Context-Oriented User-Centric Search System for the IoT Based on Fuzzy Clustering

Santosh Pattar[1], Sandhya CR[1(✉)], Darshil Vala[1], Dharamendra Chouhan[1], Rajkumar Buyya[2], Venugopal KR[3], S. S. Iyengar[4], and L. M. Patnaik[5]

[1] IoT Lab, University Visvesvaraya College of Engineering, Bengaluru, India
sandhya.cr7592@gmail.com
[2] University of Melbourne, Melbourne, Australia
[3] Bangalore University, Bengaluru, India
[4] Florida International University, Miami, USA
[5] National Institute of Advanced Studies, Bengaluru, India

Abstract. The Internet of Things (IoT) paradigm envisions to support the creation of several applications that aids in the betterment of the society from various sectors such as environment, finance, industry *etc.* These applications are to be user-centric for their larger acceptance by the society. With the increase in the number of sensors that should are getting connected to the IoT infrastructure, there is an augmented increase in the amount of data generated by these sensors. Therefore it becomes a fundamental requirement to search for the sensors that produce the most applicable data required by the application. In this regard, context parameters of the sensors and the application users can be utilized to effectively filter out sensors from a large group. This paper proposes a sensor search scheme based on semantic-weights and fuzzy clustering. We have modified the traditional fuzzy c-means clustering algorithm by incorporating the semantic and context attributes of the sensors to obtain fuzzy clusters. During the query resolution phase, the query is directed to the most appropriate cluster. These clusters are formed through the use of linguistic variables rather than quantitative attributes and thus aid in effective user-centric search results. Experimental results indicate that the proposed scheme achieves better performance when compared to the existing techniques.

Keywords: Context-aware · Fuzzy clustering · Internet of Things · Resource discovery · Semantic similarity · Sensor search

1 Introduction

The infrastructure of Internet of Things (IoT) is undergoing an expeditious development, wherein several physical components (such as sensors, actuators, *etc.,*) inclusive of the user are being linked to the digital world. These components

© Springer Nature Singapore Pte Ltd. 2020
A. Saha et al. (Eds.): ICCISIoT 2019, CCIS 1192, pp. 343–356, 2020.
https://doi.org/10.1007/978-981-15-3666-3_28

collaborate to capture, analyze, communicate and exchange data among themselves, allowing the development of intelligent applications that are beneficial to human life [18]. As the number of devices being connected to the IoT environment is increasing, the volume of data that is being continuously generated and exchanged by them is also increasing. This makes it difficult for the applications to fetch the relevant IoT resource in the search space and thereby extract the required data to provide solutions. Therefore, there is a need for an efficient search system in the IoT that aids in extracting appropriate IoT resources from the huge amount of devices connected in the IoT environment [12].

In regard to an IoT scenario, the term context refers to information that describes a situation, circumstance or characteristics of an entity. Contemplating this information in the IoT search techniques, enables application users to specify their requirement in terms of sensor attributes predefined by the system. These attributes include location, accuracy, type, battery life, *etc* [16]. Although there are search techniques that consider context-based parameters of both the user and IoT devices to find the relevant IoT resources, they are prone to two main challenges namely, *i.e,* (i) performance of the search system decreases with the increase in the number of connected devices in the IoT network (scalability), and (ii) time required to resolve a given query increases with increase in the size of the search space (communication overhead).

In this paper, we focus on designing a search system for the IoT based on fuzzy logic that restructures the search space. It takes into consideration the context parameters of both the users and IoT devices. Fuzzy logic deals with linguistic uncertainty which is very sensitive and plays a vital role in processing context information. Therefore, the user need not specify the requirements through a precise quantitative value, it can rather be imprecise and also expressed using natural language [22].

The rest of our paper is structured as follows. In Sect. 2, we present the literature survey on the topics, context-based search for the IoT and fuzzy-based IoT search systems. Section 3 defines the problem statement and elaborates the importance of fuzzy logic in the IoT search systems. In Sect. 4, we describe in detail, our proposed search system. Section 5 discusses on the implementation and performance evaluation of our proposed search model and we conclude our work in Sect. 6.

2 Literature Survey

In this section, we present an overview of the recent works that address the sensor search challenge in the IoT. In the first subsection, we discuss the previous studies related to the context-based search systems for IoT and in the second subsection, we illustrate the recent works that apply fuzzy theory to the IoT search applications. We put forth how our work differs from the existing works as our contributions in this field.

