Computing and Informatics, Vol. 33, 2014, 857–876

OPTIMIZING ONTOLOGY ALIGNMENTS THROUGH NSGA-II WITHOUT USING REFERENCE ALIGNMENT

Xingsi Xue

School of Computer Science and Technology
Xidian University
Xi'an, Shaanxi, 710071 China
&
School of Information Science and Engineering
Fujian University of Technology
Fuzhou, Fujian, 350118 China

Yuping Wang, Weichen Hao, Juan Hou

School of Computer Science and Technology
Xidian University
Xi'an, Shaanxi, 710071 China
e-mail: ywang@xidian.edu.cn

Abstract. Ontology is widely used to solve the data heterogeneity problems on the semantic web, but the available ontologies could themselves introduce heterogeneity. In order to reconcile these ontologies to implement the semantic interoperability, we need to find the relationships among the entities in various ontologies, and the process of identifying them is called ontology alignment. In all the existing matching systems that use evolutionary approaches to optimize their parameters, a reference alignment between two ontologies to be aligned should be given in advance which could be very expensive to obtain especially when the scale of ontologies is considerably large. To address this issue, in this paper we propose a novel approach to utilize the NSGA-II to optimize the ontology alignments without using the reference alignment. In our approach, an adaptive aggregation strategy is presented to improve the efficiency of optimizing process and two approximate evaluation measures, namely match coverage and match ratio, are introduced to replace the classic recall and precision on reference alignment to evaluate the quality of the alignments. Experimental results show that our approach is effective and can find the solutions that are very close to those obtained by the approaches using reference alignment, and the quality of alignments is in general better than that of state of the art ontology matching systems such as GOAL and SAMBO.

Keywords: Ontology alignment, NSGA-II, similarity measures, adaptive similarity aggregation, approximate evaluation measure

1 INTRODUCTION

Ontology is constructed to capture implicit, explicit and commonsense knowledge of a domain such that the knowledge can be shared, reused and consumed by autonomous computer agents [1]. However, because of human subjectivity, the available ontologies could themselves introduce heterogeneity: one entity in different ontologies may be defined with different names or in different ways. In order to reconcile these ontologies to implement semantic interoperability, we need to find the relationships that hold between these entities in various ontologies and the process of identifying them is called ontology alignment.

It is highly impractical to align the ontologies manually especially when the size of ontologies is considerably large. Therefore, several matching systems have arisen over the years. The first ones could use only one or few similarity measure techniques to determine whether the entities from separate ontologies are semantically similar. Since none of these similarity measure techniques could provide the satisfactory results independently, the matching systems of next generation focus on providing a wide suite of basic similarity measure techniques with a specific matching purpose of combining them in a flexible way. Lately, the focus is towards meta-matching, that is finding the best way to combine different basic similarity measure techniques which can be regarded as an optimization problem and be addressed by approaches like Evolutionary Algorithm (EA).

Nevertheless, all the existing evolutionary approaches optimize the parameters of meta-matching system with a prerequisite that a Reference Alignment (RA) between two ontologies to be aligned should be given in advance. Since the number of possible correspondences grows quadratically with the increasing number of entities inside the ontology, the typical approach of manually constructing the reference alignment for large scale matching tasks is infeasible. In this paper, we propose to use the adaptive aggregation strategy and multi-objective evolutionary algorithm (MOEA) to address this problem. To be specific, our contributions are as follows:

- the proposal of the adaptive aggregation strategy to aggregate various alignments obtained by the different similarity measures in order to generate an intermediate alignment;
- the utilization of NSGA-II, which is considered to be a computationally fast elitist MOEA based on non-dominated sorting approach, to find an optimal

threshold for filtering the correspondences in the intermediate alignment to optimize the quality of the ontology alignment. NSGA-II aims to obtain a well distributed approximation set of points that are close to the pareto front. Both, closeness and diversity, are addressed in the selection operator, where the population is sorted using non-domination ranks as primary sorting criterion, and crowding-distances as secondary sorting criterion. With the properties of a fast non-dominated sorting procedure, an elitist strategy and a parameterless approach, NSGA-II is suitable for a wide range of multi-objective problems including ontology alignment optimization problem. To the best of our knowledge, this is the first time to utilize multi-objective evolutionary algorithm to optimize the ontology alignment. The multi-objective optimization model for optimizing the ontology alignment problem is given and details of the problem-specific NSGA-II are presented;

- the introduction of several new evaluation measures of alignment to replace the classic evaluation measures on RA in the process of optimizing the alignments.

The rest of this paper is organized as follows: Section 2 is devoted to present the related work such as the existing ontology matching algorithms; Section 3 provides a detailed description of the basic concepts of ontology and ontology alignment; Section 4 presents the multi-objective optimal model for optimizing multiple ontology alignments with no reference alignment; Section 5 describes the details of the NSGA-II used to address the ontology alignment problem without using reference alignment; Section 6 shows the experimental results of our approach on which we give an analysis; in Section 7, we draw conclusions and propose further improvements.

