AIOT-Arch: Furthering Artificial Intelligence in Big Data IoT Applications

Sandra Said., Rowan Ihab., Ossama Hesham., Islam Abou Tabl., Nu Maged., Sherin Youssef., Mazen Elagamy.

1Department of Computer Engineering, College of Engineering and technology Arab Academy for Science and Technology, Alexandria, Egypt.

*Corresponding author: sandrasaidmarc1994@gmail.com

Abstract. Applications for the Internet of Things (IoT) have evolved in excessive numbers, producing a vast amount of data needed for intelligent processing. The varying IoT infrastructures such as cloud and IoT application layer protocol limitations in the transmission/receiving of messages become the barriers in the implementation of intelligent IoT apps. In this paper, we review the importance of Big data, cloud computing and fog computing in IoT and the challenges of using machine learning in IoT. Finally, we discuss the general statistics of using artificial intelligence in IoT applications.

1. Introduction:
The word "smart" fascinates us but the tools that we are using today are far from being smart like a human being. Artificial intelligence is designed to make computers do human reasoning that why we need machine learning [1] and data analysis [2] to be combined in one system. Machine learning creates methods to make the network's component automatic and self-sufficient while data analysis analyzes the data to identify the historical trends and be more effective and accurate in the future. IoT [3] is the main object of this trend that will provide a word full of intelligent devices called “smart objects” [4] integrated through the internet, Bluetooth or infrared. In this paper, we research and evaluate the role of machine learning in promoting data analytics for the IoT systems.

2. Big Data in IoT environments:
Intrinsically, IoT data is a type of big data. The widespread use of sensors for data collection, the utilization of the data collected over long periods of time and the need to evaluate them on a scale to help decision taking means that they overlap with many dimensions of big data [5]. In the following section we will describe the main characteristics of big data found in IoT and how these characteristics are considered as a challenge when using traditional machine learning techniques. Figure 1 shows the IoT data characteristics, the left part of the figure concerning big data characteristics will be discussed in the following section.

3. Characteristics of big Data which lead to inefficiency of Traditional Machine Learning Techniques:
3.1. Velocity
velocity is one of the main properties of big data hence IoT data. The velocity dimension refers to how quickly the data is being produced and how fast the application needs to process it. "Fast data" is endemic to IoT, since the generated data is in most cases a continuous and unbounded stream from a multitude of sensors [6]. The velocity of IoT data is affected by many factors. It’s affected directly by the sampling rate and the number of sensors in a given environment. Very high sampling rates can also often lead to redundancy if the observations are changing slowly. Nonetheless, a certain amount of redundancy will be optimal with some degree of trust for detecting the events with some level of confidence [5].

3.2. Volume

Volume of data is not less important than velocity. Big data implies enormous volumes of data. One view is the volume of data generated per sensor and the number of sensors (distributed). But these sensors are constantly observing, and the streaming data is potentially unbounded.

3.3. Variety

The variety component of big data is representative of the various potential types of data representations and protocols. IoT data heterogeneity is common, and can be due to different data sources, data types, or networks. IoT applications collect data from a wide range of sensors and in a wide range of formats due to the variety of environment variables being monitored.

3.4. Veracity

Veracity refers to the biases, noise and abnormality in data. The question here is whether the data that is being stored, and mined is relevant to the problem being analyzed. Figure 2 shows the characteristics of Big Data [31].

4. Why Cloud Computing and Fog Computing?

4.1. Role of Cloud Computing in IoT:

Cloud computing is the on-demand of computer resources like data storage and computing power over the internet. Companies prefer to use a cloud service provider [7] rather than using their own data centers. Cloud computing helps in the storage and the analysis of the large amount of data generated by the IoT applications and connects objects and people together to provide a high visibility [8]. IoT Links existing devices and smart devices to the internet and reduces gaps between Information Technology and Operational technology teams to create a single systems/data view. Nowadays the deployment of comprehensive hardware and the management of networks in IoT systems is not the main problem with the help of cloud computing.

