IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

SPECIAL SECTION ON SMART CACHING, COMMUNICATIONS, COMPUTING AND CYBERSECURITY FOR INFORMATION-CENTRIC INTERNET OF THINGS

Received March 2, 2019, accepted April 7, 2019, date of publication April 11, 2019, date of current version April 25, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2910736

Feature Data Selection for Improving the Performance of Entity Similarity Searches in the Internet of Things

SUYAN LIU[1,2], YUANAN LIU[1], FAN WU[1,2], (Member, IEEE), AND WENHAO FAN[1,2], (Member, IEEE)

[1]School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Beijing Key Laboratory of Work Safety Intelligent Monitoring, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding authors: Yuanan Liu (yuliu@bupt.edu.cn) and Fan Wu (wufanwww@bupt.edu.cn).

This work was supported in part by the National Natural Science Foundation of China under 61821001, in part by the Beijing Key Laboratory of Work Safety Intelligent Monitoring, and in part by the YangFan Innovative and Entrepreneurial Research Team Project of Guangdong Province.

ABSTRACT Sensors are used to sense the state information of physical entities in the Internet of Things (IoT). Thus, a large amount of dynamic real-time data is generated. The entity similarity search based on the quantitative dynamic sensor data is thus worth studying. To the best of our knowledge, there is no research on the entity similarity search based on feature data selection for the quantitative dynamic sensor data in the IoT. This paper proposes a selection mechanism for the entity main features (SMEF). The SMEF is a feature data selection method based on the quantitative dynamic sensor data. It uses the feature matrix to delete the irrelevant entity features, applies an improved relief algorithm (iRelief) to calculate the feature data relevance and proposes a three-component storage relation table of the entities, models, and features (TEMF) for the dynamic feature weights calculation. The experimental results show that the similarity search algorithm based on feature data selection can improve the average search accuracy by more than 10%, as well as increase the search speed and reduce the data transmission and storage costs.

INDEX TERMS Entity similarity search, feature data selection, feature relevance, Internet of Things.

I. INTRODUCTION

With the advent of modern sensor technology, we can use text to define the relationships between sensors and real-world objects [1]. Users are mainly interested in entities related to the Internet of Things (IoT) and their states rather than in sensors and their raw output [2]. Thereby, sensors are used to sense the physical entity state information. The physical entities may have multiple state information, such as the temperature and humidity of a room. Each physical entity can correspond to multiple types of sensors. Each type of sensor corresponds to a physical entity attribute. Sensors are used as the entity features to achieve an entity similarity search in the IoT [3].

The number of connected devices is expected to increase to 50 billion by 2020 [4]. The amount of data generated has been continuously growing from global sensor sources [5]. The massive quantities of data are raising the critical challenge of efficiently and effectively searching for and selecting the sensors most related to a particular need [6]. Therefore, it becomes a major challenge for the entity similarity search. Meanwhile, the irrelevant and redundant features in high-dimensional data not only result in high computational complexity but also seriously reduce the efficiency of the entity similarity search methods. The irrelevant features of the entity similarity search methods of the IoT are features that are not related to customized queries. For example, if we search for conference rooms with the same temperature and humidity, then noise and light are irrelevant features. Redundant features are the features with poor classification abilities for entity similarity queries. For example, in a constant greenhouse, the temperature attribute has no classification ability.

To cope with these problems, we propose an entity selection mechanism of main features (SMEF) to achieve a reduction in the feature dimensions and remove redundant data. First, the irrelevant entity features are deleted by using the

The associate editor coordinating the review of this manuscript and approving it for publication was Jun Wu.

feature matrix. In addition, we propose an improved Relief algorithm (iRelief) to calculate the dynamic data relevance of the sensors in the feature data selection process. Lastly, the features of the customized user query are extracted. A similarity computing model is created dynamically according to the three-component storage relation table of the entities, models and features (TEMF) to improve the efficiency and accuracy of the entity similarity search.

The contributions of this paper are as follows: (i) An architecture is designed to efficiently search similar entities in the IoT. (ii) The feature matrix is created to delete the irrelevant entity attributes. (iii) We propose the iRelief algorithm. The iRelief algorithm can process the quantitative sensor data to calculate the relevance of the feature data for feature data selection. (iv) We define the TEMF to store the corresponding relationships between the entity models and entity features. A dynamic similarity query model for customized queries can be built based on the TEMF to reduce the impact of redundant features on queries. (v) We propose the SMEF method which is a feature data selection method for the entity similarity search.

