International Conference on Intelligent Technologies & Science - 2021
(ICITS-2021)

Nonlinear Dimension Reduction: Edge Computing Data Analytics in IoT networks

Rupa Gupta[1], Abhishek kumar Saxena[2]
Computer Science and Engineering,
Bansal Institute of Engineering and Technology, Lucknow, India
rupa24021996@gmail.com, abhisaxena0212@gmail.com

Abstract: The data sent over the network is large then the application would slow down if proper network connectivity is not available. The data includes the data generate by sensors, cameras and other modules. The data is complex and can have both linear and non-linear relation among them. Hence the concept of edge computing is introduced which reduces the data dimension and retains the information by encryption and sent over the network to the cloud based servers and there it is decrypted and used for the further processing. Researchers have used several machine learning, deep learning based applications the encryption and decryption. Majorly used algorithms are PCA (principal component analysis) which is a machine learning based dimension reduction technique suitable for data which has linear relationship among themselves, Autoencoders which is a deep learning based dimension reduction technique which is suitable for handling the linear and non-linear complex relation among data, but the Auto encoders takes much time to process and it is complex to handle. In our proposed work we will use the Kernal PCA algorithm for the encryption and decryption of data.

Keywords: IOT, Edge detection, Machine learning, PCA

1. Introduction:

Edge computing has been gaining popularity, especially with applications that require fast response time and those with limited bandwidth because it locates computation close to the data sources. Applications of edge computing including smart street lamps [31], face identification [32], smart manufacturing [33], and vehicular networks [34] have demonstrated great success and prompted further investigations.

Wang et al. presented a survey on mobile edge networks focusing on computingrelated issues, edge offloading, and communication techniques for edge-based systems. The use cases highlighted in their study include IoT, connected vehicles, content delivery, and big data analysis. At the same time, Wang et al. identified real-time analytics as one of the open challenges. Similarly, Abbas et al. surveyed mobile edge computing and also identified big data analytics as a future research direction. While Wang et al. and Abbas et al. examined mobile edge computing, El-Sayed et al. focused on IoT applications of edge computing. They compared the characteristics of cloud, multi-cloud, fog, and edge computing and identified low bandwidth utilization and latencies as the main edge computing advantages. Mao et al. see edge computing as a key enabling technology for realizing the IoT vision and, similar to Wang et al. and Abbas et al., recognize data analytics as one of the future research directions in edge computing.

Discussed surveys [7, 21, 35, 36] note the potential of edge computing in data analytics and point out the importance of edge computing in IoT for handling the rapid increase of the number of connected devices. This thesis contributes to employing edge computing for data analytics by combining edge and cloud computing for the delivery of ML applications. Several studies present various scenarios with accompanying edge-based architectures demonstrating edge computing capabilities. Sinaeepourfard et al. [37] proposed the fog to cloud (F2C) data management architecture incorporating the data preservation block to provide faster data access than the cloud. To illustrate the possible benefits of the proposed architecture, they calculated the potential reduction in the data transfer volume and latency decrease, taking the city of Barcelona as an example. Sinaeepourfard et al. did not run real-world experiments. Jararweh et al. proposed a hierarchical model composed of mobile edge computing servers and cloudlets, small clouds located close to the edge of the network. Their experiments consisted of simulation scenarios; varying numbers of requests were generated with the objective of demonstrating how the offloading impacts the power consumption and the incurred delay. While Sinaeepourfard et al. and Jararweh et al. demonstrate potentials of edge computing, this thesis is concerned with embedding intelligence in the edge for data analytics tasks. Edge computing has been used to reduce network traffic in a variety of applications, while video compression has been a common way of dealing with video downloads, uploads, and streaming. Video compression is based on the understanding that information between consecutive frames changes very slowly. For example, the background may not need to be encoded for each frame but can be reused. As the name indicates, video compression targets specificallyvideos and takes advantage of relationships between video elements. In contrast, our approach is designed for IoT data with the objective of reducing network traffic specifically for machine learning applications. In our approach, autoencoders find relationships between readings within the same timestep and well as between consequent timesteps through the sliding window approach.

In addition to studies investigating edge computing architectures, potentials, and advantages in general scenarios, a number of studies investigated applications of edge computing in specific domains. The three most commonly discussed areas of edge computing applications are smart homes and cities, healthcare, and manufacturing; therefore, the following subsections discuss edge computing applications in those domains.

2. Related Work:

Mohiuddin, Irfan, et al. [30] talked about the issues encompassing capacity units in server farms. An extraordinary arrangement strategy to guarantee loads are similarly scattered during allotment, and our key commitment is on the VM-based relocation approach. The VM movement is expected to merge the VMs relying upon the responsibility, diminish the utilization of assets, and empower green processing. In that capacity, we should call the methodology Workload Aware Virtual Machine Consolidation Method (WAVMCM). The creator additionally confirms the proposed strategy by standing out it's anything but an AI-subordinate probabilistic technique, including Simulated Annealing, Genetic Algorithm, and a test to look at the meandering rates between cells. Analyses show that WAVMCM diminishes the quantity of working workers by 9%, saving 15% of the CPU's electrical utilization than hereditary calculation based methodologies.