2 RELATED WORK

In recent years, numerous fully automatic or semi-automatic meta-matching systems have been developed. Meta-matching does not use parameters from the experts, but selects them according to a training benchmark, which is a set of ontologies that have been previously aligned by the experts. Some of those algorithms based on meta-matching techniques have investigated the use of computational intelligence techniques such as EA to implement automatic ontology matching processes.

Among those meta-matching systems using evolutionary algorithm, the most notable system is GOAL [2] (Genetics for Ontology Alignments). Although GOAL does not directly compute the alignment between two ontologies, it determines, through a Genetic Algorithm (GA), the optimal weight configuration for a weighted average aggregation of several similarity measures by considering a given RA. A typical approach is to build it manually; however, when it comes to large scale matching tasks, manual construction of the reference correspondences is infeasible. Hence, some semi-automatic matching systems using Partial Reference Alignment (PRA), i.e. some of the correct mappings between entities are given or have been obtained for tuning parameters, are proposed in a more recent paper. Among those semi-automatic matching systems based on PRA, the most notable one is SAMBO [24]. SAMBO uses PRA as anchors to give hints for partitioning larger ontologies in a pre-processing step, as well as for filtering those incorrect mappings in a post-processing step. Another semi-automatic matching system exploiting PRA and applying machine learning methods is LSD [4]. It asks the user to provide the semantic mappings for a small set of data sources, then uses these mappings together with the sources to train a set of learners. Nevertheless, technique of PRA is not so mature, e.g. it is difficult to build a PRA that is the set of representative sample mappings in RA. Therefore, the requirement of matching ontologies without using RA or PRA, which is still able to obtain the high-quality alignments, is of great urgency.

3 BASIC CONCEPTS

In this section, we first give the definitions of ontology, ontology alignment and ontology alignment process. Then the used similarity measures, i.e. the syntactic measure, the linguistic measure and the taxonomy based measure, are introduced. After that, an adaptive similarity aggregation strategy called Harmony is presented. Finally, the metrics we used to evaluate the quality of the alignments are introduced.

3.1 Ontology, Ontology Alignment and Ontology Alignment Process

There are many definitions of ontology, but the most frequently referenced one was given by Gruber in 1993. According to Gruber's definition, an ontology is an explicit specification of a conceptualization. For the convenience of our work, an ontology is defined as follows [6]:

Definition 1 (Ontology)**.** An ontology O is a triple

$$O = (C, P, I) \tag{1}$$

where:

- C is the set of classes, i.e. the set of concepts that populate the domain of interest;
- P is the set of properties, i.e. the set of relations existing between the concepts of domain;
- I is the set of individuals, i.e. the set of objects of the real world, representing the instances of a concept.

In general, classes, properties and individuals are referred to as entities.

An ontology defines a common vocabulary for information interchange in a knowledge domain [5] and is used as a solution for enabling interoperability across heterogeneous systems and distributed applications. But, because of human subjectivity, people may use different terms for the same meaning or may use the same term to mean different things. Therefore, instead of reducing heterogeneity, merely using ontologies themselves raises heterogeneity problem to a higher level. So, it is necessary to find the correspondences between semantically related entities of ontologies. The process of finding correspondences, which may include equivalence, consequence, subsumption and disjointness, is called ontology matching and the resulting set of correspondences is called an alignment. An alignment between two ontologies can be defined as follows [6]:

Definition 2 (Alignment)**.** An alignment A between two ontologies is a set of mapping elements. A mapping element is a 4-tuple, (e, e', n, R), where:

- e and e' are the entities of the first and the second ontology, respectively;
- n is a confidence measure in some mathematical structure (typically in the $[0, 1]$ range) holding for the correspondence between the entities e and e';
- R is a relation (typically the equivalence $=$) holding between the entities e and e'.

More in detail, the closeness value n (related to a given relation R) between the entities e and e' in the range $[0, 1]$ is determined by a similarity measure, which is used to compute a mapping element in the ontology alignment process. 0 stands for complete inequality and 1 for complete equality.

In a formal definition, the input of the ontology alignment process includes two ontologies O and O' to be matched, additional and optional inputs, such as a partial alignment A, some parameters p, such as weights and thresholds, and some external resources r such as dictionaries and databases. Therefore, the ontology alignment process can be given as [6]:

Definition 3 (Ontology Alignment Process)**.** The matching process can be seen as a function ϕ which, from a pair of ontologies O and O' to align, an input alignment A, a set of parameters p, a set of resources r, returns a new alignment A' between these ontologies:

$$A' = \phi(O, O', A, p, r). \tag{2}$$

The ontology alignment process computes a mapping element by using a similarity measure, which determines the closeness value n (related to a given relation R) between the entities e and e' in the range $[0, 1]$, where 0 stands for complete inequality and 1 for complete equality.

3.2 Similarity Measure

Generally, the most used similarity measures can be classified into three categories: syntactic measure, linguistic measure, and taxonomy based measure.

3.2.1 Syntactic Measure

Syntactic measures refer to different methods of string comparison or edit distances. In the context of ontology alignment, they are applied over the names, labels or comments associated with the entities. In our work, we employ two most widely used syntactic measures: Levenshtein distance [7] and Jaro distance [25].

