CAC40). Assaad *et al.* [17] presented a boosting algorithm for forecasting time series with RNNs. Kuremotoa *et al.* [18] used Deep Belief Nets, which are probabilistic generative neural networks composed of three layers of Restricted Boltzmann Machine for time series forecasting. Many research works used the LSTM RNNs to predict future values from a time series. However, it was initially introduced by Bontemps *et al.* [19] for time series analysis as a method for detecting anomalies. Gers *et al.* [20] used a time window approach for the LSTM to predict the time series. In this study, we used this approach with a 7-day window.

B. SVR

A SVM was first used to resolve classification issues. Later, this machine learning method was developed to be the perfect fit for regression problems [20].

Suppose we are given training data of the form $\{(x_i, y_i); i=1\ldots n\} \subset \Omega \times \mathbb{R}$, where Ω denotes the input space. The input date is transformed from Ω to a Reproducing Kernel Hilbert Space $\mathfrak{H}$ via the feature map ϕ [15]. The data is then transformed into $\{(\phi(x_i), y_i); i=1\ldots n\}$.

Our aim is to find a nonlinear function f defined on Ω such that

$$\mathrm{f}(x) = \langle w, \phi(x) \rangle_{\mathrm{H}} + b \tag{1}$$

where $w \in \mathfrak{H}$, and $b \in \mathbb{R}$ denotes the weight vector and b is a constant.

To estimate the parameters, SVR is formulated as an optimization problem. A loss function is minimized subject to some constraints.

The problem is formulated as follows:

$$\min_{w,b} \frac{1}{2}\|w\|^2 + \frac{C}{n}\sum_{i=1}^{N}\left(\xi_i + \hat{\xi}_i\right) \tag{2}$$

$$\text{contraints}\begin{cases} \langle w, \Phi(x_i) \rangle + b - y_i \le \varepsilon + \xi_i \\ y_i - \langle w, \Phi(x_i) \rangle - b \le \varepsilon + \hat{\xi}_i \\ 0 \le \hat{\xi}_i, \xi_i \end{cases} \tag{3}$$

where C defines a constant that represents a trade-off between the tolerance of the estimation ε and the flatness of the solution.

The solution is obtained using the Karush-Kuhn-Tucker (KKT) algorithm, considering the dual problem:

$$\max_{\alpha,\hat{\alpha}}\begin{cases} -\frac{1}{2}\sum_{i,j=1}^{n}\left(\hat{\alpha}_i + \alpha_i\right)\left(\hat{\alpha}_i + \alpha_i\right)\kappa\left(x_i x_j\right) \\ -\varepsilon\sum_{i=1}^{n}\left(\hat{\alpha}_i + \alpha_i\right) + \sum_{i=1}^{n} y_i\left(\hat{\alpha}_i + \alpha_i\right) \end{cases} \tag{4}$$

$$\text{contraints}\begin{cases} \sum_{i=1}^{n}\left(\hat{\alpha}_i + \alpha_i\right) = 0 \\ \alpha_i, \hat{\alpha}_i \in \left[0, \frac{C}{n}\right] \end{cases} \tag{5}$$

where α_i and $\hat{\alpha}_i$ are the Lagrange multipliers corresponding to the two constraints of the primal problem and κ is the chosen Kernel function [21]. Using the fact that $w = \sum_{i=1}^{n}(\hat{\alpha}_i - \alpha_i)\phi(x_i)$, we can prove that the solution is as follows:

$$\mathrm{f}(x) = \sum_{i=1}^{n}\left(\hat{\alpha}_i - \alpha_i\right)\kappa(x_i, x) + b \tag{6}$$

b is determined by the KKT algorithm.

Mapping the input data into a Reproducing Kernel Hilbert Space $\mathfrak{H}$ can be done using different ways. It allows us to extend the class of the decision functions to the nonlinear case. We do not need to know the explicit mapping function ϕ [22]. The kernel function is sufficient to resolve the learning problem since

$$\kappa(x, y) = \phi(x)\phi(y) \tag{7}$$

In the input space, we can define the distance by

$$\begin{aligned} d^2(x, y) &= (\phi(x) - \phi(y))^2 \\ &= \kappa(x, y) - 2\kappa(x, y) + \kappa(x, y) \end{aligned} \tag{8}$$

We conclude that the kernel function is a way to measure the similarity between *x* and *y* and this is what makes the kernel choice very significant in regression modeling [23].

