An Efficient Kernel Density Based Algorithm of Big Data in Cybersecurity for Enhancing Smart City

https://doi.org/10.3991/ijoe.v18i01.27875

Khaled H. Alyoubi
Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia
kalyoubi@kau.edu.sa

Abstract—Smart cities are attracting much interest in terms of future development. As new technologies come on stream, ordinary towns are reshaping themselves as smart cities, where technology is used to improve connections between all elements of the town. The technology can be embedded everywhere and can harvest data for dedicated smart city applications. Smart cities will have a huge number of different devices running these applications. There will be a substantial amount of data associated with these devices. In the interlinked smart city environment, many different messages could be shared between them. Such devices will be associated with many security risks and privacy issues, as many of the shared statistics could also hold personal data. A substantial review of research has been recently undertaken to ensure that data will be safe in the smart city environment. This review has included all the latest research in the area and is intended to ensure that all the data required to run green smart cities and the devices required for them will remain secure and confidential.

Keywords—big data, cybersecurity, smart cities, auditing, SVM algorithm, KDE

1 Introduction

Big data represents a massive collection of a variety of data sets. In the current era, big data is a central part of our everyday lives: from personal users to substantial companies, big data is everywhere. Whenever substantial amounts of data are held, cybersecurity becomes an issue. All data is valuable and has to be kept secure from hackers or other malicious actors. Previously, data was kept in hardcopy form, i.e., on paper, a system that had many drawbacks [1].

With the development of new technologies, written records became digitized, to the extent that now we have the concept of the IoT where it is possible for humans to have direct communication with electronic devices without needing to know how to program them in any way. The IoT will be central to the technological revolution that will allow for new entities such as smart cities. Within smart cities, sensors will allow for smart environmental management, smart water management, smart parking facilities, and many more [2]. Although there are many upsides to the concept of smart cities, there

are also drawbacks, such as the potential for cyber-attacks. If the system is to remain credible, it has to be guaranteed that data cannot be accessed by malicious actors. This means that continuous monitoring of all processes and carrying out audits is essential. To meet this need, the SVM algorithm is used to analyze a smart city data set, demonstrating how important monitoring will be [3]. A review of the activities needed for a smart city provides us with a conclusion that the danger of cyber-attacks can be significantly mitigated through continuous monitoring of all aspects of the system.

2 Applications of smart cities

Figure 1 shows a number of smart city applications. Smart cities have many different facets, ranging from smart cards that will allow easy payment and authentication all the way to smart mobility management that will make the visitor experience more pleasant and reduce CO2 emissions, and smart control of resources such as electricity or water [4].

Other areas that can be managed by smart cities are:

- T Managing traffic
- Managing air pollution
- Monitoring and managing radiation levels
- Managing public transport
- Locating lost and stolen items
- Diagnosing problems with vehicles
- Managing natural disasters

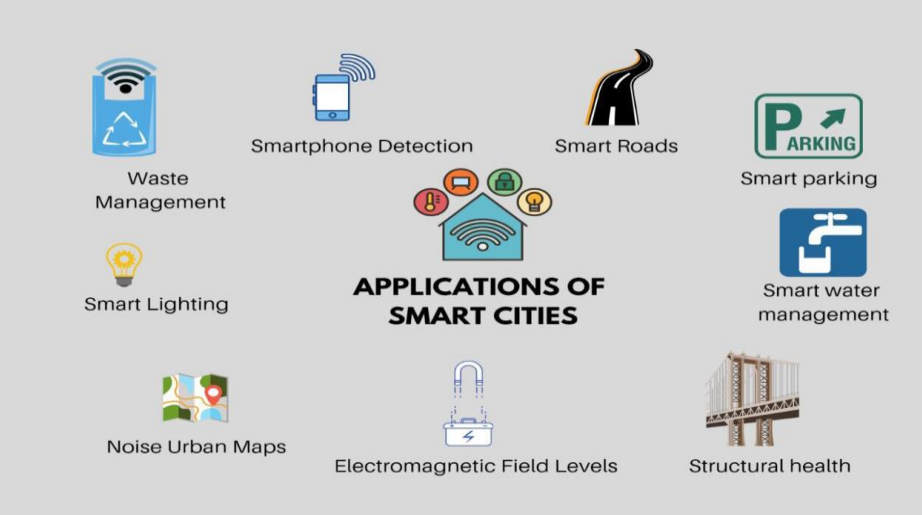

Fig. 1. Applications of smart cities [2]

3 Big data security

Nearly 62% of security breaches are caused by hackers, with 51% being malicious attacks and 43% employing social media interfaces to discover ways in which information or access can be obtained [5].