The Levenshtein distance measures the minimum number of token insertions, deletions, and substitutions required to transform one string into another. Based on Levenshtein distance, a similarity function has been proposed:

$$Sim_{\text{Levenshtein}}(s,t) = \max\left(0, \frac{\min(|s|,|t|) - d(s,t)}{\min(|s|,|t|)}\right) \tag{3}$$

where:

- $|s|$ and $|t|$ are the length of string s and t, respectively;
- $d(s,t)$ is the Levenshtein distance between s and t.

Another similarity measure utilizes Jaro distance, which is based on the number of the common characters between two strings and the positions in which they appear. Given two strings s and t, Jaro is defined as follows:

$$Jaro_{Dist}(s,t) = \frac{1}{3}\left(\frac{\mathrm{com}(s,t)}{|s|} + \frac{\mathrm{com}(s,t)}{|t|} + \frac{(\mathrm{com}(s,t) - \mathrm{inv}(s,t))}{\mathrm{com}(s,t)}\right) \tag{4}$$

where:

- $\mathrm{com}(s,t)$ is the number of common characters of string s and t;
- $\mathrm{inv}(s,t)$ is the number of pairs consisting of common characters that appear in different positions.

3.2.2 Linguistic Measure

Linguistic measures make use of external linguistic resources, such as dictionaries, to calculate linguistic similarity between ontology entities. In our work, we use the WordNet [9], which provides a synonym, hyponym or hypernym set for a given word. Based on these resources, a linguistic similarity function can be presented by the following equation:

$$\mathrm{Sim}_{Linguistic}(w_1,w_2) = \begin{cases} 1, & \text{if the word } w_1 \text{ belongs to the set of} \\ & \text{synonym of the word } w_2 \text{ or vice versa;} \\ 0.5, & \text{if the word } w_1 \text{ is a hyponym or hypernym} \\ & \text{of the word } w_2 \text{ or vice versa;} \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

3.2.3 Taxonomy Based Measure

These measures rely on the internal structure of the ontology. The intuition behind taxonomic measures is that subsumption relation connects terms that are already similar, therefore their neighbors maybe also somehow similar. Let us suppose that c_1 and c_2 are the classes of ontologies O_1 and O_2, respectively, and ss_1 and ss_2 are the direct subclass sets of the classes c_1 and c_2, respectively. If the similarity between ss_1 and ss_2 has already been determined then the taxonomy distance between c_1 and c_2 can be defined by the following function:

$$\frac{\sum_{i=1}^{m} \max_{j=1\cdots n}(\mathrm{sim}(ss_{1i}, ss_{2j})) + \sum_{j=1}^{n} \max_{i=1\cdots m}(\mathrm{sim}(ss_{1i}, ss_{2j}))}{m+n} \tag{6}$$

where:

- m and n is the cardinality of ss_1 and ss_2, respectively;
- ss_{1i} is the i^{th} direct subclass of c_1 and ss_{2j} is the j^{th} direct subclass of c_2;
- sim is the similarity function which returns the similarity value of ss_{1i} and ss_{2j}.

This formula uses for every direct subclass of c_1 the highest similarity to a direct subclass of c_2 and vice versa. These maxima are added and the resulting sum is divided by the number of all direct subclasses.

For all entities in two ontologies O_1 and O_2, each similarity measure will generate a similarity matrix M, whose rows and columns are formed by the entities in O_1 and O_2, respectively. Then a similarity aggregation strategy, which is presented in the next section, aggregates various similarity scores (or similarity matrices) generated by different similarity measures into one final similarity.

3.3 Aggregation Strategies

In this section, a similarity aggregation strategy called Harmony [10] is presented. Harmony assigns the harmony values to each similarity measure as the weight in the aggregation, so that the weights of different similarity matrices can be tuned automatically. Based on Harmony, the weights of similarity measures can be assigned according to their reliability.

3.3.1 Harmony

Due to different similarities having different importance and reliability, setting different parameters to aggregate different similarities is necessary. However, doing this work manually is impractical, because of the inability to adapt to different ontology mapping tasks. The fact that similarities have their own advantages and shortcomings in different situations motivates us to find a measure that can tell us which similarity is more reliable and trustful so that we can give it a higher weight during aggregation and filter out false mappings that with similarity.

Ideally, for 1-1 mapping, the similarity score of two truly mapped entities should be larger than that of all other pairs of entities that share the same row or column with the two entities in the similarity matrix, which implies that the two entities of this pair mutually prefer each other. On this basis, Harmony can be defined as follows:

$$h = \frac{s\_max}{\min(|O_1|, |O_2|)} \tag{7}$$

where:

- s_max is the number of the entity pairs which has the highest and the only highest similarity in its corresponding row and column in the similarity matrix;
- $|O_1|$ and $|O_2|$ are the numbers of entities in the ontology O_1 and O_2, respectively.

