Research Article

Using Competitive Binary Particle Swarm Optimization Algorithm for Matching Sensor Ontologies

Lei Xiao ¹, Junhong Feng ¹, Xishuan Niu ¹ and Jian-Hong Wang ²

¹School of Computer Science and Engineering, Yulin Normal University, Yulin 537000, China

²Department of Computer Science and Information Engineering, National Chin-Yi University of Technology, Taichung 411030, Taiwan

Correspondence should be addressed to Junhong Feng; jgxyfjh@126.com and Xishuan Niu; cookes2000@163.com

Received 14 December 2021; Accepted 26 January 2022; Published 9 February 2022

Academic Editor: Jianhui Lv

Copyright © 2022 Lei Xiao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Developing sensor ontologies and using them to annotate the sensor data is a feasible way to address the data heterogeneity issue on Internet of Things (IoT). However, the heterogeneity issue exists between different sensor ontologies hampers their communications. Sensor ontology matching aims at finding all the heterogeneous entities in two ontologies, which is a feasible solution for aggregating heterogeneous sensor ontologies. This work investigates swarm intelligence (SI)-based sensor ontology matching techniques and further proposes a competitive binary particle swarm optimization algorithm (CBPSO)-based sensor ontology matching technique. In particular, a guiding matrix (GM) is proposed to ensure the population's diversity and a competitive evolutionary framework is presented. The experiment uses ontology alignment evaluation initiative (OAEI)'s benchmark and three real sensor ontologies to test CBPSO's performance. The experimental results show that the competitive evolutionary framework is able to help CBPSO effectively optimize the alignment's quality, and it significantly outperforms other SIs at 5% significant level.

1. Introduction

Since most Internet of Things (IoT) [1] devices work in diverse environments, the heterogeneity problem of underlying devices makes it difficult to provide a uniform way of representing sensor data. Developing sensor ontologies [2] and using them to annotate the sensor data is a feasible way to address these issues. Currently, various sensor ontologies, such as the CSIRO sensor ontology (CSIRO) [3], semantic sensor network ontology (SSN) [4], and MMI device ontology (MMI) [5], have been widely used in the IoT domain. These sensor ontologies own lots of overlapped information, but there also exists the heterogeneity problem between them since one sensor concept might be represented with different terminologies, granularities, or contexts. Sensor ontology matching technique is able to map the identical ontology entity pairs, which enables the processing, interpretation, and sharing of sensor data that are organized with different ontological schemes [6].

Inspired by the success of metaheuristics-based matching technique in ontology matching domain, this work further proposes a competitive binary particle swarm optimization algorithm (CBPSO) and uses it to address the sensor ontology matching problem. To better trade off the algorithm's searching performance, CBPSO uses guiding matrix (GM) to describe the solutions' distribution and guide newly generated particle's initialization; then, two competitive subpopulations are introduced to, respectively, focus on the exploitation and exploration. The contributions made in this work are as follows: (1) we present the mathematical formula for the sensor ontology matching problem; (2) we propose a CBPSO-based sensor ontology matching technique, which uses the GM to ensure the population's diversity, and two competitive subpopulations to trade off the algorithm's exploitation and exploration; (3) we employ CBPSO on ontology alignment evaluation initiative (OAEI)'s benchmark and three real sensor ontologies' matching tasks; the results reveal that CBPSO is able to determine high-quality sensor alignments.

The rest paper is organized as follows. The sensor ontology matching problem is defined in Section 3; CBPSO is presented in Section 4 in detail; the experimental results are shown in Section 5; finally, Section 6 draws the conclusion.

2. Related Work

Determining high-quality sensor ontology alignment is a challenge since there exist rich semantic relationships among sensor concepts [7]. To face this challenge, in recent years, different researchers have proposed various intelligent matching techniques. FuzzyAlign [8] integrates the semantic sensor web with a fuzzy theory-based method, and it uses evolutionary algorithm (EA) to find the optimal threshold for filtering the final alignment's quality. Differential evolution-based ontology matching (DEOM) [9] first uses the neural network to train the ontology matcher, and then, DE is used to find the correct correspondences. Co-evolutionary algorithm-based ontology matching (cEAOM) [10] uses multiple populations' trade off the algorithm's performance and improves the searching efficiency. Evolutionary tabu search algorithm (ETSA) [11] improves the algorithm's converging speed by combining the tabu search algorithm [12] as the local search strategy with EA. Due to the limitation of f-measure [13], the classic EA might optimize the solution's quality by improving recall (or precision) while sacrificing the other [14]. To overcome this drawback, multiobjective evolutionary algorithm-based ontology matching (MOEAOM) [15] is proposed to simultaneously optimize two objectives. Comparing with single-objective EA, MOEAOM is able to provide the decision makers with more nondominated options.