TABLE I: Used Kernels

Linear Kernel	$\kappa(x, y) = \phi(x)\phi(y)$
RBF Kernel	$\kappa_\eta(x, y) = \exp\left(-\eta\|x - y\|^2\right)$
Sigmoid Kernel	$\kappa_{r,\eta}(x, y) = \tanh(-\eta\phi(x)\phi(y) + r)$
Polynomial Kernel	$\kappa_{d,\eta}(x, y) = (\eta\phi(x)\phi(y) + r)^d$

C. LSTM

A LSTM neural network, introduced in [24], is a specific type of RNN that is more appropriate when it comes to modeling long-range dependencies. As compared to RNNs, the LSTM's architecture contains memory blocks instead of hidden units.

A memory block contains one or more *memory cells* that are modulated by nonlinear sigmoidal gates and are applied multiplicatively. In order to reduce the parameters, memory cells are sharing the same gates. These gates determine whether the model keeps the values at the gates (if the gates evaluate to 1) or discards them (if the gates evaluate to 0). In this way, the network can exploit long-range temporal contexts [24].

Given the input sequence $x=(x_1 \dots x_T)$, the network computes a mapping sequence to the output $y=(y_1 \dots y_T)$. The following equations determine the unit activations:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i)$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f)$$
$$c_t = f_t c_{t-1} + i_t \tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c)$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o)$$
$$i_t = o_t \tanh(c_t)$$

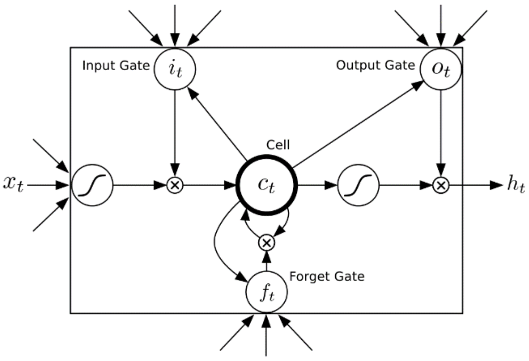


Fig. 1. LSTM cell.

where σ is the logistic sigmoid function; gates *i*, *f*, *o*, and *c* are the input gate, forget gate, output gate, and cell activation vector, respectively. All these vectors have the same size as the hidden vector *h*. The W terms denote the weight matrices from the cell to gate vectors (e.g., W_{si}). Here, tanh denotes the output activation function [25].

As far as the backward pass is concerned, our LSTM network compiles an iterative gradient descent algorithm, minimizing the usual root mean squared error [24].

III. Experimental Results

This study is based on the daily phone prices in one particular marketplace, namely, *amazon.fr*. The forecast models were tested on different time series. In this paper, our target is the time series available from the April 3, 2016, to March 23, 2017, involving prices related to a specific product, namely, the Samsung Galaxy S7 (32 GB). We used Python in our implementation with different application programming interfaces (APIs). For machine learning and deep learning, we used *Scikit-learn* and *TensorFlow* as a backend to the *keras*. The training and prediction calculi were made with a parallel computing platform and API called CUDA (NVIDIA).

In order to compare the results from different models, we used the RMSE of prediction. RMSE appends a relatively high weight to large errors [26]. In our case, large errors are particularly undesirable; this is why RMSE is a good fit to our approach.

At first, we studied the statistics of the time series. For instance, we calculated the standard deviation, and the first and third quartiles (Q1 and Q3, respectively). We also calculated the median and mean values of the time series. The price yields a mean value that is almost equal to the median value.

TABLE II: Descriptive Statistics to the Samsung Galaxy S7 32GB

Standard deviation	72.39
Minimum	449
Q1	603.95
Median	659.42
Mean	660.44
Q3	699
Maximum	940.58

The constructed time series used for the study considered the daily mean values. With machine learning techniques, time series decomposition into trends, residues, and seasonal components is not needed, which is not the case in other statistical methods like ARIMA. We feed the time series as an input to the tested models.

During our experiments, we built our SVR model using the *Scikit-learn* library. The LSTM model was implemented under the *keras* library with *TensorFlow* backend for parallel GPU computing. Both the libraries are available in Python.