3.3.2 Adaptive Similarity Aggregation

Due to the feature of representing the importance and reliability of the similarity, the harmony of the similarity matrix can be used as the weight to aggregate various similarities. Therefore, the final similarity of the pair of entities (e_{1i}, e_{2j}) can be defined by the following equation:

$$\mathit{FinalSim}(e_{1i}, e_{2j}) = \frac{\sum_{k=1}^{n} h_k \times \mathrm{Sim}_k(e_{1i}, e_{2j})}{n} \tag{8}$$

where:

- h_k is the harmony of the k^{th} similarity matrix;
- n is the number of similarity matrices;
- $\mathrm{Sim}_k(e_{1i}, e_{2j})$ is the similarity of the pair of entities e_{1i} and e_{2j} in the k^{th} similarity matrix.

In our work, the whole process of similarity aggregation is as follows:

- convert the input similarities into their corresponding similarity matrices;
- calculate the final similarity matrix through the above equations (7) and (8);
- convert the final similarity matrix back to its corresponding similarity.

Besides, harmony also provides a way to filter out noise from similarity matrix. If a similarity matrix has a high harmony, we can firmly believe that the lowest ranked similarity in each row and column is noise. Before aggregation we filter out a proportion of lower ranked similarities in each row and column. The number of similarities filtered equals to:

$$p = \min(L - 1, h \times L) \tag{9}$$

where:

- p is the number of the lowest ranked similarities filtered out in each row or column;
- L is the length of a row or column;
- h is the harmony of the similarity matrix.

3.4 Alignment Evaluation

Before we present our algorithm, we must first discuss the metrics we used to evaluate the quality of our alignments. In this section, three traditional alignment assessing measures are introduced first. However, all of these measures work with a prerequisite that a reference alignment between two ontologies to be aligned should be given in advance. Since most of the real scenarios are characterized by the absence of reference alignments, two approximation metrics, i.e. match coverage and match ratio, are then presented to approximate recall and precision respectively.

3.4.1 Alignment Evaluation on Reference Alignment

In order to compare which alignment is better it is necessary to evaluate quality of the alignment, which depends on correctness and completeness of the correspondences it has found. Recall (or completeness) measures the fraction of correct alignments found in comparison to the total number of correct existing alignments. A bigger value of recall indicates the found alignments have more correct ones, but the number of additionally falsely identified alignments is not known. Precision (or correctness) measures the fraction of found alignments that are actually correct. A bigger value of precision indicates that the found alignments are more precise, but it does not imply that more alignments are found. However, recall and precision are often balanced against each other with the so-called f-measure, harmonic mean of recall and precision.

Given a reference alignment R and some alignment A, recall, precision and f-measure are given by the following expressions:

$$\text{recall} = \frac{|R \cap A|}{|R|} \in [0,1] \tag{10}$$

$$\text{precision} = \frac{|R \cap A|}{|A|} \in [0,1] \tag{11}$$

$$f\text{-measure} = 2 \cdot \frac{(\text{recall} \cdot \text{precision})}{(\text{recall} + \text{precision})} \in [0,1] \tag{12}$$

where $|R|$, $|A|$ and $|R \cap A|$ represent the cardinality of R, A and $R \cap A$, respectively.

3.4.2 Alignment Evaluation on No Reference Alignment

Although the alignment evaluation measures recall, precision and f-measure, which use a reference alignment, can reflect quality of the resulting alignment, the reference alignment between two ontologies which is exactly what we are looking for, is not used nor created in our work. Therefore, we use rough approximations for recall and precision based on the relative quality of the obtained resulting alignment.

Match Coverage (MC) [11], the fraction of entities which exist in at least one correspondence in the resulting alignment in comparison to the total number of entities in the ontology, is used as a substitute for recall. The formulas of MC are presented as follows:

$$MC_1 = \text{MatchCoverage}_{O_1} = \frac{|O_1 - \text{Match}|}{|O_1|} \in [0,1] \tag{13}$$

$$MC_2 = \text{MatchCoverage}_{O_2} = \frac{|O_2 - \text{Match}|}{|O_2|} \in [0,1] \tag{14}$$

where:

- $O_1 - \text{Match}$ and $O_2 - \text{Match}$ are the sets of matched entities of ontology O_1 and O_2, respectively;
- O_1 and O_2 are the sets of all entities of ontology O_1 and O_2, respectively.

Match Ratio (MR) [11], the ratio between the number of found correspondences and the number of matched entities, is the estimation for precision. The MR is defined by the following expressions:

$$MR_1 = \text{MatchRatio}_{O_1} = \frac{\text{Corr}_{O_1-O_2}}{O_1 - \text{Match}} \in [1,+\infty) \tag{15}$$

$$MR_2 = \text{MatchRatio}_{O_2} = \frac{\text{Corr}_{O_1-O_2}}{O_2 - \text{Match}} \in [1,+\infty) \tag{16}$$

where:

- $\text{Corr}_{O_1-O_2}$ is the set of correspondences in a resulting alignment;
- $O_1 - \text{Match}$ and $O_2 - \text{Match}$ are the sets of matched entities of ontology O_1 and O_2, respectively.