2.1 Context-Based Search Systems for the IoT

Context-aware information is a core component of ubiquitous computing. A bird's eye view of the recent works on context-oriented search systems for locating the IoT resources are as follows. Search functionality becomes critical as and when more sensors are deployed into the IoT environment. To address this problem, Perera *et al.* [17] presented a model called CASSARAM, which could effectively select a small group of relevant sensors from a large group of sensors bearing similar functionality and capabilities. On receiving the input query from the user, a score called Comparative-Priority based Weighted Index (CPWI) is calculated by comparing the sensor's context property and user's priority values for each sensor in the search space. Finally, the sensors are indexed using the generated CPWI score and returned to the user. Even though this framework made significant progress, the process is extremely complex as it is inefficient when applied to a large number of sensors. To overcome this challenge, machine learning techniques can be utilized. Ebrahimi *et al.* [9] proposed an adaptive meta-heuristic cluster-based approach for sensor search that increases the scalability and efficiency of context-aware search algorithms. An ant-based clustering technique is used to create clusters containing sensors that are identical to each other based on their context properties. On receiving the query from the user, it is forwarded only to a suitable cluster that matches the query and thus returns the result to the user. This process reduces the size of search space thereby increasing the efficiency of the search technique as mentioned earlier. This method also proved to have lower execution time in comparison with CARASSAM.

2.2 Fuzzy-Based Systems for the IoT

Few recent works on fuzzy-based IoT search systems are discussed in this subsection. Based on the data generated by the sensors, Truong *et al.* [19] put forth a content-based search approach built through fuzzy theory. Although this work achieves low communication overhead, it does not provide user-centric results. Similarly, Truong *et al.* [20] in one of their other works have proposed an algorithm for computing similarity index of the sensors deployed in Web of Things (WoT) using fuzzy rules. This approach is highly accurate in computing similarity parameters between sensors of the same region but failed to compute similarity among sensors between two different regions. Jun *et al.* [21] presented a fuzzy logic-oriented information retrieval mechanism in IoT integrated Intelligent Transport System (ITS) application. However, this mechanism handles only simple fuzzy relations that are to be defined beforehand in the ontology. In Table 1, we compare the different search techniques proposed for the IoT systems.

From the above works, it can be inferred that the context-based search techniques developed by several authors could not handle user requirements specified through natural language or linguistic variables. Also, the existing fuzzy-based search algorithms lack the applicability of context parameter requirement of the user and the sensor attributes. In this regards, we have devised a search system based on context-oriented weighted fuzzy c-means clustering algorithm to reduce the sensor search space in the IoT and thereby decrease the communication overhead for efficient query resolution. Our approach is also capable of handling the queries given by the user through linguistic variables.

Table 1. Comparison of Different Existing Search Techniques in the IoT

Authors	Approach	Advantages	Drawbacks
Context-based search techniques			
Perera *et al.* [17]	CARRASAM framework	Distributed search and rank model based on context information	Inefficient in the aggregation and management of context-properties for a large number of sensors
Ebrahimi *et al.* [9]	A meta-heuristic Ant-Clustering algorithm for search space clustering	User queries are forwarded to a particular matching cluster only	Offline cluster formation process is time consuming
Fuzzy-based Search Techniques			
Truong *et al.* [19]	Content-based fuzzy logic technique	Achieves low communication overhead	Fails to provide user-centric information since their preferences are not considered
Truong *et al.* [20]	Computation of similarity index between the sensors using fuzzy logic	Highly accurate query resolution	Could not compute similarity index of sensors between two different regions
Jun *et al.* [21]	Information retrieval system based on fuzzy logic for ITS	User's preference was taken into consideration	Applicable only to pre-defined simple fuzzy relations

3 Problem Definition and Background

In this section, we outline the sensor search problem considering user queries and sensor properties. In the first subsection, we formulate the problem mathematically along with an example. We discuss the importance of fuzzy logic applicability for IoT search systems in the second subsection.