Zhang, Wei-Zhe, et al. [42] suggested a Joint Load Balancing and Mobile Edge Computing (MEC) Offloading Strategy, adding another security layer to alleviate conceivable security issues. Then, a heap adjusting calculation is proposed to reallocate sBS cell phone clients (MDUs) adequately. Furthermore, another high level encryption standard (AES) cryptographic innovation is introduced as an insurance layer for defending information weakness during transmission with an electrocardiogram (ECG) signal-based encryption and unscrambling. An upgraded model for load balance, estimation offloading, and insurance is frequently considered as a worry to lessen the framework's time and energy necessities. Point by point test discoveries show that, contrasted and neighborhood executions, our machine utilization with and without the extra security levels will save some 68,2% and 72,4%.

Riad, Khaled, et al. [43] The Multi-Dimensional Access Control (MD-AC) programming to permit and eliminate clients in the Cloud progressively through different specialists has been embraced. The exploratory discoveries show that MD-AC will decide demands for access in a sensible and suitable period. The normal encryption and decoding times are 18 and 10ms, individually, regardless of exceptionally confounded research center conditions and various exchanges. The proposed conspire is additionally tried and appeared differently in relation to best in class plans as of late. The discoveries show that the proposed system against various notable assaults is quick and stable. Likewise, MD-AC can be utilized to secure IoT administrations' protection in the cloud world.

3. Methodology:

PCA stands for principle component analysis. PCA is used for the dimension reduction. It learns the linear relationship among the data and then reduces the dimension. The PCA algorithm firstly calculate the mean of the data at axis 0 that is row wise mean. The mean is subtracted from the original data then the covariance of the data is obtained. The Eigen vector of the covariance is calculated and then the resulting vector is multiplied with the original dataset and the result is returned as the transformed value. The PCA algorithm firstly scales the data using the scales the value by using the standard scalars which will use the standard deviation and the data is scaled by using the standard deviation. Let's take up an example suppose we have dataset x with shape of n rows and k features. The data set is used to construct a variable space as shown in the figure 1.

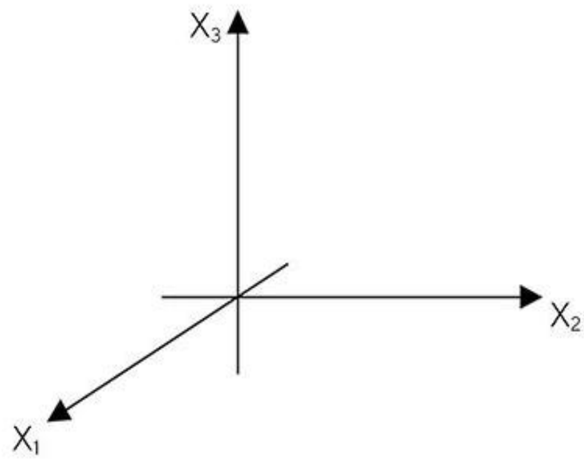


Fig. 1: variable space of dataset

The original data is placed on this variable space as shown in the figure 2.

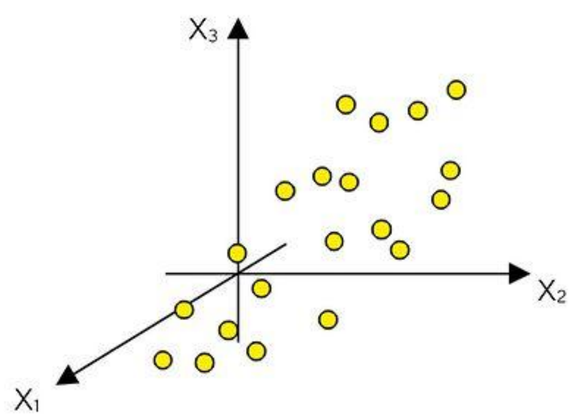
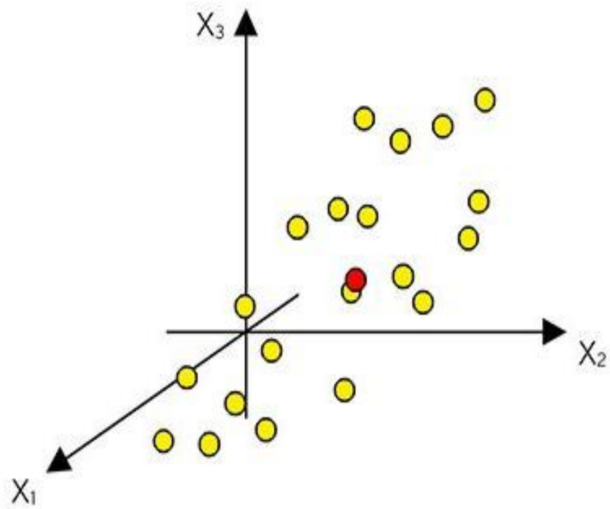


