A Framework of Automatic Alignment of Concept in Ontology with Confidence Score based on Inner Concept Information

Panadda Jaiboonlue[1], Supot Nitsuwat[1], Wasan Na Chai[2], Prasert Luekhong[3], Taneth Ruangrajitpakorn[2] and Thepchai Supnithi[2]

[1] Faculty of Information Technology
King Mongkut's University of Technology North Bangkok, Thailand
pjaiboonlue@gmail.com, sna@kmutnb.ac.th
[2] Language and Semantic Technology Laboratory
National Electronics and Computer Technology Center (NECTEC), Pathumthani, Thailand
{ wasan.na_chai, taneth.rua, thepchai.supnithi}@nectec.or.th
[3] College of Integrated Science and Technology
Rajamangala University of Technology Lanna, Chiang Mai, Thailand
prasert@rmutl.ac.th

Abstract. This paper presents a framework to align synonymous concepts of multiple ontologies. It applies the information attached to the concept including label, description and property relations. Label is a feature to consider for likeness of concept's name which can be the same, partial alike, or totally different. Description is an optional feature in case the given definition of the concepts is similar. Properties of the concept are the major feature to indicate the equivalent relation of the concepts to another concepts and their datatype. After the equivalent concepts are assigned, confidence score is calculated to provide a confidence value of the alignments. From the result, the system gains the impressive result as it can align synonymous concepts as same as the manual mapping concept list.

Keywords: ontology alignment, ontology matching, sericulture

1 Introduction

Ontological products become more popular nowadays due to ontology advantage [1] such as re-usability, interoperability of human and machine, etc. and several supported tools for ontology development such as ontology editor [2][3], inference engine [4], etc. Hence, there are many implemented ontologies using in active researches at the moment. From the observation, several ontologies in the same topic were developed and published freely for re-using and knowledge-sharing. However, ontology and ontology-based system developers often ignore the existing ontologies and decide to design and develop their own ontology since the scope of ontology of the same topic is slightly or subjectively different. This leads to the increasing number of several new ontologies in the same topic and the reusability and extendable benefit of ontology cannot practically be explicit as claimed.

Creating a new ontology is not the hardest part in the development process, but to include the well-designed class and properties of the existing one. Normally, ontology developers review the existing relevant ontologies in the topic as a reference to over-run the weakness or fill out their own interesting scope. However, the reference on-tologies can contain a large number of non-relevant concepts and their relations, and as aforementioned, the number of ontologies in such topic can be numerous. Hence, the assisting tools to help on finding out the classes in ontologies can be useful to indicate the interesting concepts. Moreover, to review many ontologies in the same topic can be helpful on comparing the coverage and missing applicable concepts, but the number of concepts to examine can be greatly burden to reviewers or developers who want to extend existing ontologies.

For comparing several ontologies, it is simple to acknowledge the equivalent class with the same or similar label. However, there are the cases which are 1) ontological classes are equivalent in different label, and 2) classes refer to different concepts with the same label. These issues require much knowledge and understanding in the field from the readers.

Since the ontological concepts include the essential attributes such as concept la-bel, properties of concepts and hierarchical structures, the mentioned information is the hint to inform the likeness of ontological concept. The more similar the infor-mation is, the more likely those concepts are synonym to each other. Thus, we use the information as a clue to identify the likeness of concepts from between ontologies to develop the assisting tools to align synonym concepts.

In this paper, the rest is organised as following. Section 2 gives information on re-lated work on existing concept alignment systems. Section 3 provides the methodolo-gy of the proposed framework. Experiment setting and results are given in Section 4. Section 5 is filled with discussion over the results and methodology. Section 6 con-cludes the paper and lists the plan for future development.

2 Related work

This section shows the existing matching ontology approaches. Several matching application ware proposed such as SAMBO[5], Falcon[6], DSsim[7], RiMOM[8], ASMOV[9] and Anchor Flood [10]. The efficiency of those approves were described in Table 1.

