International Journal of Scientific Research in Computer Science, Engineering and Information Technology

ISSN : 2456-3307

OPEN ACCESS

Available Online at : www.ijsrcseit.com

doi : https://doi.org/10.32628/IJSRCSEIT

Improving Amazon EC2 Spot Instances Price Prediction using Machine Learning Algorithm

[1]M. Prasanthi, [2]G.Chishma, [3]P. Padmavathi, [4]K.Reethika, [5]A.Chandra Sekhar

[1]Assistant Professor, [2,3,4,5] UG Students

Department of CSE, Sri Vasavi Institute of Engineering and Technology, Nandamuru, Andhra Pradesh, India

ARTICLEINFO

Article History:

Accepted: 10 April 2024

Published: 22 April 2024

Publication Issue

Volume 10, Issue 2

March-April-2024

Page Number

713-720

ABSTRACT

Spot instances were introduced by Amazon EC2 in December 2009 to sell its spare capacity through auction based market mechanism. Despite its extremely low prices, cloud spot market has low utilization. Spot pricing being dynamic, spot instances are prone to out-of bid failure. Bidding complexity is another reason why users today still fear using spot instances. 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 cloud users 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. Simulations with 12 months real Amazon EC2 spot history traces to forecast future spot prices show the effectiveness of the proposed technique. Comparison of RRFs based spot price forecasts with existing non-parametric machine learning models reveal that RRFs based forecast accuracy outperforms other models. We measure predictive accuracy using MAPE, MCPE, OOBError and speed. Evaluation results show that $MAPE$ <= 10% for 66 to 92% and $MCPE$ <= 15% for 35 to 81% of one-day-ahead predictions with prediction time less than one second. $MAPE$ <= 15% for 71 to 96% of one-week-ahead predictions. Index Terms— Amazon EC2, Compute instances, One-day-ahead prediction, One-week-ahead prediction, Regression Random Forests, Spot instances, Spot price prediction.

Keywords : Amazon EC2, Compute instances, One-day-ahead prediction, One-week-ahead prediction, Regression Random Forests, Spot instances, Spot price prediction

I. INTRODUCTION

The on-demand scalability characteristic of cloud computing forces cloud service providers to overestimate their resources to meet peak load demand of its customers which happens at different time periods and may not overlap. Due to over-estimation, a large number of cloud resources are idle during off peak hours. Cloud providers also face the problem of allocating resources, keeping in view user"s different job requirements and data center capacity. Different types of users, multiple types of requirements further alleviate the resource allocation problem. Also, demand for cloud resources fluctuate due to today"s usage based pricing plans. In order to manage these demand fluctuations more flexible pricing plans are required to sell resources according to real time market demand. Spot pricing was introduced by Amazon EC2 in December 2009 to minimize operational cost, combat under utilization of its resources and make more profit. Similar to on-demand instances, spot instances offer several instance types comprising different combinations of CPU, memory, storage and networking capacity. Amazon Web Service (AWS) is not the only participant in the spot instance realm. Google Compute Engine launched its preemptible Virtual Machines on September 8, 2015 designed for such type of workloads that can be delayed and are fault tolerant at the same time. Users can bid for spot instances (SIs) where prices are charged at lowest bid price, whereas, pricing on Google Preemptible VMs is fixed at per hour rate. The distinguishing feature of Amazon Elastic Compute Cloud (EC2) spot instance is its dynamic pricing. From customer's perspective, spot instances offer prospects of low cost utility computing at a risk of out-of-bid failure at any time by Amazon EC2. Spot instance reliability depends on the market price and user's maximum bid (limited by their hourly budget). Spot prices vary dynamically with real-time based on demand (user's bid) and supply (resource availability) for spot instance capacity in the data centers across the globe. User's bids for spot instances and control the balance of reliability versus monetary cost. The price for spot instances sometimes can be as low as one eighth of the price of on-demand instances. On the other hand, it is also not uncommon that spot prices surpass on-demand prices in cloud data centers. When the demand is low, spot prices are low because less numbers of users are bidding for the same instance. Therefore, a bidder"s probability of incurring less monetary cost is higher. On the other hand, when the demand is high, users are willing to pay high to get access and hence spot prices increase.

Amazon Web Services (AWS) provides virtual computing environments via their EC2 service. You can launch instances with your favourite operating system, select pre-configured instance images or create your own.Why this is revelant to data sciensits is because generally to run deep learning models you need a machine with a good GPU. EC2 can be configured with a P2/P3 instance and can be configured with up to 8 or 16 GPUs respectively!However, you can request Spot Instance Pricing. Which basically charges you for the spot price that is in effect for the duration of your instance running time.They are adjusted based on long-term trends in supply and demand for Spot instance capacity. Spot instances can be discounted at up to 90% off compared to On-Demand pricing.