Fig. 2: variable space of dataset

The model calculates the mean of the variable space and subtracts from the original data and the residual data vector is a point of same dimension as of the data set. the mean of data in the variable space would look like the figure 3



Figu. 3: variable space of dataset after subtracting the mean

After subtracting the mean with the original dataset the data will rearrange and reach at the origin and other data points on their respective axis the data visualization would be same as the figure 4.

.

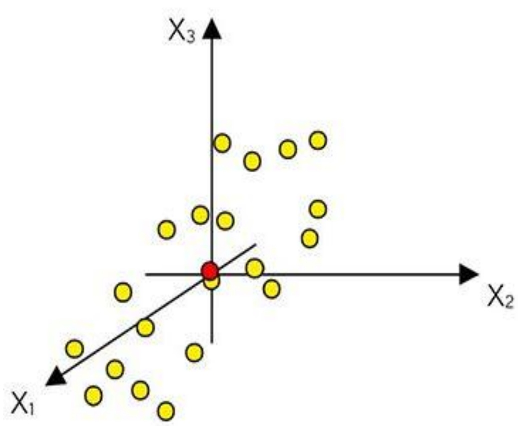


Fig. 4: variable space of dataset after subtracting the mean

After shifting the coordinates to the mean of dataset and the dataset on the axis. Now the data can be used to calculate the 1st principal component. The line of the axis in the variable space is the first principal component which represents the least square. We can clearly see that all the data points are now place on this axis which is called as the first principal component. The coordinate value obtained is also called as the scores. The graph of the first principal component is shown in the figure 5.

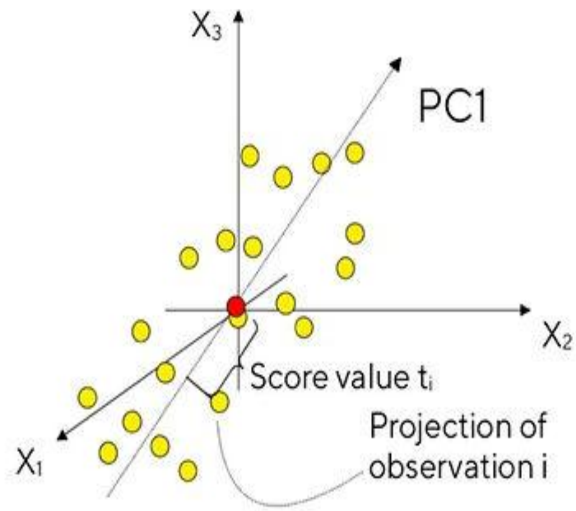


Fig. 5: first principal component.

A line orthogonal to the first principle component is termed as the second principal component. The average points fall on the second principal component thus improves the approximation of the dataset. The second principal component is also calculated by using the variable space. The second principal component is obtained by taking the covariance of the data set which is subtracted by the mean of the data set. The figure 6 shows the second principal component.

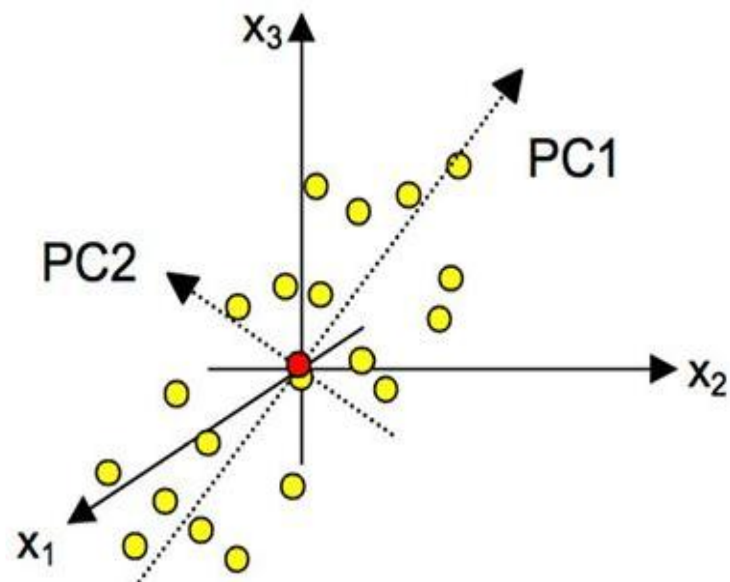


Fig 6 second principal component.

