

Hybridizing Genetic Algorithms and Hill Climbing for Similarity Aggregation in Ontology Matching

Giovanni Acampora and Uzay Kaymak
School of Industrial Engineering, Information Systems
Eindhoven University of Technology
P.O. Box 513, 5600 MB, Eindhoven, The Netherlands
Email: {g.acampora, u.kaymak}@tue.nl

Vincenzo Loia and Autilia Vitiello
Department of Computer Science
University of Salerno
Fisciano, 84084, Italy
Email: {loia, avitiello}@unisa.it

Abstract—Ontology Matching aims at finding correspondences between two different ontologies with overlapping parts in order to bring them into a mutual agreement. The set of correspondences, called *alignment*, is obtained by computing an aggregated similarity value for all pairs of ontology entities through a weighted approach. Unfortunately, the similarity aggregation task is a very complex optimization process, above all, when no information is known about ontology characteristics. This work presents a hybrid approach which aims at efficiently optimizing the weights for the similarity aggregation task without knowing *a priori* the ontology features. The effectiveness of our approach is shown by aligning ontologies belonging to the well-known OAEI benchmark dataset and by executing a comparison based on the Wilcoxon's signed rank test which highlights that our proposal statistically outperforms both its genetic counterpart and a traditional no evolutionary approach.

I. INTRODUCTION

Nowadays, ontologies play a prominent role for enabling data and knowledge sharing within heterogeneous and distributed systems thanks to their capability of formally describing the semantics of a particular domain of interest. However, in spite of their large exploitation, the ability of ontologies to manage disparate information could be limited by the so-called *semantic heterogeneity problem*. This problem is due to the enormous variety of ways that a domain of interest can be conceptualized which leads to the creation of different ontologies with contradicting or overlapping parts[1]. In order to address the semantic heterogeneity problem, a so-called *ontology matching* is required to detect correspondences between semantically related entities of different ontologies [2] to lead them into a mutual agreement. The set of correspondences is called *alignment*. Typically, an alignment is produced by computing the similarity level for pairs of entities belonging to the ontologies to be aligned by exploiting a so-called *similarity measure* or *matcher* and, successively, by applying a threshold-based operator. In literature, there exist a lot of matchers which can be categorized in *lexical*, *linguistic* and *structural* comparison approaches [3][4]. Currently, it is recognized that a combination of several similarity techniques leads to better alignment results than using only one at a time [5]. The task of combining more similarity measures in order to compute a unique similarity value for a pair of entities belonging to ontologies under alignment is known

as *similarity aggregation*. Typically, the similarity aggregation task is computed by exploiting a weighted average function. Therefore, the best aggregation of different similarity measures results in identifying the most suitable combination of weights. Unfortunately, finding the most appropriate values for these weights is a very complex optimization process, above all, when no information is known about ontology characteristics.

Starting from this consideration, this work proposes an evolutionary hybrid approach which aims at efficiently optimizing the weights for the similarity aggregation task without knowing *a priori* the ontology characteristics. In particular, our proposal exploits an emergent hybrid optimization method, known as Memetic Algorithm (MA). MAs [6] are population-based search methods which combine evolutionary algorithms with local search strategies. Therefore, they have the complementary advantages of the global search (generality, robustness and efficiency) and the local improvement (rapid convergence toward local minima). These features make MAs seemingly suitable to be applied for performing a complex process like the similarity aggregation task. In addition, differently from other approaches [7][8], our proposal does not require any information about ontology characteristics or a reference alignment to perform the ontology alignment process. The applicability of our approach is shown by means of a set of experiments where our memetic algorithm is used to align pairs of ontologies belonging to the well-known Ontology Alignment Evaluation Initiative (OAEI) benchmark dataset¹, whereas, its effectiveness is highlighted by means of the Wilcoxon's signed rank test [9] which states that our proposal statistically outperforms both its genetic counterpart and a traditional no evolutionary approach.

II. ONTOLOGY MATCHING: BASIC CONCEPTS

According to [10], an ontology matching is a function f which, given a pair of ontologies O_1 and O_2 , returns an alignment A . However, there are also some optional inputs that can be expressed: (i) an input alignment A' , which is to be completed; (ii) the matching parameters, such as weights or thresholds; and (iii) external resources, such as common

¹<http://oaei.ontologymatching.org/>

knowledge or dictionaries. Hence, ontology matching can be represented as in figure 1.