The rest of this paper is organized as follows: In section II, we survey the sensor similarity search and feature data selection of the IoT, discuss the existing problems. Section III illustrates the proposed architecture of the entity similarity search of the IoT. Section IV illustrates the procedure for the SMEF method. Experimental results of the proposed approach are presented in section V. Finally, section VI concludes the paper.

II. RELATED WORK

The massive real-time sensor data not only increases the cost of data storage and transmission but also brings challenges to IoT search engines [7]. We use a classification algorithm to process the sensor data for the entity similarity search, which can improve the query speed and search accuracy of IoT search engines [3]. We will introduce the similarity search algorithms and feature data selection methods of the IoT in detail.

A. SENSOR SIMILARITY SEARCH OF THE IOT

The similarity search refers to the technology of querying similar contents in a data set for a given sample. The similarity search has numerous uses, and has been studied extensively, such as for the content-based retrieval services for moving objects [8], the image similarity search [9], optimizes users' products and helps them to find potential consumers [10], web services for recommendations [11], and so on. With the development of network and sensor technologies, sensors have undergone rapid growth and are now collecting more complex multivariate data. Traditional Internet search engines fail to meet the needs of searching the massive real-time sensor data that is produced. The sensor similarity search method for the IoT is thus worth studying.

Truong *et al.* [12] calculated the similarity of different rooms based on the fuzzy set method (FUZZY). The FUZZY method calculates the temperature probability and density functions according to a time interval which leads to a strict timing constraint. That is, we calculate the similarity of sensors S and V in the time interval $[t_1, t_2]$. We need to also calculate the density function in the time interval $[t_1, t_2]$. Then, we change the query interval to $[t_3, t_4]$; therefore, we need to recompute the density function of the sensors. As a result, the redundant data increases the number of calculations. In [6], time-dependent fitting models are used to calculate the sensor similarity. However, the single-feature entity similarity search method has a low sensor similarity search accuracy in a closed search space with similar feature data. The study of the multi-feature entity similarity search is imperative to solve the problems of the single-feature similarity searches [3]. The high dimensionality and complex data in the IoT increase the complexity of the multi-feature similarity calculation and reduces the search efficiency.

B. THE IOT DATA FEATURE SELECTION

Feature extraction and feature selection are the two main categories of dimensionality reduction [13]. Feature extraction is applied to use the original features to get a new low-dimensional feature space. Feature selection is based on the original feature set to obtain a low-dimensional feature subset space that satisfies certain conditions [14]. The massive sensor data can be processed by feature selection to achieve dimensionality reduction.

There are a few studies on dimensionality reduction in the IoT. In [14], the maximum information coefficient (MIC) is used to reduce the data dimensions. The MIC can recognize the relationships between sine, hyperbolic and linear functions, but the value of the MIC is the same. In addition, the impact of noise on the MIC is independent of the functional relationships between the variables. Zhao and Dong *et al.* [15] proposed the FMPE feature selection algorithm based on a potential entropy evaluation criteria. In [16], the FCBFiP algorithm is proposed on the basis of the FCBF. The FCBFiP algorithm is a feature selection algorithm that can be quickly filtered based on the feature correlation. The quantitative sensor data is highly dynamic and real-time. The above feature selection methods use information theory and probability distribution algorithms to reduce the data dimensions and are not suitable for high-dimensional, dynamic, time-dependent sensor data dimension reduction.

In this paper, the feature data selection problem for dynamic sensor data is studied based on the IoT entity similarity search methods. First, we construct the feature matrix based on the entity attributes to compute the common feature set. Second, the feature data relevance and weights are calculated according to the iRelief algorithm. We specify that the attribute data that has a weight that is less than a threshold value will be removed from the feature data set. Finally, we use the customization query to search for redundant features which do not contribute much to the similarity search.

III. SYSTEM DESCRIPTION

In this section, we propose the architecture of the sensor search. It is important to design a low-cost search framework, based on original sensor data, to improve the efficiency of the sensor search process. The IoT sensor search system consists of seven types of modules: client, gateway, similarity calculation, model database, feature data selection, TEMF, and wireless sensor network (WSN). Sensors periodically collect the values of real-world objects and automatically identify and report these values to local gateways. We define the sensor data format of this paper as $\left(X_{t_j}, Y_{t_j}\right)$, where X_{t_j} is the time value, Y_{t_j} is the data value, and t_j is the time sequence. The local gateway deletes the redundant features through feature data selection methods and stores the TEMF for the dynamic similarity model construction. In addition, we then use the similarity calculation algorithm to calculate the similarity model of the entities according to the dynamic data of sensors. Users customize the query through the global gateway and give the query time. The global gateway obtains the target object query model, $S_q = \left(\{a_i \,| i \in N^*\}, T, \left(R_q\right)\right)$, from the local gateway, where a_i refers to the entity feature, T is the query time interval, and R_q is the search result that needs to be returned to the user. Then, the similarities between other entities and the target entity are calculated according to the query model. Ultimately, the resulting list is fed back to the global gateway based on the similarity and the matching algorithm. The architecture of the sensor search system is illustrated in Figure 1.

