GFSOM: Genetic Feature Selection for Ontology Matching

Hiba Belhadi[1], Karima Akli-Astouati[1], Youcef Djenouri[2], and Jerry Chun-Wei Lin[3]

[1] Computer Science Department, University of Science and Technology Houari Boumediene (USTHB), Algiers, Algeria
[2] Computer and Information Sciences Department, Norwegian University of Science and Technology (NTNU), Trondheim, Norway
[3] Department of Computing, Mathematics, and Physics, Western Norway University of Applied Sciences (HVL), Bergen, Norway
{*hbelhadi*, *kakli*}@*usthb.dz*, *youcef.djenorui@ntnu.no*, *jerrylin@ieee.org*

Abstract. This paper studies the ontology matching problem and proposes a genetic feature selection approach for ontology matching (GFSOM), which exploits the feature selection using the genetic approach to select the most appropriate properties for the matching process. Moreover, three strategies are proposed. The genetic algorithm is first performed to select the most relevant properties, the matching process is then applied to the selected properties instead of exploring all properties of the given ontology. To demonstrate the usefulness and accuracy of the GFSOM framework, several experiments on DBpedia ontology database are conducted. The results show that the ontology matching process benefits from the feature selection and the genetic algorithm, where GFSOM outperforms the state-of-the-art ontology matching approaches in terms of both the execution time and quality of the matching process.

Keywords: Semantic Web, Ontology Matching, Feature Selection, Genetic Algorithm

1 Introduction and Related Work

Ontology matching is the process to find the correspondences between different ontologies represented by the set of of instances, where each instance is characterized by different properties. It is applied in diverse fields such as biomedical data [1], e-learning [2], and Natural Language Processing [3]. Ontology matching is a polynomial problem in terms of number of instances and number of properties,, where the trivial algorithm for ontology matching compares each instance of the first ontology with each instance of the second ontology by taking into account all the properties of both ontologies. However, for some high dimensional data like DBpedia ontology [1], the runtime of the trivial algorithms became high time consuming. To overcome this drawback, some evolutionary approaches have been

[1] http://wiki.dbpedia.org/Datasets

developed. The proposed systems presented in [4, 5] aim to find the similarities between concepts of the two ontologies. The useless of the genetic algorithm is to achieve an approximation case close to the optimal alignment between the two ontologies. The works suggested in [6, 7], proposed several fitness computing for the ontology matching problem on the genetic process, including maximizing precision, recall, Fmeasure, and optimizing weights for aggregating more similarities, where the work developed in [8] aims to reduce the memory consumption by using hybrid genetic algorithm and incremental learning process. These evolutionary-based approaches improved considerably the runtime performance of the ontology matching problem. However, the overall performance of these algorithms are still low when dealing with high dimensional data. To deal with this challenging issue, and motivated by evolutionary techniques, well applied for solving real world complex problems [9–12], this paper proposes a feature selection approach called GFSOM that explores the genetic process for solving the ontology matching problem. To the best of our knowledge, this is the first work that explores both feature selection and genetic algorithm as pre-processing step for the ontology matching problem. An intensive experiments have been performed to demonstrate the usefulness of the suggested framework. The results reveal that GFSOM outperforms the state-of-the art ontology matching algorithms on the well-known DBpedia database.

The rest of this paper is organized as follows. Section 2 presents a detail explanation of the GFSOM framework. The evaluation of the GFSOM performance is provided in Section 3. Section 4 gives the conclusions and perspectives for our future work.

2 GFSOM: Genetic Feature Selection for Ontology Matching

In this part, we present the main components of the proposed framework called GFSOM (Genetic Feature Selection for Ontology Matching). The aim of GFSOM is to improve the ontology matching based instance problem by taking into account the relevant features of the two ontologies to be aligned. This reducing allows on the one hand to boost the matching process for finding the common instances between two ontologies, on the other hand, it aims to improve the quality of the resulted alignment. GFSOM is mainly composed into two steps: feature selection and matching process steps (See Figure 1 for more details). The feature selection step is first performed to the set of attributes for each ontology, which results an optimal subset of attributes that represents perfectly the two ontologies. This step is considered as pre-processing step, (It will be executed only one time). To do so, an archive folder will be constructed for each ontology in the ontology base system. In this step, a genetic algorithm is performed to improve the feature selection process without losing on the quality of the resulted features. The process starts by generating randomly a $PopSize$ individuals from the set of m properties. Each individual is a binary vector of m elements, the i^{th} element is set to 1, if the i^{th} property is selected, otherwise, it is set to 0.

