

Classification Aspects of the Data Offloading Process Applied to Fog Computing

Sávio Melo[1(✉)], Cícero Silva[2], and Gibeon Aquino[1]

[1] Federal University of Rio Grande do Norte, Natal 59078-970, Brazil
savio.melo.016@ufrn.edu.br, gibeon@dimap.ufrn.br
[2] Federal Institute of Education, Science and Technology of Paraíba, Catolé do Rocha 58884-000, Brazil
cicero.alves@ifpb.edu.br

Abstract. The process of data offloading in fog-based environments can offer advantages such as energy savings, freeing up storage space, reduction in decision-making time, network usage reduction, among others. Thus, it is important that the data offloading process must be governed by rules and policies which will define how and when the process will be executed in the solution. Therefore, it is necessary to know the aspects of the data offloading process and discover the importance of each one of them, aiming to select the best techniques for policies definition in fog-based systems. So, this paper presents an approach through structuring in the form of a taxonomy, which classifies and organizes the knowledge needed to develop policies governing the data offloading process. Overall, fourteen (14) aspects that influence decisions for the data offloading process were classified. Finally, a practical example of how this approach can be instantiated in real-world scenarios is provided.

Keywords: Data offloading · Fog computing · Taxonomy

1 Introduction

The term called fog computing was defined in 2014 by the CISCO [6]. It refers to a decentralized computing architecture that manages data and processing closer to the network's edge. In summary, fog computing extends cloud computing capabilities by enabling the development of applications for the Internet of Things. Fog computing works between the data source and the cloud, providing several advantages: latency reduction with minimized response times and data transmission reduction from edge servers to cloud. However, fog computing has some bottlenecks: storage and local processing power, which are mandatory requirements for this architecture to work adequately [4].

As fog computing is close to the data origin, it must consider its storage limitations since the archiving and treating of some data is done locally. Therefore, it is necessary to think about ways that can help mitigate the problems caused

© Springer Nature Switzerland AG 2021
O. Gervasi et al. (Eds.): ICCSA 2021, LNCS 12957, pp. 340–353, 2021.
https://doi.org/10.1007/978-3-030-87013-3_26

by these limitations. One alternative would be to transfer this data to entities with larger storage capacities, a strategy known as computational offloading. Offloading is performed when an entity cannot complete a specific task with its resources due to an infrastructure limitation. When this happens, the most common is that this task is transferred to another more robust entity such as the cloud. The cloud can be used effectively to store, analyze, and use accumulated process data [2]. Data offloading from the fog nodes to the cloud happens in several cases. The most common of them is when there is a need for longer-term storage [1].

Policies need to be defined to conduct the data transfer process between persistence entities on the data offloading task. These policies determine security factors, storage, timeout, and issues involving data management and privacy. Therefore, their correct elaboration can help develop more secure and efficient applications in fog computing environments.

The realization of the process of data offloading has several purposes. In [1], are listed some of them: load balancing, long-term storage, latency, data management, among others. In the same study, such goals are called "criteria for performing data offloading". However, since each criterion depends on the architecture and domain characteristics, defining a standard policy for data offloading in diverse scenarios would be a difficult task since there could be numerous policy possibilities that meet the various aspects to be considered.

Because of the onerousness of designing policies for data *offloading*, this paper suggests a way that facilitates the selection of essential concepts and techniques of the data offloading process, both from the data perspective and from the system infrastructure perspective. Thus, the main contribution of this work is to provide a classification structure that organizes aspects of the data offloading process, enabling policy development in fog-based solutions. Furthermore, such a framework systematizes knowledge about this process and supports its execution to be composed of the best techniques. Therefore, this serves as a support for the solutions in the fog environment to enjoy the benefits of data offloading, such as energy savings, freeing up storage space, decreased network usage, decreased decision-making time, among others.

The rest of this paper is organized as follows: In Sect. 2, the works related to the theme of this article are described. Section 3 discusses the approach in the form of a taxonomy. Section 4 demonstrates the application and instantiation of the proposed approach through a case study. Finally, Sect. 5 talks about the main findings and future work.