Besides EA, another category of metaheuristics, i.e., swarm intelligence (SI), is also applied to match the sensor ontologies. Borrowing the idea from cEA, a co-Firefly Algorithm (cFA) [16] is proposed to optimize the sensor ontology alignment's quality. It uses two subpopulations, i.e., the better subpopulation with higher-quality elite, and the worse subpopulation whose elite's fitness value is lower. These two subpopulations, respectively, use the exploitation and exploration, and when the worse subpopulation's elite's quality outperforms the better subpopulation, they adaptively switch the searching strategy. More recently, particle swarm optimization (PSO) [17] is also presented to aggregate different sensor ontology matchers, which borrows the idea from ETSA and introduces the simulated annealing (SA) [18] to execute the local search process.

3. Sensor Ontology Matching Problem

A sensor ontology O is a 3-tuple (C, P, R) , where C , P , and R , respectively, represent the sets of concept, data-type property, and relationship between two concepts [19]. Matching sensor ontologies aim at finding the sensor ontology alignment, which is a set of sensor entity correspondences. Each correspondence is a 4-tuple $(e_1, e_2, \text{rel}, \text{conf})$, where e_1 and e_2 are, respectively, two sensor ontologies' entities, rel is two entities' relationship, and conf is a real number in $[0, 1]$ that denotes to what extent the correspondence holds [20]. Often, conf is measured by the similarity value sim Value

between two entities. If $\text{sim Value} = 1$, two entities are the same, and if $\text{sim Value} = 0$, they are different.

There are three categories of ontology matcher for measuring the entities' similarity, which are, respectively, based on string, linguistic, and ontology structure [21]. This work uses a hybrid similarity measure that is proposed by Xue and Wang [22], which combines three kinds of ontology matchers to improve the result's confidence. Given an alignment A , the more correspondences it has ($f_1(A) = \text{normalize}(|A|)$) and the higher mean similarity value it owns ($f_2(A) = \sum \text{sim}_i/|A|$), the better quality it has. On this basis, we use the following metric to evaluate the quality of [23]

$$f(A) = \frac{2 \times f_1(A) \times f_2(A)}{f_1(A) + f_2(A)}. \quad (1)$$

To optimize the alignment's quality, the sensor ontology matching problem is defined as follows:

$$\begin{cases} \max F(M), \\ M = [m_{i,j}]_{|O_1| \times |O_2|}, \\ \text{s.t.} \\ m_{i,j} \in \{0, 1\}, \end{cases} \quad (2)$$

where M is the matching matrix, $m_{i,j} = 1$ means the i th entity in ontology O_1 is mapped with the j th entity in ontology O_2 , and $F(M)$ use the function $f()$ to measure M 's corresponding alignment's quality.

4. Competitive Binary Particle Swarm Optimization Algorithm

For trading off the algorithm's exploitation and exploration, CBPSO uses two subpopulations, where the one with better (worse) global elite particle is called the better (worse) subpopulation. In this work, the better subpopulation mainly focuses on the exploitation, and the worse one tends to explore the unknown domain.

4.1. Initialize Particle. CFPA uses the binary encoding mechanism to encode a particle, which is denoted by a 0-1 matrix, whose row and column are, respectively, two entity sets, and its element 1 means two corresponding entities are mapped, while 0 means not. When the scale of entities is huge, this matrix is a sparse one, i.e., most of its elements are 0. Therefore, we need to uniformly assign the elements with value 1 in each subpopulation's matrices on the whole. To this end, we introduce a guiding matrix, which is of help to ensure the diversity of each subpopulation. For the sake of clarity, given a guiding matrix GM and a matching matrix MM [24], we show the pseudocode of initializing a particle in the subpopulation in Algorithm 1.