Table 1. Analytical comparison of the recent matching systems from[11]

System	Input	Output	Terminological	Structural
SAMBO	OWL	1:1 Alignments	n-gram, Edit distance, UML, WordNet	Iterative structural similarity base on *is-a*, *part-of* hierarchies
Falcon	RDFS,OWL	1:1 Alignments	I-SUB, Virtual Documents	Structural proximities, Clustering, GMO
DSsim	OWL,SKOS	1:1 Alignments	Tokenization, WordNet Monger- Elkan, Jaccard	Graph similarity base on leaves
RiMOM	OWL	1:1 Alignments	Edit distance, WordNet Vector distance	Similarity Propagation
ASMOV	OWL	n:m Alignments	Tokenization, WordNet String Equality, UML Levenstein distance	Iterative fix point computation, hierarchical, Restriction similarities
Anchor Flood	RDFS,OWL	1:1 Alignments	Tokenization, WordNet String Equality, Winkler-base sim.	Internal/External similarities, Iterative anchor-based similarity propagation
Agreement Maker	XML,RDFS, OWL,N3	n:m Alignments	TF IDF, Edit distance, Substring, WordNet	Descendant, sibling similarities
Propose System	OWL	1:1 Alignments	String similarity, WordNet, Description	Structural Properties

Those approaches are considered the fine systems as they were publically used and tested in several ways. They have their own advantages and disadvantages as shown in Table 1. However, they can decide the matching of the concepts with their criteria, but none of them can give a confident reason to endorse their decision. To solve such problem, we propose a concept matching system which provides the automatic matching of synonym concepts with confidence score in this paper.

3 Methodology

In this work, the new alignment process for matching synonym concepts between the multi-ontologies is proposed. The system employs the alignment function to identify the synonymy concepts between two ontologies based on the information within a concept. The features to identify synonymy concepts consist of four parts: label of the concept, description of a concept, object and data properties of a concept, and the hierarchical structure of a concept. All features together are used as a measurement to determine the semantic relation that holds between two concepts that express the same meaning. Each feature alone, such as label, cannot conclude the synonymy result since a label is an apparent concept name which can ambiguously be polysemy. These features are a certain hint to scope the synonym and similarity to each other concept among several ontologies. As aforementioned features, the framework is designed into five modules to handle each feature separately and to sum up the similarity score. The expected result is the list of concepts which are synonymy in the different ontologies. The overview of the proposed framework is illustrated in Fig. **1**.

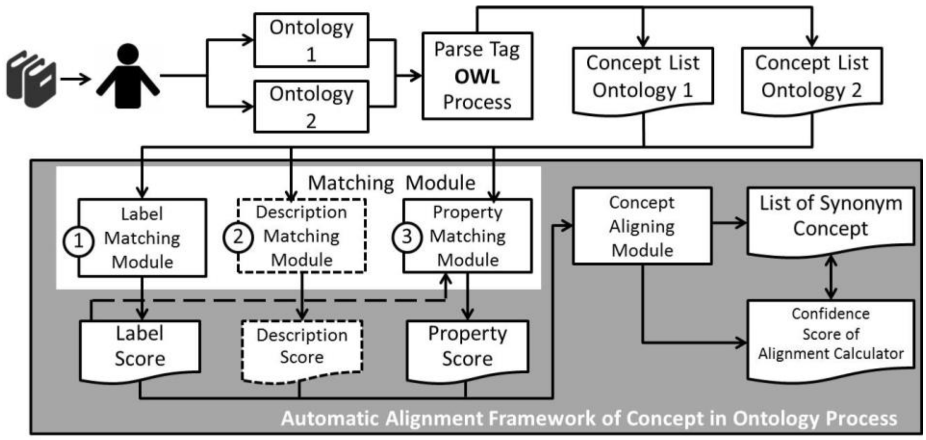

Fig. 1. Overview of the proposed framework

An input for the framework is a list of concepts in two or more ontologies with their corresponding information such as description, object property, data property and its hierarchy. These details are extracted by the exploited of owl parsing. The notations using in this paper are assigned as following:

Let *O1* and *O2* are ontologies that are considered. Ontology can be defined as follows:

$$O1 = \{C11, C12, C13, \ldots, C1p\} \text{ and } O2 = \{C21, C22, C23, \ldots, C2q\}$$

where Cij is a concept in ontology i^{th} and order at j^{th}. *p,q* are the number of concept in each ontology. For any concepts in ontology, it composes of properties which are categorised into two types; 1) part-of property or object property (*PP*) and 2) attribute-of property or data property (*PA*). Those are defined as *Cij* = {*PPij, PAij*}.