3.1 Problem Definition

An IoT application typically comprises of a large number of sensors. Depending on the service that the system needs to provide, the application has to choose a subset of sensors from the available large set. Due to this, there arises a need to search and select sensors in the most efficient and effective possible way. The problem at hand can be illustrated as follows. Consider a set of sensors $S = \{s_1, s_2, s_3, \ldots, s_n\}$, with dynamic size, *i.e.,* where n is number of sensors present in the search space (collection of sensors over which search is to be performed). Let, $A = \{a_1, a_2, a_3, \ldots, a_m\}$ be the set of properties possessed by the sensors (such as sensor type, data output, geographical location, life-time, accuracy, error rate, bandwidth, reliability, *etc*). These properties are used as parameters in the search process to select an appropriate set of resultant sensors. They are classified into two types namely, content-properties and context-properties. The former set of properties consists of the parameters of the data emitted by the sensors (represented by the set O), while the latter set describes the situation based properties (context) of the sensors (represented by the set C). It is necessary for us to find out an optimum subset $P \subseteq S$ such that it satisfies the conditions specified by the user. Some of the attributes of set A change over time. Thus, a sensor $s \in S$ can be represented as,

$$s \colon T \rightarrow F \tag{1}$$

where, T represents the set of time intervals within which sensor reading are extracted and F is the set of attribute instances of a sensor, where $F \subseteq A$ *i.e.,* the sensor readings can either belong to output-data set (O) or context-based attribute set (C). An IoT application has to capture user requirements in the form of a query in order to perform search and select operation on the sensor set S. We use query Q to represent the user-requirements, that in-turn specifies the sensor properties $x \in A$ required by the user along with the relationships among them. This query is used to discover the "optimal subset" of p_n sensors that match the user-requirements. Thus the query Q, can be formulated by the function as follows,

$$Q \colon T \times X \times p_n \rightarrow F \tag{2}$$

where, the query poses a request to locate p_n sensors from S, such that they have $x \in A$ properties at time-stamp $t \in T$. $F \subseteq S$ defines the subset of "optimal sensors" obtained after the query has been executed.

Example. Let us consider an IoT application designed for an ITS system with n sensors. The objective of this application is to offer search service to the user and help him/her in locating appropriate sensors for their use. Thus, S represents the set of sensors deployed in the ITS environment. Let the attribute set defined for the sensor set S be $A = \{reading - data, latitude, longitude, battery - life, reliability, time - stamp\}$. For a given set of four temperature sensors, let us assume that set $O = \{\{(30\text{deg}, 1.00\,\text{pm}), (36\text{deg}, 1.02\,\text{pm})\}, \{(40\text{deg}, 1.05\,\text{pm})\}, \{(42\text{deg}, 1.10\,\text{pm})\}\}$ consists of temperature measured by the sensors at a particular time interval. Similarly, C contains such readings for two context-properties, battery-life and reliability. A user query on such a search space can take form like, To retrieve sensors that measure temperature (sensor type context-property) between the time interval 1.00 pm to 1.15 pm (content property). Thus, a query can have constraints on both the set of content as well as context-properties.

3.2 Fuzzy Logic and Its Importance in IoT

Everything present in the world we live in is a matter of measure. We encounter many problems in our real life that are challenging to understand, measure or solve using the concept of crisp logic (binary: 2 values only). This complexity arises due to the uncertainty associated with the problems, making it difficult to arrive at an accurate solution. These challenges pave the way for fuzzy logic-based approaches. Fuzzy logic is an effective technique that handles the ambiguity present in complex problems, thereby providing the best possible solution. Unlike binary logic that accepts only either of the two values 0 or 1 (true or false), fuzzy logic is multi-valued. It deals with approximation rather than precision. Fuzzy logic also provides a medium to represent linguistic variables (in natural language) that in-turn express human's reasoning capabilities [8].

Let us take an example from an IoT search requirement scenario to better understand the importance of fuzzy logic and how its association with context parameters uplifts the competence of the search system. Two users Ravi and Ranjan intend to acquire road traffic congestion data captured by the sensors in a certain city, say Bengaluru, for different purposes. Ravi prefers to make use of data collected from highly accurate sensors (say, 95% of accuracy) while Ranjan is interested in using the information aggregated by the moderately reliable sensors (say, 70% of reliability). Here, the terms 'accurate/accuracy', 'reliable/reliability' are referred to as "linguistic descriptor" that captures the user requirements.