2 Related Works

Alelaiwi et al. [3] and Pisan et al. [22] use the data offloading process to meet their applications' requirements better. So, they consider strategies and policies that set the direction of such a process, serving to mitigate the limitations of the fog environment.

The process of data offloading in fog computing is a topic under discussion. Alelaiwi et al. [3] used machine learning algorithms that select data to be sent

to the cloud. The data offloading from an edge platform is only performed after deep learning techniques are executed in this approach. Pisan et al. [22] consider data that needs to be balanced among the fog nodes as a way to decide the offloading execution according to the available storage. However, these studies do not discuss the volume and nature of data, such as size and criticality, respectively.

Aazam et al. [1] classify the existing types of offloading, defining different situations where the process can occur and showing its variations. Whereas, He et al. [13] ponder that security and load balancing aspects are significant in the data offloading process. However, neither paper relates the data properties and infrastructure.

Still dealing with security, He et al. [13] propose a security model for fog data storage, highlighting the possible vulnerabilities that these data may be subject. Among other points found, we can highlight the proposal of Verma et al. [29], which proposes an adaptive processing balancing model for the fog data context. However, the authors do not specify the techniques used in the processes and their possible impacts.

To summarize, several works in the literature propose solutions for the data offloading process in fog computing, which gather proposals for different problems. However, many of them do not consider the properties of the data: security, management, nature, and volume of the data, nor the characteristics of the infrastructure, such as network, storage, processing, and power. Thus, unlike other works, this study seeks to classify the aspects, concepts, techniques, and particularities involved in the process so that it is possible to define more substantiated policies that allow the execution of data offloading in fog-based solutions.

3 Taxonomy of Aspects of the Data Offloading Process

Offloading is a technique that aims to assist devices with limited capabilities by migrating processing and/or data from these devices to other entities with larger computational power and storage. In general, offloading occurs when the network situation is favorable and when there are gains in the migration of data or processing. On the other hand, when no connection is available between the device and the external entity, the storage is performed locally.

There are two types of offloading: processing and data [10]. Data offloading is a topic that has been discussed in the fog computing paradigm [1]. This happens due to the resource constraints of the distributed entities that usually form the fog. This process is governed by rules that dictate how it should occur, commonly called policies. Thus, its realization is conditioned to policies that describe how the data is managed during the procedure. Therefore, based on the criteria defined by Aazam et al. [1], a taxonomy was proposed to assist in the process of eliciting essential aspects for the definition of rules and policies for data offloading.

3.1 Taxonomy Organization

First, the defined aspects were divided into two perspectives for organizational reasons. Thus, the classes and subclasses from the taxonomy were organized according to the following views: data and infrastructure.

The **data** view represents the basis of the data offloading process in fog computing because of the properties of the data. For example, some studies suggest using deep learning techniques to recognize such properties [3], while others consider data size an essential element for executing the process [22]. Therefore, this view brings together the characteristics directly associated with the data being generated, transferred, and processed in the context of fog computing. The classes defined to detail this category were: *security, management, nature, and volume.*

The **infrastructure** view was considered to represent the data offloading process because of the physical characteristics of the environment's infrastructure. In the study of Hao et al. [12], for example, the data offloading decision is based on the intensity of network resource utilization, while other studies only consider predetermined times as a parameter for the execution of the process. Thus, this category concerns the requirements of the environment where the data is located, that is, the properties to be considered in the nodes that form the fog. So, this category has only one class called *resources.*

3.2 Taxonomy Structure

Figure 1 presents a taxonomy that allows listing aspects to develop policies for data offloading in fog computing. The main goal of this taxonomy is to guide the development of policies for data offloading, providing the main criteria and aspects to be considered in the process. This way, industry and researchers could understand the importance of data offloading policy development. That said, the subsequent sections will detail the classes that form the proposed taxonomy.

3.3 Security

Many security problems in personal information can happen in distributed fog nodes. This happens due to incorrect approaches and the existence of malicious users [16]. However, the correct practice of the four pillars of information security (confidentiality, availability, integrity, and privacy) ensure the secure traffic of data generated in the fog environment, resulting in improving application security.