In case of part-of properties, it will be linked to a concept within its ontology to define a constraint on the range of properties and thus we assign the linked class of each part-of property as *LC*. We define a pair of property label and linked class (ppij, LCij) in each property below:

$$PPij = \{(pp11, LC11), (pp12, LC12), \ldots, (ppPPk, LCPPk)\}$$

For the attribute-of properties, they link the properties to a defined data-type symbol (*s*) such as, integer, float, string and boolean. Hence, we define each property as a paired list of property label and symbol (paij, sij) as below:

$$PAij = \{(pa11, s11), (pa12,s12), \ldots, (paPAk, sPAk)\}$$

3.1 Label Matching Module

This process is designed to find a similarity of the labels which are a given surface word of the concept. To compare likeliness of the label, three types of a comparable concept are identified as 1) exact sameness, 2) partial sameness and 3) none sameness. Though the labels of two concepts are completely different in terms of characters, they can mean to the same concept as a synonym. Hence, the label matching of concepts is invented to two separated calculating functions.

String based Similarity Matching.

To consider the sameness of the apparent concept names which are exactly the same and partially alike, string similarity calculation proposed by [12] is exploited to calculate the score.

A merge of normalised longest common subsequence (*NLCS*), maximal consecutive longest common subsequence starting at character 1 ($NMLCS_1$) and maximal consecutive longest common subsequence starting at any character n ($NMCLCS_n$) are applied in this module. Where *label-c1i* and *label-c2j* are a label of concept in ontology 1 and a concept in ontology 2 respectively, The formulae are obtained as:

$$v_1 = NLCS(label - c_{1i}, label - c_{2j}) = \frac{length\left(LCS(label - c_{1i}, label - c_{2j})\right)^2}{length(label - c_{1i}) \times length\left(label - c_{2j}\right)} \tag{1}$$

$$v_2 = NMLCS_1(label - c_{1i}, label - c_{2j}) = \frac{length\left(NMLCS_1(label - c_{1i}, label - c_{2j})\right)^2}{length(label - c_{1i}) \times length\left(label - c_{2j}\right)} \tag{2}$$

$$v_3 = NMLCS_n(label - c_{1i}, label - c_{2j}) = \frac{length\left(NMLCS_n(label - c_{1i}, label - c_{2j})\right)^2}{length(label - c_{1i}) \times length\left(label - c_{2j}\right)} \tag{3}$$

The weighted sum of these individual values *v1*, *v2* and *v3* is used to determine string similarity score, where w_i is weights with the sum of w_i = 1. *w* value is set by using an EM algorithm to find a significance of each parameter by v. Therefore, the string similarity of the two concepts is:

$$Sim(label - c_{1i}, label - c_{2j}) = w_1 v_1 + w_2 v_2 + w_3 v_3 \tag{4}$$

From the abovementioned formulae, the score of the string similarity calculation is at maximum as 1.0 in case of exact sameness whilst the partial sameness will gain the decreasing score based upon the apparent difference. With string similarity calculation, the labels with little different writing style such as plurality form, gerund form, capitalisation and localising form (American - British English) can be handled systemically. For example given in Fig. **2**, the exact sameness example is the class “Method” in both ontology#1 and 2 which is equivalent in label therefore the score is calculated as 1.0. Furthermore, the class “Rope” from ontology#1 and the class “Ropes” from ontology#2 are partially different so the calculation returns the score as 0.8 based on the equation (4). However, string similarity calculation cannot determine the completely different surface of the concept name as exemplified in a line with X mark in Fig. **2**.

