Ontology Alignment Optimization Method Based on NSGA-II

Shaorong Feng

School of Information Science and Engineering, Xiamen University, Xiamen 361005, China

shaorong@xmu.edu.cn

Keywords: Ontology; Alignment; NSGA-II; Aggregation of similarity measure

Abstract: In this paper, we propose a novel approach based on NSGA-II to address the problem of optimizing the aggregation of three different basic similarity measures (SyntacticMeasure, Linguistic Measure and Taxonomy-based Measure). Comparing with conventional genetic algorithm, the proposed method is able to realize three goals simultaneously, i.e. maximizing the alignment recall, the alignment precision and the f-measure, andthe resulted ontology alignment could avoid bias to recall or precision value. Experimental results show that the proposed approach is effective.

1. Introduction

Because ontology enables data and knowledge to be shared and reused more effectively, it is widely used in the interaction between heterogeneous information sources in the Semantic Web. However, due to human subjectivity, different ontologies in the same application domain may define the same entity object in different names or ways, which leads to the so-called ontology heterogeneity problem. To solve this problem, we need to determine the corresponding relationship between entities in different ontologies, which is often called ontology mapping.

When the scale of ontology is large, it is unrealistic to determine the corresponding relationship between entities in ontology manually. Therefore, in recent years, a large number of ontology mapping systems have been developed to determine the similarity values between entities in different ontologies by automation or semi-automation, and to further determine whether the two entities are semantically similar by using these similarity values. Since no similarity measurement technology can independently provide satisfactory measurement results, most ontology mapping systems use the results of integrating different similarity measurement technologies to obtain the final mapping results to improve the quality of ontology mapping results.[1-4]In ontology mapping process, how to determine the integration weight ofappropriate similarity measurement results and the filtering threshold of mapping results to obtain satisfactory ontology mapping results is called meta-matching problem. This problem can be regarded as an optimization problem and solved by techniques such as genetic algorithm. However, the current meta-matching method based on genetic algorithm usually determines the weight of ontology mapping system by means of single objective,[5-9]This may lead to the final mapping result tending to prefer the evaluation value of a certain mapping result (such as recall or precision). In order to solve this problem, this paper proposes to use NSGA-II (Non-dominated Sorting Genetic Algorithms-II) algorithm to determine the set of parameters of the system, in order to achieve the purpose of optimizing ontology mapping results. In the method proposed in this paper, we can optimize three objectives at the same time, namely, maximizing the recall rate, precision rate and f-measure value of the mapping, and the ontology mapping results obtained can avoid the preference of recall rate or precision rate to the greatest extent.

NSGA-II is considered as a flexible and robust optimization algorithm, which is good at quickly determining the non-dominant solutions of various problems.Firstly, in the evolutionary process, the algorithm uses standard crossover and mutation operators to perform evolutionary operations on the population. Then, the algorithm uses fast non-dominated sorting technology and congestion distance sorting to select the next generation of population. Finally, the solution of the problem is

  DOI: 10.25236/isrme.2019.019

determined by considering the factors of non-dominance and diversity. Therefore, NSGA-II is very suitable for integrating different similarity values and obtaining a variety of globally non-dominant optimal ontology mapping results.

2. Basic concepts

2.1. Ontology and Ontology Mapping.

There are many definitions of ontology, but the most frequently cited one is Gruber's standardized explanation of defining ontology as a clear concept in 1993. In this article, for the sake of introduction, we define ontology as follows:

Definition 1. Ontology is a triple $O=(C, I, P)$,

Among them: C is the set of concepts; P is the set of attributes, that is, the set of relations between concepts in thefield; I is the set of instances, that is, the set of objects in the real world. Concepts, attributes and instances are collectively referred to as entities in ontology.

Ontology is regarded as a solution to the problem of data heterogeneity in Semantic Web. However, heterogeneity may exist among different ontologies. Ontology mapping is a method to solve heterogeneous problems among ontologies. The results of ontology mapping can be defined as follows:

Definition 2.The mapping result between ontologies is a set of mapping elements. The mapping element is a quaternion (e, e', n, R),

Among them: e and e' are entities in two ontologies respectively; n is the value of similarity between entities to e and e' obtained by some similarity measure technology (usually in the range of [0,1]); R is the semantic relationship between entities to e and e'(usually equivalent).