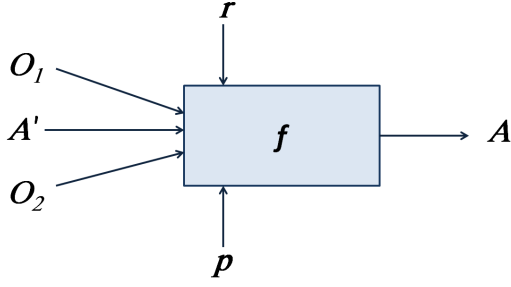


Fig. 1. The ontology alignment process

The output alignment A is a set of semantic correspondences between two entities, each one belongs to one of ontologies under alignment. Mathematically, let consider an alignment A between two ontologies O_1 and O_2 as follows:

Definition 1: An alignment A is a set of k correspondences, where each correspondence c_l (with $l = 1, 2, \dots, k$) represents a pair of entities defined as follows:

$$c_l = (e_i, e_j) \text{ with } i \in \{1, 2, \dots, |O_1|\}, j \in \{1, 2, \dots, |O_2|\}$$

where e_i is the i^{th} entity of ontology O_1 , whereas, e_j is the j^{th} entity of ontology O_2 and the implied relation is the equality.

Each correspondence has associated a so-called *confidence value* (typically in the $[0, 1]$ range) which represents the similarity level existing between the two entities composing the correspondence. In order to compute this confidence value, different matchers exist in literature categorized in lexical, linguistic and structural similarity measures. Since the application of a single matcher is often not enough to produce an acceptable output alignment, the common technique is to combine different matchers to compute a single aggregated similarity value. The process of aggregating different similarity measures is commonly known as *similarity aggregation*. Formally, let consider an alignment A , a correspondence c belonging to the alignment A and h similarity measures, the aggregated similarity value for c can be defined as follows:

$$sim_{aggregate}(c) = \sum_{i=1}^h w_i \times sim_i(c) \quad \text{subject to} \quad \sum_{i=1}^h w_i = 1$$

where w_i is the weight associated to the i^{th} similarity measure and $sim_i(c)$ is the similarity value computed by the i^{th} similarity measure.

The aim of this work is to optimize the combination of the weights w_i (with $i = 1, \dots, h$) by exploiting an emergent optimization method such as the memetic algorithms in order to produce an optimal alignment between the two input ontologies O_1 and O_2 .

The rest of the section is devoted to present the five similarity measures taken into account in this work and extracted by the matchers presented in [11]. In details, the considered matchers are all distance-based similarity measures, i.e., they

compute the distance rather than the similarity value between two ontology entities. In particular, we consider:

- two matchers belonging to lexical category named *Entity Name Distance Measure* and *Comment Distance Measure* which compute a string metric for ontology alignment introduced in [1] between, respectively, the names of ontology entities and the comments associated to them;
- two matchers belonging to the structural category named *Hierarchy Distance Measure* and *Domain and Range Restrictions Distance Measure* which compute the distance between ontology entities by taking into account, respectively, the existing hierarchy relations among ontology entities and the Domain and range restrictions existing on properties;
- a matcher for the the linguistic category named *Word Net Synonymy Name Distance Measure* which computes a synonymy-based distance between the names of the ontology entities by exploiting the WordNet dictionary².

III. AN EVOLUTIONARY HYBRID APPROACH FOR THE SIMILARITY AGGREGATION TASK

This paper aims at presenting a novel technique to perform the similarity aggregation step by exploiting an emergent hybrid method named Memetic Algorithms (MAs). The MAs are population-based search methods which integrate local search processes within the global search successive generations in order to refine population individuals. The MAs, representing the marriage between global search and local improvement, succeed to merge the corresponding benefits: generality, robustness and global search efficiency derived by EAs and rapid convergence toward local minima characterizing all local search methods. In general, the structure of a MA is shown in figure 2. As similar to genetic algorithms, MAs try to solve an optimization problem by manipulating a *population* of potential solutions. Precisely, they operate on encoded representations of the solutions, called *chromosomes*. The initial population of chromosomes is typically randomly generated. The algorithm evolution progresses successive *generations*. In each generation, a *selection process* provides the mechanism for selecting better solutions to survive. Each solution is evaluated by means a *fitness function* that reflects how good it is. In each generation, a recombination process of genetic material is simulated through two operators: *crossover* that exchanges portions between two randomly selected chromosomes and *mutation* that causes random alteration of the chromosome genes. Moreover, different from genetic algorithms, MAs execute a local search process within each generation. The algorithm evolution terminates when prefixed conditions such as the maximum number of generations are reached.