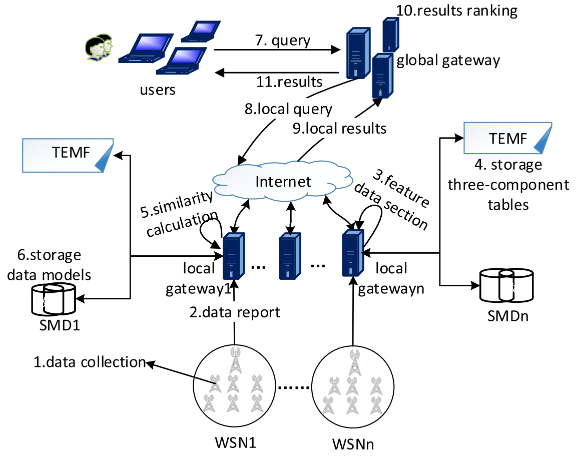


FIGURE 1. The architecture of the entity search system.

IV. PROPOSED THE SMEF METHOD

The redundant and irrelevant data increases the amount of computational effort and storage for the similarity computing model. The focus of this paper is the use of feature data selection to optimize the entity similarity search methods. We proposed a feature matrix, the iRelief algorithm and the TEMF to implement the SMEF, which is a feature data selection method.

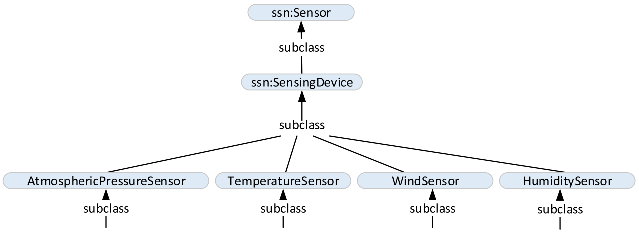


FIGURE 2. Schematic diagram of sensor ontology fragment.

A. THE ENTITY FEATURE MATRIX

In this section, entity features are defined according to the SSN ontologies [17]. A simple short description of the ontology is defined, as shown in Figure 2. An entity may correspond to multiple sensors, because a certain type of sensor is used to sense a particular feature of the entity. Some types of sensors are used to sense multiple physical entities and physical environments.

A feature matrix is proposed to obtain the entity's common features. We construct the entity feature matrix M. The rows represent the physical entities and the columns represent the sensors that sense the physical entities. E_i denotes entity i, m is the number of the entity, C_j denotes the j-th sensor of the feature, n is the number of the feature. A one-dimensional array, c, is used to store the sensor feature sets.

$$c[j] = C_j \mid \quad j = 1, 2, \ldots, n \tag{1}$$

We define i as the row indices and j as the column indices of the entity feature matrix. The $m \times n$ entity feature matrix M is as follows:

$$M = \left(m_{ij}\right) \mid \; i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n \tag{2}$$

$m_{ij} = 1$ when the entity E_i includes the sensor feature C_j, otherwise the $m_{ij} = 0$.

B. THE IMPROVED RELIEF ALGORITHM

The Relief algorithm [18] is a feature weighting algorithm, which is mainly used for two-category searches. The feature weighting algorithm assigns different weight to each feature according to the relevance of each feature and the category, and the feature with a weight less than a certain threshold value is removed. The Relief algorithm randomly selects a sample S from the training set D, then searches for the nearest neighbor sample, NH, from the samples of the same kind as S and the nearest neighbor sample, NM, from the samples different from S, and then updates the weight of each feature according to the following rules: the feature is beneficial to distinguish the nearest neighbor of the same kind from the one of a different kind when the feature distance between S and NH is less than the distance between S and NM. The weight of the feature is then increased. If the opposite is true, then the weight of the feature is reduced. The greater the weight of the feature is, the stronger the classification ability of the feature is. The lower the weight of the feature is, the weaker the classification ability of the feature is.