In Algorithm 1, we initialize a new particle according to GM , whose elements represent the distribution of current subpopulation's particle positions. The larger value of GM 's element is, the more likely this dimension has been explored by the existing particles, whose probability of being 1 in newly generated particle should be smaller. In this way, we are able to ensure the diversity of subpopulation to the maximum extent.

```

(1) if (GM is not given)
(2)   set all GM's elements as zero;
(3) end if
(4) for int  $j = 0$  to  $\text{rand} \times |O_1| \times |O_2|$ 
(5)    $[(m, n), (m', n')] = \text{randomly select two elements from MM};$ 
(6)   if  $(\text{GM}_{m,n} < \text{GM}_{m',n'})$ 
(7)      $\text{MM}_{m,n} = 1;$ 
(8)      $\text{GM}_{m,n} = \text{GM}_{m,n} + 1;$ 
(9)   else
(10)     $\text{MM}_{m',n'} = 1;$ 
(11)     $\text{GM}_{m',n'} = \text{GM}_{m',n'} + 1;$ 
(12)   end if
(13) end for
(14) return MM

```

ALGORITHM 1: Initialization.

4.2. Update Particle. In each iteration, particle MM is updated according to the following equations:

$$\text{MM}^{t+1} = \omega_t \text{MM}^t + c_1^t (\text{MM}_{\text{local}}^* - \text{MM}^t) + c_2^t (\text{MM}_{\text{global}}^* - \text{MM}^t), \quad (3)$$

$$s(\text{MM}_{i,j}^{t+1}) = \frac{1}{1 + e^{-\text{MM}_{i,j}^{t+1}}}, \quad (4)$$

$$\text{MM}_{i,j}^{t+1} = \begin{cases} 1, & \text{if } \text{rand} \leq s(\text{MM}_{i,j}^{t+1}), \\ 0, & \text{otherwise,} \end{cases} \quad (5)$$

where $\text{MM}_{\text{local}}^*$ and $\text{MM}_{\text{global}}^*$ are, respectively, the particle's local optima and its subpopulation's global optima and ω_t is the inertia weight in t th generation, which is determined by the following equation:

$$\omega_t = \omega_{\text{start}} + (\omega_{\text{start}} - \omega_{\text{end}}) \left(1 - \frac{t}{\max T}\right). \quad (6)$$

c_1^t and c_2^t are two learning factors, whose values are, respectively, calculated by the following equations:

$$c_1^t = c_{1,\text{start}} + (c_{1,\text{end}} - c_{1,\text{start}}) \frac{t}{\max T}, \quad (7)$$

$$c_2^t = c_{2,\text{start}} + (c_{2,\text{end}} - c_{2,\text{start}}) \frac{t}{\max T}, \quad (8)$$

where $\omega_{\text{start}} = 0.9$, $\omega_{\text{end}} = 0.4$, $c_{1,\text{start}} = 2.5$, $c_{1,\text{end}} = 0.5$, $c_{2,\text{start}} = 0.5$, $c_{2,\text{end}} = 2.5$, and $\max T$ is the maximum generation. Here, we use different annealing strategies on ω_t , c_1^t , and c_2^t to trade off the algorithm's exploration and exploitation. All these parameters are configured empirically, which enables the algorithm determine the highest f-measure in average on all testing cases.

4.3. The Pseudocode of Competitive Binary Particle Swarm Optimization Algorithm. The pseudocode of CBPSO is presented in Algorithm 2. The initialization of CBPSO consists of three steps: (1) two subpopulations P_{better} and P_{worse} 's particles are initialized according to Algorithm 1, (2)

P_{better} and P_{worse} 's global optimal particles are initialized by selecting the particles with the best objective function values, and (3) each particle's local optima is initialized with newly generated particle. During each iteration, each particle is first updated according to equations (3)–(8); then, their corresponding local optima and each subpopulation's global optima are updated accordingly. After that, P_{better} and P_{worse} are competed through their local optimal particles, and if P_{worse} wins, we will switch P_{better} and P_{worse} . After the competition, two subpopulations' guiding matrices $\text{GM}_{\text{better}}$ and GM_{worse} are calculated by summing all the subpopulation's particles' matching matrices together. Finally, we use $\text{GM}_{\text{better}}$ and GM_{worse} to, respectively, reinitialize 10% and 50% particles in P_{better} and P_{worse} according to Algorithm 1. When reaching the maximum iteration number $\max T = 3000$, the algorithm terminates and returns $\text{MM}_{\text{global,better}}^*$.