II.RELATED WORK

Title: Layer Recurrent Neural Network Based Power System Load Forecasting

This paper presents a straight forward application of Layer Recurrent Neural Network (LRNN) to predict the load of a large distribution network. Short term load forecasting provides important information about the system's load pattern, which is a premier requirement in planning periodical operations and facility expansion. Approximation of data patterns for forecasting is not an easy task to perform. In past, various approaches have been applied for forecasting. In this work application of LRNN is explored. The results of proposed architecture are compared with other conventional topologies of neural networks on the basis of Root Mean Square of Error (RMSE), Mean Absolute Percentage Error (MAPE) and Mean Absolute Error (MAE). It is observed that the results obtained from LRNN are comparatively more significant.

Title-EMOTION RECOGNITION IN HUMANS AND MACHINE

This paper presents a straight forward application of Layer Recurrent Neural Network (LRNN) to predict the load of a large distribution network. Short term load forecasting provides important information about the system's load pattern, which is a premier requirement in planning periodical operations and facility expansion. Approximation of data patterns for forecasting is not an easy task to perform. In past, various approaches have been applied for forecasting. In this work application of LRNN is explored. The results of proposed architecture are compared with other conventional topologies of neural networks on the basis of Root Mean Square of Error (RMSE), Mean Absolute Percentage Error (MAPE) and Mean Absolute Error (MAE). It is observed that the results obtained from LRNN are comparatively more significant.

Title-A Double Auction Mechanism to Bridge Users' Task Requirements and Providers' Resources in Two-Sided Cloud Markets

Double auction-based pricing model is an efficient pricing model to balance users' and providers' benefits. Existing double auction mechanisms usually require both users and providers to bid with the unit price and the number of VMs. However, in practice users seldom know the exact number of VMs that meets their task requirements, which leads to users' task requirements inconsistent with providers' resource. In this paper, we propose a truthful double auction mechanism, including a matching process as well as a pricing and VM allocation scheme, to bridge users' task requirements and providers' resources in two-sided cloud markets. In the matching process, we design a cost-aware resource algorithm based on Lyapunov optimization techniques to precisely obtain the number of VMs that meets users' task requirements. In the pricing and VM allocation scheme, we apply the idea of second-price auction to determine the final price and the number of provisioned VMs in the double auction. We theoretically prove our proposed mechanism is individual-rational, truthful and budget-balanced, and analyze the optimality of proposed algorithm. Through simulation experiments, the results show that the individual profits achieved by our algorithm are 12.35 and 11.02 percent larger than that of scaleout and greedy scale-up algorithms respectively for 90 percent of users, and the social welfare of our mechanism is only 7.01 percent smaller than that of the optimum mechanism in the worst case.

Title-Dynamic and Heterogeneous Ensembles for Time Series Forecasting

This paper addresses the issue of learning time series forecasting models in changing environments by leveraging the predictive power of ensemble methods. Concept drift adaptation is performed in an active manner, by dynamically combining base learners according to their recent performance using a non-linear function. Diversity in the ensembles is encouraged with several strategies that include heterogeneity among learners, sampling techniques and computation of summary statistics as extra predictors. Heterogeneity is used with the goal of better coping with different dynamic regimes of the time series. The driving hypotheses of this work are that (i) heterogeneous ensembles should better fit different dynamic regimes and (ii) dynamic aggregation should allow for fast detection and adaptation to regime changes. We extend some strategies typically used in classification tasks to time series forecasting. The proposed methods are validated using Monte Carlo simulations on 16 realworld univariate time series with numerical outcome as well as an artificial series with clear regime shifts. The results provide strong empirical evidence for our hypotheses. To encourage reproducibility the proposed method is publicly available as a software package. Keywords-dynamic ensembles; time series forecasting.

Title-Bidding Strategies for Amazon EC2 Spot InstancesA Comprehensive Review

Spot instance bid price is among one of the major issues in cloud computing. Bid price is aimed at complete execution of the work flow on spot instances while considering several job requirements such as hardware, latency, deadline and budget constraints. Several state-of-the-art spot bidding strategies have been proposed in the literature for executing jobs on Amazon EC2 spot instances. This paper presents different spot bidding strategies proposed by the authors. It highlights the objectives of each and provides the suitability of each of the proposed bidding strategies based on the type of application, its fault tolerance, job requirements and other constraints.