The combined Match Ratio can be defined as follows:

$$\text{MatchRatio} = \frac{2 \cdot |\text{Corr}_{O_1-O_2}|}{|O_1 - \text{Match}| + |O_2 - \text{Match}|} \in [1,+\infty). \tag{17}$$

Match ratio that is too high indicates entities mapped to many other entities, and suggests low precision. Match ratio close to 1.0 indicates the highest precision.

Therefore, for the convenience of our work, we define Frequency which is the rough approximation for precision.

$$\text{Frequency} = \frac{1}{\text{MatchRatio}} \in [0,1]. \tag{18}$$

In our work, we use three evaluation measures, i.e. MC_1, MC_2 and Frequency, to estimate the quality of the alignment.

4 MULTI-OBJECTIVE OPTIMAL MODEL FOR OPTIMIZING MULTIPLE ONTOLOGY ALIGNMENTS WITH NO REFERENCE ALIGNMENT

In Definition 3, we have defined the ontology alignment process. Next, we formally formulate it as an optimization problem as follows:

Definition 4 (Ontology Alignment Optimization Problem)**.** The alignment optimization problem is a quintuple (O, O', A_{set}, X, F), where:

- O and O' are the ontologies to align and A_{set} is the set of various alignments determined by diverse similarity measures beforehand;
- X is the set of all possible thresholds which are used for filtering the alignment, and $X_i \in [0,1]$, $i = 1 \ldots |X|$;
- $F : X \to S, S_i \in [0,1], i = 1, 2, 3,$ is the objective function for evaluating the quality of a threshold $x \in X$:

  $$F(x) = (MC_1(A), MC_2(A), \text{Frequency}(A)) \tag{19}$$

  where A is an alignment determined by first aggregating the alignments in A_{set} and then filtering the aggregated alignment through threshold x. $MC_1(.)$, $MC_2(.)$ and Frequency$(.) : A \to [0,1]$ determines the match coverage of A in O, O' and the match ratio of A, respectively.

The larger the values of MC_1 and MC_2 are, the larger is the value of recall on RA, and the greater the value of Frequency is, the greater is the value of precision on RA [11]. Therefore, in our work, we take maximizing MC_1, MC_2 and Frequency simultaneously as our goals to replace the goals of maximizing the recall and precision on RA simultaneously. Our proposal of NSGA-II will exploit this definition for implementing an efficient searching approach and achieving better performances than the existing approaches.

5 NSGA-II FOR ONTOLOGY ALIGNMENTS WITHOUT REFERENCE ALIGNMENT

NSGA-II is considered to be a flexible and robust technique, which is good at finding various non-dominated solutions quickly. First, the algorithm applies the standard crossover and mutation operators in the evolution of current population. Then, it uses the fast non-dominated sorting technique and a crowding distance to rank and select the next generation. Finally, the best individuals in terms of non-dominance and diversity are selected as the solutions.

Before using NSGA-II, we need to apply the similarity measures on the two input ontologies and store the results in XML format. After that, the adaptive similarity aggregation presented in Section 3.3.2 is employed to aggregate the aligning results of the separate similarity measures and generate the intermediate alignment which is also stored in XML format. This is done only to avoid recalculating the final similarity during the process of running NSGA-II and improve the efficiency. Finally, we determine an optimal threshold by using NSGA-II. Four basic steps, namely population initialization, genetic operator, generation of new population, and elite strategy, are presented in detail.

5.1 Population Initialization

Only a threshold which determines whether a pair of entities is an alignment or not is encoded in a single chromosome. We utilize the Good-Lattice Point Method (GLPM) [12], a method of approximate uniform design, to initialize the population. The Good-Lattice Point Method can be described as follows: let $c = \{(x_1, x_2, \cdots, x_n) | 0 \leq x_1, x_2, \ldots, x_n \leq 1\}$, $\langle \alpha \rangle$ be the decimal part of α, p_1, p_2, …, p_n be the first n prime numbers and $(\gamma_1, \gamma_2, \cdots, \gamma_n) = \left(\sqrt{p_1}, \sqrt{p_2}, \cdots, \sqrt{p_n}\right)$, q be the number of uniformly distributed points in c, then $\{(\langle k\gamma_1 \rangle, \langle k\gamma_2 \rangle, \ldots, \langle k\gamma_n \rangle) \mid k = 1, \ldots, q\}$ are the q uniform distributed points in c. Especially, the dimension n of the problem equals one.

5.2 Genetic Operators

5.2.1 Selection

The aim of the selection operator is to select out two chromosomes as two parents which will be used in crossover operator. First, the selection operator randomly selects two individuals from the current population. Then, it chooses the better one as the first parent by comparing the candidates using their fitness and crowd distance values. Similarly, the other parent is selected in the same way.

5.2.2 Crossover

Let us denote parents selected by the selection operator as parent_1 and parent_2. We check if the crossover could be applied according to the crossover probability, a parameter of the genetic algorithm, and if it is, two children can be generated according to the following formula:

$$\text{child}_i = \text{rand}_i \times \text{parent}_1 + (1 - \text{rand}_i) \times \text{parent}_2, i = 1, 2 \tag{20}$$

where:

- child_i is the i^{th} child generated by parents, parent_1 and parent_2;
- rand_i is a random number between 0 and 1.