5. Experiment

5.1. Experimental Setup. In this work, ontology alignment evaluation initiative (OAEI)'s benchmark and three real sensor ontologies are used to test CBPSO's performance. In Tables 1–3, we compare CBPSO with five SI-based sensor ontology matching techniques, i.e., differential evolution algorithm (DE) [9], co-evolutionary algorithm (cEA) [10], evolutionary tabu search algorithm (ETSA) [11], co-firefly algorithm (cFA) [16], and simulated annealing particle swarm optimization (SA-PSO) [18] in terms of recall, precision, and f-measure [25], respectively. Their results are the average of thirty independent runs. Table 4 briefly describes OAEI's benchmark and three real sensor ontologies.

5.2. Statistical Experiment. We utilize statistical testing method T -test [26] to compare different competitors' performance in terms of recall, precision, and f-measure, respectively. Tables 1 and 2 show six SI-based sensor ontology matching techniques' mean, recall, precision, and f-measure and the corresponding standard deviation on all the testing cases, and Table 3 presents the t -value on recall, precision, and f-measure.

```

(1) *** Initialization **
(2) Initialize two subpopulations  $P_{\text{better}}$  and  $P_{\text{worse}}$ ;
(3) Initialize  $P_{\text{better}}$  and  $P_{\text{worse}}$ 's global optima,  $MM_{\text{global,better}}^*$  and  $MM_{\text{global,worse}}^*$ ;
(4) Initialize each particle's local optima  $MM_{\text{local}}^*$ ;
(5) generation  $t = 0$ ;
(6) while  $t < \max T$ 
(7)   update each particle according to equations (3)–(8);
(8)   update each particle's local optima  $MM_{\text{local}}^*$ ;
(9)   update  $MM_{\text{global,better}}^*$  and  $MM_{\text{global,worse}}^*$ ;
(10)  *** Competition **
(11)  [winner, loser] = compete( $MM_{\text{global,better}}^*$ ,  $MM_{\text{global,worse}}^*$ );
(12)  if(winner ==  $MM_{\text{global,worse}}^*$ )
(13)    switch ( $P_{\text{better}}$ ,  $P_{\text{worse}}$ );
(14)  end if
(15)  update  $P_{\text{better}}$ 's guiding matrix  $GM_{\text{better}}$ ;
(16)  update  $P_{\text{worse}}$ 's guiding matrix  $GM_{\text{worse}}$ ;
(17)  *** Trade off Exploitation and Exploration **
(18)  use  $GM_{\text{better}}$  to randomly reinitialize 10% particles in  $P_{\text{better}}$ ;
(19)  use  $GM_{\text{worse}}$  to randomly reinitialize 50% particles in  $P_{\text{worse}}$ ;
(20)   $t = t + 1$ ;
(21) end while
(22) return  $MM_{\text{global,better}}^*$ 

```

ALGORITHM 2: Competitive binary particle swarm optimization algorithm.

TABLE 1: Comparison among swarm intelligence-based sensor ontology matching techniques in terms of alignment's quality. 1XX, 2XX, and 3XX refer to those starting with number 1, 2, and 3, respectively, and the symbols f , r , and p , respectively, stand for f-measure, recall, and precision.

Testing case	cEA $f(r, p)$	ETSA $f(r, p)$	cFA $f(r, p)$	DE $f(r, p)$	SA-PSO $f(r, p)$	CBPSO $f(r, p)$
1XX	1.00 (1.00, 1.00)	0.81 (0.72, 0.90)	0.95 (0.94, 0.95)	1.00 (1.00, 1.00)	1.00 (1.00, 1.00)	1.00 (1.00, 1.00)
2XX	0.92 (0.91, 0.93)	0.72 (0.84, 0.61)	0.82 (0.79, 0.86)	0.91 (0.88, 0.94)	0.73 (0.72, 0.74)	0.96 (0.96, 0.96)
3XX	0.78 (0.81, 0.74)	0.42 (0.37, 0.48)	0.71 (0.67, 0.77)	0.89 (0.85, 0.94)	0.82 (0.85, 0.79)	0.93 (0.91, 0.95)
MMI-SSN	0.90 (0.86, 0.94)	0.92 (0.90, 0.95)	0.92 (0.90, 0.95)	0.94 (0.92, 0.95)	0.88 (0.90, 0.87)	0.96 (0.94, 0.97)
CSIRO-SSN	0.92(0.89, 0.95)	0.94 (0.94, 0.95)	0.94 (0.94, 0.94)	0.94 (0.94, 0.95)	0.90 (0.88, 0.93)	0.96 (0.95, 0.96)
MMI-CSIRO	0.86 (0.88, 0.91)	0.90 (0.87, 0.94)	0.90 (0.87, 0.94)	0.92 (0.90, 0.93)	0.90 (0.87, 0.94)	0.93 (0.94, 0.93)
Average	0.89 (0.88, 0.92)	0.78 (0.77, 0.80)	0.87 (0.85, 0.90)	0.93 (0.91, 0.95)	0.87 (0.87, 0.87)	0.96 (0.95, 0.96)

