A Hybrid Framework for Effective Prediction of Online Streaming Data

A Hybrid Framework for Effective Prediction of Online Streaming Data

K Kanagaraj[1] **and S Geetha**[2]

[1,2]Department of Computer Applications, MEPCO Schlenk Engg. College, Sivakasi, Tamilnadu, India

kanagaraj@mepcoeng.ac.in

Abstract. In this paper, we present a hybrid model to perform the training and testing of prediction model with online streaming data. Prediction of online streaming data is a time critical task. Huge volume of data that is being generated online need to be ingested to a prediction model and to be used to train and test the prediction model dynamically which improves the learning rate. The existing approaches for dynamic training and testing use the local infrastructure or virtual machines from the cloud infrastructure to increase the learning rate of the prediction model with streaming data. Recently many applications prefer serverless cloud infrastructure than virtual machines. However, using the serverless infrastructure for the entire prediction process will have time and space tradeoffs due to its autonomic feature. Hence in this paper we propose a hybrid approach that uses the three different environments such as the local infrastructure, virtual machine and serverless cloud for different stages. A novel approach to select the suitable environment to train and test the LSTM based air quality prediction model with stream data is proposed with increased learning rate and reduced resource utilization.

1. Introduction

The capability of an application to process a data while it is being generated is called as stream processing. The input for this application will be from various sources. In most case the data is collected continuously from different real-time sensors. Due to this features stream data requires parallel and distributed computing resources for quick data processing. Also selecting a suitable environment for processing stream data is a big challenge. With the introduction of cloud, most of the stream processing applications have move to cloud. However, selecting a suitable type of cloud service for different parts of the streaming application workflow will have considerable cost and time benefits. Particularly using serverless cloud computing architecture for streaming applications will be most gainful for users. Initially online streaming was used in stock market for forecasting the Sensex and Nifty using distributed processing. Recently stream processing is popularly used for real-time data processing. As it is the imminent and suitable technology for IoT sensors, there is a rapid growth in the number of applications using these two technologies.

As people are more concerned about the quality of the air, they breathe every day, real-time air quality monitoring becomes an important application. Due to the higher cost, setting up an air quality monitoring system in all the areas is not possible and are implemented in areas where high spatial variability is found. However effective methods for fast processing of the air quality data at a cheaper cost are the need of the day.

Hence, in this research we have proposed a novel air quality prediction model using LSTM technique. The novelty in this approach is selecting appropriate resources at every stage of the prediction model. The major contributions of this research are

1. Novel air quality prediction model using LSTM
2. Hybrid resource selection for different stages of the prediction model
3. Using serverless cloud for reduction of cost and time for prediction

1.1 Serverless Cloud

Serverless cloud is the current buzzword for cloud users. The overheads in provisioning and releasing the virtual machines are replaced by serverless cloud computing. Users enjoy the freedom of invoking everything as a function and pay only for the period during which computing resources are utilized to complete the task. The configuration of the resources will be decided based on the memory requirements of the applications. Cost of serverless functions depends on response time rounded to nearest 100ms, Function size and overhead per execution. The function completion time also depends on the input as well as the function execution in different context. Because of its inherent features the serverless cloud is very suitable for scheduling workflows. The dynamic structure resource requirements of tasks in a workflow will be fulfilled by the serverless cloud infrastructure.

Many popular cloud service providers offer serverless computing service. The AWS Fargate from amazon allocates the right amount of compute resources, eliminating the requirement to select an instances and scaling capacity. The users have to pay only for the resources required to run their containers. There is no need to pay for additional servers and eliminates over provisioning and under provisioning. It runs each task in its own kernel thus providing an isolated environment for each of them. The workload isolation and enhanced security features has attracted several customers to run even their mission critical applications in fargate. The architecture of serverless clouds using distributed containers proposed by (Soltani et al., 2018) describes the benefits of serverless cloud architecture in a distributed environment. An optimized approach to reduce the execution cost in serverless infrastructures (Pérez et al., 2018) using container based architecture is also a very good step to understand the feasibility of serverless computing. The serverless computing is also suitable for processing big data applications. The experiment by (Giménez-Alventosa et al., 2019)for execution map reduce applications in AWS lambda was proved to be effective.