A. Univariate Models

We first started by forecasting the time series only using the values of the variable being forecast. Our regression model considered the past observations as the input and the future values as the output. The window approach [20] involves the following:

$$\hat{x}_t = f(x_{t-1} + x_{t-2}, \dots, x_{t-n}) \quad (9)$$

where $\hat{x}$ is the forecast value, $x_{t-1} + x_{t-2}, \dots, x_{t-n}$ denotes the training values, and *n* is the window size. After different attempts, we chose to fix the window size at 7 for this case study. Fig. 2 shows the predicted values in red; the actual values are denoted in blue.

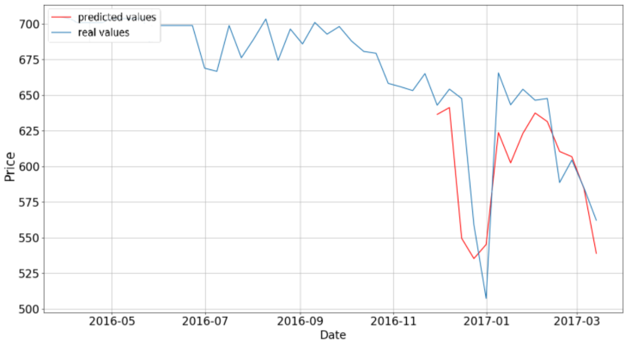


Fig. 2. Univariate SVR model.

We can see that although there is a difference between the two curves, the SVR model with a linear kernel is able to detect the variations perfectly.

In order to study the robustness of this method, we tested these two univariate forecasting models with different phone products in the same time period. Table IV lists the results of the RMSE obtained for Huawei P9, iPhone 6 Plus (16 GB), iPhone 6 (16 GB), Galaxy S6 (64 GB), and Galaxy S6 (32 GB). For the SVR model, the RMSE varies between 15 euros and 49 euros. For the LSTM model, it varies between 16.4 euros and 56.09 euros. With these tests, we can conclude that the SVR univariate model is more accurate than the LSTM model. Nevertheless, the difference in the RMSE results between these two approaches, in terms of euros, is minimal.

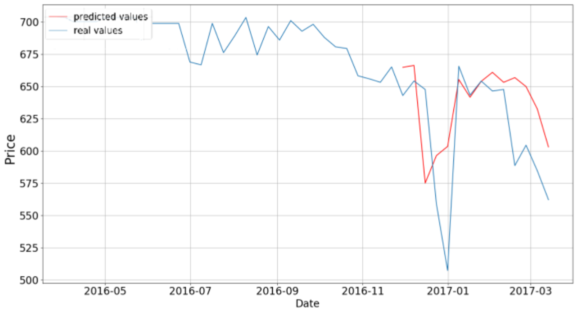


Fig. 3. Univariate LSTM model.

TABLE III: COMPARING UNIVARIATE MODELS

Model	*RMSE*
SVR (Linear)	33.430
LSTM	40.656

TABLE IV: MEAN RMSE VALUES WITH UNIVARITE SVR AND LSTM MODEL

Phone models	*RMSE(LSTM)*	*RMSE(SVR)*
Huawei P9	56.093	49.096
iPhone 6 Plus 16Gb	35.634	15.992
iPhone 6 16Gb	16.415	15.744
Galaxy S6 64Gb	34.042	35.528
Galaxy S6 32Gb	16.774	15.282
Galaxy S7 32Gb	40.656	33.430

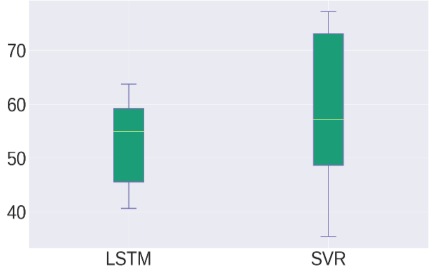


Fig. 4. RMSE boxplot for univariate models.

In terms of RMSE, the SVR model with linear kernels yields a better prediction. We notice that when the time series is stationary, the LSTM model prediction is very accurate, but when it comes to forecasting sudden peaks, the model does not seem to provide the perfect solution.

The RMSE boxplot provides us with an exhaustive view of the models' accuracies. After applying both the models on different time series, we saved the RMSE for each one and then created the boxplot. It is evident that although the SVR model gives the minimal error, the variance in RMSE for the LSTM model is still smaller. The medians are barely equal to each other. As a conclusion, we can say that the LSTM model is more appropriate in forecasting our time series.

B. *Multivariate Models*

After studying and comparing univariate models, our second approach was to introduce other data to the initial model inputs. Thus, we consider the correlations between different time series in the phone market, which are due to competition, in order to improve the model's accuracy.