Aiming at data *confidentiality*, authentication protocols can be applied between the nodes in the fog to obtain the identity of the connected nodes [14]. The deployment of these protocols prevents unauthorized people from accessing the data traffic between these nodes. On the other hand, to ensure that the fog nodes or the cloud servers are always available, techniques that promote the *availability* of the data [14] must be applied. Thus, the use of techniques such as

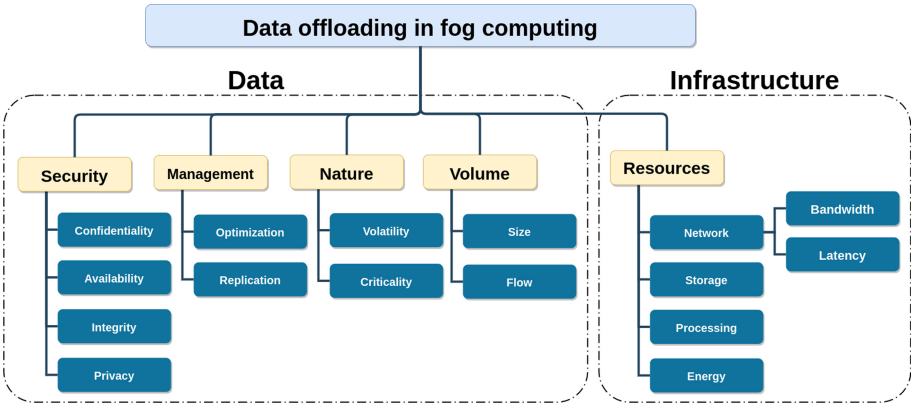

Fig. 1. Taxonomy for data offloading process in fog computing.

data replication and load balancing can help in this assurance. The work developed by Verma et al. [29], for example, uses the technique of data replication in conjunction with load balancing to decrease the environment's dependence on large cloud servers.

Data *integrity* ensures that any unauthorized attempts to modify transmitted or stored data are detected [14]. It is common to use the *blockchain* technique in addressing this vulnerability in fog computing [23,28]. Another property that helps in data preservation is *privacy*, which protects sensitive personal information [20]. Silva et al. [27] propose an approach based on *blockchain* to ensure the privacy of patient medical data in the context of fog computing. In the face of all these solutions, it is common to find approaches that use *blockchain* techniques, anonymization, and others.

3.4 Management

Data are commonly moved from one device to another in fog computing, so this flow requires appropriate data management [1]. The main motivation for considering management in the data offloading process is that the fog nodes are geographically distributed, making it somewhat challenging to identify the location of the data. Therefore, one of the first moments in the data management process is the data *optimization* task, followed by the data *replication* procedure. This set of techniques can distribute data among the nodes in the fog more efficiently since the requested data may not be available at nearby nodes.

The process of data size optimization is closely related to local storage and processing constraints, a feature quite common in fog computing [5]. According to Rahman et al. [24], two optimization techniques stand out in the fog layer: data filtering and data compression. Data filtering helps remove noise in the data acquisition phase, while the data compaction technique aims to reduce latency and the energy consumed during operations. Commonly, solutions use optimization techniques and data replication on their nodes, often aiming to reduce data

size and promote data availability on nearby nodes. However, one should also consider that data replication can generate large data redundancy, which can compromise the storage factors of the fog nodes. Likewise, optimization techniques can require more processing from these nodes.

3.5 Nature

Fog computing environments generally do not use proprietary hardware or communication technologies, such as non-volatile storage devices [30]. Usually, resources such as storage and networking are leveraged from devices near the network's edge, such as smartphones, smart TVs, or connected vehicles. This implies a challenge which is the availability of resources that are on fog nodes, as it may not be continuous due to its volatile nature [15]. Therefore, these aspects of infrastructure in fog environments can lead to data taking on volatile and non-volatile natures.

Several applications also use fog computing to meet their real-time operation needs. Therefore, large computing infrastructures are increasingly dependent on critical data, which is also stored in databases arranged in the cloud [9]. So, when manipulating/transferring data generated by cloud layer devices, one must consider their different degrees of criticality. Critical data, for example, could be found in a scenario of monitoring the vital health information of a patient. In contrast, some other data in the same solution could be less relevant and even discarded. In this way, data generated at the edge of the network can also be volatile and/or critical.