In addition, the ontology mapping process can be defined as follows:

Definition 3. Ontology mapping process can be regarded as a function φ, given two mapping ontologies O and O', an existing mapping result A_I, a parameter set P, and an external resource set R, returning a new mapping result between ontologies:

$$A'=\varphi(O, O', A_I, P, R)$$

Ontology mapping process calculates the similarity value of mapping elements by similarity measurement technology. The similarity value of 0 represents two entity of mapping elements are completely different, and the similarity value of 1 represents two entity of mapping elements are equivalent.

2.2. Similarity Measurement Technology.

Generally speaking, similarity measurement technology can be divided into three categories: grammar-based similarity measurement technology, linguistic-based similarity measurement technology and taxonomic-based similarity measurement technology.

2.2.1. Grammar-based similarity measurement technology

The grammatical similarity measurement technique calculates the editing distance between two strings, the most commonly used of which is *Levenstein* distance. Specifically, the *Levenstein* distance between the two strings s_1 and s_2 is defined as follows[10]:

$$Levenstein(s_1, s_2) = \max(0, \frac{\min(|s_1|, |s_2|) - d(s_1, s_2)}{\min(|s_1|, |s_2|)})$$

Among them, $|s_1|$ and $|s_2|$ are the lengths of strings s_1 and s_2, respectively. $d(s_1, s_2)$ is the minimum number of operations needed to convert s_1 into s_2.

2.2.2. Linguistic-based similarity measurement technology

Linguistic metrics calculates the similarity of two strings by considering linguistic relationships (such as synonyms, epistasy, etc.). In the work of this paper, WordNet[11](An Electronic Language

Database Covering a Collection of Synonyms of Various Vocabularies)It is used to calculate the similarity measure of string based on synonyms.Given two words $w_1$and w_2 , the linguistic similarity measure between them *LinguisticMeasure*(w1, w2) is equal to:

$$LinguisticMeasure(w_1, w_2)= \begin{cases} 1 & \text{If w1 and w2 are synonyms;} \\ 0.5 & \text{If w1 is the superposition of w2, vice versa;} \\ 0 & \text{Other circumstances.} \end{cases}$$

2.2.3. Similarity measurement technology based on Taxonomy

The similarity measurement technology based on Taxonomy utilizes the similarity value between neighbors of entity pairs in different ontologies to calculate the similarity value of entity pairs. For example, if the concepts of a pair of concepts are similar between the concepts of parent or child or brother, then the concepts should also be similar. Assuming that c_1 and c_2 are two concepts of ontology O_1 and O_2 respectively, s_1 and s_2 are parent or child concepts of c_1 and c_2 respectively. If there is a mapping $c=(s_1, s_2)$ between s_1 and s_2, and the similarity value of the mapping is n, then $TaxonomyMeasure\ (c_1, c_2) = f\ (c)$.

In this paper, the weighted average integration method is used to integrate the above similarity measurement results. The method is defined as follows:

$$\theta(\vec{s}(c), \vec{w}) = \sum_{i=1}^{n} w_i s_i(c) \quad with \quad \sum_{i=1}^{n} w_i = 1 \quad and \quad w_i \in [0,1]$$

Among them, $\vec{s}(c)$ is the result vector of similarity measurement technology, $\vec{w}$ is the weight vector, and n is the number of similarity measurement technology used by mapping system. () scw n

Since the quality of mapping results (i.e. correctness and completeness) needs to be evaluated, then some evaluation methods of ontology mapping results quality are introduced. These methods are derived from the field of information retrieval.