Phone markets have different price ranges and the prices are volatile. We considered the products with similar price ranges from different brands and studied their influence on the Samsung Galaxy S7 (32 GB).

In fact, exploring the relations between these time series provides the model with additional information, which might be significant for regression modeling [27]. In addition to historical data from the time series describing the Samsung Galaxy S7 prices, we added data related to the Samsung Galaxy S6 (32 GB), Samsung Galaxy S6 (64 GB), iPhone 6 (16 GB), iPhone 6 Plus (16 GB), and Huawei P9.

We conducted a Granger causality study [28] to analyze the effect between the time series. The notion of causality plays an important role in the multivariate approach. We tested the causality of all the different series. For clarity, we refer to the Samsung Galaxy S6 (32 GB), Samsung Galaxy S6 (64 GB), iPhone 6 (16 GB), iPhone 6 Plus (16 GB), and Huawei P9 as *A*, *B*, *C*, *D*, *E*, and *F*, respectively, in Table III.

From these results, we can conclude that there are causal effects between the series system and the iPhone 6s (16 GB), Huawei P9, and Samsung Galaxy S7 (32 GB) variables. However, the causality is not verified for Samsung Galaxy S6 (32 GB), Samsung Galaxy S6 (64 GB), and iPhone 6 Plus (16 GB).

In addition to the Granger causality effect, we conducted a sensitivity analysis with an impulse response function in order to better understand the market price evolution.

From the result of the impulse analysis, we can observe that the Huawei P9 and iPhone 6s Plus variables have negligible effects on the targeted variable. A unitary shock on the iPhone 6 (referred in Fig. 5 by "*iphone*") affects the Samsung Galaxy S7 (32 GB) (referred in Fig. 5 by "*s7*") price by slightly increasing it on the third day and decreasing it on the sixth day.

TABLE V: RESULTS OF GRANGER CAUSALITY TEST

Variable	*A*	*B*	*C*	*D*	*E*	*F*
Statistic	1.109	0.872	1.350	1.009	1.631	2.171
p-value	0.265	0.814	0.017	0.466	0	0
Causality	NotVerified	Not Verified	Verified	Not Verified	Verified	Verified

An increase in the price of the "Samsung Galaxy S6 (32 GB)" (referred in Fig. 5 by "*s664*") generated an increase in the targeted variable price that lasted for 10 days. The response of the price of the Samsung Galaxy S6 (64 GB) to an impulse is a remarkably immediate decrease followed by an increase by the fourth day. Finally, the effect of the impulse on the value decreases till it cancels out completely.

These two analyses allowed us to verify the existence of effects between these time series products. For the multivariate study, a matrix with these six variables will be considered as an input to our model.

The SVR with a linear kernel multivariate model can predict the change in the peaks when the amplitude of the variation is not important (Fig. 6). However, when the changes are important (e.g., in the end of December 2016), it yields wrong predictions. We obtained a RMSE of 39.912, which is higher than the results obtained in the univariate SVR model. The SVR model shows promising results in the univariate approach, considering the fact that additional data does not seem to improve the RMSE. The results are better when the time series are stationary.

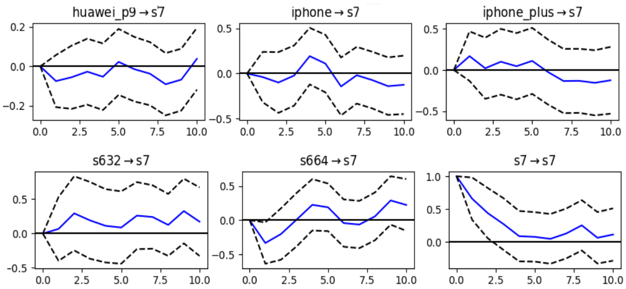


Fig. 5. Effect of impulse responses on the Samsung Galaxy S7 32Gb.

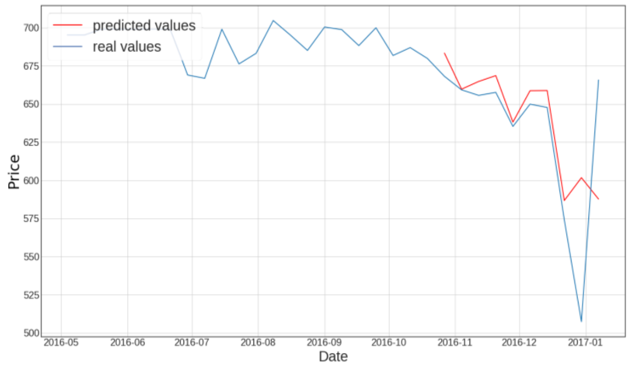


Fig. 6 Multivariate SVR model.