3.1 Data security requirements

Figure 2 shows the three essential elements of data security, these being confidentiality, availability, and integrity [4].

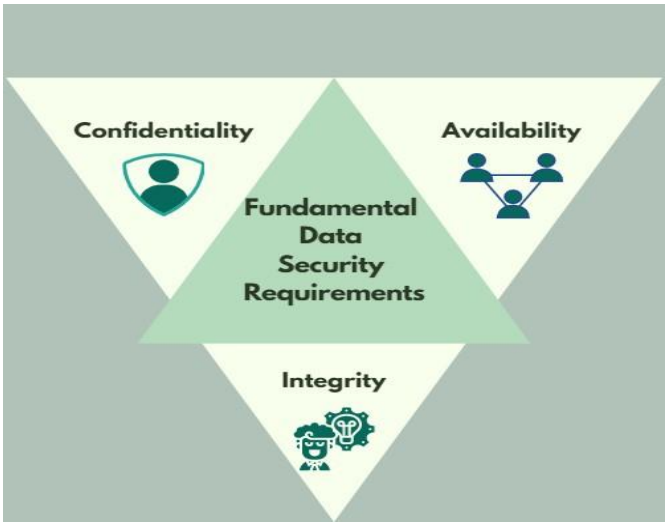

Fig. 2. Fundamental data security requirement [4]

3.2 Big data challenges

Building trust is central to smart city concepts because the necessary technology and structures must be able to securely communicate huge quantities of data in an ecologically sound and reliable manner [5, 6]. New software and hardware have been developed that can control the flow of information as it facilitates communication between every level of government and citizens.

The fundamental risks of smart technologies should be examined from both political and social perspectives that consider the preferred ideas of technology designers [7]. Smart city initiatives often use a one size fits all approach with technology; customization should be applied to ensure that the technology suits every potential user.

Many new trends in Wi-Fi and sensor developments have created the potential for the widespread dissemination of IoT Technology to create a working smart city [8]. The technology must have sufficient built-in protections in order to protect the vulnerable and ensure that data remain secure. All smart cities are complex entities and will

always carry some risk of data being acquired, used, and shared by parties who should not have access to it.

One of the major demands on making smart city projects sustainable is the requirement to make allowances for human behavior and motivation. The smart city will be a supplier of enormous innovation, providing answers to the demands of increased urbanization and the greater focus on sustainability, power-sharing, security, health, and mobility [9]. The increased use of sustainable energy in order to improve sustainability and manage natural resources can have a very significant influence on people's lifestyles and answer the moral demands of contemporary society.

4 Cyber security methods to reduce risks

4.1 Employing the principle of least privilege

Tiered admission procedures should be employed for manipulating and enforcing the principle of least privilege (PoPL). This principle is based on imposing user restrictions that only allow the minimum level of access required for a system to function properly. In other words, consumers should only be given the privileges that they need to access the services they require; this is a good way of preventing malicious actors from illegally harvesting data through the mining activity.

4.2 Employ cutting edge antivirus measures

Numerous different providers of antivirus software provide protection, with many having a particular focus on the use of big data. The entire big data environment should be protected using the latest available antivirus software. All manufacturer-supplied updates and patches should be installed at the very earliest opportunity [10].

4.3 Timetable periodic audits

Big data is a growing market and technology is evolving at a rapid pace, meaning that the technology available at present will not always be able to meet future demands. Undertaking periodic audits will allow you to assess which technology is becoming outdated and what new technology could replace it, meaning that your security measures will always match up to the latest standard [11].

4.4 Predictions using machine learning

Machine learning algorithms can use the data gathered from protection mechanisms to look at both historical and contemporary ways of assessing and making predictions regarding new hazards. These techniques can help to find out the ways malicious actors may behave before they have a chance to put their plans into action [12].

4.5 Automation and large-scale monitoring

A substantial proportion of cyberattacks occur due to an organization's staff not being properly trained regarding the threats the organization is facing. In many cases employees are not aware of the cyber threats they face and do not know how they should react when an attack is instigated, meaning that they leave an open goal for malicious actors [7].

4.6 Real-time detection of intruders

It can be hard to identify and deal with attacks in real-time, but using mass data analysis could resolve this problem by employing automated techniques on a large scale. Intrusion detection systems (IDS) could be extremely useful for undertaking real-time analysis with huge capacities for identifying any malicious intrusions in a system [7]. Such protections can repel attacks before any unauthorized system of the mission is gained by the attacker. One example of such security would be the integration of various data sets using proxy logs, creating secure domain names, and fine-tuning structural suitability.