2.3. Mapping Result Evaluation.

The quality of ontology mapping results is usually evaluated by recall and precision. Recall ratio (also known as integrity) is used to measure the proportion of the correct mapping results found to all the correct results. The value of recall ratio of 1 means that all the correct mapping results have been found. However, recall does not provide the number of mapping errors found in the mapping results. Therefore, recall rate needs to be considered together with precision rate (also known as correctness), which is used to measure the proportion of the correct mapping results in the found mapping results. The accuracy of 1 means that all the mapping results found are correct, but this does not mean that all the correct mapping results have been found. Therefore, recall rate and precision rate must be considered at the same time, which can be achieved by f-measure (that is, weighted harmonic mean of recall rate and precision rate).

Given a reference mapping R and a mapping result A, *recall*, *precision* and *f-measure* can be calculated by the following formulas:

$$recall = \frac{|R \cap A|}{|R|}$$

$$precision = \frac{|R \cap A|}{|A|}$$

$$f - measure = 2 \cdot \frac{recall \cdot precision}{recall + precision}$$

3. NSGA-II algorithm

Given two ontologies as input, before using NSGA-II to optimize ontology mapping results, we first save the ontology mapping results obtained by a single similarity measure technology in an XML file. The purpose of this method is to avoid the repeated calculation of entity similarity values

obtained by a similarity measurement technology in the operation of NSGA-II, so as to improve the efficiency of the algorithm. Following are four basic steps of NSGA-II algorithm for optimizing ontology mapping results.

3.1. Individual Coding.

Individual coding includes the weights of similarity measurement technology and threshold information used to filter the final results. Therefore, the coding of an individual can be divided into

two parts: one represents the integrated weight of similarity measurement technology, and the other represents the filtering threshold of mapping results. According to the characteristics of the weights mentioned in section 1.2, we indirectly represent them by defining segmentation points in the [0,1] interval. For example, assuming that p is the number of weights required, the set of partitioning points can be expressed as $c'=\{c'_1, c'_2,... c'_{p-1}\}$. The process of individual decoding is as follows: Firstly, the elements in c' are sorted in descending order, and $c=\{c_1,c_2,\ldots,c_{p-1}\}$ is obtained; Then the weight values are calculated according to the following formulas:

$$w_k=\begin{cases} c_1, & k=1 \\ c_k-c_{k-1} & 1<k<p \\ 1-c_{p-1} & \end{cases}$$

Therefore, the total length of individual coding is (n-1)*$cutLength$+$thresholdLength$, in which the number of weights is n. $cutLength$ and $thresholdLength$ are the coding length of segmentation points and the coding length of threshold respectively.

3.2. Fitness Function.

Fitness function is an objective function used to evaluate the result of ontology mapping obtained by using the weight and threshold of coding in the individual. In this paper, two objective functions are used to evaluate the recall and precision of ontology mapping results.

3.3. Genetic operators

3.3.1. Selection operator

Like natural selection, the best individuals should have more opportunities to replicate themselves to the next generation. Individuals with the best population have the best fitness values and genetic information, which can potentially provide the best solution to the problem. However, in order to ensure the diversity of individuals in the population, individuals with low fitness should not be completely deprived of the opportunity to replicate themselves. In this paper, in order to ensure the diversity of the population and accelerate the convergence speed of the algorithm, the selection operator first ranks the individuals in the population in descending order according to the fitness value, and randomly selects and replicates the first 1/2 individuals at atime until a new population is formed.

3.3.2. Crossover operator

The crossover operator selects two paternal individuals to produce two offspring, which is accomplished by mixing the genes of the paternal individuals. Crossing operation occurs under certain probability, which is one of the parameters of genetic algorithm. In this paper, we use the common single-point crossover method for population. Firstly, a crossing point is randomly selected in the parent, and then the second half of the crossing point is exchanged to form two sub-individuals.

3.3.3. Mutation operator

The mutation operator ensures the diversity of the population and avoids premature convergence. In this paper, we first determine the mutation bit of individual coding, and change the value of thebit from 0 to 1 or from 1 to 0 when performing mutation operation.

3.4. Generating the Next Generation of Population.

First, we put the current population together with the newly generated population and remove the duplicate individuals. Later, the new population is obtained by non-dominant ranking and congestion calculation. See details[12-14].