5.2.3 Mutation

Mutation operator assures diversity in the population and prevents premature convergence. If the mutation is applied according to the mutation probability, the new generated individual can be obtained through the following expression:

$$\text{Individual}_{new} = \begin{cases} \text{rand} \times \text{Individual}_{old}, & \text{if } r < 0.5 \\ \text{Individual}_{old} + \text{rand} \times (1 - \text{Individual}_{old}), & \text{otherwise} \end{cases} \tag{21}$$

where:

- Individual_{old} and Individual_{new} are two individuals before and after mutation, respectively;
- rand and r are two random numbers between 0 and 1.

Random number r is used to determine the old individual should become bigger or smaller after mutation. Besides, the formula (21) assures the new generated individual remains in the range $[0, 1]$.

5.3 Generation of New Population

In generation of new population, both closeness and diversity are addressed, where the parent population and the current population are put together and sorted lexicographically using non-domination ranks as primary sorting criterion, and crowding-distances as secondary sorting criterion. In non-dominated-sorting, first the non-dominated set is identified and its members get assigned rank 0. Then, the non-dominated set from the remaining individuals is computed, and assigned rank 1, and so on, until all chromosomes are ranked. To determine the crowding distance, a measure of diversity contribution, the circumference of the box touching the nearest neighboring solutions is computed. Finally, the new population is selected from sorted population. For more details about non-dominated-sorting and crowd-distance-sorting algorithm see also [13].

5.4 Elite Strategy

Elitist strategy puts the best individual (elite) of the current population unaltered in the next population. This assures the survival of the elite that has been obtained up to the moment. In our work, we propose a new evaluation measure, which is presented by the following expression that is similar to the definition of f-measure, to choose the best individual from the final generation.

$$\frac{\text{Frequency} \times \max(MC_1, MC_2)}{\alpha \times \text{Frequency} + (1 - \alpha) \times \max(MC_1, MC_2)} \tag{22}$$

In our work, we set $\alpha = 0.5$ to favor neither $\max(MC_1, MC_2)$ nor Frequency. In our algorithm, the individual with the highest value of the expression (22) found so far will be saved as the elite of the current generation. When the algorithm terminates, the best individual saved will be recommended to the user.

6 EXPERIMENTAL RESULTS AND ANALYSIS

In our work, we use the well-known benchmark provided by the Ontology Alignment Evaluation Initiative (OAEI) 2011 [14]. Table 1 shows a brief description of the benchmarks.

ID	Brief 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 flattened hierarchy
223	A regular ontology and other with an 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
304	A regular ontology and other with a real ontology which is quite close

Table 1. Brief description of benchmarks

6.1 Experiments Configuration

The similarity measures we used are Levenstein distance, Jaro distance, Linguistic distance and Taxonomy distance, and the population size is set as 20 chromosomes, crossover probability as 0.8, mutation probability as 0.05 and the max evolutionary generation as 5 in NSGA-II. In addition, we use Java to implement the proposed approach, and Alignment API [25] to read in and manipulate ontologies.

6.2 Results and Analysis

Table 2 shows the values of $\max(MC_1, MC_2)$, Frequency and the according recall and precision values when we evaluate the solutions by RA. Table 3 shows the recall, precision and f-measure values obtained by NSGA-II using RA and without RA, respectively. Table 4 shows the results obtained by other state of the art systems in ontology matching and our approach. In Table 2, Table 3 and Table 4, the symbols R and P stand for recall and precision value, respectively.

ID	$\max(MC_1, MC_2)$	Frequency	R	P
101	1.00	1.00	1.00	1.00
103	1.00	1.00	1.00	1.00
104	1.00	1.00	1.00	1.00
201	0.92	1.00	0.90	0.98
203	0.98	1.00	0.98	1.00
204	0.99	1.00	0.98	0.99
205	0.91	1.00	0.89	0.99
206	0.57	0.94	0.49	0.87
221	1.00	1.00	1.00	1.00
222	0.96	1.00	1.00	1.00
223	1.00	1.00	0.96	0.96
224	1.00	1.00	1.00	1.00
225	1.00	1.00	1.00	1.00
228	1.00	1.00	1.00	1.00
230	0.95	1.00	1.00	0.99
231	1.00	1.00	1.00	1.00
301	0.53	1.00	0.44	0.90
302	0.47	1.00	0.40	0.95
304	0.77	1.00	0.88	0.99

Table 2. The values of $max(MC_1, MC_2)$, Frequency, and the according recall and precision values

As can be seen from Table 2, except benchmark number 206, the frequency value of each test case equals to 1.00, which indicates that the alignments obtained are all 1-1 mappings. Therefore, the values of $O_1 -$ Match and $O_2 -$ Match are equal and we only need to consider MC_1 or MC_2. In all benchmarks but 206, 301 and 304, the $\max(MC_1, MC_2)$ and Frequency values of the resulting alignments are very close to their recall and precision values on RA.