In the IoT sub-query in the local gateway, we search entities that are similar to the target entity. Therefore, the entity similarity search can be simplified into two-category search problems. Thus, the Relief algorithm is improved for the IoT entity similarity search. Sensors periodically report data to the local gateway. The data set of the feature f is denoted as $D^f = \left\{s_1^f, s_2^f, \ldots, s_m^f\right\}$, where s_i^f is dataset of the feature f's i-th sensor and m is the number of sensors. We denote $s_i^f = \left\{\left(x_{t_1}^i, y_{t_1}^i\right), \left(x_{t_2}^i, y_{t_2}^i\right), \ldots, \left(x_{t_n}^i, y_{t_n}^i\right)\right\}$, where $x_{t_j}^i$ is the time, $y_{t_j}^i$ is the value of the sensor i, t_j is the time series, and n is the number of data points. The minimum value difference between sensors of the same kind as s_i^f is $s_{i,nh}^f$, and the minimum value difference between sensors of different classes from s_i^f is $s_{i,nm}^f$. The relevant statistic corresponding to feature f is:

$$\delta_f = \sum_{i=1}^{m}\left(-diff\left(s_i^f, s_{i,nh}^f\right)^2 + diff\left(s_i^f, s_{i,nm}^f\right)^2\right) \tag{3}$$

In addition, the definition of the $diff$ function depends on the type of feature f.

f is the qualitative sensor attribute:

$$diff\,(s_a^f, s_b^f) = \begin{cases} 0, & \sum_{j=1}^{n}\left|y_{t_j}^a - y_{t_j}^b\right| \Big/ n < \Delta \\ 1, & otherwise \end{cases} \tag{4}$$

$\Delta \in [0, 1]$ is the threshold of the sensor data value difference. $y_{t_j}^a$ is the value of sensor a.

f is the quantitative sensor attribute:

$$diff\,(s_a^f, s_b^f) = \sum_{j=1}^{n}\left|y_{t_j}^a - y_{t_j}^b\right| \Big/ v_f \tag{5}$$

v_f is:

$$v_f = \sum_{i=1}^{m}\sum_{j=1}^{n} y_{t_j}^i \Big/ mn \tag{6}$$

Formula (1) is the relevance statistical component of the feature f. The larger the component value is, the stronger the classification ability is.

C. THE TEMF DEFINITION

We propose TEMF to store the corresponding relationships between the entity models and entity features. The TEMF is used to create the dynamic similarity calculation model to avoid the influence of irrelevant features on the entity similarity search in the user-customized query. It is defined as:

$$T = \left\{l_f, l_{w_f}, l_e\right\} \tag{7}$$

l_f is the feature list of the entities. This list is the complete set of features in the local gateway. l_{w_f} is the feature weight list, and l_e is the entity set list. Entities in l_e are all sensed by sensors in l_f.

The feature weight list constraint function is:

$$\sum l_{w_f} = 1 \tag{8}$$

When a feature in l_f changes, the weight in l_{w_f} changes accordingly. When l_{f_i}, $0 < i < \left|l_f\right|$ is missing in l_f, the calculation method of the remaining feature list is:

$$l_{w_{f_j}} = w_{f_j} \Big/ \sum w_{f_j}, \quad f_j \notin l_{f_i}, \, f_j \in l_f \tag{9}$$

Formula (9) satisfies the constraint of (8).

D. THE SMEF METHOD

The focus of this paper is to select the entity attribute with a strong classification ability to optimize the similarity search. The SMEF is a feature data selection method. It removes redundant and irrelevant feature data to improve the accuracy and speed of the similarity search and reduce the sensor data storage requirements. It executes the following steps sequentially:

Step 1: Sensors periodically collect data from real-world objects and automatically report these values to the local gateway. To compare different units or orders of magnitude, we use formula (10) to process the sensor data.

$$y'_{t_j} = y_{t_j} / \,\bar{y} \tag{10}$$

We define $\bar{y}$ as the feature arithmetic mean value and use y'_{t_j} instead of the feature value y_{t_j} to calculate the attribute relevance.

Step 2: We process the feature and entity information according to the sensor data to form the feature matrix M in part A of this section. The common feature information set, $\left\{l_f, l_e\right\}$, of the entities is calculated by the and operation of the matrix row, and the redundant feature information is deleted.

Step 3: According to the calculation algorithm of relevant features in part B of this section, we calculate the relevant features in $\left\{l_f, l_e\right\}$, delete the irrelevant features, calculate the weight of features, w_f, build the feature list, l_{w_f}, and form the TEMF, $T = \left\{l_f, l_{w_f}, l_e\right\}$, in part C of this section.