III.PROPOSED SYSTEM

The objective of this work is to present and evaluate a predictive model for spot price prediction that can predict future prices with increased accuracy and speed, minimize forecasting errors and predict spot prices sufficiently far in advance to assist cloud spot users in bid decision making process with increased reliability. We compare prediction accuracy of the state of the art non-parametric supervised machine learning algorithms with Regression Random Forests (RRFs) model.

Advantages:

1. The purpose of proposed system is An analysis of the length of time epoch durations when spot price is less than on-demand price to raise users confidence level in opting for spot instances.
2. Spot price forecasting. We resort to machine learning based ensemble method namely RRFs for one-week ahead and one-day-ahead spot price prediction. The approach focuses on both prediction accuracy and speed is high.

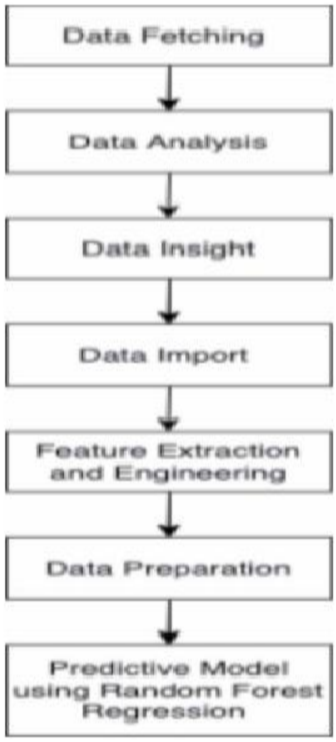

Fig 1 : Proposed work flow

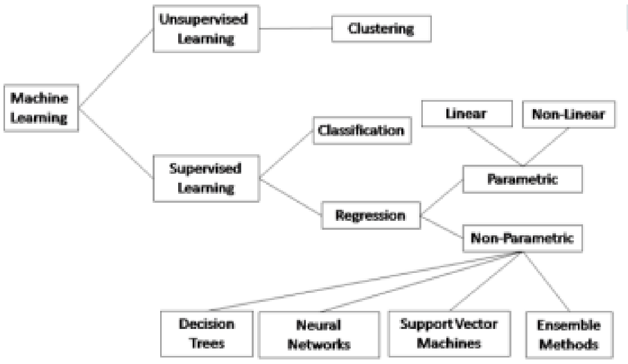

Fig 2 : Classification

Classification predicts categorical (discrete, unordered) labels, whereas prediction or regression is most often used for numeric prediction. Algorithms that do not make strong assumptions or make fewer assumptions about the form of the mapping function are called nonparametric algorithms. By not making assumptions, they are free to learn any functional form from the training data and automatically adapt easily to changes in the underlying time dynamics with varying characteristics. Popular non-parametric regression machine learning algorithms are:

- Support Vector Machines
- Multilayer Feed forward Neural Networks
- Decision Trees (Classification/Regression)
- Regression Tree Ensembles
- Random Forests

"Wisdom of crowds" refers to the phenomenon in which aggregated predictions from a large group of people can be more accurate than most individual judgments and can rival or even beat the accuracy of subject matter experts. Ensemble methods are learning models that achieve performance by combining the opinions of multiple learners. In doing so one can often get away with using much simpler learners and still achieve great performance in terms of increased robustness and accuracy.

IV. RESULTS AND DISCUSSION

```
In [*]:
import numpy as np
import pandas as pd
from subprocess import check_output

In [26]:
central_df = pd.read_csv("ca-central-1.csv");

In [27]:
central_df.head()

Out[27]:
```

	2017-05-06 17:29:01	c4.large	Linux/UNIX	ca-central-1a	0.0139
0	2017-05-06 17:29:01	m4.4xlarge	Windows	ca-central-1b	0.8328
1	2017-05-06 17:29:00	m4.4xlarge	Linux/UNIX	ca-central-1b	0.1051
2	2017-05-06 17:29:00	m4.2xlarge	Windows	ca-central-1b	0.4152
3	2017-05-06 17:29:00	m4.2xlarge	Linux/UNIX	ca-central-1b	0.0532
4	2017-05-06 17:28:49	m4.4xlarge	Linux/UNIX	ca-central-1b	0.1060

Fig 3. Results screenshot

```
In [21]:
from sklearn.model_selection import train_test_split

In [22]:
X1_train, X1_test, y1_train, y1_test = train_test_split(X1, y1, test_size=0.2,random_state=42 )

In [*]:
from sklearn.linear_model import SGDRegressor
clf1 = SGDRegressor()
clf1.fit(X1_train,y1_train)

y1_rbf = clf1.predict(X1_test)
```