The existing search systems designed for the IoT require queries to be in a crisp format and thus fails to provide a user-centric query resolution mechanism. Fuzzy logic can be used to handle this challenge. In traditional crisp logic methodology, the linguistic descriptor is expressed either as 'accurate' (value: 1) or 'not accurate' (value: 0), while in fuzzy systems, the linguistic variable is expressed as 'less-accurate', 'accurate', 'more-accurate' and so on, where each variable is associated with a truth value between 0 to 1 operating on the basis of membership functions. If a sensor has an accuracy value of 0.9, then 0.9 is called as the "degree of membership" of the sensor in a fuzzy set that represents accuracy context-property.

In this work, we design a fuzzy logic-based search mechanism for the IoT system. Initially, properties of the sensors are used to create clusters that hold similar sensors (through our proposed semantic weight-based fuzzy c-means search-space clustering algorithm). Then, a user query that expresses search constraints in linguistic terms is mapped to one of the appropriate cluster (through the selection of cluster head). Finally, a subset of sensors that matches the given user requirements are selected and returned. This approach offers user-centric results by considering the fuzzy representation of both sensor properties and user constraints. Also, the overall communication overhead for query resolution is effectively reduced due to the use of clustering approach.

4 The Proposed Sensor Search Scheme

In this section, we describe in detail the execution flow and construction of the proposed sensor search model delineating the techniques used for query processing, search space structuring and query resolution.

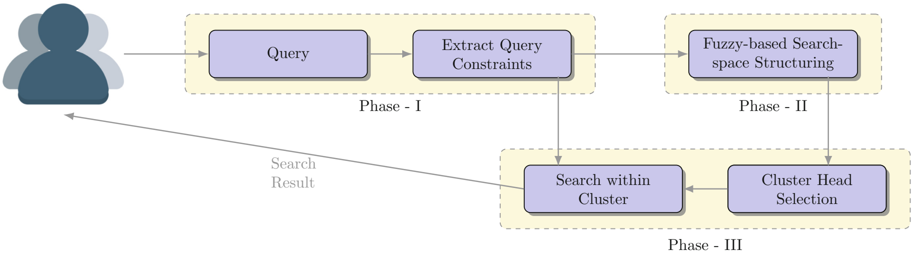

Fig. 1. Overall workflow of the proposed sensor search scheme

4.1 Execution Flow of the Search System

Our proposed sensor search scheme executes in three phases as depicted in Fig. 1. The processes carried out in each phase is explained in the following.

- *Phase - I: Query Processing.* Initially, user submits a query specifying the search constraints. A graphical user interface captures this query, where two kinds of user requirements are to be input *i.e.*, (i) indefinite - these are non-negotiable constraints, and (ii) definite - these requirements are negotiable. Further, these constraints are expressed in the form of values for linguistic variables that are predefined by the search system.
- *Phase - II: Search Space Structuring.* This is a offline phase where the data from the sensors are gathered and used for clustering. To group similar sensors based on their context-data and functionality, we use the semantic weighted fuzzy c-means clustering algorithm which is explained in the subsequent subsection.
- *Phase - III: Query Resolution.* In this phase search results containing appropriate sensors that match the given query constraints are constructed in two subsequent sub-phases. First, a relevant cluster (represented by the cluster head) is chosen and then the query is forwarded to it. Here, the cluster head is selected by calculating the semantic similarity score (discussed in Sect. 4.2) between the query and cluster head. In the second sub-phase, we employ the multi-dimensional Euclidean distance to select the appropriate sensors in the chosen cluster.

To abet the search technique, it is essential to annotate the sensors with semantic knowledge using ontology [15]. Hence, we have used Semantic Sensor Network Ontology (SSN) to interpret the attributes of the sensors and establish a link between them [11,14]. It also enables us to achieve interoperability among heterogeneous devices and applications in the IoT [7]. The proposed search scheme allows the user to query using imprecise values that are modeled as linguistic variables. Through the proposed fuzzy clustering mechanism these linguistic terms aid to obtain user-centric results. Also, the communication overhead of the proposed search system is drastically reduced by incorporating the search space structuring and cluster selection process, thus resulting in an efficient search performance.

4.2 Fuzzy-Based Search Space Cluster Formation

We have employed a semantic weight-based fuzzy c-means algorithm to form clusters, wherein each cluster consists of sensors with similar functionality and properties. Fuzzy C-Means (FCM) approach was first put forth by Bezdek *et al.* [6]. The drawback of this standard algorithm is that it treats every sensor in the search space equally *i.e.*, every sensor is given the same importance and thus increases the processing time. Hence, a Weight-based Fuzzy C-Means (WFCM) clustering approach is introduced by Hung *et al.* [10]. In this approach, weight is calculated for each sensor in the search space where the weight defines the sensor's importance.