4.7 Risk management reporting

Continual assessment of risk is vital to keep protection against cyber-attacks viable, and continuous analysis and reporting can help with this. Massive information analysis looks at every structure and data store and can propose actions that will assist with building protection [13]. Examples of metrics that may be reported would be any incidents that occur, levels of authentication, individual access, system usage outside commercial hours, etc.

5 Methodology

5.1 Plotting a dataset traditionally

Because logistic regression does not have the capacity of creating an assumption regarding class distribution within feature spaces it has been rejected as out of date. In Python, a Count plot is employed to undertake logistic regression.

Countplot is a standard and well-recognized means of plotting a dataset with representations using a variety of colored bars with no specificity. In Figure 3 the x-axis shows the SmartCity_Index_relative_Edmonton; Countplot cannot use a y-axis at the same time, making it less useful for estimations.

```
In [44]: sns.countplot(x=data['SmartCity_Index_relative_Edmonton'])
Out[44]: <matplotlib.axes._subplots.AxesSubplot at 0x21c8e493b00>
```

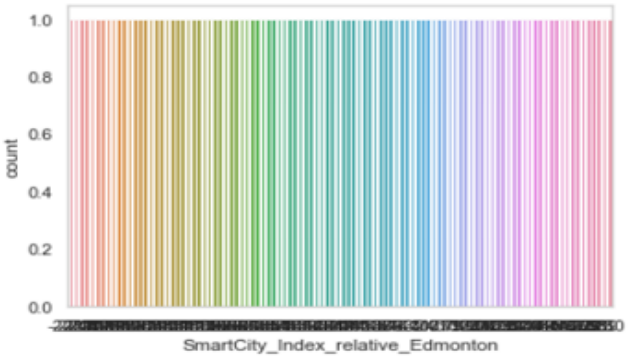

Fig. 3. SmartCity index relative Edmonton

5.2 Using Python to plot a dataset

The KDE plot provides greater accuracy for making estimations in comparison to the standard logistic regression.

A plot of a sample diagram is made in Figure 4 employing Jointplot [14] with the given kind "KDE" where the x-axis is SmartCity_Index_relative_Edmonton and y-axisis SmartCity_Index:

```
sns.jointplot(data['SmartCity_Index_relative_Edmonton'],data['SmartCity_Index'],kind="kde")
<seaborn.axisgrid.JointGrid at 0x249e98efeb8>
```

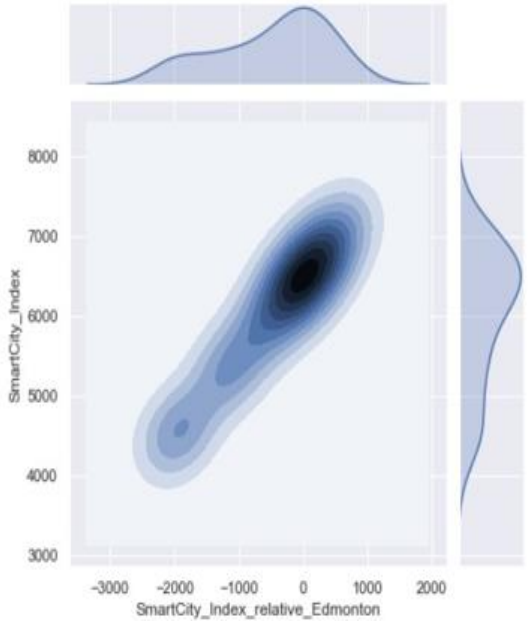

Fig. 4. The joint plot of smart city index relative Edmonton and smart city index

6 Experimental results

- **Support Vector Machine Algorithm:** SVM (Support Vector Machine) is one of the supervised learning algorithms with the greatest relevance implied for performing regression problems and classification. It is primarily employed to resolve classification issues in machine learning [3].
- **The KDE Algorithm:** Kernel Density Estimation is an algorithm used to estimate the probability density function that can be essential in allowing detailed examination of the research opportunity distributions compared to employing a standard histogram [3].

Although the title Kernel Density Estimation might appear complex, it is extremely useful in analyzing statistics [3]. Frequently abbreviated as KDE, it's a means of creating a clean curve when provided with accurate raw data.

With the KDE algorithm, bandwidth is used as a parameter, providing a smooth curve.