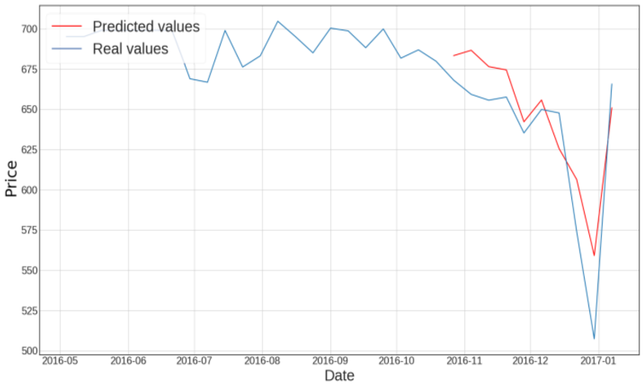


Fig. 7 Multivariate LSTM model.

However, by introducing additional time series in the LSTM model, we can obtain more accurate results and it reduces the RMSE of the forecast prices. In fact, in addition to decreasing the RMSE, which is 23.640, the LSTM model using 200 memory blocks is able to perfectly detect the peaks.

Furthermore, the RMSE boxplot (Fig. 8) used for evaluating errors for different time series shows that the SVR's RMSE is remarkably higher than that of LSTM. However, the medians are very close to each other. We can conclude that the LSTM multivariate model is more accurate than the SVR one in predicting the e-commerce time series under investigation.

TABLE VI: COMPARING MULTIVARIATE MODELS

Model	***RMSE***
SVR (Linear)	39.912
LSTM	23.640

TABLE VII: MEAN RMSE VALUES WITH MULTIVARIATE SVR AND LSTM MODEL

Phone models	***RMSE(LSTM)***	***RMSE(SVR)***
Huawei P9	34.695	39.957
iPhone 6 Plus 16Gb	35.634	38.619
iPhone 6 16Gb	25.119	31.621
Galaxy S6 64Gb	36.112	26.129
Galaxy S6 32Gb	29.852	31.539
Galaxy S7 32Gb	23.640	39.912

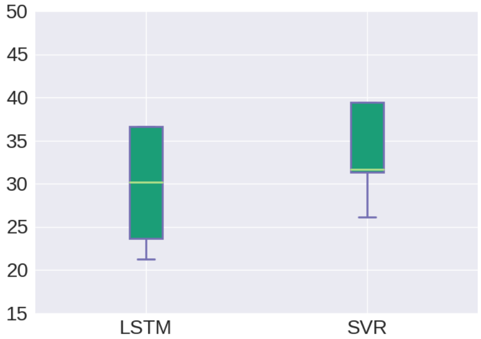


Fig. 8. RMSE boxplot for multivariate models.

IV. CONCLUSION

Machine learning techniques are used nowadays in a variety of fields such as clustering, classification, anomalies detection, speech recognition, self-driving cars, etc. Recently, many new models are emerging and their use in the industry is affecting our lives. The use of machine learning in forecasting has also made great improvements. Achieving reasonably accurate forecasts of a time series is a very important, yet challenging task. In this work, we presented a comparison between SVR and LSTM models with regard to time series prediction considering both univariate and multivariate approaches. For the multivariate approach, we studied the causality of different phones from competing brands having similar price ranges. The experiments on e-commerce data related to the phone market show that multivariate LSTM is more accurate than multivariate SVR in predicting such time series. It yields the lowest RMSE of 23.640. However, if we do not consider external time series, the SVR model performs better than LSTM. In our future work, we will test this model on other market data, progressing toward real-time price forecasting.

REFERENCES

[1] J. Abraham, R. van Welie, and R. Willemsen, Global Ecommerce Foundation Report, 2016.

[2] D.C. Montgomery, C. L. Jennings, and M. Kulahci, *Introduction to Time Series Analysis and Forecasting*, New Jersey, 2008.

[3] G. Kirchg ässner and J. Wolters, *Introduction to Modern Time Series Analysis*, Springer, 2007.

