Identifying Common Research Interest through Matching of Ontological Research Profiles

Nawarat Kamsiang and Twittie Senivongse

***Abstract*— In this paper, we present an ontology-based methodology for identifying common interest among researchers. The methodology uses an ontology building algorithm to build researchers' ontological research profiles from publication keywords. Then an ontology matching algorithm is used to find similarity between research profiles. Our ontology matching also considers location of the ontological terms within the two profiles that are matched. That is, the terms that are located near the bottom of the ontologies should indicate specialization of researchers, and more attention should be paid to matching of such terms than to the matched terms that are closer to the top of the ontologies. Particularly, this paper experiments with cross-field collaboration through matching of research profiles from different fields, and reports an evaluation using an ontology matching benchmark and a performance evaluation of the methodology. We consider that the methodology is useful as it can quantify similarity of research interest as well as give practical similarity results.**

***Index Terms*—ontology building, ontology matching, profile matching, research expertise.**

I. Introduction

Identifying common research interest is a challenging task for promoting research collaboration. Researchers seek collaboration with one another for sharing ideas and resources, complementing one's expertise with others', as well as increasing visibility of the research work and the researchers themselves. Collaboration between researchers in the same field can specifically strengthen the work within the field while collaboration between different fields may lead to useful innovative work with wider application across fields.

The basis of identifying shared research interest is analyzing researchers' expertise and finding correspondence between research areas. Primarily, association between researchers can be drawn using bibliographic data of their publications [1]. Researchers who, for example, co-author publications, cite similar publications, or use similar keywords in publications can be identified as sharing common interest. Another approach taken by a great number of related works is gathering researchers' information from electronic sources, e.g., online libraries, Web sites, blogs, and project documents, to build research profiles and mine researchers' expertise.

This paper presents a methodology for identifying common interest among researchers. The methodology uses an ontology building algorithm to build researchers' ontological research profiles from keywords of previous publications. Then an ontology matching algorithm is used to determine the degree of similarity between research profiles and identify matched ontological terms as the shared area of interest. Here the ontology matching algorithm is particularly interested in matching of the terms that are located near the bottom of the ontological profiles, since they are specialization of the researchers, and should represent common interest better than matched terms that are near the top of the ontological profiles. We present a performance evaluation and an evaluation using an ontology matching benchmark to support the methodology.

Section II discusses research work related to this paper. Section III gives the detail of the methodology through an example of researchers in different fields. The evaluation of the methodology is presented in Section IV followed by a conclusion in Section V.

II. Related Work

Information from electronic sources, such as bibliographic databases, Web sites, and discussion forums, has been used widely for analyzing expertise and connection between people. A well-known search and mining tool called ArnetMiner [2] can provide search services including researcher profile search, expert finding, active researchers for conferences, and researcher ranking. To build researcher social network, it extracts researchers' information from the Web to create semantic-based profiles for the researchers and has their bibliographic data from several digital libraries integrated with the profiles. Zhang et al. [3] analyze asker-helper interaction in the Java Forum threads by considering the number of replies any user has posted to help others and whom the user has helped. The analysis uses network-based ranking algorithms, including PageRank and HITS, to identify users with expertise. Punnarut and Sriharee [4] build semantic-based researcher profiles based on ACM computing classification and compute expertise scores, find researchers who share expertise, as well as rank them. Trigo [5] finds researchers with similar interest by using the DBLP bibliographic database and research Web pages as the sources for extracting researcher information, and applying text mining techniques to discover relations between them. Yang et al. [6] construct a social network of researchers by analyzing four types of data, i.e., publication keywords, personal interest, themes of the conferences where papers are published, and co-authorship. An interesting finding is that

Manuscript received June 26, 2012.
N. Kamsiang is with the Office of Computer Service, Sripatum University Chonburi Campus, Chonburi 20000 Thailand (phone: +66 38 743690-9 ext. 3159; email: nawarat.km@east.spu.ac.th).
T. Senivongse is with the Department of Computer Engineering, Faculty of Engineering, Chulalongkorn University, Bangkok 10330 Thailand (phone: +66 2 2186996; fax: +66 2 2186955; e-mail: twittie.s@chula.ac.th).

publication keywords can represent research interest better than co-authorship data. Motivated by these related works, we explore another approach to determining research similarity by representing researchers' profiles as ontologies, which are built upon publication keywords. Then we compare the profiles using an ontology matching algorithm.