By analyzing the general structure of a MA, it is clear that in order to apply MAs for solving the similarity aggregation problem, the following components must be defined:

- the chromosome structure representing a solution to our problem;

²<http://wordnet.princeton.edu/>

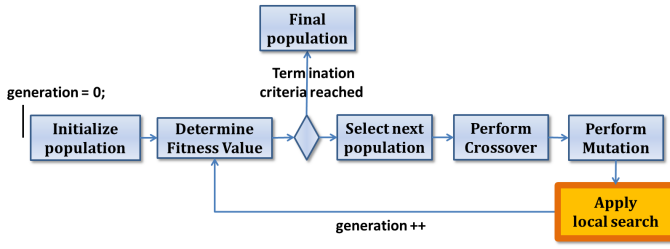


Fig. 2. The general structure of a memetic algorithm

- the fitness function which allows the evaluation of the explored solutions;
- the integrated local search process and all its features.

Hereafter, each of these points will be deepened and discussed as a necessary step for building our memetic approach.

A. The chromosome structure

By considering that each chromosome of the population represents a potential solution to our problem, it must contain the values for the set of weights indicating the contribution of each similarity measure used in the similarity aggregation task. In addition, in order to give a better optimization of the whole similarity aggregation process, our proposal considers as part of chromosome also the threshold value t used to decide if a pair of entities must belong to alignment or not. Because all the considered values (weights and threshold) are numbers belonging to the interval $[0, 1]$, the chromosome is defined as a double vector where the first genes represent the weights and the last gene represents the threshold. Therefore, by considering h similarity measures, our chromosome has a length equal to $h + 1$. A graphical representation of the chromosome structure is given in figure 3. However, since the sum of all weights has to be equal to 1, it is necessary a further step in order to obtain the real value of weights to use in the similarity aggregation step. In detail, each one of the first h values of the chromosome is scaled conforming to the sum of all first h values. Formally:

$$w_i = \frac{g_i}{\sum_{i=1}^h g_i} \quad (1)$$

where g_i is the i^{th} value of the chromosome and w_i is the i^{th} weight used to perform the similarity aggregation task.

Starting from each chromosome, it is possible to build an alignment by performing the following steps:

- computation of the aggregated similarity $sim_{aggregate}$ for each pair of entities between the two ontologies to be aligned by considering the weights w_i , with $i = 1, 2, \dots, n$;
- selection of pairs of entities with an aggregated similarity greater or equal to the threshold value t .

Since in this work the considered matchers are distance measures, in our approach, the aggregated similarity $sim_{aggregate_i}$ for the i^{th} pair of entities is computed as $1 - d_{aggregate_i}$ where $d_{aggregate_i}$ represents the aggregated distance between the entities belonging to the i^{th} pair.

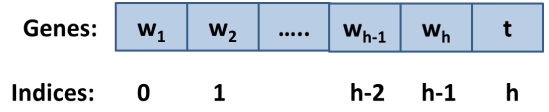


Fig. 3. Graphical representation of a chromosome

B. The Fitness Function

The fitness function evaluates a chromosome by estimating the quality of the corresponding alignment. In particular, this evaluation is not based on common quality measures such as precision, recall and F-measure and, as a consequence, it does not require the exploitation of a reference alignment. In our approach, the fitness function evaluates the alignment by taking into account the number of the correspondences belonging to the alignment and their similarity values. More in detail, similar to [12], our evaluation is based on these two observations:

- a higher average of the similarity values of the correspondences reflects a better alignment;
- by considering the same average of the similarity values, a higher number of correspondences reflects a better alignment.

However, since the considered matchers are distance measures, our fitness function performs a minimization process based on these opposite observations:

- a lower average of the distance values of the pair of entities corresponds to a better alignment;
- by considering the same average of the distance values, a lower difference between the number of correspondences and the cardinality of the smaller ontology under alignment corresponds to a better alignment.