[4] W. S. Jevons, *The Theory of Political Economy*, Macmillan and Company, 1879.

[5] W. M. Person, "Indices of business conditions," *Review of Economics and Statistics*, vol. I, pp. 5-107, 1919.

[6] T. Wang, Y. Cai, L. Song, and Q. Chang, "Financial time series forecasting using directed-weighted chunking SVMs," *Mathematical Problems in Engineering*, p. 7, Apr. 2014.

[7] P. Jong, "State space models in actuarial science," *Research Rep.*, July 2005.

[8] E. H. de Alc ântara, J. L. Stech, J. A. Lorenzzetti, and E. de M. Novo, "Time series analysis of water surface temperature and heat flux components in the Itumbiara Reservoir (GO), Brazil," *Acta Limnologica Brasiliensi*., vol. 23, No. 3, pp. 245–259, Feb. 2012.

[9] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*, 4th ed. John Wiley & Sons, Inc. 2013.

[10] R. F. Engle, "Autoregressive conditional hetero-scedasticity with estimates of the variance of United Kingdom inflation," *Econometrica*, vol. 50, issue 4, pp. 987-1008., July 1982

[11] V. Akgiray, "Conditional heteroscedasticity in time series of stock returns: Evidence and forecasts," *The Journal of Business*, vol. 62, issue 1, pp. 55–80, 1989.

[12] R. D. Martin, J. T. Connor, and L. E. Atlas, "Recurrent neural networks and robust time series prediction," *IEEE Transactions on Neural Networks*, vol. 5, issue 2 pp. 240-254, Mar. 1994,.

[13] X. Wu, P. C. Y. Chen, J. Liu, Z. Zhao, and W. Chen, "LSTM network: a deep learning approach for short-term traffic forecast," *IET Intelligent Transport Systems*, vol. 11, no. 2, pp. 68–75, 2017.

[14] S. Mukherjee, E. Osuna, and F. Girosi, "Nonlinear prediction of chaotic time series using support vector machines," in *Proc. the 1997 IEEE Workshop on Neural Networks for Signal Processing VII*.

[15] C. L. Giles, S. Lawrence, and A. C. Tsoi, "Noisy time series prediction using recurrent neural networks and grammatical inference," *A.C. Machine Learning*, vol. 44, issue 161, 2001.

[16] L. J. Cao and F. E. H. Tay, "Support vector machine with adaptive parameters in financial time series forecasting," *IEEE Transactions on Neural Networks*, vol. 14, issue 6, Nov. 2003.

[17] M. Assaad, R. Bone ́, and H. Cardot, "A new boosting algorithm for improved time-series forecasting with recurrent neural networks," *Information Fusion*, vol. 9, pp. 41–55, 2008.

[18] T. Kuremotoa, S. Kimuraa, K. Kobayashib, and M. Obayashia, "Time series forecasting using a deep belief network with restricted Boltzmann machines," *Neurocomputing*, vol. 137, pp. 47-56, August 5, 2014.

[19] L. Bontemps, V. L. Cao, J. McDermott, and N. A. Le-Khac, "Collective anomaly detection based on long short-term memory recurrent neural networks," *Lecture Notes in Computer Science*, vol. 10018, 2016.

[20] F. A. Gers, D. Eck, and J. Schmidhuber, "Applying LSTM to time series predictable through time-window approaches," *Perspectives in Neural Computing*, Springer, London, 2002.

[21] S. R. Gunn, "Support vector machines for classification and regression," *Tech. Rep.*, University of Southampton, May 1998.

[22] C. H. Ho and C. J. Lin, "Large-scale linear support vector regression," *Machine Learning Research*, vol. 13, pp. 3323–3348, 2012.

[23] S. Ruping, "SVM kernels for time series analysis," *Tech. Rep. 43*, CS Department, AI Unit, 44221 Dortmund, Germany: University of Dortmund, 2001.

[24] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, Dec. 1997.

[25] J. Koutnik, R. Bas, S. K. Greff, R. K. Srivastava, and J. Schmidhuber, "LSTM: A search space odyssey," *Tech. Rep.*, Mar. 2015.

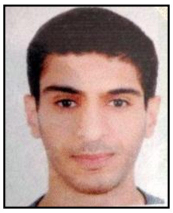
[26] N. L. Shcherbakova, A. P. Tyukov *et al*., "A survey of forecast error measures," *World Applied Sciences Journal*, pp. 171–176, Sept. 2013.