III. Methodology

Given two researchers, our methodology determines the degree of similarity and the area of interest that they share. The methodology comprises two steps: building ontological research profiles and matching the profiles. A preliminary report on the methodology and its application to finding shared interest between two researchers in close fields of Computer Science (i.e., machine learning vs. data mining) can be found in [7]. Here we revisit the methodology and experiment with the case of researchers in different fields.

A. Building Ontological Research Profiles

A researcher's profile is built upon publication keywords. A researcher's keywords in a particular subject area during a certain period are taken from ISI Web of Knowledge database [8]. As an example, we select researchers in different subject areas named Kijsirikul, B in Computer Science and Howard, JH in Psychology. We take top five keywords of their publications during the year 2002-2011 as starting terms for building their ontological profiles as depicted in Fig. 1-2. In this section, we describe how to build such profiles. Note that the five starting terms of each researcher are at the bottom of the corresponding profile.

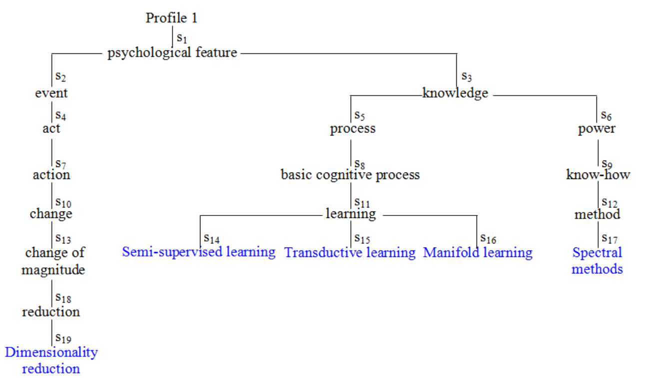

Fig. 1. Kijsirikul, B's ontological profile.

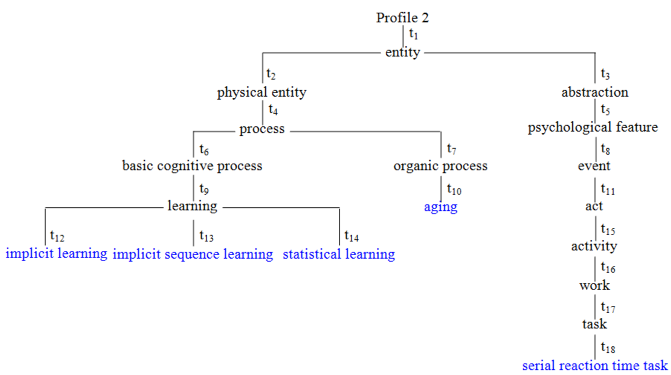

Fig. 2. Howard, JH's ontological profile.

We adopt the Obtaining Domain Ontology (ODO) algorithm proposed by An et al. [9] which can automatically derive a domain-specific ontology from items of information (i.e., keywords in this case). Starting with each keyword, we repeatedly find terms and hypernym (i.e., parent) relation from WordNet [10] to build an ontology fragment as a Directed Acyclic Graph. Since a term may have several hypernyms, for simplicity, we select one with the maximum tag count which denotes that the hypernym of a particular sense (or meaning) is most frequently used and tagged in various semantic concordance texts. For example, in Fig. 2, *aging* has *organic process* as its hypernym, and *organic process* has *process* as its hypernym, and so on. If the term is not found in WordNet but is a noun phrase consisting of a head noun and modifier(s), we generalize the term by removing one modifier at a time to look up in WordNet. If found, that generalized form becomes the hypernym. Otherwise, the researcher's subject area is used as the hypernym. For example, in Fig. 1, the noun phrase *Semi-supervised learning* is not found in WordNet, so we use the head noun *learning* as its hypernym. After we obtain all ontology fragments, we then join together identical terms in different fragments to form an ontology representing a profile of research interest. In Fig. 1, five ontology fragments based on five starting terms are joined at the terms *learning*, *knowledge*, and *psychological feature* respectively.