The main objective of the proposed clustering technique is to create a membership matrix for the given search space (*i.e.*, S) by considering the attributes of the sensors (*i.e.*, A). As it is impossible for a user to precisely define the query constraints quantitatively which leads to imperfect similarity search. Being motivated by the semantic techniques like ontologies that perfectly capture the descriptive features of the sensor data, we utilize them to define a similarity score [13]. Given two sensors, s_i and s_j, the similarity between them is represented by $sim(s_i, s_j)$. We extract the ontological concepts for these sensors (p denotes the number of properties that are similar between the sensors) and calculate the score using the below equation.

$$sim(s_i, s_j) = \begin{cases} \sum_{k=1}^{p} \frac{C_{sim}(s_i,s_j)}{p}, & p > 0 \\ 0, & p = 0 \end{cases} \tag{3}$$

Here, C_{sim} is the semantic similarity between two given ontological concepts (*i.e.*, linguistic variables) and is calculated using the Eq. 4.

$$C_{sim}(c_i, c_j) = \begin{cases} 1, & \text{if } linguistic\ variables\ have\ same\ terms \\ 0, & \text{if } linguistic\ variables\ have\ different\ terms \\ \frac{\zeta}{|height(c_i) - height(c_j)|}, & \text{otherwise} \end{cases} \tag{4}$$

where ζ is the balancing factor that controls the contribution of the number of levels between the given concepts to the similarity score.

We modified the technique presented by Hung *et al.* [10] in order to make it suitable for sensor search problem. Algorithm 1 outlines the steps of our proposed clustering scheme. Our objective is to cluster the sensor set S into z clusters. The proposed approach is an iterative algorithm where initially random values between 0 and 1 are assigned to the membership matrix U (Step-1 in Algorithm 1), and then cluster heads, denoted by set H, are calculated. Here, the semantic similarity score is used as a weight to calculate H. Generally, the semantic score of sensors that possess similar properties is high and thus it is multiplied with

Algorithm 1. Semantic Weight-based Fuzzy C-means Search-space Clustering

```
Input  :  Sensor Search Space, $S = \{s_1, s_2, \ldots s_n\}$;
          Number of Clusters, $z$;
Output:   Cluster Heads, $H$;
$k \leftarrow 0$
repeat
    $k \leftarrow$ k + 1
    # Step-1: Setup the membership matrix.
    for $i \leftarrow 1$ to $z$ do
        for $j \leftarrow 1$ to $n$ do
            $u_{ij} \leftarrow$ random value between 0 and 1.
        end for
    end for
    # Step-2: Calculate Cluster Heads.
    for $i \leftarrow 1$ to $z$ do
        $h_i \leftarrow$ calculate cluster head using Equation 5.
        $H \leftarrow H \cup h_i$
    end for
    # Step-3: Update the Membership Values.
    for $i \leftarrow 1$ to $z$ do
        for $j \leftarrow 1$ to $n$ do
            $u_{ij} \leftarrow$ update membership value using Equation 6.
            $U \leftarrow U \cup u_{ij}$
        end for
    end for
until $\|U^k - U^{k-1}\| < \beta$
return H
```

the membership value to obtain the updated weighted cluster heads. Equation 5 is used for this purpose, where h_i is the cluster head for i^{th} cluster, m is the fuzzification parameter, and s_j is a particular sensor among n sensors.

$$h_i = \sum_{k=1,k\neq j}^{n} \frac{sim(j,k) * u_{ij}^{m}}{\sum_{t=1,k\neq j}^{n} sim(j,t) * u_{tj}^{m}} s_j, \qquad j = 1, 2, \ldots n. \tag{5}$$

In the third step, the membership values are updated using the below equation.

$$u_{ij} = \frac{\|s_j - h_j\|^{-1/(m-1)}}{\sum_{i=1}^{z} \|s_j - h_i\|^{-1/(m-1)}} \tag{6}$$

The above steps are repeated until the difference between the membership matrix of current and previous iterations are below a pre-defined tolerance value (i.e., β).

5 Experiments and Results

In this section, we describe the methods of experimentation, assumptions taken into consideration and datasets used to evaluate our proposed technique.