Hereafter, the fitness function is formally defined. Let consider an alignment A as formulated in definition 1, the fitness function F for the alignment A is defined as follows:

$$F(A) = \beta \cdot \Phi(abs(min\{|O_1|, |O_2|\} - |A|)) + (1 - \beta) \cdot f(A)$$

where $|O_1|$ and $|O_2|$ are respectively the cardinalities of the ontologies O_1 and O_2 , $|A|$ is the cardinality of the alignment under evaluation, Φ is a function of normalization in range $[0, 1]$, abs is a function which computes the absolute value, f is the function which computes the average of distances characterizing the correspondences belonging to the alignment A . Precisely, f is defined as follows:

$$f(A) = \frac{\sum_{i=1}^{|A|} d_{aggregate_i}}{|A|}$$

where $d_{aggregate_i}$ is the aggregated distance obtained by using the considered similarity measures and the weights computed starting from the chromosome under evaluation. The function F is a sum weighted by β , a real value in $[0, 1]$ acting as tuning parameter useful for generating ontology alignments characterized by high precision ($\beta < 0.5$) or high recall ($\beta > 0.5$). In our approach, β is empirically set to 0.6.

C. The integrated local search process

In order to design an efficient MA, a lot of issues must be faced about the integrated local search process. The principal of these issues may be stated as [13]: *What is the best trade-off between local search and the global search provided by evolution?* This issue leads naturally to questions such as the following:

- *Local search frequency*: How often should local search be applied within the evolutionary cycle?
- *Order with respect to genetic operators*: When should local search be applied?
- *Individual selection mechanism*: Which individuals in the population should be improved by local search?
- *Local search intensity*: How much computational effort should be allocated to each local search?
- *Local search method*: Which local search procedure should be used?

Therefore, in order to complete the design of our memetic algorithm for the similarity aggregation problem, the aforementioned issues must be addressed. In detail, our proposal is to build a memetic algorithm which has the following features:

- *Local search frequency* = 1, i.e., the local search is applied within each evolutionary cycle;
- *Order with respect to genetic operators* = after, i.e., local refinement is executed after crossover and mutation operators;
- *Individual selection mechanism* = only the best, i.e., only the best chromosome of population is improved by the local search process;
- *Local search intensity* = equal to n local search iterations, i.e., each local search process ends after running n iterations.
- *Local search method* = stochastic hill climbing, i.e., the selected local search method is the stochastic version of the Hill Climbing search.

The rest of subsection is devoted to give more details about the exploited local search method. In general, the local search strategies accomplish an iterative search to try an optimum solution in the *neighborhood* of a candidate. Starting from an initial candidate solution, these methods use *local* information to select in the neighborhood the next current solution. In each iteration, typically, the current candidate is improved through a mutation operator. The process ends when the solution can not be better or after a prefixed number of iterations. In particular, the Hill Climbing algorithm is a local search iterative method which attempts, during each iteration, to find a better solution by incrementally changing a single component of the current one. There are several variants of Hill Climbing search depending on how the next solution is tried. Our proposal is to use the stochastic version of Hill Climbing which selects a neighbor at random, and then decides whether to move to that neighbor or to examine another by considering the amount of improvement.

IV. EXPERIMENTS AND RESULTS

In order to show the applicability of our proposal, our memetic approach has been used to align, by optimizing the combination of weights used in the similarity aggregation task, pairs of ontologies belonging to the well-known OAEI benchmark dataset 2010. In particular, the benchmark tests exploited in our experiments lie in the four categories reported in Table I.

TABLE I
BENCHMARK DESCRIPTIONS

Benchmark No.	Description
# 101-104	The ontologies under alignment are the same or the first one is the "OWL Lite restriction" of the second one.
# 201-210	The ontologies under alignment have the same structure, but different lexical and linguistic features.
# 221-247	The ontologies under alignment have the same lexical and linguistic features, but different structure.
# 301-304	The ontologies under alignment are real world cases.