4.2. Role of Fog Computing in IoT:

Fog computing is a platform that uses edge devices to conduct a large amount of local and routed storage, computation and communication over the internet [7]. It is a distributed environment that is
related to the cloud computing and IoT. Figure 3 shows different IoT components, and how analytics apply to these components [9]. Fog computing aims to form low latency network links between devices and the endpoints of their analyses and reduces the bandwidth required [10]. An additional advantage is the advanced security features that users can implement in a fog network such as network traffic segmentation and virtual extension of firewalls to provide network security.

5. IoT-Generated Data Processing for Machine Learning Analysis
Because IoT devices send data repetitively, big data challenges are experienced [14,15]. To get past this problem, distributed processing can be leveraged to partition data into chunks and assign each partition to a different computer for processing. Different frameworks can be used to achieve distributed processing like Hadoop and Spark, which can be used in machine learning. When using distributed computing, the following benefits are gained:
1) Network load is lessened.
2) Faster processing of data.
3) Lower energy consumption.
4) Processing of huge volumes of data becomes feasible. [5]

6. Challenges in Developing Intelligent IoT Solutions
6.1. Security
Communication technologies used by IoT devices such as Zigbee, Z-Wave and Wi-Fi have their own limitations in terms of security, which hinders the security of IoT applications [11]. The widespread use of IoT devices increases the vulnerability of an IoT system to attacks. Moreover, devices used in IoT systems usually are resource limited, which makes the use of advanced security techniques to protect IoT devices hardly achievable.

6.2. Performance
Some machine learning algorithms like Deep Convolutional Neural Networks can achieve high accuracies, but they have high computational and memory requirements, which make it hard to implement and sometimes infeasible on the resource-constrained devices used in IoT systems, for example in safety-critical applications like autonomous driving, which requires real-time image processing [11]. The following table 1 shows frame rates of YOLO algorithm, which is a state-of-the-art algorithm used in object detection applications, accomplished using Nvidia Jetson TX1 embedded module [12], which is made to be used in visual computing applications [35].

| Algorithm     | Frame Rate |
|---------------|------------|
| Fast YOLO     | 17.85      |
| O-YOLOv2      | 11.8       |
| YOLOv2        | 5.4        |

Table 1. Results of testing YOLO algorithms on Nvidia Jetson TX1

As shown in the Table 1, the frame rate didn’t exceed 17.85, which is too slow for applications that require real-time processing, which requires a frame rate of 30 [11].

6.3. Reliability
This is another core issue in machine learning algorithms. Machine learning algorithms like neural networks are not entirely accurate, which makes it unreliable for applications with high sensitivity to accuracy, like self-driving vehicles, and cancer diagnosis. Moreover, deep learning is being introduced in many industrial applications including data mining & analytics, where certain industrial standards need to be met, like IEC 61508 [13], which specifies reliability specifications to be met in industrial applications. Therefore, deep learning-based systems reliability needs to be ensured [13].
7. Deep Learning Machine Learning for IOT:
Nowadays, the trend of Artificial intelligence has opened new windows to build a variety of deep learning frameworks. Deep learning platforms e.g. TensorFlow are frequently used for text analysis in multimedia systems. The texts are analyzed using deep learning techniques such as CNN and RNN architectures. Some examples include detection of actions in video, recognition of disasters in the web crawled datasets. Figure 4 shows the difference between machine learning and deep learning.

![Figure 4](image_url)

7.1. Deep learning Layered Architecture:
The deep learning architecture is an 8 layers convolutional neural network (CNN) composed of 3 convolutional layers, three max-pooling (subsampling) layers and two fully connected layers as shown in Figure 5.

![Figure 5](image_url)

7.1.1. Convolutional Layer
The major building blocks used in convolutional neural networks are the conventional layers. A convolution is simply applying a filter to an input resulting in an activation. Repeated application of the same filter to an input results in an activation map called a feature map, which indicates the location and strength of a detected feature in an input such as an image[16]. The innovation of convolutional neural networks is the ability to automatically learn, under the constraints of a specific predictive modelling problem, such as image classification, a large number of filters in parallel specific to a training dataset. The result is highly specific features, which can be detected on input images anywhere [17].