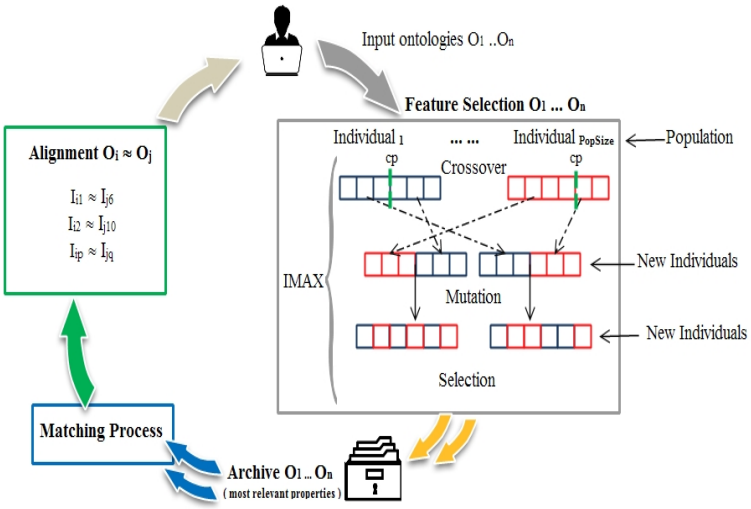
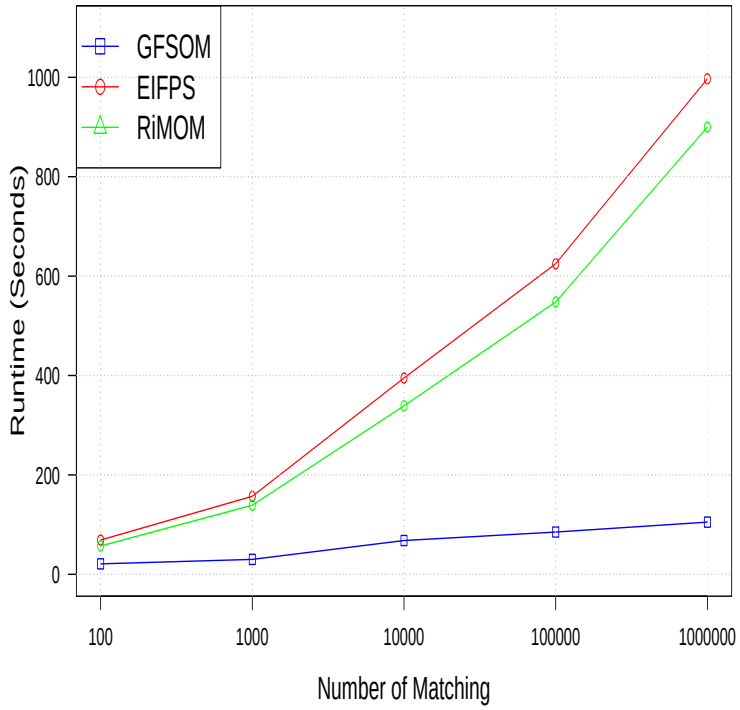