Fig 4. Results screenshot

Random Forests

```
In [24]:
from sklearn.ensemble import RandomForestRegressor
clf = RandomForestRegressor(max_depth=2, random_state=0)

In [*]:
clf.fit(X1_train,y1_train)

Pred = clf.predict(X1_test)
```

Fig 5. Results screenshot

V. CONCLUSION

Spot pricing encourages users to shift execution of flexible workloads from provider's peak hours to off-peak hours and thus obtain monetary incentives. Analysis of one year spot price history data shows that there are sufficient number of time epochs of duration ranging from 30 days to more than 100 days and even longer when spot prices are up to 6 to 8 times cheaper than on-demand prices. It is therefore reasonable for users to shift their workloads from on-demand to spot instances. This work presents application of RRFs for Amazon EC2 spot price prediction. We compare several non-parametric machine learning prediction algorithms for spot price prediction in terms of various forecasting accuracy measures and conclude that RRFs outperforms other methods. Evaluation results show that $MAPE$ <= 10% for 66 to 92% and $MCPE$ <= 15% for 35 to 81% of one-day-ahead predictions in different regions. $MAPE$ <= 15% for 71 to 96% for One-week-ahead predictions in different regions. Instance types with lower infrastructure show better prediction accuracy than those with higher infrastructure. The prediction results suggest that spot price predictions using RRFs are more accurate than other machine learning algorithms. Prediction time for all predictions is less than 1 second. This indicates that the RRFs based predictions are fast enough to predict spot prices. Furthermore, RRFs when used for spot price predictions have only a few adjustable parameters namely number of regression trees and leaf size. One-week-ahead and one-day-ahead predictions along with feature importance can be effectively used by spot users to plan job executions in advance and bid effectively leading to significant cost savings and reducing out-of-bid failure probability of spot instances.

VI. FUTURE WORK

Amazon EC2 spot historical price to predict the price of 1-day-ahead and 1-week-ahead prices for spot instance by establishing a random forests regression. Experiments are performed and compared with neural network, support vector machine regression, regression tree and other methods. In this case, if the distance between x[axis] and node[axis] is less than the maximum distance between k nearest nodes set and X ,we should search neatest node in node 's right subtree (line [17][18][19]. This is a similar case when x[axis] is more than node[axis]
(line [21][22][23][24][25][26][27]. Finally, this algorithm will return the set of k nearest nodes. Evaluated algorithms In this paper, we use 5 algorithms as comparison methods which are Linear Regression (LR) [22],Support Vector Machine Regression (SVR) , Random Forest (RF) [25], Multi-layer Perception Regression.

REFERENCES

[1]. Agmon Ben-Yehuda, O., Ben-Yehuda, M., Schuster, A., and Tsafrir, D. Deconstructing amazon ec2 spot instancepricing. ACM Transactions on Economics and Computation 1, 3 (2013), 16.

[2]. Amazon Web Services. AWS EC2 Spot Instance Price Histories. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-spot-instances-history.html Accessed Jul-2018.

[3]. Amazon Web Services. New AWS EC2 Spot Instance Pricing. https://aws.amazon.com/about-aws/whats-new/2017/11/amazon-ec2-spot-introduces-new-pricing-model-and-the-ability-to-launch-new-spot-instances-via-runinstances-api/Accessed Jul-2018.

[4]. Amazon Web Services. Amazon EC2. https://aws.amazon.com/ec2/, 2018. [Online; accessed July-2018].

[5]. Amazon Web Services. Amazon ec2 service level agreement, 2018. https://aws.amazon.com/ec2/sla/ accessed July2018.

[6]. Amazon Web Services. Amazon ec2 spot instances, 2018. https://aws.amazon.com/ec2/spot/ accessed July 2018.

[7]. Amazon Web Services. Amazon instance pricing, 2018. https://aws.amazon.com/ec2/pricing/on-demand/ accessedJuly 2018.

[8]. Amazon Web Services. Amazon simple storage service, 2018. https://aws.amazon.com/s3 accessed July 2018.

[9]. Amazon Web Services. How spot instances work, July 2018.

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/how-spot-instances-work.html accessed July 2018.

[10]. Amazon Web Services. New amazon ec2 spot pricing model, July 2018. https://aws.amazon.com/blogs/compute/new-amazon-ec2-spot-pricing/ accessed July 2018.

[11]. Amazon Web Services. New amazon ec2 spot pricing model: Simplified purchasing without bidding and fewer interrup tions, 2018. https://aws.amazon.com/blogs/compute/new-amazon-ec2-spot-pricing/ accessed August 2018.