7.1.2. Pooling or Subsampling layer
Convolution layer in CNN is often immediately followed by the pooling or subsampling layer. Its role is to down sample the output of a convolution layer along both the height and width dimensions of the
space [19]. For example, an operation of 2 x 2 pooling on top of 12 feature maps will produce an output size tensor [16 x 16 x 12].

7.1.3. Fully Connected Layer
Fully Connected Layer can be defined as a feed forward neural network. Fully Connected Layers are considered from the last few layers in the network [18]. The output from the final pooling or Convolutional layer is the input to the fully connected layer, which is flattened first then fed into this last layer [19].

8. Comparisons between different machine learning Techniques for IoT Application:

| Use Case for IoT | Machine learning algorithm (Data processing tasks) | Advantages | Disadvantages |
|------------------|-----------------------------------------------------|------------|---------------|
| Traffic and Air Control monitoring | K-mean (Clustering) | Better for huge variables and also produces tighter clusters | K- value difficult and difficult for data of different densities |
| Agriculture data analysis | Naive Bayes (Classification) [21] | Simple to implement. Needs less training data | Assumption can be wrong |
| Lameness detection | K-Nearest Neighbor (Classification) [23] | It will be helpful even if the training data is noisy | The value for k must be known as accurately |
| System to estimate personal thermal comfort | Support Vector Machine (Classification) [20] | Data can be visualized in more than 2 dimensions. Decision taking system is nearly perfect | Selection of a good kernel value is not easy, and also, if the data set is large then it will take more time for training |
| Monitoring Public Places, Useful in Fault Detection. | Principal Component Analysis (Feature extraction) | Lack of redundancy, Reduced complexity in images | The covariance matrix is difficult to evaluate and even the simplest invariance could not be captured |
| Low energy consumption, Healthcare Analysis and Forecasting | Neural Network (Regression/Classification) | Neural Network can model | Parameters are hard to interpret. It becomes very complex sometimes. |
6

| Clustering/Feature extraction) | nonlinear and complex relations. | Needs a large dataset |
|-------------------------------|---------------------------------|-----------------------|
| Predicting energy consumption using a smart-metering architecture | Random Forrest (Classification/Regression) [24] | considered as good for takings thousands of input variables | Noise in the datasets |
| Predict the energy usage | Linear Regression (Regression) [22] | simple to implement, susceptible to over-fitting but it can be avoided using some dimensionality reduction techniques | outliers can have huge effects, assumes a linear relationship |
| Temperature and humidity data prediction | Support Vector Regression (Regression) | Works with non-linear problems, Not biased by outliers | Not a familiar mode, Quite hard to understand |
| customer attrition prediction | Bagging (Classification/Regression) [25] | customer attrition prediction, reduces model over-fitting, missing values in the dataset do not affect the performance | final prediction based on the mean predictions. |

Table 2. Comparison between different machine learning techniques in IoT applications

9. Application Case-Study in Smart Cities:

9.1. Smart City
Smart cities use IoT devices such as sensors to collect data and cloud computing systems to store and analyze data [26]. The IoT devices are connected to each other using Wi-Fi connection or Bluetooth or infrared to create a huge single network that improves the citizen's lifestyle and public services such as energy consumption and water management. Smart city uses the data collected from IoT devices to monitor and manage the city's assets. Smart City is the main use case of IoT applications because of the following three reasons: Firstly, 60% of the publications examined is on the Smart City sector, secondly, smart City includes many other IoT use cases and finally, huge number of open datasets for Smart City applications exist that are easily accessible to researchers.

9.2. Case-Study of IoT in smart cities

9.2.1. Smart Road
Municipalities use IoT systems to provide smart road that helps people to navigate through places rapidly and safely. Different Sensors and Data collected from smart phone's GPS help smart road applications to determine the vehicles number, position and speed. Smart road manages the traffic light
that is connected to a cloud platform depending on the actual traffic situation [27]. By the time, the smart road system can predict the time and the location of the traffic and prevent that automatically.