5.1 Implementation

We have implemented our proposed prototype search technique in Java. The SSN ontology required for sensor data modeling is handled by Apache Jena Framwork (Version 3.12.0) [3] and SPARQL querying language [5]. We have also made use of libraries like Commons Math from Apache [2] to calculate semantic weights and euclidean distance. For our experimental purposes, we have used environmental datasets MesoWest [4] and Air Quality Sensor Dataset [1] that capture the air quality measurement reading recorded from sensors. As at present there does not exists public dataset that houses context-properties of the sensors, we have randomly generated ten context-properties for each of the sensors in the above datasets. These properties include location, accuracy, type, battery life, life-time, error rate, bandwidth, reliability, energy, and availability. To evaluate and measure the performance of the proposed search system, we used a computer with Intel(R) Core i5-4200M 2.50 GHz processor with 8 GB RAM. The dataset constructed consists of a total 1,00,000 readings.

After, conducting several empirical tests we have set the optimal values of the parameters as $\zeta = 0.7$, $z = 10$, $m = 1.75$ and $\beta = 0.5$. In the following, we discuss them: (i) The balancing factor (ζ) in semantic score calculation controls the contribution of sensor attributes to the cluster formation, thus it decides the quality of the clusters. In this work as we used ten context-properties which mostly have three linguistic terms, the ontology thus has a tree with a maximum depth of 10 and at most 30 nodes (concepts). Here, a factor of 0.7 covers all the context-properties with these values of the ontology's depth and the number of nodes. (ii) As the number of context-properties in the dataset is limited to 10, the number of clusters (z) at this value achieves better search results. (iii) The fuzzification parameter (m) controls the magnitude of membership value. A higher value will magnify the addition of context-property, while a lower value decreases the role of the context-property to clustering process. It is observed that a 1.75 value balances the membership matrix with all the context-properties. (iv) The tolerance value (β) controls the number of iterations required for cluster formation. We noted that at 0.5 value of β, the difference in membership updation is no longer decreasing and thus effectively stops the clustering process.

5.2 Performance Evaluation and Analysis

In this subsection, we compare the proposed fuzzy based search technique with one of the recent search technique in the area of sensor search which is based on ant-colony optimization (AntClust). Among many works, this is the only technique that uses a meta-heuristic approach for clustering of sensors along with considering many sensor context-properties for implementation [9]. We have constructed a total of five queries by combining different number of sensor-properties along with several combinations of their linguistic terms and report the mean of 10 executions. For comparison we have considered some common parameters that are the characteristic features of the search system which are detailed as follows:

Processing Time. It is the time taken by the search system to resolve a given query (*i.e.*, search space structuring, query processing, and resolution). As seen in Fig. 2, the proposed technique takes comparatively less time than AntClust. This is attributed to the fact that AntClust involves a two-phase cluster formation process where at first level sensors are grouped based on their type and further at second level they are clustered based on their attributes. However, in our approach, only one phase is involved where semantic-based clustering technique groups the sensors with similar descriptive (linguistic) properties, we thus eliminate the two-way cluster formation and search. Due to this, the overall communication overhead in the search system is drastically reduced.

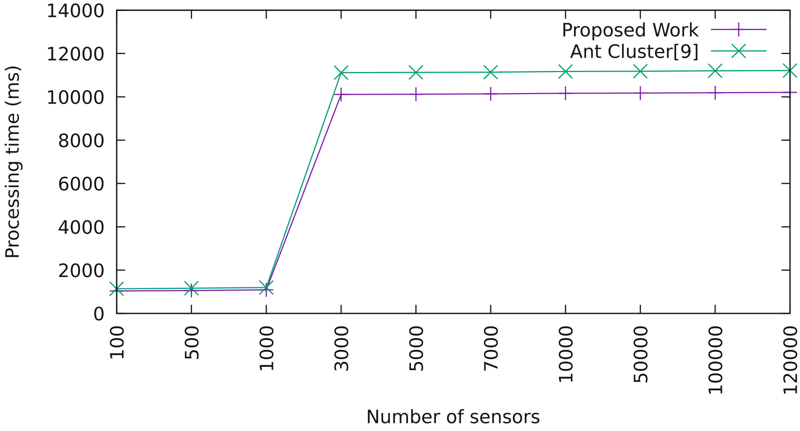

Fig. 2. Performance comparison of proposed work with AntClust.