2. Review of Literature

There are abundant research contributions from researchers towards online stream processing. The research contribution by Barddal et al., (2020) used the Kolmogorov–Smirnov as well as the Population Stability metrics for comparing the performance of batch processing against online streaming. The ensemble-based algorithm proposed by (Junior & Nicoletti, 2019) uses an updating technique that enhances the model flexibility with quick knowledge acquisition that overcome the error rate. Nguyen et al., (2019) propose a decay mechanism to identify the age of the incomes data and compared them with several bench mark algorithms.

Tavasoli et al., 2019, proposed technique for processing complex data streams that are produced by non-stationary stochastic methods. The self-adjusting learning model to utilize the multiplication-based update algorithm to update the data whenever a new element occurs is also explained. Din & Shao, 2020, propose a new data stream classification technique to capture the time-changing concept, to dynamically maintain a set of online micro-clusters and learns. The work proposed by (Sun et al., 2020) achieves better prediction performance than the state-of-the-art online structure learner for SPNs, while promising speed up in the order-of-magnitude.

A study of the three batch classification algorithms C4.5, k-nearest Neighbor, and Support Vector Machine was presented (Shi et al., 2019). Ghomeshi et al., (2020) proposed a novel layer based ensemble technique called RED-PSO to seamlessly adopt to the different concept drifts in non-stationary data stream prediction tasks. The work containing an inbuilt forgetting mechanism of neural networks proposed by Ksieniewicz et al., (2019) focused on active learning, that asks for labels particularly for interesting examples, crucial for appropriate model upgrading.An optimized solution using Lambda function to predict the arrival frequency of stream data was presented by y (Reza Hoseinyfarahabady et al., 2018).

The work proposed by (Chaudhry et al., 2020) uses serverless computing to combine MEC and NFV by orchestration. Bhardwaj et al., (2019), introduced a decentralized directory service useful for connecting edges from multiple stakeholders. Recently the use of application programming interfaces on the new service introduced by Watson Machine Learning is also gaining momentum for transforming a huge quantity of data generated by organizations. The use of distance algorithm to synthesize and refine correlation data using machine learning is also a noteworthy technique.

Luckow & Jha, (2019) used pilot streaming for serverless environments. It provides a combined solution for managing resources in high performance computing and serverless cloud. Apart from eliminating the need to write resource specific code it provides hassle free scaling of resources. The claim by (Becker et al., 2019) outlines various challenges that may occur during elastic resource provisioning. A model driven DevOps framework for developing and managing applications that are based on serverless computing was developed by (Casale et al., 2020). Fine grained independent applications based on microservices optimized for FaaS and container technologies were proposed. An idea of reusable microservices to avoid vendor lock in was also developed.

Gundu et al., (2020) discussed the need of a cloud computing environment that can use the cloud for HPC applications with increased scalability capable of handling sudden fluctuations in workload were also insisted. The research contribution by (Kritikos et al., 2019) presents an architecture for an environment that realizes the vision of multi cloud provisioning. This environment involves innovative components that supports the cross orchestration of cloud services as well as the multilevel monitoring and adaptation of batch processors. It also supports adaptive provisioning of resources. The survey made by (G. Nguyen et al., 2019) provides a recent time based comprehensive overview along with comparisons and trends in development and usage of recent technologies. It also provides an overview of massive parallelism support that is capable of scaling computation effectively.