The following mathematical expression is used to weight the distances of observations from a specific point:

$$f(x) = \sum_{observation} K\left(\frac{x - observations}{bandwilsth}\right) \quad (1)$$

K represents the kernel function, with a variety of different estimates being produced from a variety of kernel functions [3].

Step By Step Procedure:

- STEP 1: First and foremost set the observed continuous process as definitelike {x1,x2,⋯,xN}{x1,x2,⋯,xN}.
- STEP 2: Now consider (x_o) as left bound, h as the width of the bin and calculate the bin k probability estimator f_h(k):
- STEP 3: Bin k is represented as [xo + (k − 1)h, xo + k × h)]
- STEP 4: f^h(k) is calculated using the below formula:

$$f^{h(k)} = \frac{\sum_{i=1}^{N} I\{(k-1)h \le x_i - x_0 \le k \times h\}}{N} \quad (2)$$

- STEP 5: An event function I{.} is returned as 1 if the condition is true, otherwise zero.
- STEP 6: Lastly, we can describe the histogram method thus:
- STEP 7: The occurrence probability of each event is equal to 1/N which is statistically independent.
- STEP 8: fh(k)fh(k) is found from the sum of the observations or the probabilities in each bin.

7 Implementation

To undertake analysis, we will take a dataset derived from Kaggle.com containing 10 different columns of unique entries. Table 1 shows the country-wise statistical data.

Table 1. Country-wise statistical data

City	Country	Smart People	Smart City Index	Smart City_ Index relative Edmonton
Oslo	Norway	8618	7138	666
Bergen	Norway	8050	7296	823
Amsterdam	Netherlands	7098	7311	839
Copenhagen	Denmark	5780	7171	698
Stockholm	Sweden	6743	6812	340
Montreal	Canada	8465	7353	880
Vienna	Austria	8580	6771	298
Odense	Denmark	6955	6886	414
Singapore	Singapore	9695	6813	341
Boston	United States	6573	6852	380
Zurich	Switzerland	7200	6984	512
Trondheim	Norway	7558	7039	567

Aalborg	Denmark	6955	6720	247
Washington, DC	United States	5930	6662	190
Stavanger	Norway	7595	6818	346
Los Angeles	United States	7498	6437	-35
Helsinki	Finland	7523	6920	448
Vancouver	Canada	7725	7152	679

Employing the Jupyter notebook from Python Figure 5 plots "smart people" on the X-axis and "smart city index" on the y axis.

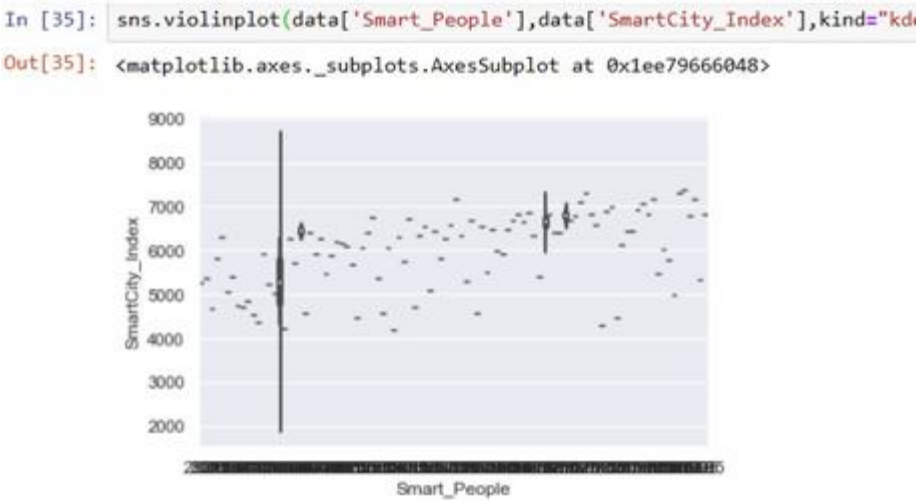

Fig. 5. Violinplot of smart people and smart city

SNS is an abbreviation for seaborn, a visualization library allowing statistical plotting employing Python. Violinplot is another form of plot employed for quantitative data distribution [3].