The combination of the first and the second principal component defines the score plot. The score plot is the area of the data with low dimensional subspace which is shown in the figure 7. The score plot is calculated by using the covariance, Eigen vector of the covariance is called as the score plot. The box formed in the graph is called as the score plot or the Eigen vector.

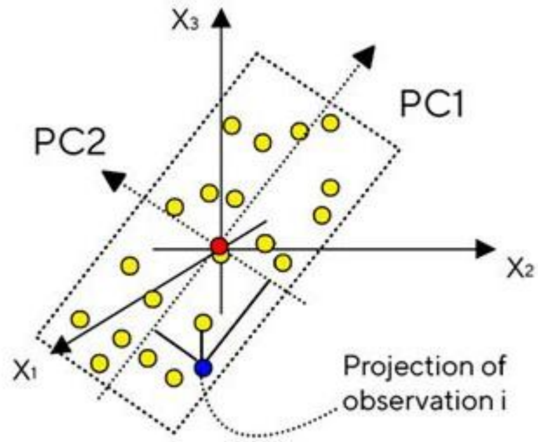


Fig. 7: score plot.

The biggest drawback of the PCA algorithm is that it is useful when used with the dataset which contains some linear relationship among the features of the feature set.
In The proposed methodology Firstly the human activity data set is encrypted (dimension reduction) by using the kernel PCA. We will reduce the dataset dimension to 75%, 50% and 25% .and sending the data to the server. The second phase is to decrypt the data to original size and training a machine learning based model to check the accuracy of the model. A comparison will be made between the accuracy of the machine learning model trained with the original data set and the dataset obtained after encrypting and decrypting the model. We will consider three scenarios, in first scenario the data dimension is reduced and sent to the server then the machine learning model is trained by using the data received as shown in the fig 7.in second scenario the data generated by the sensors is directly sent to the server and the machine learning model is trained by using the data received as shown in the fig 8. In third scenario the data is encrypted on the edge node of the sensors and then sent to the server, on server we decrypt the data to restore the original dimension. For the same purpose we also use the Autoencoders and PCA algorithm to measure the performance of the proposed technique. A multidimensional result analysis of the machine learning model is performed which includes parameters like F1-score, precision, recall, accuracy.

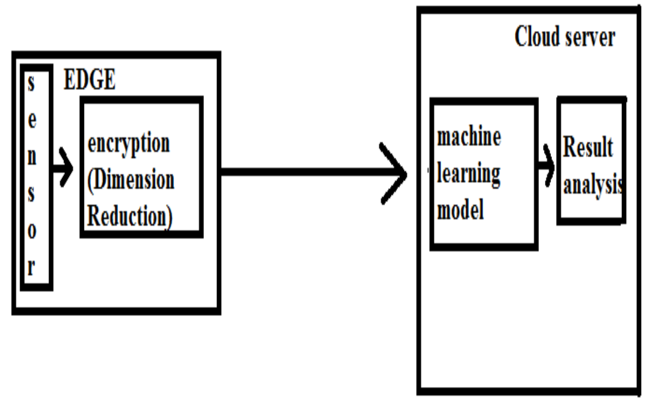


Fig 7: Block diagram of scenario 1.

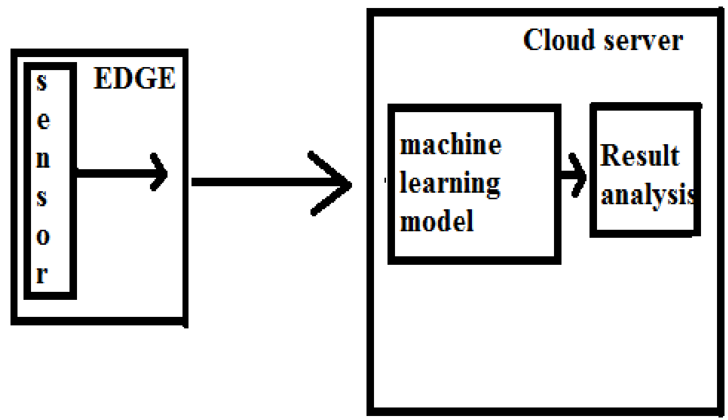


Fig 8: Block diagram of scenario 2.