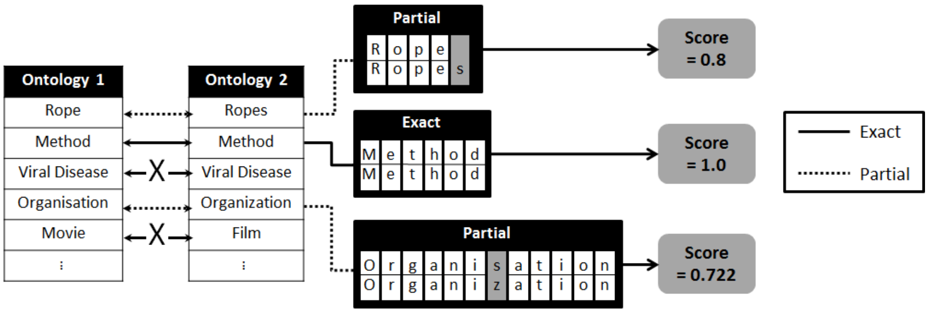

Fig. 2. An example of String Based Matching Result

Sense Based Similarity Matching

This process is designed to deal with the completely different surface of the semantically equivalent concept. WordNet [13], [14] is chosen as a source for lexical relations. The relations include synonym with in the given entry and the relation across POS type. Normally, a label of a concept in an ontological product is a phrasal expression. To employ WordNet, those phrases should be split into words. Each word is searched with the headword in WordNet entry and is examined the related information given in WordNet as a medium to another label in another ontology. For more detail, please see examples in Fig. 3.

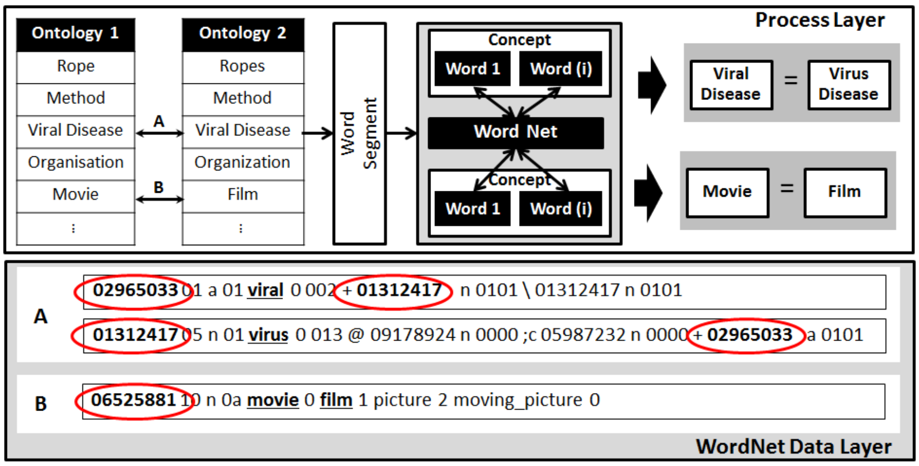

Fig. 3. An example of Sense Based Matching Result

In the WordNet, relations of lexicon are assigned by the pattern which links two or more senses with sense ID. As shown in Fig. 3 A, the lexicon “virus” in noun file is assigned with sense ID “01312417”, and there is a link (signal as the circle in Fig. 3) to another sense ID as “02965033” in adjective file which is where the word “viral” is located. The link informs that the word sense relation has the related meaning but different part-of-speech. Thus, this information provides us that the word “virus” and “viral” have the relevantly equivalent meaning. In Fig. 3 B, the word “movie” from ontology#1, once is searched through WordNet is found that it is in the sense ID “06535881” entry which has several another words such as “film” in synonym set (SynSet). This information hence can be concluded that the label “movie” and “film” are semantically equivalent.

Cases of relation of the sense between concepts are exact, partial and non-matching. Each concept label was segmented to list of string. Then calculate the sense based similarity matching (Sense) for two concepts as follows:

$$\mathrm{Sense}(\mathrm{label}-\mathrm{c1i},\mathrm{label}-\mathrm{c2j}) = \frac{\text{amount of equivalent senses}}{\text{maximum length of string}} \tag{5}$$

3.2 Description Matching Module

This function is designed as an optional score in case there is a description (Des) attached to the concept. Naturally, the description of the concept is according to ontology developer to select the description from well-known reference, and it can be chosen freely. Therefore, there will be less chance to capture synonymous meanings to identify the equivalent concepts. However, in case that the descriptions of two concepts are exactly the same as using the same reference of meaning, they can be assured that those two concepts are the synonymy to each other. Thus, the matching description is consider as a positive information as a hint for informing the equivalent concept.

The string matching is applied to capture the sameness of description in this work. The Des value can solely be 1.0 if the descriptions of both concepts are the same. The exact sameness will only be counted as a plus score towards the total score while other cases will be ignored by the system. For the case of the Des value is not 1.0, the non-matched description will not be calculated in the total score.