For Figure 6, smart people are on the x-axis, with the smart city index on the y axis. As this figure is employing kdeplot univariate/multiple variables are combined in the plot.

```
In [39]: sns.kdeplot(data['Smart_People'],data['SmartCity_Index'])
Out[39]: <matplotlib.axes._subplots.AxesSubplot at 0x1ee798f8dd8>
```

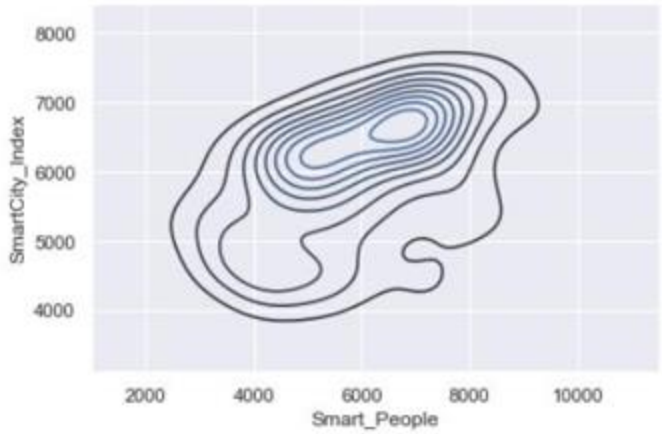

Fig. 6. kdeplot of smart people and smart city

When we compare the techniques, we can see that the KDE Technique has greater efficiency than logistic regression. As the KDE algorithm is capable of defining a smooth curve employing kernel it produces a clear picture to make estimates and offers considerable accuracy. Figure 7 says a graph comparing the KD algorithm and logistic regression; The four bars on the left represent the KDE algorithm, with the four on the right representing logistic regression. Table 2 shows the comparison of algorithms.

Table 2. Comparison of algorithm

Techniques/Algorithm	Accurate estimation	Studied Opportunity distribution	Smooth curve	Probability Density
KDE technique	4.8	3.9	3.5	4
Logistic regression	2	2.4	2	1

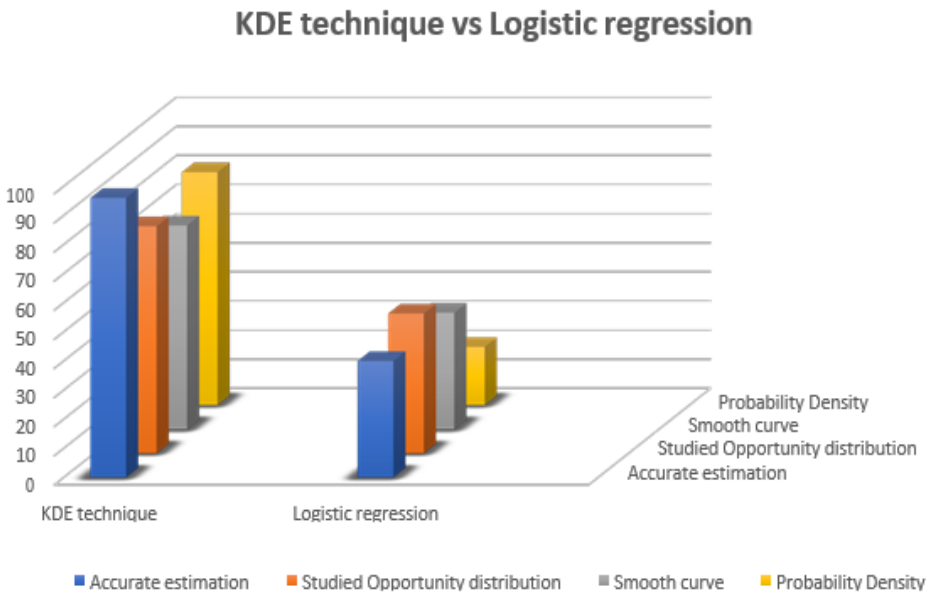

Fig. 7. Efficiency of KDE technique

8 Limitations and remedies

The algorithm employed has certain limitations, as follows:

- As it requires more training time, it does not perform as well with substantial data sets.
- It is always problematic to select a "good" kernel function.
- Noise may be present with overlapping target classes which reduces performance.
- Understanding and interpreting the subsequent model is a complex process.
- KDE only presents vague data representations, causing disruption, and SVM does not guarantee probability estimates [3].

We can mitigate these limitations to a degree by:

- Employing the Probability Density of a continuous variable as the preferred means of visualization;
- Using the "best kernel" trick to ensure the non-linear learning algorithm for SVM;
- Undertaking additional study and analysis to adapt an efficient kernel, which is central to SVM;
- Show more patience whilst building results from SVM.