TABLE 2: Comparison in terms of standard deviation. 1XX, 2XX, and 3XX refer to those starting with number 1, 2, and 3, respectively, and the symbols f_d , r_d , and p_d , respectively, stand for the standard deviation of f-measure, recall, and precision.

Testing case	cEA $f_d(r_d, p_d)$	ETSA $f_d(r_d, p_d)$	cFA $f_d(r_d, p_d)$	DE $f_d(r_d, p_d)$	SA-PSO $f_d(r_d, p_d)$	CBPSO $f_d(r_d, p_d)$
1XX	0.01 (0.01, 0.01)	0.03 (0.02, 0.02)	0.01 (0.03, 0.02)	0.01 (0.01, 0.01)	0.01 (0.01, 0.01)	0.01 (0.01, 0.01)
2XX	0.01 (0.02, 0.01)	0.04 (0.02, 0.04)	0.02 (0.01, 0.03)	0.02 (0.02, 0.02)	0.03 (0.02, 0.03)	0.01 (0.02, 0.01)
3XX	0.02 (0.01, 0.03)	0.05 (0.02, 0.04)	0.02 (0.01, 0.03)	0.01 (0.02, 0.01)	0.02 (0.02, 0.01)	0.02 (0.01, 0.02)
MMI-SSN	0.01 (0.02, 0.02)	0.03 (0.03, 0.02)	0.03 (0.03, 0.02)	0.02 (0.03, 0.01)	0.04 (0.03, 0.02)	0.01 (0.01, 0.01)
CSIRO-SSN	0.03 (0.01, 0.01)	0.03 (0.02, 0.01)	0.03 (0.02, 0.02)	0.05 (0.03, 0.02)	0.03 (0.02, 0.02)	0.02 (0.01, 0.02)
MMI-CSIRO	0.03 (0.02, 0.01)	0.03 (0.01, 0.01)	0.02 (0.02, 0.03)	0.01 (0.01, 0.01)	0.04 (0.02, 0.04)	0.02 (0.02, 0.01)

In Tables 1 and 2, alignments determined by CBPSO are much better than other sensor ontology matching technique. Due to subpopulations' competitive mechanism, CBPSO is able to better trade off the algorithm's exploitation and exploration, which ensures not only the solution's quality but also the

algorithm's stability. In Table 3, the degree of freedom is 61 (two samples' scales are both 33), the significant level is 5%, and except those results that are the same, the p -value of the rest are all smaller than 0.005, which show that CBPSO significantly outperforms other competitors on 5% significant level.

TABLE 3: T -test's t -value. 1XX, 2XX, and 3XX refer to those starting with number 1, 2, and 3, respectively, and the symbols f_t , r_t , and p_t , respectively stand for the t -value on f -measure, recall, and precision.