4. Result and Discussion:

For implementation of the proposed model python programming language is used. Python is a popular technology due its versatility of been used in the field of web site development, app development, IoT (internet of things) and in the field of research and analytics. For implementation of our model we have used the various tools of python like scikit-learn, tensorflow, keras, matplotlib, seaborn, jupyter notebook.
The proposed model is implemented on the minist data set which is originally a handwriting dataset, it contains 70000 images with high dimensions. The data is multidimensional. The data is converted to 2 dimensional and the data is scaled by using the minmax scalers. The data is split in to 80:20 ratio for training and testing purpose. The original dataset is used to build a logistic regression model. The model is trained on the training set and accuracy is measured by using the testing set.
Firstly the auto encoders are used for dimension reduction i.e. encoding and decoding. The auto encoders are built by using the neural network model with relu as activation function and Adams loss function. The dataset is encrypted to 128, 64, 32

features and then the data is decrypted. The encrypted and decrypted data are used to train the logistic regression model and the test data is used to measure the prediction accuracy of the model. The results are as shown in figure 9.

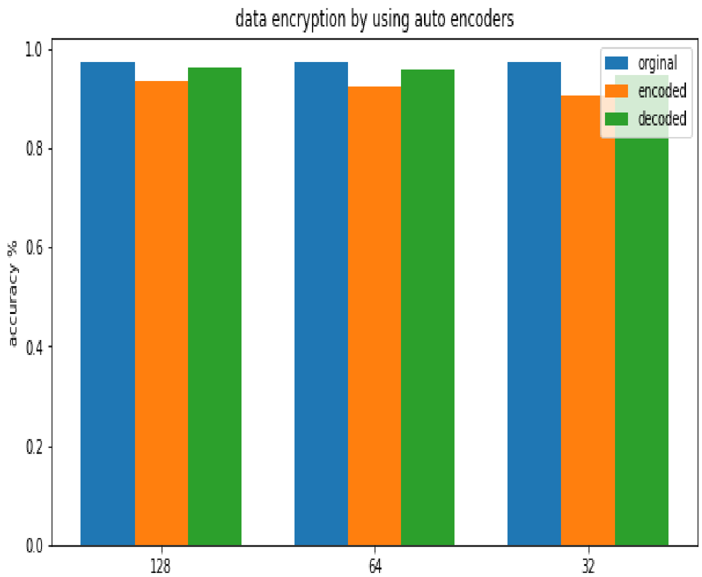


Fig. 9: Results of Autoencoders

The same dataset is encrypted and decrypted by using the PCA algorithm. The data is encrypted into 128, 64, 32 features and then decrypted. Bothe the encrypted and decrypted dataset is used to train the logistic regression model and the results are shown in figure 10.

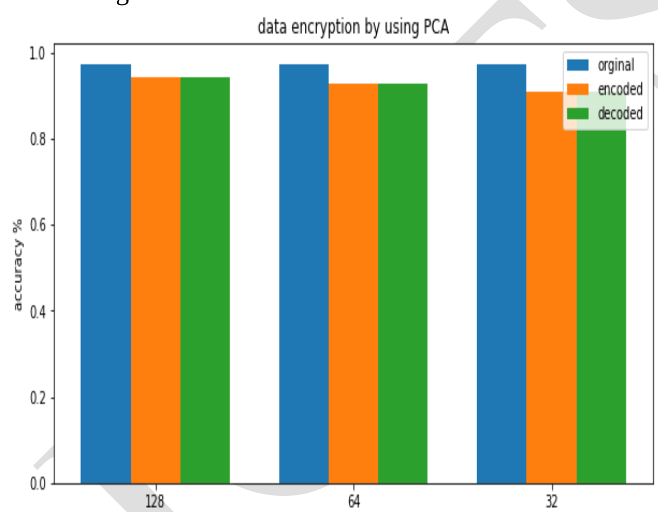


Fig. 10: Results of PCA

The same data set is encrypted and decrypted by using the KernelPCA algorithm with cosine kernel. For decryption we need to provide on the fit_inverse_transform parameter. The kernelPCA is used to encrypt the dataset resulting 128, 64, 32 features. The encrypted and decrypted dataset is used to train the logistic regression model and the results are shown in figure 11.

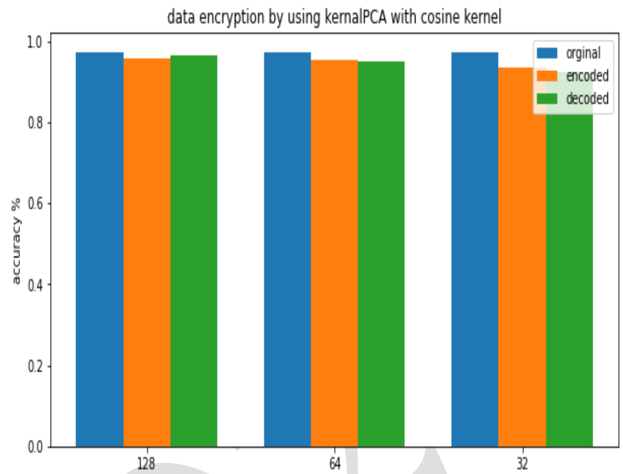


Fig. 11: Results of KernelPCA

5. Conclusion:

In our proposed work we have implemented edge computing for data dimension reduction. The encrypted data is sent over the network to the cloud platform and then the data is decrypted and the data set is used to train the logistic regression model. And the result shows that the KernelPCA is a better model for data dimension reduction and the size of the encrypted dataset is just 20 percent due to which the latency of the data over the network is decreased and hence the network congestion is also very low due to which there will be a proper and smooth connectivity between the cloud server and the iot devices. The Iot device's which are used in the critical and emergency services can be benefited the most. the KernalPCa algorithm can easily learn both the linear and non linear relation