9.2.2. Smart parking

Smart parking applications create a real-time map of parking spots using sensors implemented in the parking spots and data collected from driver's GPS [28]. This data is sent to the server which is a cloud system that collects and analyzes the data. After analysis, the server sends notifications of the available parking spots to the driver's smart phone application. The drivers are capable of paying by cash or visa (using the smart phone's application) as shown in Figure 6.

9.2.3. Smart Street lighting

Smart Street lighting system consists of sensors in the lighting poles. The poles are connected together using Wi-Fi connection. The data collected from these sensors is sent to the cloud system (control and management system) that analyzes the data and control the street lighting schedule [29] as shown in Figure 7. The sensors collect data concerning the movement of citizens in the road, the movement of cars and the degree of illuminance. The lighting poles gets brighter when a walker crosses the road or a bus arrives to the bus stop.

10. General statistics:

10.1. Datasets:

10.1.1. Transportation Mode Detection

The dataset of the transportation mode is collected using smartphones. Three main sensors are used to collect data which are the accelerometer, gyroscope and sound. The idea is to detect/recognize what type of vehicle the user uses, as there are five classes which are car, bus, train, walking and still. The process is considered as (HAR) Human Activity Recognition (HAR) [30]. The Specifications of this dataset are shown in Table 3.

| Number of Samples | Number of Features | Number of Target Classes |
|-------------------|--------------------|--------------------------|
| 5893              | 14                 | 5                        |

Table 3. Specifications of the Transportation Mode detection dataset

10.1.2. Rain Prediction

The dataset is collected from Australia through the observation of weather stations. The dataset includes many attributes such as: rainfall, evaporation, sunshine, temperature, direction, wind speed, pressure,
clouds and humidity. The dataset is used to predict whether there will be rain the next day or not [31]. The Specifications of this dataset are shown in Table 4.

| Number of Samples | Number of Features | Number of Target Classes |
|-------------------|-------------------|--------------------------|
| 142000            | 24                | 2                        |

Table 4. Specifications of the Rain detection dataset

10.2. Performance Evaluation Metrics:

| Metric         | Formula | Description |
|----------------|---------|-------------|
| Precision      | \( \frac{TP}{TP + FP} \) | number of relevant selected data items that are relevant. |
| Recall         | \( \frac{TP}{TP + FN} \) | number of relevant data items that are selected. |
| F1-score       | \( 2 \times \frac{Precision \times Recall}{Precision + Recall} \) | The harmonic mean of precision and recall. |
| Accuracy       | \( \frac{TP + TN}{TP + FN + TN + FP} \) | The ratio of accurately classified data items to the total number of observations. |
| Confusion Matrix | | Allows to measure Recall, Precision, Accuracy, and AUC-ROC curve. Its main usage is when a classification problem has two or more outputs. |
| ROC-AUC score  | | Used for discriminating the positive and negative classes in binary classification and demonstrates how good a model is. |

Table 5. Performance Metrics

10.3. Results

10.3.1. Performance comparison of the algorithms on Transportation Mode Detection dataset:
The accuracy rate is the most reliable measure in this case because for each target class in the dataset there are equal numbers of labels which means that the dataset is balanced [32] as shown in Table 6.

| Algorithms | Precision % | Recall % | F1-Score % | Training Set Accuracy % | Test Set Accuracy (Avg/Highest) % | Execution Time |
|------------|-------------|----------|------------|--------------------------|-----------------------------------|----------------|
| LR         | 63          | 62       | 62         | 64                       | 63/65                             | 0.68           |
| KNN        | 83          | 83       | 83         | 87                       | 80/82                             | 0.005          |
| DT         | 80          | 79       | 79         | 100                      | 76/79                             | 0.067          |
| RF         | 87          | 87       | 87         | 100                      | 85/87                             | 0.71           |
| SVM        | 80          | 80       | 80         | 88                       | 79/81                             | 9.9            |
| CNN        | 82          | 80       | 80         | 83                       | 82/82                             | 16.03          |