In Table 3, the alignments achieved using no reference alignment approach are equal to those obtained using reference alignment except benchmarks 206, 223, 230, 301, 302 and 304. Besides, the quality of the alignment is even better than that using reference alignment in benchmark number 304, and the results are very close in benchmarks 223 and 230.

In Table 4, 1XX stands for the benchmarks in Table 1 whose number beginning with the prefix digit 1 and so are 2XX and 3XX, and the results obtained by our approach are the mean value of the corresponding precision and recall in Table 2. Among several state of the art ontology matching systems, we picked GOAL and SAMBO [24] because GOAL is also a matching system based on evolutionary algo-

ID	with reference alignment			without reference alignment		
	f-measure	R	P	f-measure	R	P
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	0.98	0.99	0.98	0.99	0.98
205	0.93	0.89	0.99	0.93	0.89	0.99
206	0.70	0.67	0.73	0.63	0.49	0.87
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	0.97	0.98	0.97	0.96	0.96	0.96
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	1.00	1.00	1.00	0.99	1.00	0.99
231	1.00	1.00	1.00	1.00	1.00	1.00
301	0.75	0.75	0.75	0.59	0.44	0.90
302	0.72	0.63	0.86	0.56	0.40	0.95
304	0.91	0.91	0.91	0.93	0.88	0.99

Table 3. Comparison of the results obtained by NSGA-II using reference alignment and without using reference alignment

ID	GOAL			SAMBO			Our Approach		
	f-measure	R	P	f-measure	R	P	f-measure	R	P
1XX	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
2XX	0.90	0.91	0.89	0.70	0.54	0.98	0.96	0.94	0.98
3XX	0.57	0.62	0.54	0.87	0.80	0.95	0.81	0.57	0.95
Avg.	0.82	0.84	0.81	0.87	0.78	0.98	0.90	0.84	0.98

Table 4. Comparison of the alignments obtained by our approach with those returned by GOAL and SAMBO

rithm, and SAMBO utilizes the PRA to align the ontologies. To some extent, their ideas are similar to ours.

As can be seen from Table 4, for the benchmark 1XX, the quality of our alignments is identical to GOAL and SAMBO. For the benchmarks 2XX, our approach obviously outperforms GOAL and SAMBO. For the benchmarks 3XX, our approach outperforms GOAL and is very close to SAMBO by considering the value of f-measure. Finally, from the average value in Table 4, the quality of the alignments obtained by our approach, which achieves the highest f-measure, recall and precision, is in general better than GOAL and SAMBO.

To sum up, we may draw the conclusion that the use of MC and MR in NSGA-II to approximately evaluate the ontology alignments is effective.

7 CONCLUSIONS AND FUTURE WORK

Ontology alignment plays an increasingly important role in ontology engineering. Although many studies have been carried out on this problem, there still exist a lot of problems to be solved urgently. One of the most significant issues is how to use EA to optimize the alignment without using RA.

In this work, we first use an adaptive aggregation strategy to aggregate the different similarity measures into a single similarity metric, then the NSGA-II to optimize the alignments using the approximate evaluation measures: MC and MR. Experimental results show that our approach effectively finds the approximate optimal solutions which are very close to the optimal solution obtained by NSGA-II using RA and the quality of alignments is in general better than state of the art ontology matching systems such as GOAL and SAMBO.

In continuation of our research, we are interested in designing an ontology matching system capable of working on a dataset to match at both schema and instance level and the matching execution becomes dynamic. This means that the various techniques featuring a matching tool can be invoked alone or in combination to satisfy the specific need of the considered matching scenario.

Acknowledgments

This work was supported by National Natural Science Foundation of China (No. 61272119, No. 61472297).

REFERENCES

[1] Wong, A.—Yip, F.—Ray, P.—Paramesh, N.: Towards Semantic Interoperability for IT Governance: An Ontological Approach. Computing and Informatics, Vol. 27, 2008, pp. 131–155.

[2] Martinez-Gil, J.—Alba, E.—Aldana-Montes, J. F.: Optimizing Ontology Alignments by Using Genetic Algorithms. Nature Inspired Reasoning for the Semantic Web (NatuReS 2008), Vol. 419, 2008, pp. 31–45.

[3] Lambrix, P.—Liu, Q.: Using Partial Reference Alignments to Align Ontologies. Proceedings of the 6th European Semantic Web Conference (ESWC) 2009, pp. 188–202.

[4] Doan, A.—Domingos, P.—Halevy, A.: Reconciling Schemas of Disparate Data Sources: A Machine-Learning Approach. Proceedings of the 2001 ACM SIGMOD International Conference on Management of Data 2001, pp. 509–520.

[5] Alípio, P.—Neves, J.—Carvalho, P.: An Ontology For Network Services. Computing and Informatics, Vol. 26, 2007, pp. 543–561.

[6] Acampora, G.—Loia, V.—Salerno, S.: A Hybrid Evolutionary Approach for Solving the Ontology Alignment Problem. International Journal of Intelligent Systems, Vol. 27, 2012, No. 3, pp. 189–216.