3.3 Property Matching Module

This module is to calculate the likeness of properties related to concepts between ontologies. We assume that the concepts which contain equivalent properties in terms of property label, cardinality, and range of class are likely to be synonym to each other. Moreover, the inherited properties from mother concepts are also considered as attached properties. Please see the exemplified illustration in Fig. **4**.

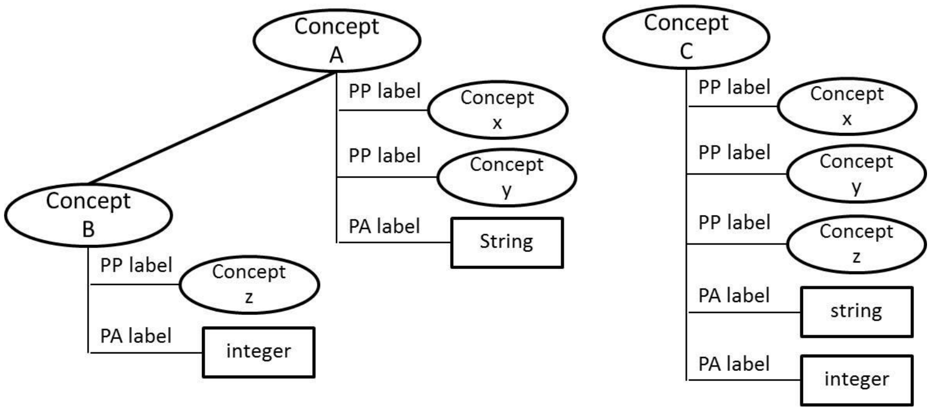

Fig. 4. An example of Property Based Matching

From Fig. **4**, Concept A and B are the concepts from ontology#1 while Concept C is from ontology#2. Each concept has its own property as shown, but please be reminded that Concept B also gets the inherited properties from Concepts A thus Concept B has five properties in total. To identify equivalence of the concepts, properties of the concepts should be the same or mostly similar. In this work, ontological property [15][16] is categorised into two types.

1. *part-of* property or object property (*PP*) – containing a constraint and number on the range of properties as an object of the relation
2. *attribute-of* property or data property (*PA*) – containing a constraint on the instantiated data by data type, i.e. string, integer, float, Boolean, etc.

From Fig. **4**, Concept B from ontolog#1 with inherited properties from Concept A and Concept C from ontology#2 contain the same PP properties in terms of linked concepts and PA properties in terms of data-type. To draw the matching method, we applied the best first search in our methodology as given in pseudo code in Fig. 5.

```
Property Similarity (O1,O2)

1. Let O1[i] and O2[j] is a list of all concepts in ontology O1 and O2 respectively
2. Let Initial C[m] is an initial list of concept which contains only attribute-of proper-
ties from O1
3. Let  Diff O1  = O1 -  Initial C
3. While concept o1[i]  in Initial C is not null
4.        For each concept o2[j] in O2
5.                Sim [i][j] = Sim[j][i] = Similarity_Calculation (o1[i], o2[j])
6.                 Calculated_C ← o1[i]
7.        End For
8. End While
9. While concept in DiffO1 is not null
10.       SimCalcCandidate = Update (DiffO1 , Calculated_C)
11.       If SimCalcCandidate is not null
12.               DiffO1  ← DiffO1 - SimCalcCandidate
13.               While concept o1[i] in SimCalcCandidate is not null
14.                       For each concept o2[j] in O2
15.                               Sim [i][j] = Sim[j][i] = Similarity_Calculation (o1[i], o2[j])
16.                               Calculated_C ← o1[i]
17.                       End For
18.               End While
19.       Else
20.       UnabletoCal ← DiffO1
21. End While

Function Update (DiffO1 , Calculated_C)

1. For each concept O1[i] in DiffO1
2.        For each PP[i,j]  in O1[i]
3.                Unless LinkClass PP[i,j] is in Calculated_C
4.                Break
5.        End For
6.        SimCalcCandidate ← O1[i]
7. End For
```

Fig. 5. Psuedo Code of Property Matching Module

From Fig. 5, the pseudo code is designed to handle PA and PP of concept from two ontologies. The PA of the concept will be handled first hand and compare with the candidate concepts in another ontology. After PAs are collected, PPs of the concept are focused for similarity calculating. Each property will be compared and once the calculation is done, the set of PA and PP will be sent to compare with another concept until all possible concepts are scored.