References:

[1] M. A. Sadeeq, S. R. Zeebaree, R. Qashi, S. H. Ahmed, and K. Jacksi, "Internet of Things security: a survey," in 2018 International Conference on Advanced Science and Engineering (ICOASE), 2018, pp. 162-166.

[2] L. M. Haji, O. M. Ahmad, S. R. Zeebaree, H. I. Dino, R. R. Zebari, and H. M. Shukur, "Impact of cloud computing and internet of things on the future internet," Technology Reports of Kansai University, vol. 62, pp. 2179-2190, 2020.

[3] S. Zeebaree and H. M. Yasin, "Arduino based remote controlling for home: power saving, security and protection," International Journal of Scientific & Engineering Research, vol. 5, pp. 266-272, 2014.

[4] A. I. Abdulla, A. S. Abdulraheem, A. A. Salih, M. A. Sadeeq, A. J. Ahmed, B. M. Ferzor, et al., "Internet of Things and Smart Home Security," Technol. Rep. Kansai Univ, vol. 62, pp. 2465-2476, 2020.

[5] A. A. Salih, S. R. Zeebaree, A. S. Abdulraheem, R. R. Zebari, M. A. Sadeeq, and O. M. Ahmed, "Evolution of Mobile Wireless Communication to 5G Revolution,"