3. Implementation

The novel approach to select the suitable environment for training and testing the air quality prediction system is proposed in this work which has a huge potential for implementation. The prediction model is developed in the stand alone desktop, trained and tested with historical dataset. Once it is achieved with expected learning rate, it can be deployed into virtual machine if the size of the ingesting stream data is huge else the serverless cloud can be used for a moderate size of ingesting stream data. When talk about the streaming data, it is not necessary to use the complete stream data for training. Instead, only the batches where there is a pattern change can be considered for training the prediction model. Because, when there is no change in the pattern of stream data, no much change in the learning rate of the prediction model. Hence, only extracted streaming data can be considered for training the prediction model with micro services. As we implement serverless cloud for training and testing process the time and cost will be reduced to a reasonable extent, thus increases the possibility of deploying the system to achieve a high learning rate. Algorithm 1 shows the rule followed in the proposed approach.

Algorithm 1: Rule followed for selecting suitable environment

```
Input  : Type of Data Set
Output : Suitable Resource for processing Data Set

    if ‘historical dataset’ then
        select “standalone desktop” – for existing dataset
    else if 'high volume of streaming data' then
        select 'virtual machine'
    else if 'moderate/low volume of streaming data' then
        select 'serverless cloud'
    else if ‘high volume of streaming data along with historical dataset” then
        select “standalone desktop” – for existing dataset
        select “virtual machine” – for high volume of streaming data
    else if ‘moderate/low volume of streaming data with historical dataset” then
        select “standalone desktop” – for existing dataset
        select “serverless cloud” – for moderate/low volume of streaming data
```

The rules are framed based on the flow of data rate. The systematic process is depicted below.

- Step 1: Sorting of average discharge of records from the largest to the smallest value, involving a total of n values.
- Step 2: Rank each discharge value (M), starting with 1 for the largest discharge value.
- Step 3: Exceedance probability (P) is calculated using equation 1.

$$P = 100 * [M / (n + 1)] \quad \text{Equation (1)}$$

Where,

P - probability that a given flow will be equalled or exceeded (% of time)
M - ranked position on the listing (dimensionless)
n - number of events for period of record (dimensionless)

The overall architecture of the proposed system is given in Figure 1. The diagram shows the different stages of the air quality management workflow and how resources are selected at different stages.The resource utilization differences can be monitored with the help of this architecture. The flow of data rate is taken into consideration for setting up the rules.