Accuracy of Cluster Formation. In this experiment, the quality of clusters formed by the search technique is measured. The obtained results indicate that our approach creates high-quality clusters than AntClust technique (Fig. 3). As the search space involves multi-modal and multi-dimensional attributes of the sensors, it is a challenging task to define a perfect boundary that separates the sensors that belong to more than one cluster. AntClust technique makes use of most similar sensor (MSS), a memory-based score to assign a sensor to a particular cluster. With a high feature search space (*i.e.*, a large number of context-properties) this approach fails to identify the belongingness of the sensor to a particular group. However, owing to the semantic similarity score our clustering approach can easily identify the sensor's affinity to a particular cluster as all the matching context-properties of that cluster are considered.

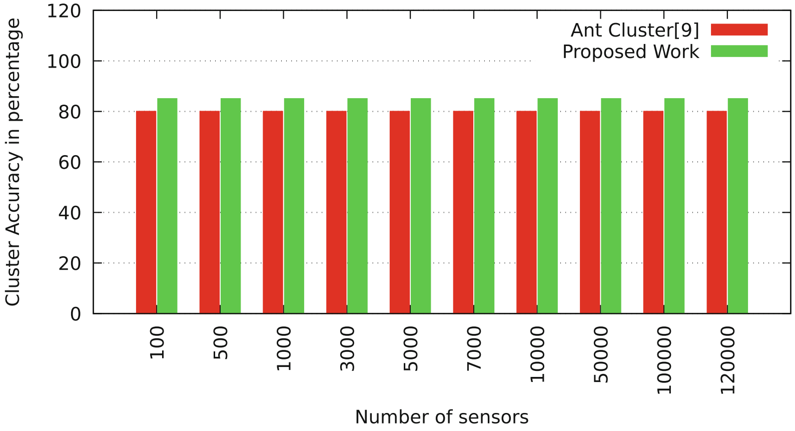

Fig. 3. Comparison of accuracy (quality) of cluster.

Search Accuracy. We conducted another experiment to measure the effectiveness of the proposed approach with respect to the search results. As seen in Fig. 4 search accuracy of the proposed work is better than the existing AntClust algorithm. As we earlier mentioned, our proposed algorithm aids in forming precise clusters and also the use query is forwarded only to the most acceptable cluster due to which the accuracy of search results are significantly higher. Thus, user-centric results are provided the proposed search scheme due to the use of semantic weights and fuzzy clustering technique.

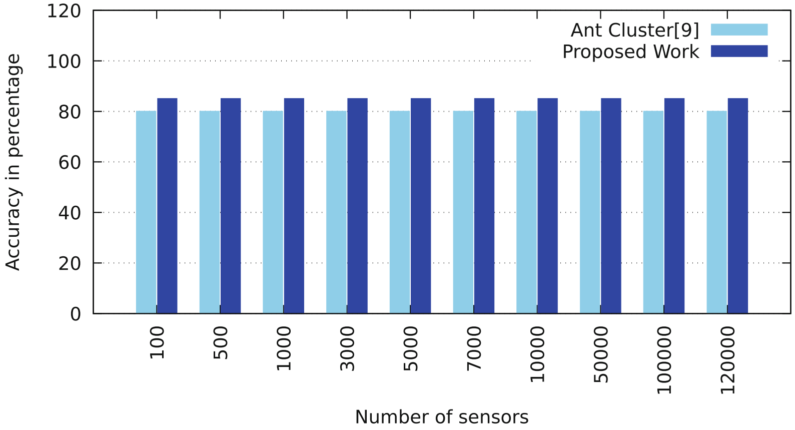

Fig. 4. Comparison of search accuracy.

6 Conclusions

Searching for relevant sensors among a large pool of connected devices in the IoT has become a significant task with the creation of several prominent applications across various sectors of the society. Thus, numerous techniques are proposed in the past to address this problem, but they do not consider the user's context and also take query in rudimentary quantity values. In this paper, we have devised a fuzzy-based context-oriented search algorithm that aids in forming clusters through semantic weights. The submitted user query is directed to the cluster that is most appropriate to the context of the query. Experimental results demonstrate that the proposed search space structuring technique drastically decrease the search processing time owing to the effective semantic-based cluster head selection in comparison to the meta-heuristic approach. It also achieves lower communication overhead through the one-way inter-cluster search. Also, when compared with the existing work our method yields effective results and thus provide a superior user-centric search system.