Technology Reports of Kansai University, vol. 62, pp. 2139-2151, 2020.
[6] A. Pichan, M. Lazarescu, and S. T. Soh, "A Logging Model for Enabling Digital Forensics in IoT, in an Inter-connected IoT, Cloud Eco-systems," in 2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4), 2020, pp. 478-483.
[7] A. S. Abdulraheem, A. A. Salih, A. I. Abdulla, M. A. Sadeeq, N. O. Salim, H. Abdullah, et al., "Home automation system based on IoT," 2020.
[8] S. I. Saleem, S. Zeebaree, D. Q. Zeebaree, and A. M. Abdulazeez, "Building Smart Cities Applications based on IoT Technologies: A Review," Technology Reports of Kansai University, vol. 62, pp. 1083-1092, 2020.
[9] O. Alzakholi, H. Shukur, R. Zebari, S. Abas, and M. Sadeeq, "Comparison among cloud technologies and cloud performance," Journal of Applied Science and Technology Trends, vol. 1, pp. 40-47, 2020.
[10] Z. N. Rashid, S. R. Zebari, K. H. Sharif, and K. Jacksi, "Distributed cloud computing and distributed parallel computing: A review," in 2018 International Conference on Advanced Science and Engineering (ICOASE), 2018, pp. 167-172.
[11] Z. Ageed, M. R. Mahmood, M. Sadeeq, M. B. Abdulrazzaq, and H. Dino, "Cloud computing resources impacts on heavy-load parallel processing approaches," IOSR Journal of Computer Engineering (IOSR-JCE), vol. 22, pp. 30-41, 2020.
[12] R. Revathi, M. Suganya, and G. M. NR, "IoT based Cloud Integrated Smart Classroom for smart and a sustainable Campus," Procedia Computer Science, vol. 172, pp. 77-81, 2020.
[13] Z. S. Ageed, R. K. Ibrahim, and M. A. Sadeeq, "Unified Ontology Implementation of Cloud Computing for Distributed Systems," Current Journal of Applied Science and Technology, pp. 82-97, 2020.
[14] P. Y. Abdullah, S. R. Zeebaree, H. M. Shukur, and K. Jacksi, "HRM system using cloud computing for Small and Medium Enterprises (SMEs)," Technology Reports of Kansai University, vol. 62, p. 04, 2020.
[15] Z. N. Rashid, S. R. Zeebaree, and A. Shengul, "Design and analysis of proposed remote controlling distributed parallel computing system over the cloud," in 2019 International Conference on Advanced Science and Engineering (ICOASE), 2019, pp. 118-123.
[16] M. Sadeeq, A. I. Abdulla, A. S. Abdulraheem, and Z. S. Ageed, "Impact of Electronic Commerce on Enterprise Business," Technol. Rep. Kansai Univ, vol. 62, pp. 2365-2378, 2020.
[17] A. M. Abdulazeez, S. R. Zeebaree, and M. A. Sadeeq, "Design and Implementation of Electronic Student Affairs System," Academic Journal of Nawroz University, vol. 7, pp. 66-73, 2018.
[18] H. Shukur, S. Zeebaree, R. Zebari, D. Zeebaree, O. Ahmed, and A. Salih, "Cloud computing virtualization of resources allocation for distributed systems," Journal of Applied Science and Technology Trends, vol. 1, pp. 98-105, 2020.
[19] L. M. Haji, S. Zeebaree, O. M. Ahmed, A. B. Sallow, K. Jacksi, and R. R. Zeabri, "Dynamic resource allocation for distributed systems and cloud computing," TEST Engineering & Management, vol. 83, pp. 22417-22426, 2020.
[20] A. Hosseinian-Far, M. Ramachandran, and C. L. Slack, "Emerging trends in cloud computing, big data, fog computing, IoT and smart living," in Technology for smart futures, ed: Springer, 2018, pp. 29-40.
[21] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the edge: A scalable IoT architecture based on transparent computing," IEEE Network, vol. 31, pp. 96-105, 2017.
[22] M. Aazam, I. Khan, A. A. Alsaffar, and E.-N. Huh, "Cloud of Things: Integrating Internet of Things and cloud computing and the issues involved," in Proceedings of 2014 11th International Bhurban Conference on Applied Sciences & Technology (IBCAST) Islamabad, Pakistan, 14th-18th January, 2014, 2014, pp. 414-419.
[23] R. Gravina, P. Alinia, H. Ghasemzadeh, and G. Fortino, "Multi-sensor fusion in body sensor networks: State-of-the-art and research challenges," Information Fusion, vol. 35, pp. 68-80, 2017.
[24] P. Tan, H. Wu, P. Li, and H. Xu, "Teaching management system with applications of RFID and IoT technology," Education Sciences, vol. 8, p. 26, 2018.
[25] S. R. Zeebaree, K. Jacksi, and R. R. Zebari, "Impact analysis of SYN flood DDoS attack on HAProxy and NLB cluster-based web servers," Indones. J. Electr. Eng. Comput. Sci, vol. 19, pp. 510-517, 2020.
[26] C. Stergiou, K. E. Psannis, B.-G. Kim, and B. Gupta, "Secure integration of IoT and cloud computing," Future Generation Computer Systems, vol. 78, pp. 964-975, 2018.
[27] S. Kianoush, M. Raja, S. Savazzi, and S. Sigg, "A cloud-IoT platform for passive radio sensing: Challenges and application case studies," IEEE Internet of Things Journal, vol. 5, pp. 3624-3636, 2018.
[28] Z. N. Rashid, S. R. Zeebaree, and A. Sengur, "Novel Remote Parallel Processing Code-Breaker System via Cloud Computing."
[29] A. B. Sallow, M. Sadeeq, R. R. Zebari, M. B. Abdulrazzaq, M. R. Mahmood, H. M. Shukur, et al., "An Investigation for Mobile Malware Behavioral and Detection Techniques Based on Android Platform," IOSR Journal of Computer Engineering (IOSR-JCE), vol. 22, pp. 14-20.
[30] I. Mohiuddin and A. Almogren, "Workload aware VM consolidation method in edge/cloud computing for IoT applications," Journal of Parallel and Distributed Computing, vol. 123, pp. 204-214, 2019.
[31] S. S. Gill, S. Tuli, M. Xu, I. Singh, K. V. Singh, D. Lindsay, et al., "Transformative effects of IoT, Blockchain and Artificial Intelligence on cloud computing: Evolution, vision, trends and open challenges," Internet of Things, vol. 8, p. 100118, 2019.
[32] P. Y. Abdullah, S. R. Zeebaree, K. Jacksi, and R. R. Zeabri, "An hrm system for small and medium enterprises