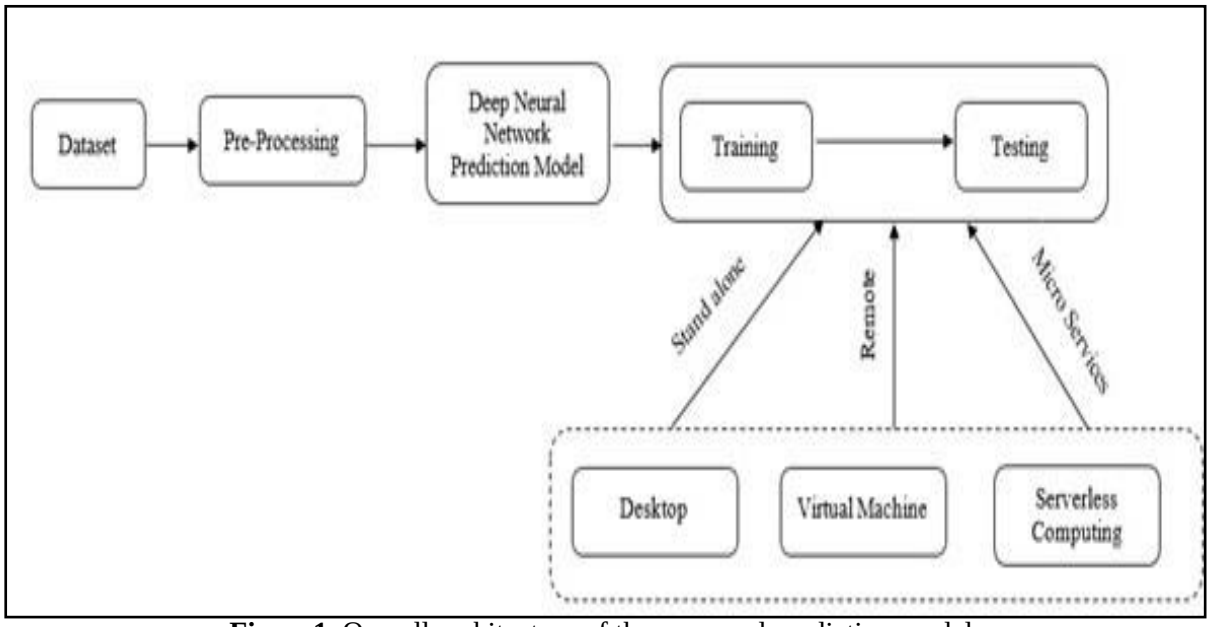

Figure1. Overall architecture of the proposed prediction model

The prediction models are generated using linear regression and LSTM. These models are trained in the standalone desktop with the historical air quality dataset. The LSTM model performed better than linear regression model with improved accuracy. Performance and accuracy of model is measured using Root Mean Square Error (RMSE), Mean Absolute Error (MAE) are used to evaluate the results. Equations for RMSE and MAE are given in equations 2 and 3 respectively.

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=0}^{n}(y_t - \tilde{y}_t)^2} \quad \text{Equation (2)}$$

where y_t is the actual value, $\tilde{y_t}$ is the predicted value and n is the total number of observations.

$$MAE = \frac{1}{n}\sum_{t=0}^{n}|y_t - \tilde{y}_t| \quad \text{Equation (3)}$$

where n is the total number of predictions, y is the predicted value.

The same models are deployed in virtual machine as well as serverless cloud using micro services. The accuracy of prediction is measured using the same Root mean square as well as mean absolute error. The accuracy of both the models are not compromised based on the deployment environments. But, the resource utilization varies between different environments for both models. Training and Testing duration are recorded for various prediction models. As well as the resource utilization is monitored and logged. All these three parameter values are taken into analysis and found that the resources are utilized efficiently in the hybrid environment than the individual environments.

4. Results and Discussion

The air quality dataset collected from Central Pollution Control Board, Delhi. The dataset contains 1hr air pollution data for the 3 months. It contains totally 2650 records. The prediction models were developed using linear regression and LSTM. These models were trained and tested in various environments to check the time complexity and resource usage. The time taken to train and test the models varies on each environment and the results are depicted in the below table.

Table 1. Time Taken for predicting the air quality data using different resource provisioning methods

Environment	Model Type	Resource Utilization	Training Time in seconds	Testing Time in seconds
Standalone Desktop	Linear Regression	33%	21.02	12.03
	Deep LSTM Model	78%	16.00	10.10
Virtual Machine	Linear Regression	76%	16.30	9.31
	Deep LSTM Model	79%	14.55	9.09
Serverless	Linear Regression	90%	10.97	6.27
	Deep LSTM Model	92%	7.27	4.55
Hybrid Environment	Linear Regression	85%	4.70	2.69
	Deep LSTM Model	88%	8.73	5.45

4.1 Prediction Models

The prediction models are designed using both Linear Regression and Deep LSTM. The figure 2 shows the linear regression model for $NO_2$and the figure 3 shows the Deep LSTM model for the same NO_2. The prediction accuracy is achieved more in the Deep LSTM model than the Linear Regression model.