3.6 Volume

The volume discusses metrics involving data centralization in fog computing, among one the *size* and *flow* of data stand out. [18]. These two metrics are the most common in the literature and the most relevant when there is the intention to work with data analysis at the nodes of the fog or external entities. Thus, identifying the volume of data in systems is advantageous because it allows the data to be integrated into more sophisticated analysis processes, providing better decision-making. Furthermore, the data volume analysis of the fog nodes allows the data offloading to the cloud to be performed only in opportune cases.

Regarding data size, it is understood how much data needs to be processed and stored in the fog environment. In some works, data size is discussed and related to the computational requirements [25]. Thus, such a property is considered aiming to mitigate the storage limitations of the fog. Another property associated with the volume of information between the fog and the cloud is data flow, which refers to the intensity with which data is sent from the source to the fog nodes. Such flow can be event-driven [17] or real-time [11]. Real-time data streams transmit data to storage at the fog nodes as they are generated, causing overload and high storage demands. Therefore, understanding the size and flow of data in fog systems is decisive in the data offloading process.

3.7 Resources

Fog computing is a very heterogeneous paradigm and has hardware constraints from its devices. However, cloud computing supports this paradigm by having almost unlimited capabilities and resources compared to the resources of fog computing-based environments. In this way, data offloading can support fog to take advantage of cloud capabilities.

Issues involving the communication *network* between devices are also pertinent for data offloading in fog computing. The fog is very heterogeneous and can be formed by different types of communication, approaches such as that of Zhou et al. [31] must be developed that propose a heterogeneous and hierarchical network architecture for fog enabling mobile communication networks in fog to achieve higher capacities than conventional communication networks. Issues like this are also related to *bandwidth* and decreasing *latency*, which are useful aspects of any fog application.

The *storage* in the fog computing paradigm is a topic already discussed in some studies [4,26]. Despite offering many other features such as data management, eliminating unnecessary transactions, and minimizing latency times, fog computing has limited local data storage. In addition, such a paradigm also performs techniques involving *processing* [19], which, despite aiming to reduce the amount of data sent to the cloud, intensifies the infrastructure's processing constraint. Moreover, although the techniques that compress and process local data serve to mitigate the restrictions of the environment, it is worth remembering that in the fog, the devices that are in the network form the nodes and that most of them are powered by battery [7]. So, it is simple to say that the overload of these devices can worsen the energy autonomy of the fog infrastructure.

In summary, the evaluation of infrastructure resources is a requirement that helps when designing policies for data offloading because the availability of resources can decide whether the flow will follow normal or data offloading will be performed [8].

3.8 Taxonomy Discussion

The aspects listed in this section bring essential concepts involved in the data offloading process. It also highlights the main properties to be considered when developing rules, policies, and strategies for data offloading in the most diverse fog scenarios, considering the characteristic of the data and the infrastructure of the environments involved. Therefore, the study classifies and exemplifies some concepts about the data offloading process, as well as exemplifies and cites the main techniques used in such a process. Thus, this classification can serve as a basis for selecting techniques and aspects to define policies for data offloading since it provides many of the concepts needed for this definition. A summary of the main aspects and techniques used in data offloading process decisions is presented in Table 1. Finally, to show that the taxonomy is applicable, in the following section, a proof of concept is provided, in a typical cloud and fog computing scenario, that instantiates the classes specified by the taxonomy.

Table 1. Data offloading aspects and techniques summary

Aspects	Techniques
Confidentiality	- Authentication
Availability	- Replication; - Collaboration between fog nodes
Integrity	- *Blockchain*
Privacy	- *Blockchain*; - Anonymization policies
Optimization	- Data filtering; - Data Compression
Replication	- Data replication
Volatility	- Local processing
Criticality	- Data prioritization
Size	- Data distribution; - Data compression
Flow	- Event-driven; - Real-time
Network	- Network hierarchy architecture
Storage	- Data distribution; - Data compression; - Data filtering
Processing	- Load balancing
Energy	- Local processing

4 Proof of Concept

A healthcare scenario was instantiated to employ the taxonomy of classification of aspects for data offloading proposed in this work. It was analyzed under the aspects listed by the taxonomy, demonstrating its use and assisting in the process of specifying policies for data offloading for this example. Finally, this section performs an adaptation of OpenFog [21] and considers that the scenario is constituted by an environment based on fog and cloud paradigms.