Table 6. Performance Metrics of different algorithms on Transportation Mode Detection dataset
10.3.2. Performance comparison of the algorithms on Rain Prediction dataset:

The Rain Prediction dataset has 24 features, three of the features are categorical features each one of them has 16 different values, for that kind of information the dataset required much more pre-processing tasks. One-hot encoding techniques are applied to the dataset to convert the categorical features to numeric ones, which lead to tripling the number of features [33] as shown in Table 7.

| Algorithms | Precision % | Recall % | F1-Score % | Training Set Accuracy % | Test Set Accuracy (Avg/Highest) % | Execution Time |
|------------|-------------|----------|------------|--------------------------|-----------------------------------|----------------|
| LR         | 72          | 79       | 74         | 85                       | 85/85                             | 8.2            |
| KNN        | 76          | 72       | 74         | 90                       | 81/83                             | 10.9           |
| DT         | 70          | 71       | 70         | 100                      | 79/80                             | 2.22           |
| RF         | 80          | 71       | 74         | 99                       | 85/85                             | 1.42           |
| ANN        | 82          | 74       | 77         | 86                       | 85/86                             | 35.00          |
| CNN        | 78          | 73       | 75         | 85                       | 84/85                             | 24.8           |

Table 7. Performance Metrics of different algorithms on Rain Detection dataset

11. Conclusion:
Nowadays, the IoT applications are integrated in everything. This paper shows the importance of the usage of artificial intelligence in Big data IoT applications using our proposed framework and shows the power of machine learning in providing benefits to the consumers. The proposed framework is examined on two datasets (Transportation Mode Detection dataset and Rain Prediction). When using different machine learning techniques with the first dataset, it gives best accuracy of 85/87% with RF algorithm but when using these same techniques with the second dataset, it gives best accuracy of 85/86% with the ANN algorithm. Our innovative architecture is opening up new opportunities for potential machine-to-machine communications work.

References
[1] R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, Machine Learning: An Artificial Intelligence Approach. Springer Science & Business Media, 2013.
[2] I. H. Witten and E. Frank, Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, 2016.
[3] Q. F. Hassan, A. R. Khan, and S. A. Madani, Internet of Things: Challenges, Advances, and Applications. Chapman & Hall/CRC Computer and Information Science Series, CRC Press, 2017.
[4] G. Fortino and P. Trunfio, Internet of Things based on Smart Objects: Technology, Middleware and Applications. Springer, 2014.
[5] Mohammad Saeid Mahdavinejad, Mohammadreza Rezvan, Mohammadamin Barekatain, Payaman Adibi, Payam Barnaghi, Amit P. Sheth; Machine Learning for Internet of Things Data Analysis: A Survey. 2018
[6] Nashez Zubair;Niranjan A;Kiranj Hebbar;Yogesh Simmhan; Characterizing IoT Data and its Quality for Use. 2019
[7] Bonomi F, Milito R, Zhu J, Addepalli S (2012) Fog computing and its role in the internet of things. In: Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, ACM, New York, NY, USA, MCC ’12, pp 13–16
[8] Joshi J, Reddy J, Reddy P, Agarwal A, Agarwal R, Bagga A, Bhargava A (2016) Cloud
computing based smart garbage monitoring system. In: 2016 3rd international conference on electronic design (ICED), IEEE, pp 70–75

[9] Aazam M, Zeadally S, Harras KA (2018) Offloading in fogcomputing for iot: review, enabling technologies, and research opportunities. Future Generat Comput Syst 87:278–289

[10] Fog computing and the internet of things: extend the cloud to where the things are (2015). https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-overview.pdf. Accessed 18 Oct 2019

[11] Fatima Hussain, Rasheed Hussain, Syed Ali Hassan, and Ekram Hossain, Machine Learning in IoT Security: Current Solutions and Future Challenges, IEEE 2020

[12] Shafiee, Mohammad Javad, et al. "Fast YOLO: A Fast You Only Look Once System for Real time Embedded Object Detection in Video." arXiv preprint arXiv:1709.05943 (2017).