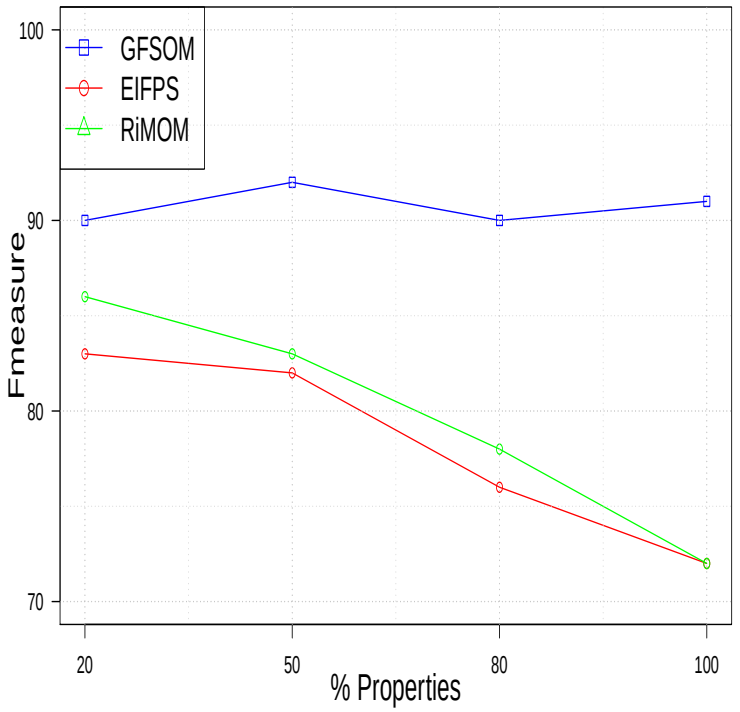
Fig. 1. GFSOM Architecture

The crossover, the mutation, and the selection operators are then performed. The crossover operator is done by merging two individuals x_1, and x_2 of the initial population, which yields two new candidate called x_{11} and x_{12} such as the first cp properties of x_1 are transferred to x_{11}, and the first c properties of x_2 are transferred to x_{12}. The remaining properties of x_1, and x_2 are transferred to x_{12}, respectively x_{11}. Note that cp is the crossover point selected randomly between 1 and m. The mutation is then applied on the new generated individuals by switching randomly a property to 0 if it is present in the given individual, 1, otherwise. At the end of each iteration, the selection operator is launched To keep the same population size, where all individuals are evaluated using a fitness function. It is determined using the information gain value of the selected properties, the aim is to maximize the fitness function value. Then, GFSOM keeps only the best $PopSize$ individuals (the others are removed). This process is repeated until the maximal number of iterations is reached. Afterwards, the matching process is applied between the instances of the ontologies by taking only the attributes selected of the above step. The K-cross-validation model is used here, where at each pass of the algorithm, the training and the test alignments are performed. For the training matching process, the proposed model is learned to fix the best parameters. If the alignment rate exceeds the given threshold, then the test alignment is started.

Fig. 2. Comparison of the runtime performance (a) and Fmeasure (b) of the GFSOM, the EIFPS, and the RiMOM using the DBpedia database



(a)



(b)

3 Performance Evaluation

To validate the usefulness of the proposed GFSOM framework, extensive experiments were carried out using the well-known DBpedia database [2]. It is a hub data that can be found on Wikipedia. This database ontology contains 4,233,000 instances and 2,795 different properties. All algorithms were implemented in Java programming language, and experiments were run on a desktop machine equipped with an Intel *I7* processor and $16GB$ memory. The quality of the ontology matching process was evaluated using the Fmeasure, for each reference alignment R, and each alignment A as follows: $Fmeasure(A,R) = \frac{2\times Precision\times Recall}{Precision+Recall}$, where $Precision(A,R) = \frac{|R\cap A|}{|A|}$, and $Recall(A,R) = \frac{|R\cap A|}{|R|}$.

The aim of this experiment is to compare the GFSOM with the state-of-the-art algorithms (EIFPS [13], and the RiMOM[14]) using the DBPedia ontology database. Figure 2.(a) show the runtime of the three approaches considering all instances, and properties. When the number of matchings varied from 100 to $1,000,000$, the GFSOM outperformed the two other approaches. Moreover, the runtime of the GFSOM stabilized at 105 seconds, where the two approaches were highly time-consuming, and need more than 900 seconds for dealing $1,000,000$ matchings in the whole DBpedia ontology database. for a large number of instances and a large number of matchings. These results were obtained using the preprocessing step, where only the most relevant features were selected using the genetic approach.

The last experiment was performed to compare the quality of matching of the GSFOM framework and the baseline algorithms (the EIFPS and the RiMOM) using the DBpedia ontology database. By varying the percentage of properties from 20% to 100%, the GFSOM outperformed the other two algorithms regarding the Fmeasure value (See Figure 2.(b)). Moreover, the results showed that the quality of the GFSOM was not affected by the increase in the number of properties. Thus, the quality of the GFSOM was up to 90%, whereas the quality of the EIFS and the RiMOM was under 72%. These results were obtained thanks to the feature selection and the genetic approach that selected the most relevant properties of the ontologies.