Testing case	(cEA, CBPSO) $f_t(r_t, p_t)$	(ETSA, CBPSO) $f_t(r_t, p_t)$	(cFA, CBPSO) $f_t(r_t, p_t)$	(DE, CBPSO) $f_t(r_t, p_t)$	(SA-PSO, CBPSO) $f_t(r_t, p_t)$
1XX	0 (0, 0)	-180.24 (-375.65, -134.16)	-106.06 (-56.92, -67.08)	0 (0, 0)	0 (0, 0)
2XX	-84.85 (-53.03, -63.63)	-174.62 (-127.27, -254.66)	-187.82 (-228.07, -94.86)	-67.08 (-84.85, -26.83)	-218.19 (-254.55, -208.71)
3XX	-159.09 (-212.13, -174.73)	-284.11 (-724.48, -315.28)	-233.34 (-509.11, -149.76)	-53.66 (-80.49, -13.41)	-116.67 (-80.49, -214.66)
MMI-SSN	-127.27 (-107.33, -40.24)	-37.94 (-37.94, -26.832)	-37.94 (-37.94, -26.83)	-26.83 (-18.97, -42.42)	-58.20 (-37.94, -134.16)
CSIRO-SSN	-33.28 (-127.27, -13.41)	-16.64 (-13.41, -13.41)	-16.64 (-13.41, -21.21)	-11.14 (-9.48, -10.60)	-49.92 (-93.91, -31.81)
MMI-CSIRO	0 (0, 0)	-180.24 (-375.65, -134.16)	-106.06 (-56.92, -67.08)	0 (0, 0)	0 (0, 0)

TABLE 4: Descriptions on the ontologies in the testing cases.

Testing case	Ontology	Scale
OAEI's benchmark	Bibliographic ontology	97 entities
	CSIRO sensor ontology (CSIRO)	33,205 entities
Real sensor ontology	Semantic sensor network ontology (SSN)	32,298 entities
	MMI device ontology (MMI)	24,034 entities

6. Conclusion

Building sensor ontologies and using them to annotate the sensor data is a feasible way to address the IoT data heterogeneity issue. However, before using these ontologies, we need to address the heterogeneity problem among them. Recently, SI-based matching techniques have been emerging as a popular method of matching heterogeneous sensor ontologies, and this work proposes a CBPSO-based sensor ontology matching technique. In particular, CBPSO uses the GM to ensure the diversity of the population, and two competitive subpopulations to trade off the algorithm's exploitation and exploration. The experiment compares CBPSO with five SI-based sensor ontology matching techniques, and the experimental results show that CBPSO's results outperform the competitors.

In the future, we will further improve CBPSO by introducing more subpopulations with different searching strategies. In addition, the competing mechanism can be further improved by referring to binary fish migration optimization algorithm (BFMO) [27]. We are also interested in further improving CBPSO to match biomedical ontologies [28], which is the large-scale matching task. Some efficiency-improving strategies will be used, such as the compact encoding, and the efficient compact evolutionary paradigm will be one of our future studies.

Data Availability

The data used to support this study can be found at <http://oaei.ontologymatching.org>.

Conflicts of Interest

The authors declare that they have no conflicts of interest in the work.

Acknowledgments

This work was supported in part by the Scientific Research and Technology Development Project of Yulin City (no. 20204012), Natural Science Foundation of Guangxi Province (no. 2021GXNSFAA220076), and Science Research Foundation for High-level talents of Yulin Normal University (no. G2021ZK17).