To score the property similarity, the following equations are obtained.

$$S_{PM} = \frac{\sum_{PP}^{ij} S_{LM}\left(PP_{k_{C_{1i}}}, PP_{m\_C_{2j}}\right) + \sum_{PA}^{ij}\left(PA_{k_{C_{1i}}}, PA_{m\_C_{2j}}\right)}{n_{pp}+n_{pa}} \quad (6)$$

Where

$PA_{k}\_C_{1i}$ is an *attribute-of* property k^{th} of concept i^{th} in ontology#1
$PA_{m}\_C_{2j}$ is an *attribute-of* property m^{th} of concept j^{th} in ontology#2
$PP_{k}\_C_{1i}$ is a *part-of* property k^{th} of concept i^{th} in ontology#1
$PP_{m}\_C_{2j}$ is an *part-of* property m^{th} of concept j^{th} in ontology#2
S_{LM} is score of label matching selected from higher score between Sim from (4) and Sense from (5)
n_{pp} is amount of *part-of*
n_{pp} is amount of *attribute-of*

3.4 Alignment

To decide which pair of ontologies is a synonym, the scores from all the features are employed. The result of this module is a list of possibly equivalent classes based on alignment score and the confident score to inform the degree of confidence which system makes. Since several features are used in this work, the alignment score can be above alignment criterion from the calculation though the pair is not guaranteed from any features. Confident score (ConfScore) is applied to distinguish the trustable pair from another.

To get alignment score, we apply equation (7) while equation (8) is designed to generate ConfScore.

$$\text{AlignmentScore}\ (c_{1i}, c_{2j}) = \frac{S_{LM}+S_{DM}+S_{PM}}{F} \quad (7)$$

$$\text{ConfScore} = \left(\text{intial} - \left((1-S_{LM}) + (1-S_{DM}) + (1-S_{PM})\right) * 100\right) + bonus \quad (8)$$

Where

AlignmentScore is a similarity score of a concept pair
ConfScore is a confidence score
C_{1i} is a concept i^{th} in ontology 1
C_{2j} is a concept j^{th} in ontology 2
S_{LM} is score of label matching selected from higher score between Sim from (4) and Sense from (5)
S_{DM} is score of description matching (in case it exists)
S_{PM} is score of properties matching from (6)
F is amount of feature apply in used

The criterion to assign equivalent concepts is the AlignmentScore is over 0.5.

ConfScore is given based on the strong score from each feature. The more strong score from features, the higher of the ConfScore will be. The initial ConfScore is 50. Once the score of the feature is found at maximum, the bonus of 50 ConfScore will be added. Otherwise, the score will be decreased from the missing point from the matching feature score.

Table 2. An Example Alignment Result and ConfScore Calculation

		O2								
		A			B			C		
		Slm,Spm	AlignScore	ConfScore	Slm,Spm	AlignScore	ConfScore	Slm,Spm,Sdm*	AlignScore	ConfScore
O1	X	1.0,1.0	1	150	0.22,0.1	0.61	78	0,0	0	not aligned
	Y	0.1,1.0	0.55	50	0.91,0.75	0.83	16	1.0,0.87,1.0*	0.96	137
	Z	0.72,0.3	0.51	0	1.0,0.7	0.85	70	0.2,0	0.1	not aligned

For example from Table 2, focusing on concept “X”, concept “A” and “B” are chosen as a equivalent concept since concept “C” do not meet the criteria from alignment score which is below 0.5. The pair of X-A obtains 150 ConfScore from two maximum feature scores which will give two of 50 bonus scores. For concept “Y”, all of concept A, B and C are aligned as Y's synonym since they all give 0.55, 0.83 and 0.956, respectively. As shown, pair of Y-C gets another a plus score (marked with asterisk symbol) from description exact matching. In details, the pair of Y-A obtains ConfScore as 50 according to ((50[1] -(90-0)[2]) = 0[3]) +50[4] . For the pair of Y-B, the ConfScore is 16. The pair of Y-C obtains ConfScore as 137. Hence, the pair from concept Y shows that the pair Y-C has the highest ConfScore.