4.1 Patient Monitoring

This scenario focuses on the analysis of cases where the health of patients needs to be continuously monitored. For example, people with diabetes, heart disease, and hypertension usually have more intense medical monitoring, and after surgical procedures, they need even more care. One of the intelligent monitoring techniques is Patient-Controlled Analgesia (PCA), a procedure to control post-operative pain in patients through self-administration of analgesics. An overview of this scenario can be seen in Fig. 2.

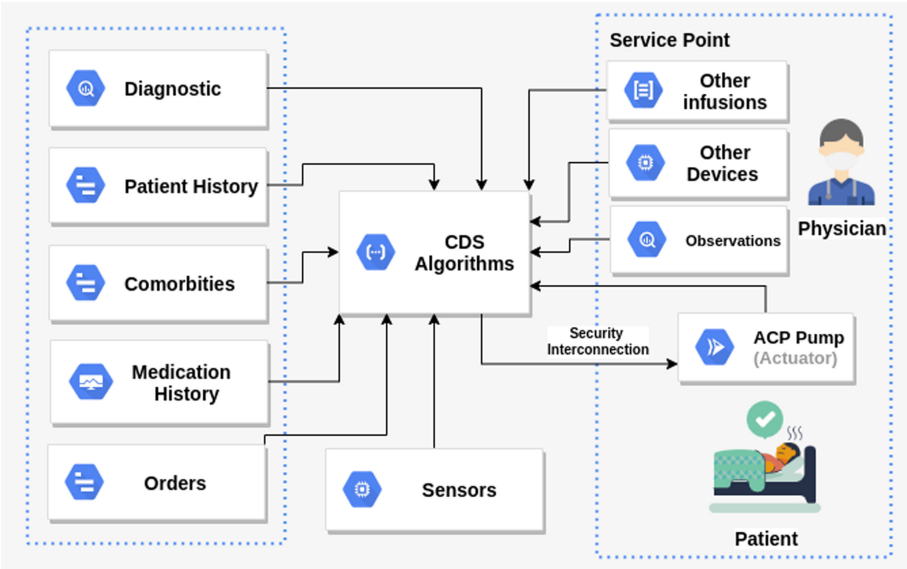

Fig. 2. Patient monitoring scenario overview.

In the fog computing environment of this scenario, data from the devices is integrated into and utilized by Clinical Decision Support (CDS) applications. This information is provided to healthcare professionals, wherever and whenever they need it. In addition, the systems have several alerting mechanisms, generating automatic responses if any anomaly is identified. The most common procedures are monitoring vital signs and stopping infusions until a healthcare professional can verify the patient's condition.

The architecture design of healthcare fog computing infrastructures must consider points that ensure the secure operation of the applications. The main ones are exemplified through the instance of the taxonomy proposed in this work presented in Fig. 3. The classes and aspects assigned for this scenario are described in the following.

4.2 Security

Healthcare fog computing environments must meet specifications that guard patient data, protecting it from unauthorized access. Aimed at this protection, the *confidentiality* of patient data can usually be provided through encryption and the use of data authentication approaches.

Another requirement of medical systems is to ensure that sensitive patient data is not exposed. For this assurance, the systems must allow the patient to control what information will be shared between the nodes and cloud, ensuring their *privacy* and preserving their personal and intimate data. Another guarantee that must implement in this type of application is the *integrity* of the data