[7] Maedche, A.—Staab, S.: Measuring Similarity Between Ontologies. Proceedings of the International Conference on Knowledge Engineering and Knowledge Management (EKAW) 2002, pp. 251–263.

[8] Euzenat, J.—Shvaiko, P.: Ontology Matching. Springer 2007, p. 80.

[9] Miller, G. A.: WordNet: A Lexical Database for English. Communications of the ACM, Vol. 38, 1995, No. 11, pp. 39–41.

[10] Mao, M.—Peng, Y. F.—Spring, M.: An Adaptive Ontology Mapping Approach with Neural Network Based Constraint Satisfaction. Web Semantics: Science, Services and Agents on the World Wide Web, Vol. 8, 2010, No. 1, pp. 14–25.

[11] Kirsten, T.—Thor, A.—Rahm, E.: Instance-Based Matching of Large Life Science Ontologies. Proceedings of the 4th International Conference on Data Integration in the Life Sciences (DILS '07), 2007, pp. 172–187.

[12] Fang, K. T.—Wang, Y.: Number-Theoretic Methods in Statistics. Chapman and Hall, London, UK 1994.

[13] Deb, K.—Agrawal, S.—Pratap, A.: A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. Proceedings of the Parallel Problem Solving from Nature VI Conference, Vol. 1917, 2000, pp. 849–858.

[14] Ontology Alignment Evaluation Initiative (OAEI). Available at `http://oaei.ontologymatching.org/2011/`.

[15] Ginsca, A.-L.—Iftene, A.: Using a Genetic Algorithm for Optimizing the Similarity Aggregation Step in the Process of Ontology Alignment. 9th Roedunet International Conference (RoEduNet) 2010, pp. 118–122.

[16] Naya, J. M. V.—Romero, M. M.—Loureiro, J. P.: Improving Ontology Alignment Through Genetic Algorithms. Soft Computing Methods for Practical Environment Solutions: Techniques and Studies 2010, pp. 240–259.

[17] Euzenat, J.—Valtchev, P.: Similarity-Based Ontology Alignment in OWL-Lite. 16th European Conference on Artificial Intelligence Proceedings, Vol. 110, 2004, pp. 333–337.

[18] Van Rijsbergen, C. J.: Information Retrieval. Butterworth, London 1975.

[19] Huang, B. Q.—Buckley, B.—Kechadi, T.-M.: Multi-Objective Feature Selection by Using NSGA-II for Customer Churn Prediction in Telecommunications. Expert Systems withApplications, Vol. 37, 2010, No. 5, 2010, pp. 3638–3646.

[20] Martinez-Gil, J.–Aldana-Montes, J. F.: Evaluation of Two Heuristic Approaches to Solve the Ontology Meta-Matching Problem. Knowledge and Information Systems, Vol. 26, 2011, No. 2, pp. 225–247.

[21] Eckert, K.—Meilicke, C.—Stuckenschmidt, H.: Improving Ontology Matching Using Meta-Level Learning. Semantic Web: Research and Applications, Vol. 5554, 2009, pp. 158–172.

[22] Mazak, A.—Schandl, B.—Lanzenberger, M.: Align++ – A Heuristic-Based Method for Approximating the Mismatch-at-Risk in Schema-Based Ontology Align-

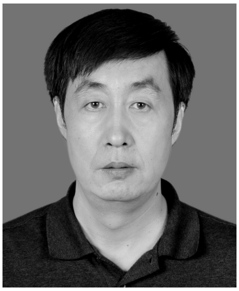
ment. Proceedings of International Conference on Knowledge Engineering and Ontology Development (KEOD 2010), pp. 17–26.

[23] Stoutenburg, S. K.—Kalita, J.—Ewing, K.—Hines, L. M.: Scaling Alignment of Large Ontologies. International Journal of Bioinformatics Research and Applications, Vol. 6, 2010, No. 4, pp. 384–401.

[24] Lambrix, P.—Tan, H.: SAMBO C – A System for Aligning and Merging Biomedical Ontologies. Journal of Web Semantics, Vol. 4, 2006, No. 1, pp. 196–206.

[25] Euzenat, J.: An API for Ontology Alignment. Proceedings of 3rd International Semantic Web Conference (ISWC '04), 2004, pp. 698–712.

Xingsi Xue is a Lecturer at Fujian University of Technology, and a Ph. D. student in Computer Applications at School of Computer Science and Technology, Xidian University, China. His research interests include ontology matching technology, intelligent expert decision system, and object-oriented technology.



Yuping Wang is a Professor and Ph. D. supervisor at School of Computer Science and Technology, Xidian University, China. He received his Ph. D. from the Department of Mathematics. Xi'an Jiaotong University, China in 1993. Currently, his research interests include optimization methods, theory and application, evolutionary computation, data mining, and machine learning.

Weichen Hao is a Master student at School of Computer Science and Technology. His research interests include software engineering, ontology matching technology, and data mining.

Juan Hou is a Master student at School of Computer Science and Technology. Her research interests include software engineering, ontology matching technology, and data mining.