We suppose that the entity E has p features and define the feature weight as:

$$w_f = \begin{cases} \delta_f \Big/ \sum_{f=1}^{p} \delta_f, & p \in R^*, \quad \delta_f \geq 0 \\ 0, & \delta_f < 0 \end{cases} \tag{11}$$

The threshold of the feature weight is:

$$\tau = (\max(w_f) - \min(w_f)) / p, \quad p \in R^* \tag{12}$$

The relevant features calculation formula for feature f is:

$$R_f = \begin{cases} 1, & w_f \geq \tau \\ 0, & otherwise \end{cases} \tag{13}$$

Step 4: The query feature list, l_{q_f}, is extracted according to the user-customized query, and then the weight of the query feature list is calculated according to the TEMF in step 3 and formula (9) in part C of this section. We obtain the specific query three-component table, $T_q = \left\{ l_{qf}, l_{w_{qf}}, l_e \right\}$, to delete the irrelevant features of the customized query.

V. EXPERIMENTS AND RESULTS

This paper assumes that the same attribute units are unified. We conduct dimensionless processing on the sensor data so that the units of the data are not marked in the figures. The Intel Lab dataset [19] is used for the evaluation and the recorded times are standardized. In the interlinked space, temperature and humidity are similar. We group by sensors 3 and 18. This paper takes 100 datasets from 2 groups of 10 sensors to use for verification, as shown in Figure 3. There are a few studies based on quantitative sensor data, such as the FUZZY algorithm [12]for the single-feature similarity search, the GFC algorithm [3]for the multi-feature similarity search, and the least squares method for the sensor data fitting [20]. This section mainly evaluates the effect of the feature selection in the FUZZY, least squares linear, least squares polynomial and GFC algorithms.

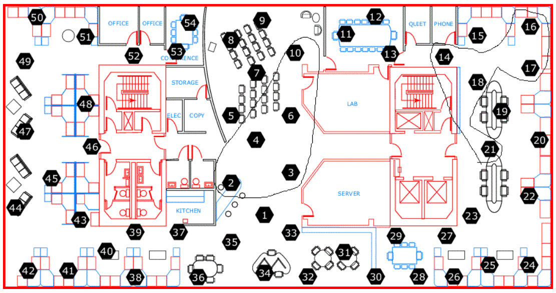


FIGURE 3. Intel lab contains information about the data collected from 54 sensors between february 28th and april 5th, 2004.

A. COMPARISON OF THE SINGLE-FEATURE SIMILARITY SEARCH METHODS

The FUZZY method uses quantitative sensor data. In [12], the sensor data is used to search for similar rooms. The FUZZY algorithm calculates the density function for the time interval to make a similarity query. We compared the search accuracies for the FUZZY algorithm before and after the feature data selection, as shown in Figure 4. It is difficult to obtain the a priori polynomial by the least squares fitting method. First, the linear function is used for data fitting. The comparison of the search accuracies is shown in Figure 5. Then, according to the feature data, the polynomial with the highest degree is used to fit the sensor data. Finally, the similarity search is carried out according to the fitted function, as shown in Figure 6.

In this section, the single-feature similarity search methods are used to evaluate the humidity data only. The search accuracies are notably improved after the feature data selection. The proportion of query accuracy greater than or equal to 0.5 is significantly increased, and the average query accuracy is improved by at least 10%.

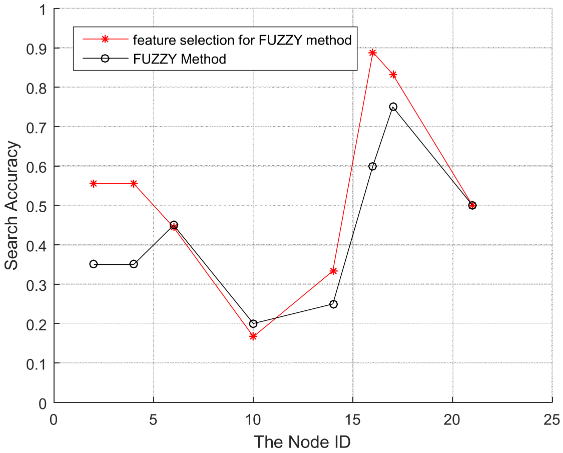


FIGURE 4. The Fuzzy method is used to compare the accuracy of the single-feature search before and after the feature data selection. The circles denote the FUZZY method's search accuracies before the feature data selection, while the stars denote the search accuracies after the feature data selection.