4 Conclusion

This paper explored both feature selection and genetic algorithm to improve the ontology matching process. It investigates GFSOM framework, where the genetic algorithm is first performed to select the most relevant properties, the matching process is then applied to the selected properties instead of exploring all properties of the given ontology. To evaluate the GFSOM framework, the intensive experiments were carried out on DBpedia database. The results show that the ontology matching process benefits from the feature selection and the genetic algorithm, where GFSOM outperforms the state-of-the-art ontology matching

[2] http://wiki.dbpedia.org/Datasets

approaches in terms of both the execution time and quality of the matching process. In our future work, we will explore other data mining techniques for ontology matching problem. In this context, we aim to use both clustering [15], frequent pattern mining [11], for dealing with big ontology databases.

References

1. Smith B, Ashburner M, Rosse C, Bard J, Bug W, Ceusters W, et al. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. Nature biotechnology. 2007;25(11):1251.
2. Cerón-Figueroa S, López-Yáñez I, Alhalabi W, Camacho-Nieto O, Villuendas-Rey Y, Aldape-Pérez M, et al. Instance-based ontology matching for e-learning material using an associative pattern classifier. Computers in Human Behavior. 2017;69:218–225.
3. Iwata T, Kanagawa M, Hirao T, Fukumizu K. Unsupervised group matching with application to cross-lingual topic matching without alignment information. Data mining and knowledge discovery. 2017;31(2):350–370.
4. Wang J, Ding Z, Jiang C. Gaom: Genetic algorithm based ontology matching. In: Services Computing, 2006. APSCC'06. IEEE Asia-Pacific Conference on. IEEE; 2006. p. 617–620.
5. Acampora G, Loia V, Salerno S, Vitiello A. A hybrid evolutionary approach for solving the ontology alignment problem. International Journal of Intelligent Systems. 2012;27(3):189–216.
6. Martinez-Gil J, Alba E, Aldana-Montes JF. Optimizing ontology alignments by using genetic algorithms. In: Proceedings of the workshop on nature based reasoning for the semantic Web. Karlsruhe, Germany; 2008. .
7. Acampora G, Loia V, Vitiello A. Enhancing ontology alignment through a memetic aggregation of similarity measures. Information Sciences. 2013;250:1–20.
8. Xue X, Chen J. Optimizing ontology alignment through hybrid population-based incremental learning algorithm. Memetic Computing. 2018;p. 1–9.
9. Djenouri Y, Belhadi A, Fournier-Viger P, Lin JCW. Fast and effective cluster-based information retrieval using frequent closed itemsets. Information Sciences. 2018;453:154–167.
10. Djenouri Y, Djamel D, Djenoouri Z. Data-Mining-Based Decomposition for Solving MAXSAT Problem: Towards a New Approach. IEEE Intelligent Systems. 2017;.
11. Djenouri Y, Belhadi A, Fournier-Viger P, Lin JCW. An hybrid multi-core/GPU-based mimetic algorithm for big association rule mining. In: International Conference on Genetic and Evolutionary Computing. Springer; 2017. p. 59–65.
12. Lin JCW, Zhang Y, Fournier-Viger P, Djenouri Y, Zhang J. A Metaheuristic Algorithm for Hiding Sensitive Itemsets. In: International Conference on Database and Expert Systems Applications. Springer; 2018. p. 492–498.
13. Niu X, Rong S, Wang H, Yu Y. An effective rule miner for instance matching in a web of data. In: Proceedings of the 21st ACM international conference on Information and knowledge management. ACM; 2012. p. 1085–1094.
14. Shao C, Hu LM, Li JZ, Wang ZC, Chung T, Xia JB. RiMOM-IM: a novel iterative framework for instance matching. Journal of computer science and technology. 2016;31(1):185–197.
15. Otto C, Wang D, Jain AK. Clustering millions of faces by identity. IEEE transactions on pattern analysis and machine intelligence. 2018;40(2):289–303.