When the algorithm terminates, we propose a selection strategy to select the representative solution, that is, the solution with the best recall rate, the solution with the best precision rate and the solution with the best *f-measure*. The specific method is that for all the solutions with the best recall rate, we choose the solution with the highest recall rate as the representative solution. Similarly, among all the solutions with the highest recall rate, the solution with the highest recall rate is chosen as the representative solution. Among all the solutions with the highest *f-measure* value, we use the max-min method to obtain the representative solution. Assuming that $x_1, x_2,\ldots, x_k$ is the set of solutions with the highest *f-measure* value in all solutions, the values of their precision and recall can be expressed as $f_r(x_i)$ as $f_p(x_i)$ and i=1~ k , respectively. We choose a better solution according to the following formula:

$$x_j = \arg\max_i \{\min(f_r(x_i), f_p(x_i))\}.$$

Next, we compare the ontology mapping results obtained by traditional genetic algorithm with the results of the proposed method through experiments.

4. Experimental results and analysis

In the experiment, we used the well-known test case set of OAEI (Ontology Alignment Evaluation Initiative) 2012[15]. Each test case in the OAEI test case set consists of two mapping ontologies and a reference mapping result for evaluating the mapping results. Table 1 provides a brief description of the OAEI 2012 test case.

Table 1 OAEI 2012 Test Case Set

ID	Case description
101	Strictly identical ontologies
103	A regular ontology and other with a language generalization
104	A regular ontology and other with a language restriction
201	Ontologies without entity names
203	Ontologies without entity names and comments
204	Ontologies with different naming conventions
205	Ontologies whose labels are synonymous
206	Ontologies whose labels are in different languages
221	A regular ontology and other with no specialisation
222	A regular ontology and other with a flatenned hierarchy
223	A regular ontology and other with a expanded hierarchy
224	Identical ontologies without instances
225	Identical ontologies without restrictions
228	Identical ontologies without properties
230	Identical ontologies with flattening entities
231	Identical ontologies with multiplying entities
301	A real ontology about bibliography made by MIT
302	A real ontology with different extensions and naming conventions

4.1. Experimental Configuration.

In the experiment, the traditional genetic algorithm and NSGA-II adopted the following parameters:

- The search space of each parameter is continuous interval[0, 1];
- Numerical accuracy= 0.01;
- The fitness value of traditional genetic algorithm is recall rate, precision rate or *f-measure*, while the fitness function of NSGA-II is recall rate and precision rate.
- Population size = 20 individuals;

- Crossover probability=0.6;
- Mutation probability=0.01;
- Maximum evolutionary algebra = 5 generations. During 30 independent runs, we observed that the results of the algorithm did not improve after 5 generations, so we set the upper limit of evolutionary algebra to 5 generations.

4.2. Experimental Results and Analysis.

Tables 2 and 3 describe the average results of traditional genetic algorithm and NSGA-II driven by recall, precision and *f-measure* in 30 independent runs respectively. Table 2 gives the results of traditional genetic algorithm driven by recall and precision. Columns 2 and 4 are the solutions of traditional genetic algorithm driven by recall and precision respectively. Columns 3 and 5 are the representative solutions of best recall and best precision in NSGA-II, respectively. Table 3 gives the solution of the genetic algorithm driven by *f-measure* and the representative solution of the best F-measure obtained by NSGA-II. In Tables 2 and 3, symbols R and P represent recall and precision respectively.

As can be seen from Table 2, except for test case 205, the best recall representative solutions of NSGA-II are better than those of recall-driven genetic algorithm. For example, in test case 201, the recall rate of the solution obtained by NSGA-II is higher than that of the recall-driven genetic algorithm, while in test case 222, although the recall rate of both is the same, the best recall rate of NSGA-II represents that of the solution driven by the recall rate is higher than that of the solution driven by the genetic algorithm. In addition, the best precision representative solution obtained by NSGA-II is superior to that obtained by genetic algorithm driven by precision in almost all test cases except test cases 206, 224 and 228. For example, in test case 103, although the precision is the same, the recall rate of solution obtained by NSGA-II is higher than that of genetic algorithm driven by precision.