The improvement provided by our proposal has been investigated both in terms of accuracy, by comparing the quality of the produced alignments with those obtained by a traditional approach, and in terms of computational performances, by comparing our memetic proposal with its genetic counterpart. Both comparisons have been carried out by means of a non-parametric statistical procedure named Wilcoxon's signed rank test. This pairwise test allows to answer this question: do two samples represent two different populations? [14]. Therefore, it is suitable to statically state that a significant difference exists between outputs of two methods. The quality of the alignments is computed by using the well-known *F-measure*. F-measure, typically formulated as in definition 2, is a conformance measure derived from the information retrieval field which represents a combination of *precision* and *recall*. It determines the quality of an alignment by comparing it with a reference alignment R .

Definition 2 (F-measure): Given a reference alignment R , F-measure M is equal to the harmonic mean of precision Pr and recall Rec , i.e.,

$$M = \frac{2 \cdot Pr(A, R) \cdot Rec(A, R)}{Pr(A, R) + Rec(A, R)}.$$

where $Pr(A, R) = \frac{|R \cap A|}{|A|}$ and $Rec(A, R) = \frac{|R \cap A|}{|R|}$.

On the other hand, the computational performances are assessed by comparing the computed fitness value after executing the same number of fitness evaluations. Hereafter, a detailed description of the performed tests is given.

A. Test I

Our first test aims at showing the improvements provided by our proposal in terms of accuracy of computed alignments. In order to achieve this goal, our memetic algorithm has been compared with a traditional method which considers a uniform distribution of weights and a prefixed threshold.

This method is typical when no information is known about ontology characteristics. Usually, the threshold value is chosen by a human expert. In our experiment, in order to select a competitive threshold, a set of real values has been tested with the intent to select the threshold value for which the best accuracy is achieved the largest number of times for the considered benchmark tests. As shown in figure 4, this value is 0.75.

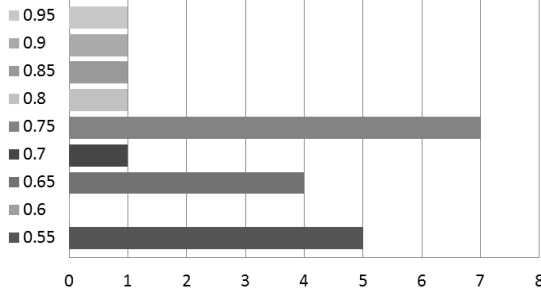


Fig. 4. A graphic reporting the number of times which the considered threshold value leads to the best F-measure value by considering a uniform distribution of weights.

With regard to our approach, the used configuration is the following one:

- population size = 30 chromosomes;
- crossover rate = 0.9;
- mutation rate = 0.02;
- maximum number of local iterations = 70;
- termination condition = 250 fitness evaluations;
- similarity measures = *Entity Name Distance Measure*, *Comment Distance Measure*, *Hierarchy Distance Measure*, *Domain and Range Restrictions Distance Measure* and *Word Net Synonymy Name Distance Measure*.

This parameter configuration has been chosen in an empirical way by performing preliminary experiments.

The comparison between our approach and the traditional method is carried out in terms of F-measure values, reported in Table II. In particular, the F-measure values for our memetic approach are obtained by computing the average value over fifteen runs.

As shown in Table II, our proposal outperforms the traditional method for the 85% of benchmark tests. However, in order to statistically verify the validity of our approach, we have performed a Wilcoxon's signed rank test by considering as sample data the F-measure values presented in the table II. The Wilcoxon's test states that our optimization method outperforms the traditional approach at 1% significance level.

B. Test II

Our second test aims at showing the improvements provided by our proposal in terms of computational performances. In order to achieve this goal, our memetic algorithm has been compared with its genetic counterpart. Both algorithms have been executed fifteen times for each benchmark test. The average fitness values are compared after 250 fitness

TABLE II
THE COMPARISON BETWEEN OUR PROPOSAL AND THE TRADITIONAL METHOD IN TERMS OF F-MEASURE VALUES OF COMPUTED ALIGNMENTS.