References

1. Air Quality Sensor Dataset. https://github.com/BathHacked/documentation/wiki/Bath-Air-Quality-Sensor-Data. Accessed 20 Aug 2019
2. Apache Commons Math. https://commons.apache.org/proper/commons-math/. Accessed 20 Aug 2019
3. Apache Jena. https://jena.apache.org/. Accessed 20 Aug 2019
4. MesoWest Dataset. http://mesowest.utah.edu. Accessed 20 Aug 2019
5. SPARQL Query Language. https://www.w3.org/TR/sparql11-query/. Accessed 20 Aug 2019
6. Bezdek, J.C., Ehrlich, R., Full, W.: FCM: the fuzzy c-means clustering algorithm. Comput. Geosci. **10**(2–3), 191–203 (1984)
7. Desai, P., Sheth, A., Anantharam, P.: Semantic gateway as a service architecture for IoT interoperability. In: IEEE International Conference on Mobile Services, pp. 313–319. IEEE (2015)
8. Dilli, R., Argou, A., Pilla, M., Pernas, A.M., Reiser, R., Yamin, A.: Fuzzy logic and MCDA in IoT resources classification. In: Proceedings of the 33rd Annual ACM Symposium on Applied Computing, pp. 761–766. ACM (2018)
9. Ebrahimi, M., ShafieiBavani, E., Wong, R.K., Fong, S., Fiaidhi, J.: An adaptive meta-heuristic search for the Internet of Things. Future Gener. Comput. Syst. **76**, 486–494 (2017)
10. Hung, C.C., Kulkarni, S., Kuo, B.C.: A new weighted fuzzy c-means clustering algorithm for remotely sensed image classification. IEEE J. Sel. Top. Sign. Proces. **5**(3), 543–553 (2011)
11. Lefort, L., et al.: Semantic sensor network. XG final report (2011)
12. Pattar, S., Buyya, R., Venugopal, K.R., Iyengar, S.S., Patnaik, L.M.: Searching for the IoT resources: fundamentals, requirements, comprehensive review, and future directions. IEEE Commun. Surv. Tutorials **20**(3), 2101–2132 (2018)

13. Pattar, S., et al.: Progressive search algorithm for service discovery in an IoT ecosystem. In: 2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 1041–1048 (2019)
14. Pattar, S., et al.: Ontology based service discovery for intelligent transport systems using Internet of Things. In: Fourteenth International Conference on Information Processing (ICInPro), pp. 223–225 (2018)
15. Perera, C., Vasilakos, A.V.: A knowledge-based resource discovery for Internet of Things. Knowl. Based Syst. **109**, 122–136 (2016)
16. Perera, C., Zaslavsky, A., Christen, P., Georgakopoulos, D.: Context aware computing for the Internet of Things: a survey. IEEE Commun. Surv. Tutorials **16**(1), 414–454 (2014)
17. Perera, C., Zaslavsky, A., Liu, C.H., Compton, M., Christen, P., Georgakopoulos, D.: Sensor search techniques for sensing as a service architecture for the Internet of Things. IEEE Sens. J. **14**(2), 406–420 (2014)
18. Roopa, M., Pattar, S., Buyya, R., Venugopal, K.R., Iyengar, S., Patnaik, L.: Social Internet of Things (SIoT): foundations, thrust areas, systematic review and future directions. Comput. Commun. **139**, 32–57 (2019)
19. Truong, C., Römer, K.: Content-based sensor search for the Web of Things. In: Proceedings of the GLOBECOM - IEEE Global Communications Conference, pp. 2654–2660 (2013)
20. Truong, C., Römer, K., Chen, K.: Fuzzy-based sensor search in the Web of Things. In: Proceedings of International Conference on the Internet of Things, (IoT), pp. 127–134 (2012)
21. Zhai, J., Liang, Y., Yu, Y., Jiang, J.: Semantic information retrieval based on fuzzy ontology for electronic commerce. JSW **3**(9), 20–27 (2008)
22. Zhang, C.: FuzWare: a fuzzy-based middleware for context-aware service. In: IEEE 2nd Advanced Information Technology Conference (IAEAC) on Electronic and Automation Control, pp. 1181–1185 (2017)