B. Matching Ontological Research Profiles

After we obtain the ontological profiles of any two researchers, we compare their profiles using ontology matching. The basis of our ontology matching is the algorithm called Multi-level Matching Algorithm with the neighbor search algorithm (MLMA+) proposed by Alasoud et al. [11] as shown in Fig. 3. It consists of initialization, neighbor search, and evaluation phases. We then enhance the initialization phase with the concept of depth weights.

```
Algorithm Match (S, T)
begin
/* Initialization phase
   K ← 0 ;
   St0 ← preliminary_matching_techniques (S, T) ;
   Stf ← St0 ;
/* Neighbor Search phase
   St ← All_Neighbors (Stn) ;
   While (K++ < Max_iteration) do
/* Evaluation phase
      If score (Stn) > score (Stf) then
         Stf ← Stn ;
      end if
      Pick the next neighbor Stn ∈ St;
      St ← St – Stn ;
      If St = ∅ then return Stf ;
   end
   Return Stf ;
end
```

Fig. 3. MLMA+ algorithm [11].

1) Initialization Phase

To match an ontology *S* with another ontology *T*, preliminary matching techniques are applied to determine similarity between terms in the two ontologies. The matching techniques used here are name matching using Levenshtein distance, and linguistic matching using WordNet.

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

Levenshtein distance determines the minimal number of insertions, deletions, and substitutions to make two strings equal [12]. For linguistic matching, we use a Perl module in WordNet::Similarity package [13] to determine semantic similarity between any two terms. For example, to match Kijsirikul's ontology S which comprises n terms with Howard's ontology T comprising m terms, a similarity matrix $L(i, j)$ of the size n x m is computed. This matrix includes values called similarity coefficients, ranging between [0,1] and denoting the degree of similarity between the terms s_i in S and t_j in T. A similarity coefficient is computed as an average of Levenshtein distance and WordNet similarity of the two terms. The similarity matrix L for Kijsirikul and Howard is shown in Fig. 4. The similarity coefficient of the terms *learning* (s_{11}) and *work* (t_{16}) is 0.533; it is an average of Lavenshtein distance (0.125) and WordNet similarity (0.941) of these two terms.

Then, a user-defined threshold *th* is applied to the matrix L to create a binary matrix $Map_{0\text{-}1}$. The similarity coefficient that is less than the threshold becomes 0 in $Map_{0\text{-}1}$, otherwise it is 1. That is, the threshold determines which pairs of terms are considered similar or matched by the user. Fig. 4 also shows $Map_{0\text{-}1}$ for Kijsirikul and Howard with $th = 0.5$. This $Map_{0\text{-}1}$ becomes the initial state St_0 for the neighbor search algorithm.

L =

	t_1	t_2	t_3	t_4 …	t_{11} …	t_{16}	t_{17}	t_{18}
s_1	0.359	0.324	0.454	0.428	0.425	0.359	0.332	0.024
s_2	0.386	0.372	0.421	0.472	0.555	0.496	0.441	0.023
s_7	0.386	0.414	0.452	0.563	0.587	0.542	0.528	0.025
s_{10}	0.405	0.339	0.446	0.438	0.512	0.447	0.437	0.022
s_{11}	0.272	0.221	0.403	0.484	0.348	0.533	0.298	0.024
s_{16}	0.032	0.036	0.034	0.032	0.03	0.032	0.03	0.024
s_{17}	0.036	0.039	0.042	0.039	0.034	0.032	0.034	0.025
s_{19}	0.025	0.03	0.03	0.022	0.023	0.022	0.021	0.023

Map_{0-1} =

	t_1	t_2	t_3	t_4 …	t_{11} …	t_{16}	t_{17}	t_{18}
s_1	0	0	0	0	0	0	0	0
s_2	0	0	0	0	1	0	0	0
s_7	0	0	0	1	1	1	1	0
s_{10}	0	0	0	0	1	0	0	0
s_{11}	0	0	0	0	0	1	0	0
s_{16}	0	0	0	0	0	0	0	0
s_{17}	0	0	0	0	0	0	0	0
s_{19}	0	0	0	0	0	0	0	0

Fig. 4. *L* and initial $Map_{0\text{-}1}$ for Kijsirikul and Howard based on MLMA+.