References

- [1] P. Pande and A. Padwalkar, "Internet of things—A future of Internet: a survey," *International Journal of Advance Research in Computer Science and Management Studies*, vol. 2, pp. 354–361, 2014.
- [2] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific American*, vol. 284, no. 5, pp. 34–43, 2001.
- [3] H. Neuhaus and M. Compton, "The semantic sensor network ontology," in *Proceedings of the AGILE Workshop on Challenges in Geospatial Data Harmonisation*, pp. 1–33, Hanover, Germany, June 2009.
- [4] C. Rueda, N. Galbraith, R. A. Morris, L. Bermudez, R. Arko, and J. Graybeal, "The MMI device ontology: enabling sensor integration," in *Proceedings of the AGU Fall Meeting Abstracts*, pp. 1–35, San Francisco, CA, USA, December 2010.
- [5] M. Compton, P. Barnaghi, L. Bermudez et al., "The SSN ontology of the W3C semantic sensor network incubator group," *Journal of Web Semantics*, vol. 17, pp. 25–32, 2012.
- [6] P. Shvaiko and J. Euzenat, "Ontology matching: state of the art and future challenges," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 1, pp. 158–176, 2011.
- [7] H. Jiang, Q. Li, Y. Jiang et al., "When machine learning meets congestion control: a survey and comparison," *Computer Networks*, vol. 192, pp. 1–28, 2021.
- [8] S. Fernandez, I. Marsa-Maestre, J. Velasco, and B. Alarcos, "Ontology alignment architecture for semantic sensor web integration," *Sensors*, vol. 13, no. 9, Article ID 12581, 2013.
- [9] X. Xue and C. Jiang, "Matching sensor ontologies with multi-context similarity measure and parallel compact differential evolution algorithm," *IEEE Sensors Journal*, vol. 21, no. 21, Article ID 24570, 2021.
- [10] X. Xue and J.-S. Pan, "A compact Co-evolutionary algorithm for sensor ontology meta-matching," *Knowledge and Information Systems*, vol. 56, no. 2, pp. 335–353, 2018.
- [11] X. Xue and J. Chen, "Using compact evolutionary tabu search algorithm for matching sensor ontologies," *Swarm and Evolutionary Computation*, vol. 48, pp. 25–30, 2019.
- [12] D. de Werra and A. Hertz, "Tabu search techniques," *Operations-Research-Spektrum*, vol. 11, no. 3, pp. 131–141, 1989.
- [13] A. Maratea, A. Petrosino, and M. Manzo, "Adjusted F-measure and kernel scaling for imbalanced data learning," *Information Sciences*, vol. 257, pp. 331–341, 2014.
- [14] X. Xue and Y. Wang, "Optimizing ontology alignments through a memetic algorithm using both MatchFmeasure and unanimous improvement ratio," *Artificial Intelligence*, vol. 223, pp. 65–81, 2015.
- [15] X. Xue, X. Wu, C. Jiang, G. Mao, and H. Zhu, "Integrating sensor ontologies with global and local alignment extractions," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6625184, 10 pages, 2021.
- [16] X. Xue, X. Wu, and J. Chen, "Optimizing biomedical ontology alignment through a compact multiobjective particle swarm optimization algorithm driven by knee solution," *Discrete Dynamics in Nature and Society*, vol. 2020, no. 7, 10 pages, Article ID 4716286, 2020.
- [17] H. Zhu, X. Xue, A. Geng, and H. Ren, "Matching sensor ontologies with simulated annealing particle swarm optimization," *Mobile Information Systems*, vol. 2021, Article ID 5510055, 11 pages, 2021.
- [18] D. Bertsimas and J. Tsitsiklis, "Simulated annealing," *Statistical Science*, vol. 8, no. 1, pp. 10–15, 1993.
- [19] X. Xue and J. Zhang, "Matching large-scale biomedical ontologies with central concept based partitioning algorithm and

- adaptive compact evolutionary algorithm,” *Applied Soft Computing*, vol. 106, pp. 1–11, 2021.
- [20] X. Xue and J. Chen, “Matching biomedical ontologies through compact differential evolution algorithm with compact adaption schemes on control parameters,” *Neurocomputing*, vol. 458, pp. 526–534, 2021.
- [21] J. Martinez-Gil and J. F. Aldana-Montes, “Evaluation of two heuristic approaches to solve the ontology meta-matching problem,” *Knowledge and Information Systems*, vol. 26, no. 2, pp. 225–247, 2011.
- [22] X. Xue and Y. Wang, “Using memetic algorithm for instance coreference resolution,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 2, pp. 580–591, 2016.
- [23] X. Xue and Q. Huang, “Generative adversarial learning for optimizing ontology alignment,” *Expert Systems*, vol. 2022, pp. 1–12, 2022.
- [24] X. Xue, P.-W. Tsai, and Y. Zhuang, “Matching biomedical ontologies through adaptive multi-modal multi-objective evolutionary algorithm,” *Biology*, vol. 10, pp. 1–16, 2021.
- [25] G. Hripcsak and A. Rothschild, “Agreement, the f-measure, and reliability in information retrieval,” *Journal of the American Medical Informatics Association*, vol. 12, no. 3, pp. 296–298, 2005.
- [26] D. Semenick, “Tests and measurements: the T-test,” *National Strength & Conditioning Association Journal*, vol. 12, no. 1, pp. 36–37, 1990.
- [27] J. S. Pan, P. Hu, and S. C. Chu, “Binary fish migration optimization for solving unit commitment,” *Energy*, vol. 226, pp. 1–11, 2021.
- [28] X. Xue, J. Lu, and J. Chen, “Using NSGA-III for optimising biomedical ontology alignment,” *CAAI Transactions on Intelligence Technology*, vol. 4, no. 3, pp. 135–141, 2019.