Benchmark No.	Traditional approach	Memetic approach	Rel. Improvement
101	0.98	1.00	2.04%
103	0.97	1.00	3.09%
104	0.98	1.00	2.04%
201	NaN	0.62	n.c.
203	0.90	0.96	6.72%
204	0.96	0.97	1.04%
205	0.50	0.79	58.00%
206	0.63	0.88	39.68%
207	0.63	0.88	39.68%
208	0.86	0.88	2.33%
209	0.35	0.33	-5.71%
210	0.51	0.49	-3.92%
221	0.98	0.99	1.02%
224	0.98	1.00	2.04%
225	0.98	1.00	2.04%
230	0.92	0.93	1.09%
247	0.97	0.68	-29.87%
301	0.69	0.70	1.45%
302	0.43	0.63	46.51%
304	0.86	0.87	1.16%

evaluations and reported in Table III. The configuration of the genetic counterpart is the following one:

- population size = 30 chromosomes;
- crossover rate = 0.9;
- mutation rate = 0.02;
- termination condition = 250 fitness evaluations;
- similarity measures = *Entity Name Distance Measure*, *Comment Distance Measure*, *Hierarchy Distance Measure*, *Domain and Range Restrictions Distance Measure* and *Word Net Synonymy Name Distance Measure*.

This setting represents the best configuration after preliminary empirical experiments.

As shown in Table III, our proposal outperforms its genetic counterpart for the 90% of benchmark tests. However, also in this case, in order to statistically validate our approach, we have performed a Wilcoxon's signed rank test by considering as sample data the fitness values presented in the table III. The Wilcoxon's test states that our optimization method outperforms the genetic approach at 1% significance level.

V. RELATED WORKS

In last years, several works have addressed how to efficiently combine similarity measures during the ontology alignment process. One of the first works which deals with the similarity aggregation problem by exploiting evolutionary algorithms is GOAL (Genetics for Ontology Alignments) [8]. This system computes, by means of a genetic algorithm, the optimal weight configuration for a weighted average aggregation of several similarity measures by considering a reference alignment. Indeed, GOAL uses a method of evaluation based on one of these conformance measures: precision, recall and F-measure.

TABLE III
THE COMPARISON BETWEEN OUR PROPOSAL AND
ITS GENETIC COUNTERPART IN TERMS OF AVERAGE FITNESS VALUES
ACHIEVED AFTER 250 FITNESS EVALUATIONS.

Benchmark No.	Genetic approach	Memetic approach	Rel. Improvement
101	0.0038	0.0035	-7.89%
103	0.0037	0.0031	-16.22%
104	0.0026	0.0028	7.69 %
201	0.2873	0.2419	-15.80%
203	0.0166	0.0071	-57.23%
204	0.0324	0.0239	-26.23%
205	0.2061	0.1401	-32.02%
206	0.1943	0.1065	-45.19%
207	0.1783	0.1190	-33.26%
208	0.0455	0.0367	-17.53%
209	0.2756	0.2608	-5.37%
210	0.2754	0.2378	-13.65%
221	0.0186	0.0085	-54.30%
224	0.0035	0.0025	-28.57%
225	0.0088	0.0046	-47.73%
230	0.0242	0.0274	13.22%
247	0.1961	0.1778	-9.33%
301	0.1644	0.1529	-7.00%
302	0.1092	0.0944	-13.55%
304	0.1154	0.0986	-14.56%

The same idea of implementing a meta-matcher to combine multiple similarity measures through genetic algorithms is developed in a more recent work [15]. In addition, also in [16], the authors try to optimize the combination of similarity measures by means of a genetic algorithm but, different from the previous works, they focus on optimizing the whole similarity aggregation step as a single unit, including the threshold value in the chromosome. For this aspect, our approach is similar to this work. Indeed, also our proposal tries to optimize the whole similarity aggregation phase by including the threshold value as part of the chromosome structure (see section III-A).

However, all cited methods have a relevant drawback which affects strongly their applicability: they require an *a priori* knowledge about ontologies under alignment in order to select the most suitable set of the weights. In fact, they align ontologies by using the optimal combination of weights obtained for a pair of ontologies with the same features whose a reference alignment is known. Instead, our approach allows to directly optimize the set of weights for the ontologies under alignment by exploiting a fitness function which does not require a reference alignment to work.

Therefore, our proposal improves the state of art because it provides the relevant benefit to optimize weights for the similarity aggregation step, and as a consequence, to align two ontologies, without having any *a priori* information about ontology nature.