2) Neighbor Search Phase

Given the initial state St_0, we search in its neighborhood. Each neighbor St_n is computed by toggling a bit of St_0, so the total number of neighbor states is $n*m$. An example of a neighbor state is in Fig. 5.

Map_{0-1} =

	t_1	t_2	t_3	t_4 …	t_{11} …	t_{16}	t_{17}	t_{18}
s_1	0	0	0	0	0	0	0	0
s_2	0	0	0	0	1	0	0	0
s_7	0	0	0	1	1	1	1	0
s_{10}	0	0	0	0	1	0	0	0
s_{11}	0	0	0	0	0	1	0	0
s_{16}	0	0	0	0	0	0	0	0
s_{17}	0	1	0	0	0	0	0	0
s_{19}	0	0	0	0	0	0	0	0

Fig. 5. One of the neighbor states of the initial $Map_{0\text{-}1}$ in Fig. 4.

3) Evaluation Phase

The initial state and all neighbor states are evaluated using the matching score function v (1) [11]:

$$v = (Map_{0-1} \cdot L)/k = \sum_{i=1}^{n}\sum_{j=1}^{m} Map_{0-1}(i,j) \cdot L(i,j) \Big/ \sum_{i=1}^{n}\sum_{j=1}^{m} Map_{0-1}(i,j) \; ; v \ge th \qquad (1)$$

where k is the number of matched pairs and $Map_{0\text{-}1}$ is St_n . The state St_n with the maximum score (i.e., St_f) is the answer to the matching; it indicates which terms in S and T are matched and the score represents the degree of similarity between S and T.

4) Modification with Depth Weights

Inspired by the concept of semantic distance between ontological terms [14], we modify the initialization phase of MLMA+ by adding the concept of depth weights. A depth weight of a pair of matched terms is determined by the distance of the terms from the root of their ontologies. We are interested particularly in matching of the terms that are located near the bottom of the ontological profiles, since they are specialization of the researchers, and should represent common interest better than matched terms that are near the top of the ontological profiles. In Fig. 4, consider the pair s_2 = *event* and t_{11} = *act* with the similarity coefficient = 0.555 and the pair s_{11} = *learning* and t_{16} = *work* with similarity coefficient = 0.533. Both pairs are considered as matched interest. Even though *event* and *act* are more similar to each other than *learning* and *work* are, they are relatively more generalized terms. We are in favor of the matched pairs that are more specialized and are motivated to decrease the similarity coefficient of the generalized matched pairs by using a depth weight function w (2):

$$w_{ij} = (rdepth(s_i) + rdepth(t_j))\,/\,2\;;\; w_{ij} \text{ is in } (0,1] \qquad (2)$$

where $rdepth(t)$ = relative distance of the term t from the root of its ontology

$$= \frac{\text{depth of the term } t \text{ in its ontology}}{\text{height of ontology}}.$$

The similarity coefficient between s_i and t_j will be multiplied by their depth weight to get a *weighted similarity coefficient*. That is, the more generalized the matched pair, the more they are "penalized" by the depth weight. Any matched pair that are both the terminal nodes of the ontologies would not be penalized (i.e., w =1). Fig. 6 shows the new similarity matrix L for Kijsirikul and Howard with weighted similarity coefficients, together with their new initial $Map_{0\text{-}1}$ where th = 0.35. For the pair s_2 = *event* and t_{11} = *act*, their depth weight is (2/8 + 5/9)/2 = 0.403 and their weighted similarity coefficient is 0.555*0.403 = 0.224. Note that they are considered similar in Fig. 4 but not in Fig. 6 since they are relatively generalized terms and their depth weight lessens the degree of their similarity. On the other hand, the pair s_{11} = *learning* and t_{16} = *work* are still considered similar even with the depth weight.