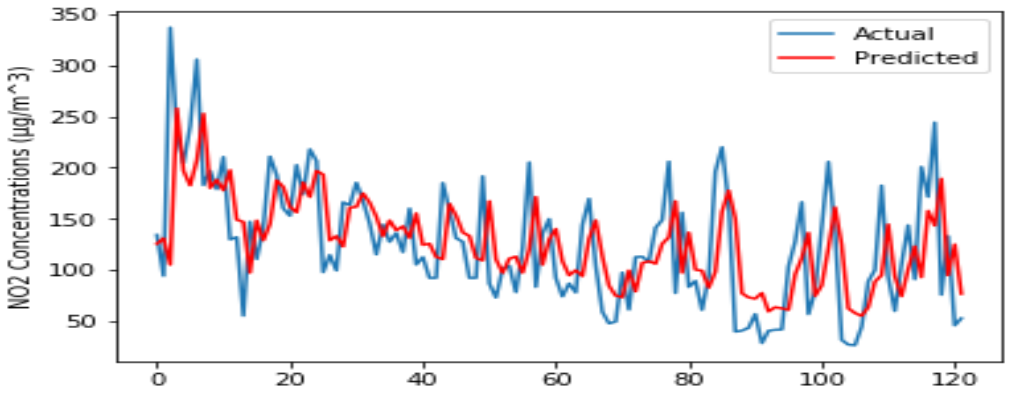

Figure 2. Prediction of NO2 using Linear Regression

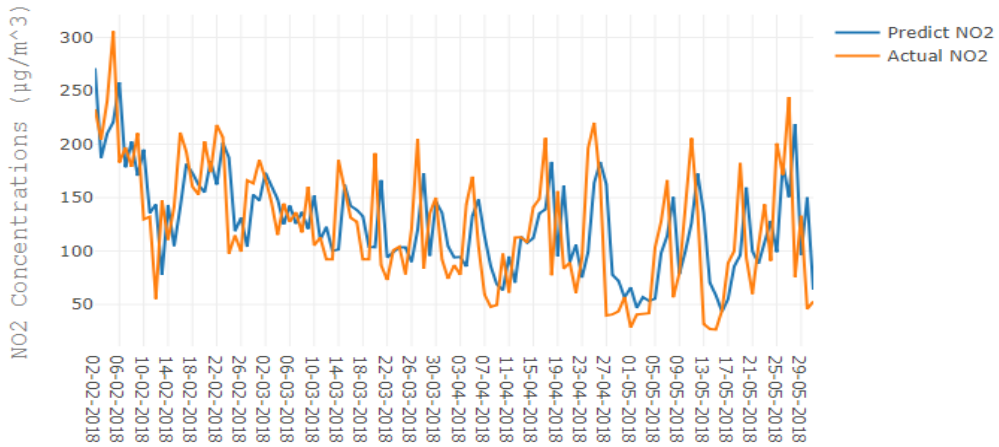

Figure3. Prediction of NO2 using Deep LSTM

4.2*Evaluation Measures*

The prediction models are evaluated using root means square error and mean absolute error.

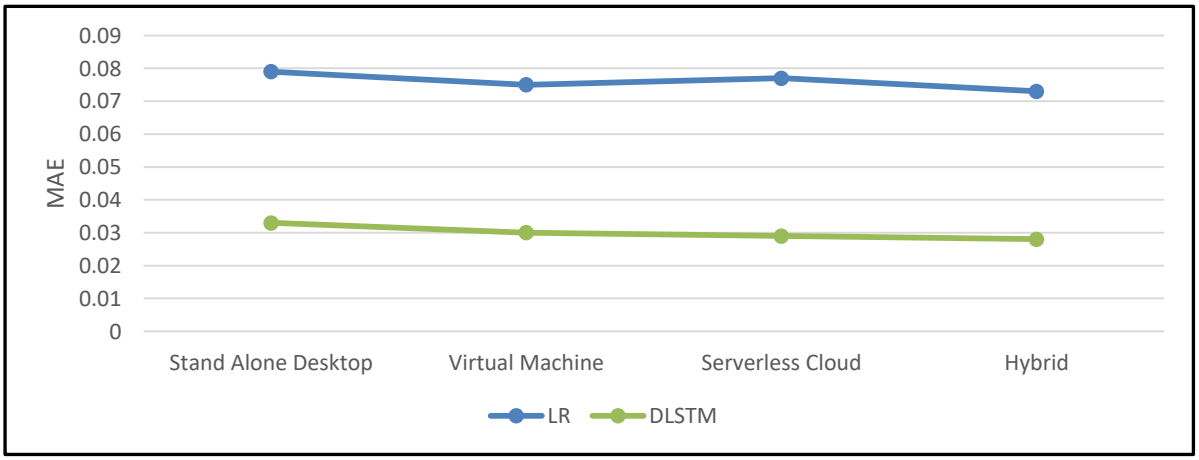

Figure 4. Mean absolute error occurred in the various environments

The LSTM model performs better than the Linear Regression models. These models were deployed in various environments. However, there is no much influence of the accuracy of the model. The MAE and RMSE occurred in the various environments are shown the Fig. 2 & 3 respectively.