Compared with genetic algorithm which only considers recall rate or recall rate, NSGA-II considers both precision rate and recall rate, so it can find better solution to ontology matching problem than traditional genetic algorithm.

In Table 3, except for test cases 205, 301 and 302, the quality of the solution of the genetic algorithm driven by F-measure is the same as that of the solution of NSGA-II. In test case 205, the F-measure value of the solution of NSGA-II is higher than that of the solution driven by F-measure genetic algorithm. In test cases 301 and 302, according to max-min method, we can judge that the solution obtained by NSGA-II is superior to that obtained by genetic algorithm driven by f-measure.

Table 2 Comparison of genetic algorithm and NSGA-II based on recall and precision

ID	R(P) (GA)	R(P) (NSGA-II)	P(R) (GA)	P(R) (NSGA-II)
101	1.00 (0.78)	1.00 (1.00)	1.00 (0.01)	1.00 (1.00)
103	1.00 (0.68)	1.00 (1.00)	1.00 (0.98)	1.00 (1.00)
104	1.00 (0.65)	1.00 (1.00)	1.00 (0.99)	1.00 (1.00)
201	0.95 (0.04)	0.98 (0.03)	1.00 (0.01)	1.00 (0.31)
203	1.00 (0.61)	1.00 (0.80)	1.00 (0.83)	1.00 (0.98)
204	1.00 (0.13)	1.00 (0.23)	1.00 (0.74)	1.00 (0.93)
205	0.98 (0.03)	0.98 (0.03)	1.00 (0.21)	1.00 (0.48)
206	0.7 (0.03)	0.73 (0.03)	1.00 (0.23)	1.00 (0.23)
221	1.00 (0.52)	1.00 (1.00)	1.00 (0.99)	1.00 (1.00)
222	1.00 (0.75)	1.00 (1.00)	1.00 (0.99)	1.00 (1.00)
223	1.00 (0.27)	1.00 (0.78)	1.00 (0.96)	1.00 (0.98)
224	1.00 (0.63)	1.00 (1.00)	1.00 (1.00)	1.00 (1.00)
225	1.00 (0.78)	1.00 (1.00)	1.00 (0.01)	1.00 (1.00)
228	1.00 (0.68)	1.00 (1.00)	1.00 (0.98)	1.00 (1.00)
230	1.00 (0.65)	1.00 (1.00)	1.00 (0.99)	1.00 (1.00)
231	0.95 (0.04)	0.98 (0.03)	1.00 (0.01)	1.00 (0.31)
301	1.00 (0.61)	1.00 (0.80)	1.00 (0.83)	1.00 (0.98)
302	1.00 (0.13)	1.00 (0.23)	1.00(0.74)	1.00(0.93)

Table 3 Comparison of genetic algorithm and NSGA-II based on F-measure

ID	F-measure(R, P) (GA)		F-measure(R,P) (NSGA-II)	
101	1.00	(1.00, 1.00)	1.00	(1.00, 1.00)
103	1.00	(1.00, 1.00)	1.00	(1.00, 1.00)
104	1.00	(1.00, 1.00)	1.00	(1.00, 1.00)
201	0.94	(0.90, 0.98)	0.94	(0.90, 0.98)
203	0.99	(0.98, 1.00)	0.99	(0.98, 1.00)
204	0.98	(0.99, 0.98)	0.98	(0.99, 0.98)
205	0.89	(0.90, 0.89)	0.94	(0.89, 0.99)
206	0.70	(0.67, 0.73)	0.70	(0.67, 0.73)
221	1.00	(1.00, 1.00)	1.00	(1.00, 1.00)
222	1.00	(1.00, 1.00)	1.00	(1.00, 1.00)
223	0.99	(0.98, 1.00)	0.99	(0.98, 1.00)
224	1.00	(1.00, 1.00)	1.00	(1.00, 1.00)
225	1.00	(1.00, 1.00)	1.00	(1.00, 1.00)
228	1.00	(1.00, 1.00)	1.00	(1.00, 1.00)
230	0.99	(1.00, 1.00)	1.00	(1.00, 1.00)
231	1.00	(1.00, 1.00)	1.00	(1.00, 1.00)
301	0.75	(0.73, 0.77)	0.75	(0.75, 0.75)
302	0.71	(0.61, 0.84)	0.71	(0.62, 0.83)