VI. CONCLUSIONS AND FUTURE WORKS

In this work, we present a memetic algorithm capable of optimizing the whole similarity aggregation task in order to produce an optimal alignment between two ontologies. Our

approach simultaneously optimizes both the combination of weights and the threshold value used to perform the cut operation. The relevant contribution of our proposal is to efficiently perform the ontology alignment process without knowing *a priori* the characteristics of the ontologies under alignment. As shown by performing a Wilcoxon's signed rank test, our memetic algorithm statistically outperforms a traditional approach, in terms of alignment accuracy, and a genetic algorithm, in terms of computational performances.

However, in order to further improve the quality of produced alignments, in the future, our idea is to enable the proposed ontology alignment system to detect not only equivalence mappings but also other kinds of correspondences based on relations such as subsumption and mismatch.

REFERENCES

- [1] Stoilos, G., Stamou, G., and Kollias, S.: A String Metric for Ontology Alignment. In: Proceedings of the ninth IEEE International Symposium on Wearable Computers, pp.624-637 (2005).
- [2] Shvaiko, P. and Euzenat, J.: Ontology matching: state of the art and future challenges. In: IEEE Transactions on Knowledge and Data Engineering (2012).
- [3] Rahm, E., Bernstein, P. A.: A survey of approaches to automatic schema matching. In: The VLDB Journal The International Journal on Very Large Data Bases, v.10 n.4, pp.334-350 (2001).
- [4] Euzenat, J., Le Bach, T., Barrasa, J., Bouquet, P., De Bo, J., Dieng, R., et al.: State of the art on ontology alignment. Deliverable D2.2.3 v1.2. Knowledge Web. Retrieved December 21, 2008, from <http://knowledgeweb.semanticweb.org/>
- [5] Ehrig, M.: Ontology alignment: bridging the semantic gap. In: Springer-Verlag New York, Inc., (2007).
- [6] Moscato, P.: On evolution, search, optimization, GAs and martial arts: toward memetic algorithms. In: California Inst. Technol., Pasadena, CA, Tech. Rep. Caltech Concurrent Comput. Prog. Rep. 826, (1989).
- [7] Keikha, M.M., Nematbakhsh, M.A., Ladani, B.T.: Adaptive Similarity Aggregation Method for Ontology Matching. In: Computer Modeling and Simulation (EMS), 2010 Fourth UKSim European Symposium on, pp.391-396 (2010).
- [8] Martinez-Gil, J., Alba, E., and Montes, J. F. A.: Optimizing ontology alignments by using genetic algorithms. In: Nature inspired Reasoning for the Semantic Web (NatuReS2008), volume 419 of CEUR Workshop Proceedings (2008).
- [9] Wilcoxon, F.: Individual comparisons by ranking methods. In: Biometrics, vol.1, pp.80-83 (1945).
- [10] Euzenat, J. and Shvaiko, P.: Ontology matching. Springer (2007).
- [11] Acampora, G., Loia, V., Salerno, S. and Vitiello, A.: A hybrid evolutionary approach for solving the ontology alignment problem. In: International Journal of Intelligent Systems, vol. 27, pp. 189-216 (2012).
- [12] Bock, J., Hettenhausen, J.: Discrete particle swarm optimisation for ontology alignment. In: Information Sciences. Article in Press (2010).
- [13] Krasnogor, N. and Smith, J.: A Tutorial for Competent Memetic Algorithms: Model, Taxonomy, and Design Issues. In: IEEE Transactions On Evolutionary Computation, vol. 9, no. 5, (2005).
- [14] Garcia, S., Molina, D., Lozano, M., Herrera, F.: A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behaviour: a case study on the CEC'2005 Special Session on Real Parameter Optimization. In: Journal of Heuristics, vol.15, pp.617-644 (2009).
- [15] Vazquez Naya, J.M., Martinez Romero, M., Pereira Loureiro, J., Munteanu, C.R., and Pazos Sierra, A.: Improving Ontology Alignment through Genetic Algorithms. In: Soft Computing Methods for Practical Environment Solutions: Techniques and Studies, pp. 240-259 (2010).
- [16] Ginsca, A.-L., and Iftene, A.: Using a genetic algorithm for optimizing the similarity aggregation step in the process of ontology alignment. In: Roedunet International Conference (RoEduNet), 2010 9th, pp. 118-122 (2010).