From the figure 4, it is observed that the mean absolute error is comparatively less in the hybrid approach. Also figure 5, clearly shows that the root means square error is also less in the hybrid approach.

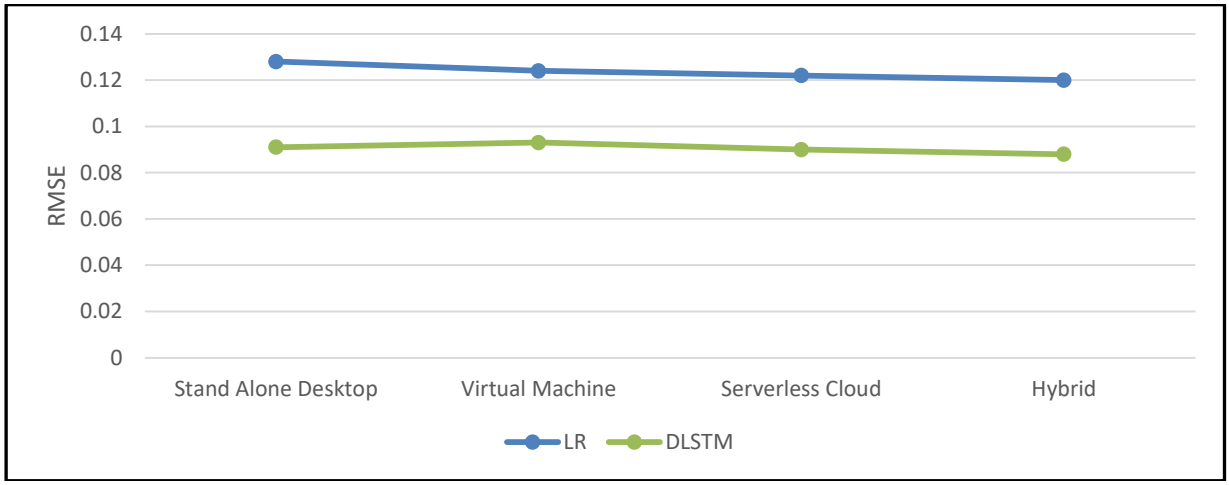

Figure 5. Root means square error occurred in the various environments

The proposed approach provides clarity of implementing the prediction model in the various environment as well as suggesting the suitable environment to reduce the cost and complete resource utilization.

5. Conclusions and Future Enhancements

The hybrid frame work proposed in this research uses different types of resources at every stage of the prediction process. Resources such as standalone computers, virtual machines and serverless cloud are selected based on the novel selection algorithm proposed in this paper. Air quality data with 2650 records is used for evaluating the proposed work. The implementation result shows that the hybrid approach increases the resource utilization and reduces the training and testing times from5% to 35%. Also the model is compared with liner regression and deep learning based LSTM Methods. The deep learning based has better performance when compared with the linear regression method. Hence using the proposed hybrid framework with deep learning based LSTM is capable of predicting the online streaming data faster and efficient that the other existing methods. In future we would like extend this work for prediction the video data set.

6. References

[1] Barddal J P, Loezer L, Enembreck F and Lanzuolo R 2020 Lessons learned from data stream classification applied to credit scoring *Expert Systems with Applications*

[2] Becker C, Julien C, Lalanda P and Zambonelli F 2019 Pervasive computing middleware: current trends and emerging challenges *CCF Transactions on Pervasive Computing and Interaction* 1(1), pp 10–23

[3] Bhardwaj K, Gavrilovska A, Kolesnikov V, Saunders M, YoonH, BondreM, BabuM, and WalshJ2019Addressing the fragmentation problem in distributed and decentralized edge computing: *A vision Proceedings - 2019 IEEE International Conference on Cloud Engineering*, pp 156–167