In summary, in the process of optimizing ontology mapping, NSGA-II can find the same or

better solution as traditional genetic algorithm. Because the method of generating new population in NSGA-II can improve the consistency between recall and precision, the solution at non-dominant frontier is obviously superior to other solutions. Therefore, compared with traditional genetic algorithm, NSGA-II increases the chances of finding a better solution.

5. Conclusion

Ontology mapping is one of the important steps to build ontology in ontology engineering. Although researchers have done a lot of work, there are still many important problems unsolved. One of the problems is how to integrate different similarity measures. To solve this problem, an ontology mapping optimization method based on NSGA-II is proposed. The experimental results show that the proposed method can automatically adjust the parameters in ontology mapping process and find solutions that are better than or equal to the traditional genetic algorithm.

Subsequent research work is transplanting NSGA-II method to real ontology mapping system. In addition, we are also considering developing an expert decision support system to assist the ontology mapping system to automatically adjust the parameters of the system.

References

[1] Do H H, Rahm E. COMA-a system for flexible combination of schema matching approaches [J]. Proceedings of the 28th International VLDB Conference. 2002, 610-621.

[2] Aumueller D, Do H H, Massmann S. Schema and ontology matching with COMA++ [J]. Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, 2005, 906-908.

[3] Drumm C, Schmitt M, Do H H. Quickming: automatic schema matching for data migration projects [J]. Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management, 2007, 107-116.

[4] Gal A, Anaby-Tavor A Trombetta A. A framework for modeling and evaluating automatic semantic reconciliation [J]. VLDB Journal, 2005, 14(1): 50-67.

[5] Martinez-Gil J, Alba E, Aldana-Montes J F. Optimizing ontology alignments by using genetic algorithms [J]. Proceedings of the workshop on nature based reasoning for the semantic Web, 2008, 419.

[6] Naya J M V, Romero M M, Loureiro J P. Improving ontology alignment through genetic algorithms [J]. Soft Computing Methods for Practical Environment Solutions: Techniques and Studies, 2010, 240-259.

[7] Ginsca A-L, Iftene A. Using a genetic algorithm for optimizing the similarity aggregation step in the process of ontology alignment [J]. 9th Roedunet Int Conference (RoEduNet), 2010, 118-122.

[8] Acampora G., Loia V. and Salerno S. A Hybrid Evolutionary Approach for Solving the Ontology Alignment Problem. International Journal Of Intelligent Systems, 2012, 27(3): 189-216.

[9] Euzenat J, Valtchev P. Similarity-based ontology alignment in OWL-Lite [J]. Proceedings of 16th European Conference on Artificial Intelligence, 2004, 333-337.

[10] Maedche A, Staab S. Measuring Similarity between Ontologies [J]. Proceedings of the International Conference on Knowledge Engineering and Knowledge Management (EKAW), 2002, 251-263.

[11] Miller, G. A.. WordNet: A lexical database for English [J]. Communications of the ACM, 1995, 38(11): 39-41.

[12] Deb K, Agrawal S, Pratap A.. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II [J], Proceedings of the Parallel Problem Solving from

Nature VI Conference, 2000, 1917: 849-858.

[13] de Oliveira, Lariza Laura,Freitas Alex A.,Tinós Renato. Multi-objective genetic algorithms in the study of the genetic code's adaptability[J]. Information Sciences, January 2018, v 425, p 48-61.

[14] Nazarahari Milad, Khanmirza Esmaeel, Doostie Samira. Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm[J]. Expert Systems with Applications, January 2019, v115, p 106-120.

[15] Ontology Alignment Evaluation Initiative (OAEI). 2013.6.18. http://oaei.ontologymatching.org/2012.