L =

	t_1	t_2	t_3	t_4 …	t_{11} …	t_{16}	t_{17}	t_{18}
s_1	0.042	0.056	0.079	0.098	0.145	0.162	0.168	0.014
s_2	0.07	0.088	0.099	0.138	0.224	0.255	0.251	0.014
s_7	0.118	0.149	0.163	0.234	0.31	0.346	0.367	0.019
s_{10}	0.149	0.144	0.189	0.21	0.302	0.314	0.331	0.018
s_{11}	0.1	0.094	0.171	0.232	0.205	0.374	0.226	0.02
s_{16}	0.014	0.017	0.017	0.017	0.02	0.024	0.025	0.021
s_{17}	0.015	0.019	0.02	0.021	0.022	0.024	0.028	0.022
s_{19}	0.014	0.018	0.018	0.015	0.018	0.02	0.02	0.023

Map_{0-1} =

	t_1	t_2	t_3	t_4 …	t_{11} …	t_{16}	t_{17}	t_{18}
s_1	0	0	0	0	0	0	0	0
s_2	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	1	0
s_{10}	0	0	0	0	0	0	0	0
s_{11}	0	0	0	0	0	1	0	0
s_{16}	0	0	0	0	0	0	0	0
s_{17}	0	0	0	0	0	0	0	0
s_{19}	0	0	0	0	0	0	0	0

Fig. 6. *L* and initial $Map_{0\text{-}1}$ for Kijsirikul and Howard based on MLMA+ with depth weights.

5) Result of Ontology Matching

As an example, Table I shows the matching result for Kijsirikul, B (Computer Science area) and Howard, JH (Psychology area). MLMA+ gives a big list of matched pairs that represent the shared interests. The list includes those very

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

generalized terms which are located near the top of the profiles. Depth weights, on the other hand, lessen the effect of similarity coefficients and hence lower the matching score. As a result, they filter out some generalized matched pairs, giving a concise list of shared interests which should be more practical for use.

TABLE I
MATCHING RESULT FOR KIJSIRIKUL AND HOWARD

Algorithm	Matching Score	Matched Pairs (s_i, t_j)
MLMA+	0.638	(psychological feature, psychological feature), (event, event), (event, act), (knowledge, process), (power,process), (power, event), (power, work), (process, process), (process, act), (process, activity), (process, work), (act, process), (act, event), (act, act), (act, activity), (act, work), (act, task), (action, process), (action, act), (action, activity), (action, work), (action, task), (basic cognitive process, basic cognitive process), (method, activity), (change, event), (change, act), (learning, learning), (learning, work), (process, organic process)
MLMA+ with Depth Weights	0.426	(process, process), (act, act), (action,task), (basic cognitive process, basic cognitive process), (method, task), (learning, learning), (learning, work)

IV. EVALUATION

This section presents a performance evaluation of the methodology and a comparison between the original MLMA+ and the modified MLMA+ with depth weights based on the profile building and matching tool that we have implemented.

A. *Processing Performance*

Obviously the size of the researcher's profile should increase with a larger number of starting keywords. Table II shows that the number of terms in Kijsirikul's and Howard's profiles increases with the increasing number of starting keywords. Also, this tends to increase the number of their similar matched pairs.

TABLE II
NUMBER OF TERMS AND MATCHED PAIRS FOR KIJSIRIKUL AND HOWARD

No.of Starting Keywords	No.of Terms in Kijsirikul's Profile	No.of Terms in Howard's Profile	No.of Matched Pairs (MLMA+)	No.of Matched Pairs (MLMA+ with Depth Weights)
5	20	18	29	7
6	24	26	37	6
7	30	33	47	7
8	36	34	62	9
9	38	35	58	17

The graph in Fig. 7 shows the time taken to build the profiles for Kijsirikul and Howard. The complexity of the ODO algorithm for building an ontology S depends on the number of terms in S and the size of the search space when joining any identical terms in S into single nodes, i.e., O($\binom{n}{2}$), where the number of ontology terms n = number of starting keywords * depth of S, given that, in the worst case, all starting keywords are of the same depth.