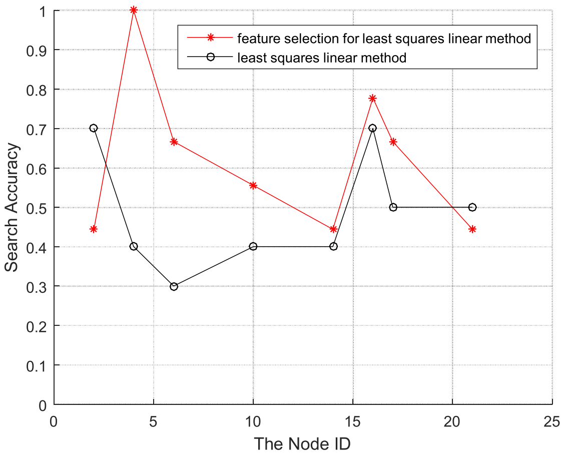


FIGURE 5. The least squares linear algorithm is used to compare the single-feature search accuracies before and after feature data selection. The circles denote the least squares linear algorithm's search accuracies before feature data selection, while the stars denote the search accuracies after feature data selection.

B. COMPARISON WITH THE MULTI-FEATURE SIMILARITY SEARCH METHOD

The multi-feature entity similarity search algorithm of the GFC algorithm is based on the dynamic quantitative feature attribute value, and the entity similarity is calculated by the similarity calculation function [3]. The formula for calculating multi-feature similarities is divided into the feature functions and the weights. The feature weights reflect the distinguishing degree of the features for the entity similarity search. The SMEF method can not only calculate the feature relevance but also calculate the feature weights to assist the GFC algorithm in calculating the formula feature weights. Search accuracies are notably improved after the feature data selection. The average query accuracy is improved by at least 10%, as shown in Figure 7.

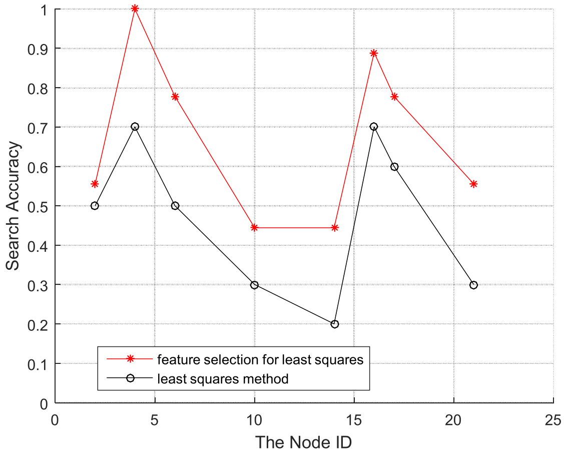


FIGURE 6. The least squares polynomial algorithm is used to compare the single-feature search accuracies before and after the feature data selection. The circles denote the least squares polynomial algorithm's search accuracies before the feature data selection, while the stars denote the search accuracies the after feature data selection.

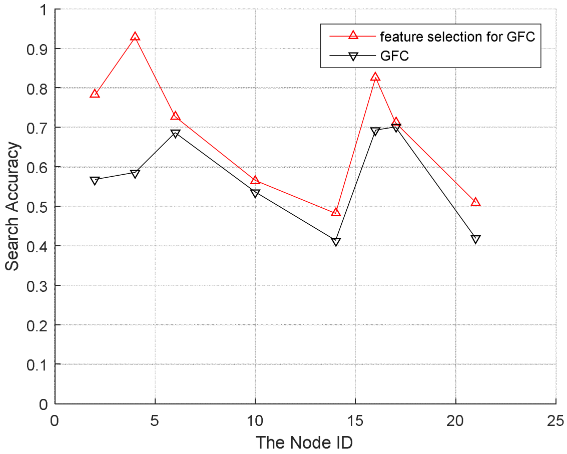


FIGURE 7. The GFC algorithm is used to compare the multi-feature search accuracies before and after the feature data selection. Inverted triangles denote the GFC algorithm's search accuracies before the feature data selection, while the triangles denote the search accuracies after the feature data selection.