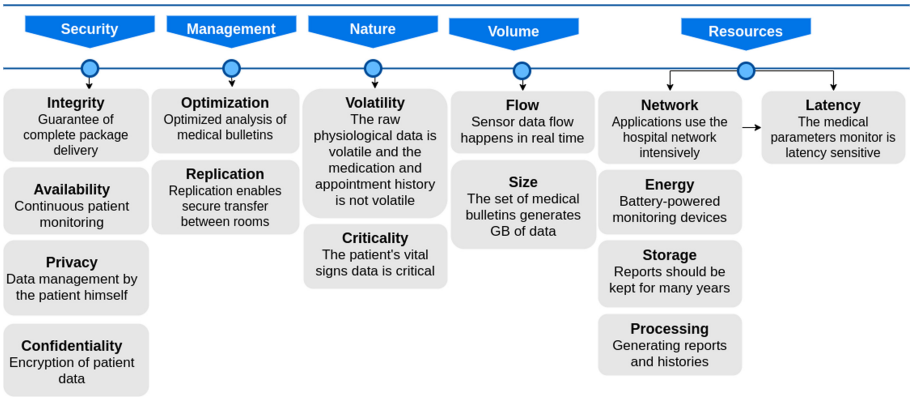

Fig. 3. Taxonomy applied to the patient monitoring scenario.

because fatal problems can happen if they are tampered with or lost, so the validation of data from offloading must be done.

Patient monitoring is a continuous action, so essential data must be available on the medical parameters monitor at all times. To ensure this *availability*, replication and load balancing techniques must be used to balance the work between the nodes and prevent the system from shutting down at critical times.

4.3 Management

In large hospitals, the data that is in fog may be geographically distributed, creating the need to use data management techniques between nodes. These environments must match data flow to patients even as the physical location and network changes during transfers between rooms. Therefore, one strategy that allows this information to be available regardless of location is *replication*, which will allow the information not to depend on the environment in which the patient is located. However, the data replication technique can harm the storage resources, so another aspect needs to be addressed, namely data optimization. Optimization will allow only essential data to be kept. So, only essential patient data will be kept in the fog environment after performing techniques such as data filtering and compression.

4.4 Nature

The nature of the data in this scenario can assume moments of *volatility*. Health care applications collect raw physiological data, which can be requested from the source and analyzed without the need for storage (*volatile*) or even synthesized in anamnesis procedures. However, data used in diagnostic parameters and patient health histories (*non-volatile*) must be kept in long-term storage infrastructures such as the cloud. This scenario also addresses the *critical* nature of the data, as

monitoring operates on patients' vital signs, i.e., data considered *critical*. When missing, this information can compromise the system's overall functioning and lead to the patient's death. Therefore, data offloading must occur adaptively. This adaptivity allows transactions to happen according to the criticality of the data. For example, in cardiac patient monitoring, the system must adapt the request priorities to heart-related parameters when detecting abnormalities. Therefore, adaptive data offloading can improve the sensitivity of these systems for critical data.

4.5 Volume

The intense *flow* of data generated by sensors arranged in the environment and on the patients happens in real-time. This fact is known to aggravate the storage constraints of the devices that form the nodes of the cloud. Therefore, the intensity with which the data will be transferred to the cloud must be well managed to avoid damage to these devices. In addition, in cases involving critical data in the medical environment, almost all the information collected from patient monitoring is accumulated, generating patient health reports. When accumulated in a hospital, these reports can have sizes that reach many gigabytes of data, therefore overloading fog nodes with storage constraints. So, these two aspects also impact how data offloading should proceed in these cases.

4.6 Resources

The connection between devices, fog nodes, and proprietary data is responsible for overloading the *network* of hospitals. The use of the hospital network can be heavy due to a large number of patients, rooms, and emergency centers. Therefore, data offloading should be restricted to essential situations, preventing the network transmission capacity from being affected by the execution of non-essential tasks.

Another aspect to consider in this scenario is its *storage* limitation. Patient reports and histories must be kept for many years, further limiting the storage environment of the fog nodes. However, this fact enables the data offloading process, which can transfer such data to external entities and free up local space. Another resource constraint in the fog is that of *processing*. However, offloading allows specific processing to be performed in the cloud. Finally, given that currently, many wireless devices used in patient monitoring are battery-powered. Knowing that recharging opportunity is not always favorable, data offloading also allows the *energy* consumption on these devices to be reduced.

In monitoring scenarios like this, *latency* requirements are critical. The delay the network generates to exchange data elements can mean the death of a patient. Integration with emergency room data, for example, must provide essential information parameters for patient care, i.e., *latency* communication between devices can cost severe consequences to patient health. Therefore, the data offloading process must always choose the best node to store and retrieve the data.