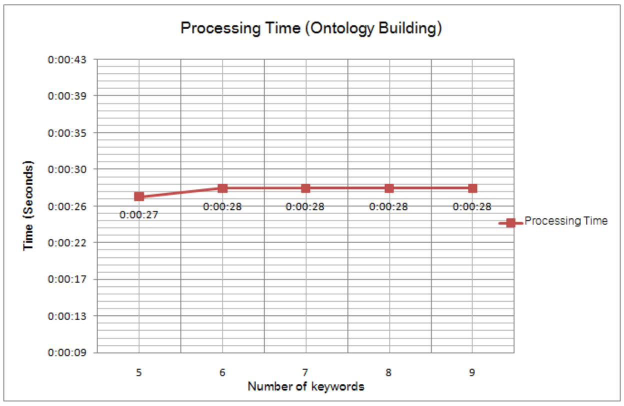

Fig. 7. *Processing time for building profiles for Kijsirikul and Howard.*

For MLMA+ and MLMA+ with depth weights, their complexity depends on the size of the search space when matching two ontologies S and T, i.e., $O((n*m)^2)$ when n and m are the size of S and T respectively. The graph in Fig. 8 shows the time taken to match Kijsirikul's profile with Howard's using MLMA+ and MLMA+ with depth weights. The latter takes less time due to less number of similar matched pairs being considered.

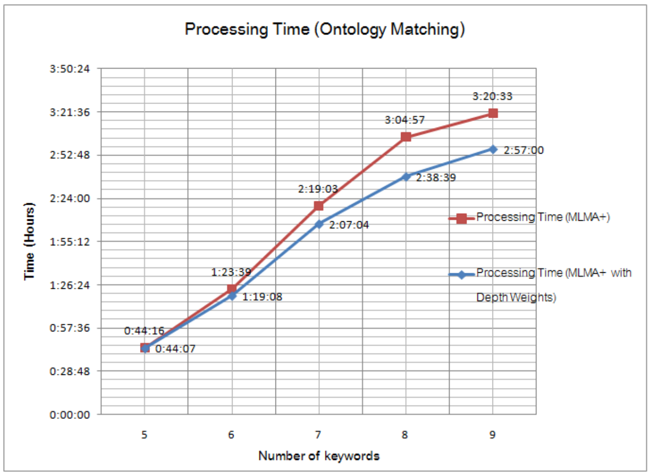

Fig. 8. *Processing time for matching profiles for Kijsirikul and Howard.*

B. *Performance Assessment Using OAEI Benchmark*

Ontology matching is evaluated using the OAEI 2011 benchmark test sample suite [15]. The benchmark provides a number of test sets in a bibliographic domain, each comprising a test ontology in OWL language and a reference alignment. Each test ontology is a modification to the reference ontology #101 and is to be aligned with the reference ontology. Each reference alignment lists expected alignments. So in the test set #101, the reference ontology is matched to itself, and in the test set #n, the test ontology #n is matched to the reference ontology. The quality indicators we use are precision (3), recall (4), and F-measure (5).

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

$$\text{Precision} = \frac{\text{no. of expected alignments found as matched by algo.}}{\text{no. of matched pairs found by algo.}} \quad (3)$$

$$\text{Recall} = \frac{\text{no. of expected alignments found as matched by algo.}}{\text{no. of expected alignments}} \quad (4)$$

$$\text{F} - \text{measure} = \frac{2 \text{ x Precision x Recall}}{\text{Precision} + \text{Recall}} \quad (5)$$

Table III shows the evaluation results with *th* = 0.5. We group the test sets into four groups. Test set #101-104 contain test ontologies that are more generalized or restricted than the reference ontology by removing or replacing OWL constructs that make the concepts in the reference ontology generalized or restricted. Test set #221-247 contain test ontologies with structural change such as no specialization, flattened hierarchy, expanded hierarchy, no instance, no properties. The quality of both algorithms with respect to these two groups is quite similar since these modifications do not affect string-based and linguistic similarities which are the basis of both algorithms. Test set #201-210 contain test ontologies which relate to change of names in the reference ontology, such as by renaming with random strings, misspelling, synonyms, using certain naming convention, and translation into a foreign language. Both algorithms are more sensitive to this test set. Test set #301-304 contain test ontologies which are actual bibliographic ontologies.