[12]. Amazon Web Services. New amazon ec2 spot pricing model: Simplified purchas ing without bidding and fewer interruptions, 2018. https://aws.amazon.com/blogs/aws/amazon-ec2-update-streamlined-access-to-spot-capacity-smooth-price-changes-instance-hibernation/accessedAugust 2018.

[13]. Amazon Web Services. New ec2 spot instance termination notices, July 2018. https://aws.amazon.com/blogs/aws/new-ec2-spot-instance-termination-notices/ accessed July 2018.

[14]. Aristotle Cloud Federation. https://federatedcloud.org [Online; accessed Aug-2018].

[15]. Chohan, N., Castillo, C., Spreitzer, M., Steinder, M., Tantawi, A., and Krintz, C. See Spot Run: Using SpotInstances for MapReduce Workflows. In Usenix HotCloud (2010).

[16]. Google Cloud Platform. Google preemptable virtual machines, July 2018. https://cloud.google.com/preemptible-vms/ accessed July 2018.

[17]. He, X., Shenoy, P., Sitaraman, R., and Irwin, D. Cutting the cost of hosting online services using cloud spot markets.In Proceedings of the 24th International Symposium on High-Performance Parallel and Distributed Computing (New York,NY, USA, 2015), HPDC '15, ACM, pp. 207–218.

[18]. Javadi, B., Thulasiram, R. K., and Buyya, R. Characterizing spot price dynamics in public cloud environments. FutureGeneration Computer Systems 29, 4 (2013), 988–999.

[19]. Khodak, M., Zheng, L., Lan, A. S., Joe-Wong, C., and Chiang, M. Learning cloud dynamics to optimize spot instancebidding strategies.

[20]. 20] Nurmi, D., Brevik, J., and Wolski, R. Qbets: Queue bounds estimation from time series. In Job Scheduling Strategiesfor Parallel Processing (2008), Springer, pp. 76–101.

[21]. Nurmi, D., Wolski, R., and Brevik, J. Probabilistic advanced reservations for batch-scheduled parallel machines. InProceedings of the 13th ACM SIGPLAN symposium on principles and practice of parallel programming (2008), ACM,pp. 289–290.

[22]. Schwarz, G. Estimating the dimension of a model. Annals of Statistics 6, 2 (1978).

[23]. Song, J., and Guerin, R. Pricing and bidding strategies for cloud computing spot instances. In 2017 IEEE Conferenceon Computer Communications Workshops (INFOCOM WKSHPS) (May 2017), pp. 647–653.

[24]. Steve Fox. New aws spot pricing model: The good, the bad, and the ugly, July 2018. http://autoscalr.com/2018/01/04/new-aws-spot-pricing-model-good-bad-ugly/ accessed July 2018.

[25]. Subramanya, S., Guo, T., Sharma, P., Irwin, D., and Shenoy, P. Spoton: A batch computing service for the spotmarket. In Proceedings of the Sixth ACM Symposium on Cloud Computing (New York, NY, USA, 2015), SoCC '15, ACM,pp. 329–341.

[26]. Wallace, R. M., Turchenko, V., Sheikhalishahi, M., Turchenko, I., Shults, V., Vazquez-Poletti, J. L., andGrandinetti, L. Applications of neural-based spot market prediction for cloud computing. In Intelligent Data Acquisitionand Advanced Computing Systems (IDAACS), 2013

IEEE 7th International Conference on (2013), vol. 2, IEEE, pp. 710–716.

[27]. Willsky, A. S., and Jones, H. L. A generalized likelihood ratio approach to the detection and estimation of jumps inlinear systems. IEEE Transactions on Automatic Control 21, 1 (1976).

[28]. Wolski, R., Brevik, J., Chard, R., and Chard, K. Probabilistic guarantees of execution duration for amazon spotinstances. In Proceedings of the International Conference for High Performance Computing, Networking, Storage andAnalysis (2017), ACM, p. 18.

[29]. Yi, S., Kondo, D., and Andrzejak, A. Reducing costs of spot instances via checkpointing in the amazon elastic computecloud. In 2010 IEEE 3rd International Conference on Cloud Computing (July 2010), pp. 236–243.

[30]. Zheng, L., Joe-Wong, C., Tan, C. W., Chiang, M., and Wang, X. How to bid the cloud. In Proceedings of the 2015ACM Conference on Special Interest Group on Data Communication (New York, NY, USA, 2015), SIGCOMM ’15, ACM,pp. 71–84.