9 Conclusions

To conclude, smart cities will be with us very soon. Employing applications for smart cities will reduce the burden of work on humans, although it may create security issues. As time progresses developers will be producing more and more applications suitable for using in smart cities. However, this will also increase the possibility of cyber-attacks occurring. In this instance, by undertaking her analysis of a dataset we have shown that small cities can be run, and their futures predicted using the SVM algorithm in combination with a IDE technique. Cybersecurity is a challenge in all of these emerging technologies. The prevention of cyber-attacks in these new domains relies on the identification of the point at which attacks will be made and strengthening them in order to repel malicious actors.

10 References

[1] Cuzzocrea, A. (2014). Privacy and security of big data: current challenges and future research perspectives. In Proceedings of the first international workshop on privacy and secuirty of big data (pp. 45-47). https://doi.org/10.1145/2663715.2669614

[2] John-Green, M. S., & Watson, T. (2014). Safety and Security of the Smart City-when our infrastructure goes online (pp. 4-21). https://doi.org/10.1049/cp.2014.0981

[3] Mahmood, T., & Afzal, U. (2013). Security analytics: Big data analytics for cybersecurity: A review of trends, techniques and tools. In 2013 2nd national conference on Information assurance (ncia) (pp. 129-134). IEEE. https://doi.org/10.1109/NCIA.2013.6725337

[4] Nelson, B., & Olovsson, T. (2016). Security and privacy for big data: A systematic literature review. In 2016 IEEE international conference on big data (big data) (pp. 3693-3702). IEEE. https://doi.org/10.1109/bigdata.2016.7841037

[5] Camp, J. (2009). Data for Cybersecurity Research: Process and" whish list". Retrieved July 15, 2013. Retrieved July 15, 2013, from http://www.gtisc.gatech.edu/files_nsf10/data-wish-list.pdf.

[6] Zhao, J., Wang, L., Tao, J., Chen, J., Sun, W., & Ranjan, R. et al. (2014). A security framework in g-hadoop for big data computing across distributed cloud data centres. Journal of Computer and System Sciences, 80: 994– 1007. https://doi.org/10.1016/j.jcss.2014.02.006

[7] Shin, D., Sahama, T., & Gajanayake, R. (2013). Secured e-health data retrieval in DaaS and Big Data. In 2013 IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom 2013) (pp. 255-259). IEEE. https://doi.org/10.1109/HealthCom.2013.6720677

[8] Sharif, A., Cooney, S., Gong, S., & Vitek, D. (2015). Current security threats and prevention measures relating to cloud services, Hadoop concurrent processing, and big data. In 2015 IEEE International Conference on Big Data (Big Data) (pp. 1865-1870). IEEE. https://doi.org/10.1109/bigdata.2015.7363960

[9] Shin, D., Sahama, T., & Gajanayake, R. (2013). Secured e-health data retrieval in DaaS and Big Data. In 2013 IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom 2013) (pp. 255-259). IEEE. https://doi.org/10.1109/healthcom.2013.6720677

[10] El-Seoud, S. A., El-Sofany, H. F., Abdelfattah, M., & Mohamed, R. (2017). Big Data and Cloud Computing: Trends and Challenges. International Journal of Interactive Mobile Technologies, 11(2). https://doi.org/10.3991/ijim.v11i2.6561

[11] Huang, T., Lan, L., Fang, X., An, P., Min, J., & Wang, F. (2015). Promises and challenges of big data computing in health sciences. Big Data Research, 2(1), 2-11. https://doi.org/10.1016/j.bdr.2015.02.002

[12] Liu, Q., & Zeng, L. (2020). Design and Application of Experimental Teaching System Driven by Big Data Technology in Economics and Management Majors. International Journal of Emerging Technologies in Learning (iJET), 15(11), 56-66. https://doi.org/10.3991/ijet.v15i11.12997

[13] Memon, M. A., Shaikh, A., Sulaiman, A., Alghamdi, A., Alrizq, M., & Archimède, B. (2021). Time and quantity based hybrid consolidation algorithms for reduced cost products delivery. Computers, Materials and Continua, 409-432. http://dx.doi.org/10.32604/cmc.2021.017653

[14] Huda, M., Maseleno, A., Shahrill, M., Jasmi, K. A., Mustari, I., & Basiron, B. (2017). Exploring Adaptive Teaching Competencies in Big Data Era. International Journal of Emerging Technologies in Learning, 12(3). https://doi.org/10.3991/ijet.v12i03.6434

11 Author

Khaled H. Alyoubi is with department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia.

Article submitted 2021-10-19. Resubmitted 2021-11-30. Final acceptance 2021-12-04. Final version published as submitted by the author.