TABLE III
PERFORMANCE ASSESSMENT USING OAEI 2011

Test Set	MLMA+			MLMA+ w. Depth Weights		
	Precision	Recall	F-measure	Precision	Recall	F-measure
#101-104	0.74	1.0	0.85	0.93	0.84	0.88
#201-210	0.35	0.24	0.26	0.68	0.18	0.27
#221-247	0.71	0.99	0.82	0.94	0.66	0.75
#301-304	0.56	0.75	0.64	0.90	0.57	0.68
Average	0.59	0.74	0.64	0.86	0.56	0.64

According to an average F-measure, both algorithms are relatively of the same quality while MLMA+ with depth weights gives better precision but lower recall. MLMA+ discovers a large number of similar matched pairs whereas depth weights can decrease this number and hence precision is higher. But at the same time, recall is affected. This is because the reference alignments only lists pairs of terms that are expected to match. That is, for example, if the test ontology and the reference ontology contain the same term, the algorithm should be able to discover a match. But MLMA+ with depth weights considers the presence of the terms in the ontologies as well as their location in the ontologies. So an expected alignment in a reference alignment may be considered unmatched if they are near the root of the ontologies and are penalized by depth weights.

The user-defined threshold *th* in the initialization phase of ontology matching can affect precision and recall. If *th* is too high, only identical terms from the two ontologies would be considered as matched pairs (e.g., (*psychological feature*, *psychological feature*)), and these identical pairs mostly are located near the root of the ontologies. We see that discovering only identical matched pairs are not very interesting given that the benefit of using WordNet and linguistic similarity between non-identical terms would not be present in the matching result. On the contrary, if *th* is too low, there would be proliferation of matched pairs because, even a matched pair is penalized by its depth weight, its weighted similarity coefficient remains greater than the low *th*. The values *th* that we use for the data set in the experiment trades off these two aspects; it is the highest threshold by which the matching result contains both the identical and non-identical matched pairs.

C. *Comparison of Matching Scores*

Using MLMA+ and MLMA+ with depth weights, we compare the matching scores they compute. For this purpose, another researcher named Einwohner, RL in Sociology area is introduced and the corresponding research profile, built from five starting keywords also, is shown in Fig. 9. Table IV compares the matching scores between Kijsirikul and Howard and between Kijsirikul and Einwohner. With MLMA+, the degree of similarity between Kijsirikul's and Howard's profiles is roughly that same as that between Kijsirikul's and Einwohner's. With MLMA+ with depth weights, Kijsirikul and Howard share more common interests than Kijsirikul does with Einwohner. It is seen that MLMA+ with depth weights can better differentiate between the matching results since the matched pairs that mostly are abstract terms near the top of the ontologies are filtered out by depth weights; in MLMA+, such abstract matched pairs still have a strong impact on ontology matching, resulting in similar matching scores.

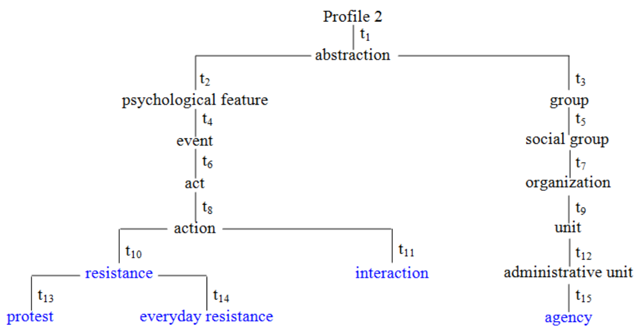

Fig. 9. Einwohner, RL's ontological profile.

TABLE IV
MATCHING SCORES FOR KIJSIRIKUL, HOWARD, AND EINWOHNER

Algorithm	Profile *S*	Profile *T*	Matching Score
MLMA+	Kijsirikul, B	Howard, JH	0.638
		Einwohner, RL	0.633
MLMA+ w. Depth Weights	Kijsirikul, B	Howard, JH	0.426
		Einwohner, RL	0.396

D. *Discussion*

Despite lower recall, we see that the concept of depth weights contributes something good to ontological profile matching since it can give a concise workable matching result. Both MLMA+ and MLMA+ with depth weights, however, do not consider the context of the terms so they can get a false-positive matching result if two ontologies contain a homograph. Another issue with the methodology is that research keywords are very technical and specific, and cannot