[27] H. Lütkepohl, *New Introduction to Multiple Time Series Analysis*, 2005.

[28] C. W. J Granger, "Investigating causal relations by econometric models and cross-spectral methods," *Econometrica*, vol. 37, issue 3, pp. 424–438, 1969.

Houda Bakir was born in Carthage, Tunisia. She received her MS/MSc degree in 2004 in electrical engineering with specialization in automatic industrial computing and a master's degree in 2009 in automatics and process engineering with specialization in automation at the Higher School of Sciences and Technology, Tunisia (Ecole Supérieure des Sciences et Techniques de Tunis (ESSTT)), and a PhD in 2016 in electrical engineering with specialization in signal and image processing from the National Higher School of Engineers, Tunisia. Her current research interests are big data analysis and machine learning techniques.

In 2004, she worked as an intern in the Laboratory of Solar Applications in the National Scientific Research and Technical Institute (Institut National de Recherche Scientifique et Technique (INRST)) for her graduation project. During the internship, she studied and realized a microcomputer system for the characterization of photovoltaic modules. In 2009, she joined the Electrical Department of Superior National School of Engineers of Tunis (ENSIT) as an Assistant Professor, where she conducted research on medical image segmentation using vector fields and active contour techniques. She also conducted a practical session in numerical computing, automatics, signal processing, numerical telecommunication, robotics, and industrial computing. From 2013 to 2015, she worked as the CEO of Grace Light, Tunisia, where she supervised projects involving smart-home modules. In 2015, she joined the École supérieure privée d'ingénierie et de Technologie (ESPRIT) as an Assistant Professor, where she led the Analytical Finance department. In 2015, she worked as a consultant with Intellixx IT Services in the field of finance. Since October 2016, she is working as the Lead Research Engineer in Datavora S.A. She is a member of the Centre de Recherche en Productique (CEREP), ENSIT.



Ghassen Chniti was born in Tunisia and has graduated from the National Engineering School of Tunis, Tunisia, in 2017 with an engineering diploma in applied mathematics specializing in finance and data science. In the summer of 2015, he joined Banque de l'Habitat as an intern. During this internship, he created a process under BPM Bonita describing the different steps involved in a bank's lending process. During the summer of 2016, he worked as an intern in Berenschot Intellerts, a consultancy firm in big data and data science located in Utrecht, Netherlands, where he analyzed data describing the employment market in order to predict vacancy times. Recently, in 2017, he completed his graduation internship in Datavora S.A., Tunisia, where he implemented time series forecasting models in the e-commerce field. Since September 2017, he has been working as an actuary for Assurances at Banque internationale arabe de Tunisie (BIAT).

Hédi Zaher was born in Tunisia in 1979. He received his engineering diploma in 2002 in computer science from the National School of Computer Science (Ecole Nationale des Sciences de l'Informatique (ENSI)) in Tunisia. He obtained his MSc degree in 2004, and a PhD in 2008 in advanced networks, knowledge, and organization from the University of Technology of Troyes (Université de technologie de Troyes (UTT)), Troyes, France.

In 2002, he was an intern in the Institut National des Télécom (INT) in Evry, France, where he worked on the Security of WiFi networks based on smart ships as his graduation project. In 2003, he joined Omniacom S.A., Tunisia, as a computer engineer. From 2004 to 2007, he worked as a research engineer in Laboratoire Tech-CICO (UTT), France. Here, he also worked as an Assistant Professor and conducted practical course sessions on algorithms, C language, Unified Modeling Language (UML), groupware, and intranet.

In 2008, he worked as the project lead in the Offshore Interactive company. From 2008 to 2010, he worked as the Chief Technical Officer (CTO) in Cogniva Europe S.A.S, Troyes, France. He has worked on the following: creation of an innovative software suite, technical specifications, contribution to functional specifications, a research and development ANR project (Miipa-Doc), supervision of interns, a thesis titled CIFFRE, and filed two patents. From 2010 to 2011, he worked at Blogspirit, Paris, France, as the lead developer and worked on Web services, highly scalable platforms and Web sites, and mobile applications for platforms on blogs and social networks. He also worked on a Research and Development ANR project (MADSPAM) and a European project (DIFAC). From 2012 to 2016, he worked as the CTO in IMDev, Tunisia. Since 2016, he is the CEO of Datavora, a big data e-commerce data provider company.