C. ANALYSIS OF SEARCH SPEED FOR FEATURE SELECTION

The minimum value difference calculation between the sensors determines the time complexity of the feature data selection. It is assumed that the sensor data number is n, and the time complexity is $O\left(n^2\right)$. The maximum time complexity of the entity similarity search algorithms that are compared in this paper is $O\left(n^2\right)$ [3]. The SMEF method does not increase the time complexity of the entity similarity search. To reduce the errors, 100 queries are performed for each search node, and the average search time is calculated. The real query time is related to the equipment, resources, and other factors; however, this paper ignores the impact of these conditions, and the sensor search functions are executed using the same hardware resources. It is difficult to obtain the fitting polynomial as an a priori condition, and the calculation time is different for different polynomials, so this part does not perform the analysis using the least squares algorithm. The search time is significantly reduced after the feature data selection, as shown in Figure 8.

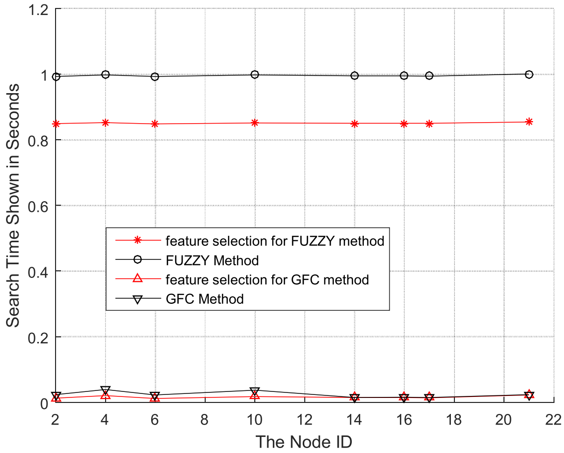


FIGURE 8. Comparison of the entity similarity search methods speed before and after the feature data selection.

D. QUANTITATIVE PERFORMANCE ANALYSIS

We assume that the data are stored as floating-point numbers and that the data transmission cost is the same for each data instance. In this paper, 100 groups of feature data are calculated. Ten groups of feature data are deemed as irrelevant after the feature data selection. First, the storage overhead is reduced compared to the original data. Furthermore, the coefficients and variables of the functions fitted by the GFC, FUZZY and least squares methods are stored. The storage requirements for the coefficients and variables are certainly reduced by reducing the 10 datasets.

VI. CONCLUSION

In this paper, feature data selection is investigated based on quantitative data (e.g., temperature and humidity). Sensors are used as entity features. We propose a feature matrix to store the relationship between the entities and features. The Relief algorithm is used for two-category searches, but it cannot be directly applied to large amounts of dynamic sensor data. We improve the Relief algorithm to enable it to process the dynamic sensor data. The TEMF is proposed to dynamically construct the multi-feature similarity computing model. The experimental results show that the similarity search algorithm with the SMEF method improves the average search accuracy by at least 10%, improves the search speed, and reduces the costs of data transmission and storage. This paper defines the data format as quantitative time series data. Our future research will focus on heterogeneous data conversion methods.

REFERENCES

[1] H. Wang, C. C. Tan, and Q. Li, "Snoogle: A search engine for pervasive environments," *IEEE Trans. Parallel Distrib. Syst.*, vol. 21, no. 8, pp. 1188–1202, Aug. 2010.

[2] B. Ostermaier, B. M. Elahi, K. Römer, M. Fahrmair, and W. Kellerer, "Dyser: Towards a real-time search engine for the web of things," in *Proc. 6th ACM Conf. Embedded Netw. Sensor Syst.*, Raleigh, NC, USA, Nov. 2008, pp. 429–430.

[3] S. Liu, Y. Liu, W. Fan, and P. Zhang, "Multi-feature sensor similarity search for the Internet of Things," *IEICE Trans. Commun.*, vol. E101-B, no. 6, pp. 1388–1397, 2018.

[4] D. Evans, "The Internet of Things how the next evolution of the internet is changing everything," cisco, San Jose, CA, USA, White Papers, Apr. 2011.

[5] S. Li, L. Xu, and S. Zhao, "The Internet of Things: A survey," *Inf. Syst. Frontiers*, vol. 17, no. 2, pp. 243–259, 2015.

[6] S. Liu, Y. Liu, F. Wu, and W. Fan, "Sensor search based on sensor similarity computing in the Internet of Things," *J. Electron. Inf. Technol.*, vol. 40, no. 12, pp. 3020–3027, 2018.

[7] P. Zhang, X. Kang, D. Wu, and R. Wang, "High-accuracy entity state prediction method based on deep belief network towards IoT search," *IEEE Wireless Commun. Lett.*, to be published. doi: 10.1109/LWC.2018.2877639.

[8] Y. Fang, R. Cheng, W. Tang, S. Maniu, and X. Yang, "Scalable algorithms for nearest-neighbor joins on big trajectory data," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 3, pp. 785–800, Mar. 2016.