4.7 Discussion

In fog scenarios like these, the elicitation of aspects pertinent to the data offloading process is very important. Through these, it is possible to make faster decisions at critical moments and efficiently take advantage of all the resources of the fog and cloud environments. To this end, the instantiation of the taxonomy allows the useful aspects of the scenario to be selected and the choice of techniques to be employed in the solution to be made with criteria and not in an empirical way. Therefore, this selection aims to ensure that the data offloading process of the solution is improved to reduce future problems. However, the financial and effort costs of this activity must be considered. The main cost identified in the applicability of this approach is related to the time and effort devoted to the selection and implementation of techniques, i.e., the most significant impact is in the requirements elicitation phase.

5 Conclusion

This paper proposes an approach to classifying aspects of the fog data offloading process through a taxonomy. The taxonomy was instantiated in a healthcare scenario and allowed prior surveys of aspects to provide the necessary background for future policy implementations. Thus, with this study, we believe that the creation of a taxonomy for understanding the aspects of data offloading provides better conditions for choosing the most appropriate techniques for the offloading process, allowing optimized data management, faster decision making, efficient use of resources, and above all, the construction of safer solutions for the fog and cloud paradigms. The work also showed that the taxonomy is instantiable, evidencing that it is helpful in fog computing scenarios. As future work, it is intended to build a guidance mechanism for policy implementation, including instantiation examples in other scenarios.

Acknowledgments. This study was financed in part by the Coordenação de Aperfeiçamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

References

1. Aazam, M., Zeadally, S., Harras, K.A.: Offloading in fog computing for IoT: review, enabling technologies, and research opportunities. Future Gener. Comput. Syst. **87**, 278–289 (2018)
2. Ahn, D.J., Jeong, J., Lee, S.: A novel cloud-fog computing network architecture for big-data applications in smart factory environments. In: Gervasi, O., et al. (eds.) ICCSA 2018. LNCS, vol. 10964, pp. 520–530. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-95174-4_41
3. Alelaiwi, A.: An efficient method of computation offloading in an edge cloud platform. J. Parallel Distrib. Comput. **127**, 58–64 (2019)

4. Azimi, I., Anzanpour, A., Rahmani, A.M., Liljeberg, P., Salakoski, T.: Medical warning system based on internet of things using fog computing. In: 2016 International Workshop on Big Data and Information Security (IWBIS), pp. 19–24. IEEE (2016)
5. Bonomi, F., Milito, R., Zhu, J., Addepalli, S.: Fog computing and its role in the internet of things. In: Proceedings of the First Edition of the MCC Workshop on Mobile Cloud Computing, pp. 13–16. ACM (2012)
6. Cisco, A.: Fog computing and the internet of things: extend the cloud to where the things are. [Electronic resource] (2015). https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-overview.pdf. Accessed 10 Mar 2019
7. Dubey, H., Yang, J., Constant, N., Amiri, A.M., Yang, Q., Makodiya, K.: Fog data: enhancing telehealth big data through fog computing. In: 2015 Proceedings of the ASE Bigdata & Social Informatics, pp. 1–6. Proceedings of the ASE bigdata (2015)
8. Enzai, N.I.M., Tang, M.: A taxonomy of computation offloading in mobile cloud computing. In: 2014 2nd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering, pp. 19–28. IEEE (2014)
9. Esteves, S., Silva, J., Veiga, L.: Quality-of-service for consistency of data geo-replication in cloud computing. In: Kaklamanis, C., Papatheodorou, T., Spirakis, P.G. (eds.) Euro-Par 2012. LNCS, vol. 7484, pp. 285–297. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32820-6_29
10. Fernando, N., Loke, S.W., Rahayu, W.: Mobile cloud computing: a survey. Future Gener. Comput. Syst. **29**(1), 84–106 (2013)
11. Giang, N.K., Blackstock, M., Lea, R., Leung, V.C.: Developing IoT applications in the fog: a distributed dataflow approach. In: 2015 5th International Conference on the Internet of Things (IOT), pp. 155–162. IEEE (2015)
12. Hao, Z., Novak, E., Yi, S., Li, Q.: Challenges and software architecture for fog computing. IEEE Internet Comput. **21**(2), 44–53 (2017)
13. He, S., Cheng, B., Wang, H., Xiao, X., Cao, Y., Chen, J.: Data security storage model for fog computing in large-scale IoT application. In: IEEE INFOCOM 2018-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 39–44. IEEE (2018)
14. Huang, C., Lu, R., Choo, K.K.R.: Vehicular fog computing: architecture, use case, and security and forensic challenges. IEEE Commun. Mag. **55**(11), 105–111 (2017)
15. Kuo, P.H., et al.: An integrated edge and fog system for future communication networks. In: 2018 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), pp. 338–343. IEEE (2018)
16. Lee, K., Kim, D., Ha, D., Rajput, U., Oh, H.: On security and privacy issues of fog computing supported internet of things environment. In: 2015 6th International Conference on the Network of the Future (NOF), pp. 1–3. IEEE (2015)
17. Lee, W., Nam, K., Roh, H.G., Kim, S.H.: A gateway based fog computing architecture for wireless sensors and actuator networks. In: 2016 18th International Conference on Advanced Communication Technology (ICACT), pp. 210–213. IEEE (2016)
18. Mahmud, R., Kotagiri, R., Buyya, R.: Fog computing: a taxonomy, survey and future directions. In: Di Martino, B., Li, K.-C., Yang, L.T., Esposito, A. (eds.) Internet of Everything. IT, pp. 103–130. Springer, Singapore (2018). https://doi.org/10.1007/978-981-10-5861-5_5
19. Maksimović, M.: Improving computing issues in internet of things driven e-health systems. In: Proceedings of the International Conference for Young Researchers in Informatics, Mathematics and Engineering 2017, pp. 14–17 (2017)