ISBN: 978-988-19251-6-9
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

be found in WordNet. We have to rely on the subject area or the generalized form of the keywords to form the ontology. Also, considering the noun phrase pattern or a tag count of a term is merely a way to resolve a problem although it may not give the most appropriate hypernym for a particular context. It is often the case that the specialized keywords are at the bottom of the ontology and all other terms built up by WordNet are more general and even abstract terms. As mentioned earlier, in our previous publication [7], we experiment with researchers in the same subject area (i.e., Computer Science). We have found that the research profiles and matching results are not very different from the case of researchers in different fields as presented in this paper (i.e., Computer Science vs. Psychology and Computer Science vs. Sociology), in the sense that the research profiles mostly comprise general and abstract terms. Therefore the matching results contain matched pairs of such general terms. Our approach is in contrast to most related work which uses research-related information sources (e.g., personal and project Web sites) or relies on taxonomies of research areas. Although WordNet is not a database of research terminologies, we still see that it is a challenge to build research profiles from this rich source of information. It should be particularly useful for the research areas in which keywords are general terms and not very technical.

V. Conclusion

This work explores the idea of an ontology-based methodology for building research profiles from ISI keywords and WordNet terms, and for finding similarity between the profiles. Relying on name similarity and linguistic similarity, the methodology can determine the degree of similarity between the profiles as well as matched terms that represent similar research interests. We adopt the ODO algorithm for ontology building and MLMA+ algorithm for ontology matching and present a modification to MLMA+ with the concept of depth weights. A number of evaluations indicate that the methodology can give useful matching results.

For future work, further evaluation using a larger corpus or the latest OAEI 2012 benchmark which has just been released can be expected. An experience report on practical use of the methodology will also be presented. In addition, it is possible to adjust the ontology matching step so that the structure of the ontologies and the context of the terms are considered.

References

[1] Y. Okubo, *Bibliometric Indicators and Analysis of Research Systems: Methods and Examples*. Paris: OECD Publishing, 1997.
[2] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, "ArnetMiner: Extraction and mining of academic social networks," in *Procs. of 14th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (KDD 2008)*, 2008, pp. 990-998.
[3] J. Zhang, M. Ackerman, and L. Adamic, "Expertise network in online communities: Structure and algorithms," in *Procs. of 16th Int. World Wide Web Conf. (WWW 2007)*, 2007, pp. 221-230.
[4] R. Punnarut and G. Sriharee, "A researcher expertise search system using ontology-based data mining," in *Procs. of 7th Asia-Pacific Conference on Conceptual Modelling (APCCM 2010)*, 2010, pp. 71-78.
[5] L. Trigo, "Studying researcher communities using text mining on online bibliographic databases," in *Procs. of 15th Portuguese Conf. on Artificial Intelligence*, 2011, pp. 845-857.
[6] Y. Yang, C. A. Yueng, M. J. Weal, and H. C. Davis, "The researcher social network: A social network based on metadata of scientific publications," in *Procs. of Web Science Conf. 2009 (WebSci 2009)*, 2009.
[7] N. Kamsiang and T. Senivongse, "An ontological analysis of common research interest for researchers," in *Procs. of 8th Int. Conf. on Computing and Information Technology (IC^2IT 2012)*, 2012, pp. 163-168.
[8] ISI Web of Knowledge, http://www.isiknowledge.com, Last accessed: June 26, 2012.
[9] Y. J. An, J. Geller, Y. Wu, and S. A. Chun, "Automatic generation of ontology from the deep Web," in *Procs. of 18th Int. Workshop on Database and Expert Systems Applications (DEXA'07)*, 2007, pp. 470-474.
[10] WordNet, http://wordnet.princeton.edu/, Last accessed: June 26, 2012.
[11] A. Alasoud, V. Haarslev, and N. Shiri, "An effective ontology matching technique," in *Procs. of 17th Int. Conf. on Foundations of Intelligent Systems*, 2008, pp. 585-590.
[12] G. Navarro, "A guided tour to approximate string matching," *ACM Computing Surveys*, vol.33, pp. 31-88, March 2001.
[13] Wordnet::Similarity, http://sourceforge.net/projects/wn-similarity, Last accessed: June 26, 2012.
[14] H. Yang, S. Liu, P. Fu, H. Qin, and L. Gu, "A semantic distance measure for matching Web services," in *Procs. of Int. Conf. on Computational Intelligence and Software Engineering (CiSE)*, 2009, pp. 1-3.
[15] Ontology Alignment Evaluation Initiative 2011 Campaign, http://oaei.ontologymatching.org/2011/, Last accessed: June 26, 2012.