[9] W. J. Scheirer, N. Kumar, P. N. Belhumeur, and T. E. Boult, "Multi-attribute spaces: Calibration for attribute fusion and similarity search," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, vol. 157, no. 10, pp. 2933–2940.

[10] P. Li, H. Luo, and Y. Sun, "Similarity search algorithm over data supply chain based on key points," *Tsinghua Sci. Technol.*, vol. 22, no. 2, pp. 174–184, Apr. 2017.

[11] S. S. Peerzade, "Web service recommendation using PCC based collaborative filtering," in *Proc. Int. Conf. Energy, Commun., Data Anal. Soft Comput.*, Chennai, India, Aug. 2017, pp. 2920–2924.

[12] C. Truong, K. Römer, and K. Chen, "Fuzzy-based sensor search in the web of things," in *Proc. 3rd IEEE Int. Conf. Internet Things*, Oct. 2012, pp. 127–134.

[13] K. Samina, K. Tehmina, and N. Shamila, "A survey of feature selection and feature extraction techniques in machine learning," in *Proc. Sci. Inf. Conf.*, Aug. 2014, pp. 372–378.

[14] G. Sun, J. Li, J. Dai, Z. Song, and F. Lang, "Feature selection for IoT based on maximal information coefficient," *Future Gener. Comput. Syst.*, vol. 89, pp. 606–616, Dec. 2018.

[15] L. Zhao and X. Dong, "An industrial Internet of Things feature selection method based on potential entropy evaluation criteria," *IEEE Access*, vol. 6, pp. 4608–4617, 2018.

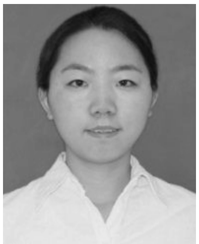
[16] S. Egea, A. R. Mañez, B. Carro, A. Sánchez-Esguevillas, and J. Lloret, "Intelligent IoT traffic classification using novel search strategy for fast-based-correlation feature selection in industrial environments," *IEEE Internet Things J.*, vol. 5, no. 3, pp. 1616–1624, Jun. 2018.

[17] P. Barnaghi *et al.* (2011). *Semantic Sensor Network XG Final Report: W3C Incubator Group Report*. Semantic Sensor Network Incubator Group. [Online]. Available: http://www.w3.org/2005/Incubator/ssn/XGR-ssn-20110628/

[18] K. Kira and L. A. Rendell, "The feature selection problem: Traditional methods and a new algorithm," in *Proc. 10th Nat. Conf. Artif. Intell.* Menlo Park, CA, USA: AAAI Press, 1992, pp. 129–134.

[19] Intel Berkeley Research Lab. *Intel Berkeley Research Lab Sensors Data*. Accessed: 2004. [Online]. Available: http://db.csail.mit.edu/labdata/labdata.html

[20] P. Zhang, Y. Liu, F. Wu, S. Liu, and B. Tang, "Low-overhead and high-precision prediction model for content-based sensor search in the Internet of Things," *IEEE Commun. Lett.*, vol. 20, no. 4, pp. 720–723, Apr. 2016.



SUYAN LIU received the M.E. degree from the Harbin University of Science and Technology, China, in 2009. She is currently pursuing the Ph.D. degree with the Beijing University of Posts and Telecommunications, China. She was a Software Engineer for six years. Her current research interests include machine learning, the Internet of Things, and sensor search. She has received a Senior Computer Engineer Qualification Certificate, in 2015.

YUANAN LIU received the M.E. and Ph.D. degrees from the Chengdu University of Electronic Science and Technology, China, in 1989 and 1992, respectively. He was involved in a Postdoctoral Research at the Beijing University of Posts and Telecommunications, China, from 1992 to 1994, where he is currently the Executive Director of the School of Electronic Engineering. He was with Carleton University, Canada, from 1995 to 1997. He focuses on EMC, mobile communications, and the Internet of Things. He is a Fellow of the IEE.

FAN WU received the Ph.D. degree from the Beijing University of Posts and Telecommunications, China, in 2009, where she is currently an Associate Professor with the School of Electronic Engineering. Her research interests include the Internet of Things, sensor search, and pervasive computing.

WENHAO FAN received the Ph.D. degree from the Beijing University of Posts and Telecommunications, China, in 2013, where he is currently an Associate Professor with the School of Electronic Engineering. His research interests include mobile terminal, cloud computing, and the Internet of Things.