[4] Casale G, Artač M, van den Heuvel W, Jvan Hoorn A, Jakovits P, Leymann F, Long M, Papanikolaou V, Presenza D, Russo A, Srirama S,N,T,amburri D,A,Wurster M,and Zhu L

2020 RADON: rational decomposition and orchestration for serverless computing Software-Intensive Cyber-Physical Systems35(1), pp 77–87
[5] Chaudhry S R, Palade A, Kazmi A and Clarke S 2020 Improved QoS at the Edge using Serverless Computing to deploy Virtual Network Functions*IEEE Internet of Things Journal*, 4662(c)1–1
[6] Din S U and Shao J 2020 Exploiting evolving micro-clusters for data stream classification with emerging class detection*Information Sciences*507, pp 404–420
[7] Ghomeshi H, GaberM M, and Kovalchuk Y 2020 A non-canonical hybrid metaheuristic approach to adaptive data stream classification*Future Generation Computer Systems*, 102 pp 127–139
[8] Giménez-Alventosa, VMoltó, Gand Caballer M 2019 A framework and a performance assessment for serverless MapReduce on AWS Lambda*Future Generation Computer Systems*97, pp 259–274
[9] Gundu S R, Panem C A and Thimmapuram A 2020 Real-Time Cloud-Based Load Balance Algorithms and an Analysis*SN Computer Science*, 1(4) pp 1–9
[10] Junior B and Nicoletti M 2019An iterative boosting-based ensemble for streaming data classification*Information Fusion* 45, pp 66–78
[11] Kritikos K, Zeginis C, Iranzo J, Gonzalez R S, Seybold D, Griesinger F and Domaschka J 2019 Multi-cloud provisioning of business processes*Journal of Cloud Computing*, 8(1) pp 1–29
[12] Ksieniewicz P, Woźniak M, Cyganek B, Kasprzak A and Walkowiak K 2019Data stream classification using active learned neural networks *Neurocomputing* pp 74–82
[13] Luckow A, and Jha S 2019 Performance Characterization and Modeling of Serverless and HPC Streaming Applications *IEEE International Conference on Big Data*, 5688–5696
[14] Nguyen G, Dlugolinsky S, Bobák M, Tran V, López García Á, Heredia I, MalíkP, and HluchýL2019 Machine Learning and Deep Learning frameworks and libraries for large-scale data mining: a survey *Artificial Intelligence Review*, 52(1) pp 77–124
[15] Nguyen, T T, Dang M T, Luong, A V, Liew A W C, Liang Tand Mc Call J 2019 Multi-label classification via incremental clustering on an evolving data stream*Pattern Recognition*95, pp 96–113
[16] Pérez A, Moltó G, Caballer M and Calatrava A 2018 Serverless computing for container-based architectures*Future Generation Computer Systems*, 83 pp 50–59
[17] Reza Hoseiny farahabady M , Zomaya A Y and Tari Z 2018 A Model Predictive Controller for Managing QoS Enforcements and MicroArchitecture-Level Interferences in a Lambda Platform *IEEE Transactions on Parallel and Distributed Systems*29(7), pp 1442–1455
[18] Shi D, Zurada J, Karwowski W, Guan J and Çakıt E 2019 Batch and data streaming classification models for detecting adverse events and understanding the influencing factors Engineering *Applications of Artificial Intelligence*, 85(May) pp 72–84
[19] Soltani B, Ghenai A Zeghib N 2018 Towards Distributed Containerized Serverless Architecture in Multi Cloud Environment Procedia Computer Science 134, pp 121–128
[20] Sun Z, Liu C L, Niu J and Zhang W 2020 Discriminative structure learning of sum–product networks for data stream classification *Neural Networks*, 123 pp 163–175
[21] Tavasoli H, Oommen BJ and Yazidi, A2 019 On utilizing weak estimators to achieve the online classification of data streams Engineering *Applications of Artificial Intelligence* pp 11–31.