(sme) s based on cloud computing technology," International Journal of Research-GRANTHAALAYAH, vol. 8, pp. 56-64, 2020.
[33] S. Zeebaree, R. R. Zebari, K. Jacksi, and D. A. Hasan, "Security Approaches For Integrated Enterprise Systems Performance: A Review," Int. J. Sci. Technol. Res, vol. 8, 2019.
[34] L. a. Tawalbeh, F. Muheidat, M. Tawalbeh, and M. Quwaider, "IoT Privacy and security: Challenges and solutions," Applied Sciences, vol. 10, p. 4102, 2020.
[35] I. Mohiuddin, A. Almogren, M. Alrubaian, and M. Al-Qurishi, "Analysis of network issues and their impact on Cloud Storage," in 2019 2nd International Conference on Computer Applications & Information Security (ICCAIS), 2019, pp. 1-4.
[36] S. K. Tayyaba, H. A. Khattak, A. Almogren, M. A. Shah, I. U. Din, I. Alkhalifa, et al., "5G vehicular network resource management for improving radio access through machine learning," IEEE Access, vol. 8, pp. 6792-6800, 2020.
[37] H. N. Almajed, A. S. Almogren, and A. Altameem, "A resilient smart body sensor network through pyramid interconnection," IEEE Access, vol. 7, pp. 51039-51046, 2019.
[38] H. Elazhary, "Internet of Things (IoT), mobile cloud, cloudlet, mobile IoT, IoT cloud, fog, mobile edge, and edge emerging computing paradigms: Disambiguation and research directions," Journal of Network and Computer Applications, vol. 128, pp. 105-140, 2019.
[39] A. Almogren, "An automated and intelligent Parkinson disease monitoring system using wearable computing and cloud technology," Cluster Computing, vol. 22, pp. 2309-2316, 2019.
[40] X. Xu, Q. Liu, Y. Luo, K. Peng, X. Zhang, S. Meng, et al., "A computation offloading method over big data for IoT-enabled cloud-edge computing," Future Generation Computer Systems, vol. 95, pp. 522-533, 2019.
[41] K. Haseeb, N. Islam, A. Almogren, I. U. Din, H. N. Almajed, and N. Guizani, "Secret sharing-based energy-aware and multi-hop routing protocol for IoT based WSNs," IEEE Access, vol. 7, pp. 79980-79988, 2019.
[42] W.-Z. Zhang, I. A. Elgendy, M. Hammad, A. M. Iliyasu, X. Du, M. Guizani, et al., "Secure and Optimized Load Balancing for Multi-Tier IoT and Edge-Cloud Computing Systems," IEEE Internet of Things Journal, 2020.
[43] K. Riad, T. Huang, and L. Ke, "A dynamic and hierarchical access control for IoT in multi-authority cloud storage," Journal of Network and Computer Applications, vol. 160, p. 102633, 2020.
[44] B. Mallikarjuna, "Feedback-Based Fuzzy Resource Management in IoT-Based-Cloud," International Journal of Fog Computing (IJFC), vol. 3, pp. 1-21, 2020.
[45] M. Anuradha, T. Jayasankar, N. Prakash, M. Y. Sikkandar, G. Hemalakshmi, C. Bharatiraja, et al., "IoT enabled cancer prediction system to enhance the authentication and security using cloud computing," Microprocessors and Microsystems, vol. 80, p. 103301, 2021.
[46] R. O. Aburukba, M. AliKarrar, T. Landolsi, and K. El-Fakih, "Scheduling Internet of Things requests to minimize latency in hybrid Fog–Cloud computing," Future Generation Computer Systems, vol. 111, pp. 539-551, 2020.
[47] B. Ali, M. A. Pasha, S. ul Islam, H. Song, and R. Buyya, "A volunteer supported fog computing environment for delay-sensitive IoT applications," IEEE Internet of Things Journal, 2020.
[48] M. Wang and Q. Zhang, "Optimized data storage algorithm of IoT based on cloud computing in distributed system," Computer Communications, vol. 157, pp. 124-131, 2020.
[49] H. Fuentes and D. Mauricio, "Smart water consumption measurement system for houses using IoT and cloud computing," Environmental Monitoring and Assessment, vol. 192, pp. 1-16, 2020.
[50] M. Abdel-Basset, G. Manogaran, A. Gamal, and V. Chang, "A novel intelligent medical decision support model based on soft computing and IoT," IEEE Internet of Things Journal, vol. 7, pp. 4160-4170, 2019.
[51] A. Mavromatis, C. Colman-Meixner, A. P. Silva, X. Vasilakos, R. Nejabati, and D. Simeonidou, "A software-defined IoT device management framework for edge and cloud computing," IEEE Internet of Things Journal, vol. 7, pp. 1718-1735, 2019.
[52] L. Ding, Z. Wang, X. Wang, and D. Wu, "Security information transmission algorithms for IoT based on cloud computing," Computer Communications, vol. 155, pp. 32-39, 2020.
[53] O. Debauche, S. Mahmoudi, S. A. Mahmoudi, P. Manneback, and F. Lebeau, "A new edge architecture for ai-iot services deployment," Procedia Computer Science, vol. 175, pp. 10-19, 2020.
[54] J. Chen, S. Chen, Q. Wang, B. Cao, G. Feng, and J. Hu, "iRAF: A deep reinforcement learning approach for collaborative mobile edge computing IoT networks," IEEE Internet of Things Journal, vol. 6, pp. 7011-7024, 2019.
[55] M. Abbasi, M. Yaghoobikia, M. Rafiee, A. Jolfaei, and M. R. Khosravi, "Efficient resource management and workload allocation in fog–cloud computing paradigm in IoT using learning classifier systems," Computer Communications, vol. 153, pp. 217-228, 2020.