[13] IEC 61508 2016. Functional Safety and IEC 61508. (2016). Retrieved Oct. 2016 from http://www.iec.ch/functionalsafety/

[14] A. Joakar, A Methodology for Solving Problems with DataScience for Internet of Things, Open Gardens(blog)(July21,2016),http://www.opengardensblog.futuretext.com/archives/2016/07/a-methodology-for-solvingproblems-with-datascienceso-finternet-oftings.html

[15] H. Zhao and C. Huang, A data processing algorithm in epc internet of things, Cyber-enabled Distributed Computing and Knowledge Discovery (CyberC), 2014 International Conference on, IEEE (2014), pp. 128-131.

[16] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. IEEE Trans Pattern Anal Mach Intell, 39(4):640–651, 2017.

[17] Farabet, C.; Couprie, C.; Najman, L.; LeCun, Y. Learning hierarchical features for scene labeling. IEEE Trans. Pattern Anal. Mach. Intell. 2013, 35, 1915–1929. [CrossRef] [PubMed]

[18] Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. arXiv 2014, arXiv:1409.1556.

[19] Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the 25th International Conference on Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.

[20] N. Cristianini, J. Shawe-Taylor, An introduction to support vector machines and other kernel-based learning methods, Cambridge university press, 2000.

[21] McCallum, K. Nigam, et al., A comparison of event models for naïve bayes text classi_cation, in: AAAI-98 workshop on learning for text categorization, Vol. 752, Citeseer, 1998, pp. 41-48.

[22] Neter, M. H. Kutner, C. J. Nachtsheim, W. Wasserman, Applied linear statistical models, Vol. 4, Irwin Chicago, 1996.

[23] T. Cover, P. Hart, Nearest neighbor pattern classi_cation, IEEE transactions on information theory 13 (1) (1967) 21-27.

[24] Breiman, Random forests, Machine learning 45 (1) (2001) 5-32.

[25] Breiman, Bagging predictors, Machine learning 24 (2) (1996) 123-140.

[26] R. Petrolo, V. Loscri, N. Mitton, Towards a smart city based on cloud of things, a survey on the smart city vision and paradigms, Transactions on Emerging Telecommunications Technologies.

[27] M. Vanis and K. Urbaniec, “Employing Bayesian Networks and conditional probability functions for determining dependences in road traffic accidents data,” 2017 Smart City Symposium Prague (SCSP), May 2017.

[28] Prabhu Ramaswamy; IoT smart parking system for reducing green house gas emission; 2016 International Conference on Recent Trends in Information Technology (ICRTIT)

[29] N. Yoshiura, Y. Fujii, and N. Ohta. Smart street light system looking like usual street lights based on sensor networks. In International Symposium on Communications and Information Technologies (ISCTT), pages 633–637, Sept 2013

[30] D. Kleyko, R. Hostettler, W. Birk, and E. Osipov, "Comparison of machine learning techniques for vehicle classification using road side sensors," in Proceedings of the IEEE 18th International Conference on Intelligent Transportation Systems, 2015, pp. 572-577.

[31] Y. K. Ever, K. Dimililer, and B. Sekeroglu, "Comparison of Machine Learning Techniques for
Prediction Problems," in Workshops of the International Conference on Advanced Information Networking and Applications, 2019, pp. 713-723.

[32] F. Alam, R. Mehmood, I. Katib, and A. Albeshri, "Analysis of eight data mining algorithms for smarter Internet of Things (IoT)," Procedia Computer Science, vol. 98, 2016, pp. 437-442.

[33] C. Carpineti, V. Lomonaco, L. Bedogni, M. Di Felice, and L. Bononi, "Custom Dual Transportation Mode Detection by Smartphone Devices Exploiting Sensor Diversity," in Proceedings of the IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), 2018, pp. 367-372.

[34] "Innovations in Smart Cities Applications Edition 2", Springer Science and Business Media LLC, 2019.

[35] Jetson TX1 Module, Nvidia Developer, https://developer.nvidia.com/embedded/jetson-tx1