20. Moysiadis, V., Sarigiannidis, P., Moscholios, I.: Towards distributed data management in fog computing. Wirel. Commun. Mob. Comput. **2018**, 1–14 (2018). https://doi.org/10.1155/2018/7597686
21. OpenFog: Openfog use cases (2020). http://openfogconsortium.org/
22. Pisani, F., do Rosario, V.M., Borin, E.: Fog vs. cloud computing: should i stay or should i go? Future Internet **11**(2), 34 (2019)
23. Podsevalov, I., Iakushkin, O., Kurbangaliev, R., Korkhov, V.: Blockchain as a platform for fog computing. In: Misra, S., et al. (eds.) ICCSA 2019. LNCS, vol. 11620, pp. 596–605. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-24296-1_48
24. Rahmani, A.M., et al.: Exploiting smart e-health gateways at the edge of healthcare internet-of-things: a fog computing approach. Future Gener. Comput. Syst. **78**, 641–658 (2018)
25. Shi, H., Chen, N., Deters, R.: Combining mobile and fog computing: using CoAP to link mobile device clouds with fog computing. In: 2015 IEEE International Conference on Data Science and Data Intensive Systems, pp. 564–571. IEEE (2015)
26. Silva, C.A., de Aquino Júnior, G.S.: Fog computing in healthcare: a review. In: 2018 IEEE Symposium on Computers and Communications (ISCC), pp. 1126–1131. IEEE (2018)
27. Silva, C.A., de Aquino Júnior, G.S., Melo, S.R.M.: A blockchain-based approach for privacy control of patient's medical records in the fog layer. In: Anais Principais do XXV Simpósio Brasileiro de Multimídia e Web, pp. 133–136. SBC, Porto Alegre (2019). https://sol.sbc.org.br/index.php/webmedia/article/view/8011
28. Tuli, S., Mahmud, R., Tuli, S., Buyya, R.: FogBus: a blockchain-based lightweight framework for edge and fog computing. J. Syst. Softw. **154**, 22–36 (2019)
29. Verma, S., Yadav, A.K., Motwani, D., Raw, R., Singh, H.K.: An efficient data replication and load balancing technique for fog computing environment. In: 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), pp. 2888–2895. IEEE (2016)
30. Yousefpour, A., et al.: All one needs to know about fog computing and related edge computing paradigms: a complete survey. J. Syst. Archit. (2019)
31. Zhou, Y., Tian, L., Liu, L., Qi, Y.: Fog computing enabled future mobile communication networks: a convergence of communication and computing. IEEE Commun. Mag. **57**(5), 20–27 (2019)