4 Experiment

To test an ability of the framework, three related ontologies were selected. The ontologies are mulberry ontology (O1), silk worm ontology (02) and trade statistic of silk-mulberry products ontology (O3). Those ontologies share several synonymous concepts since they are in the same agriculture topic. For the statistic, O1, O2 and O3 contain 303, 372, and 96 concepts respectively. For a test result, ontology developers were asked to manually align the equivalent concepts as a gold standard.

The gold standard shows that there are 59 equivalent concepts comparing O1-O2, and O1-O3 pair has 27 equivalent concepts while O2-O3 gives 12 equivalent pairs.

---

1 Initial score of ConfScore is 50

2 This is minus score from missing score of missing feature score. 90 is obtained from missing 0.9 point from feature#1 multiples with 100 while 0 is from non-missing score from 1.0.

3 The score is set as an absolute integer which does not allow negative value.

4 50 is the bonus score from existing of a maximum feature score.

The sum of all equivalent concepts from all three ontologies is 97 concepts. An example of the equivalent concept list is given in Table 3.

Table 3. An example of the equivalent concept list

Concept in O1	Concept in O2	Concept in O3
X	Acre	Acer
Viral Disease	Virus Disease	X
X	Cocoon	Cocoon
Fungal Disease	Fungal Disease	X
X	Raw Silk	Raw Silk
Mulberry tree	X	Mulberry Plant
Count Unit	Count Unit	Count Unit

From comparing to gold standard, we measured the result in terms of precision, recall and f-measure. The results from the system comparing with gold standard are given in Table 4.

Table 4. Precision, Recall and F-measure of the Matching Result

Feature	Gold Standard	System Found	Precision	Recall	F-measure
label	60	60	1	1	1
properties	7	9	0.78	1	0.88
description	0	0	0	0	0
label + description	4	4	1	1	1
label + properties	24	25	0.96	1	0.98
properties + description	0	1	-	-	-
label + properties + description	2	2	1	1	1
All matches	**97**	**101**	0.96	1	0.98

5 Discussion

From the result, we found that the proposed framework gave a good accuracy result. The system can capture all 97 equivalent concepts assigned in gold standard list. However, there are four concepts that the system returned as synonymous concept pair but not in the list. Those concepts were examined in details and found that they are the synonymous concepts which experts overlooked from manual mapping since the labels are ambiguous.

We found that label matching plays the main role for capturing 91 concepts of the result while properties matching can capture 35 concepts. In the given ontologies, there are some descriptions attached to the concept, and it helped on matching 7 concepts which were already considered as a pair by label matching or property matching. However, the description matching gave an extra confidence score to those pairs to assure the reliable aligning.

From 91 concept pairs by label matching, 69 concepts were found by the string based criteria while 22 concept pairs were recognised by sense based matching module. The examples of found pairs with the score gained from system calculation are shown in Table 5.

Table 5. An Example of found pairs by label with the score

Concept	Matched Concept	Score	Criterion
Method	Method	1.0	String similarity
Viral Disease	Virus Disease	1.0	Sense - WordNet
Rope	Ropes	0.8	String similarity
Mulberry Plant	Mulberry Tree	1.0	Sense - WordNet
Food	Nutrition	1.0	Sense - WordNet
Bacterial Disease	Bacteria infection	0.5	Sense - WordNet

Based on property matching module, we found that properties can be a great method to capture phrasal terms. All of the nine concepts that property matching can solely capture are a phrasal label with domain-specific terms as exemplified in Table 6. From examples in Table 6, the first row is the concepts with exactly same range of concepts and data-types while the second row shows the concepts that required sense based criteria to map the range concept.

Table 6. An Example of found pairs with the score and matching type

<table>
<tr><th>Concept</th><th>Matched Concept</th><th>Score</th></tr>
<tr><td>Product (in silk ontology)
PP – range_class: Material, label: made_of
PP – range_class: Country, label: import_to
PA – datatype: integer, label: has_retail_price</td><td>Silk Goods (in Trading Stat ontology)
PP – range_class: Material, label: made_of
PA – datatype: integer, label: has_retail_price
PP – range_class: Country, label: import_to</td><td>1.0</td></tr>
<tr><td>Harvesting (in Trading Stat ontology)
PP – range_class: Season, label: has_season
PP – range_class: Cocoon_Product, label: has_output</td><td>Product (in silk ontology)
PP – range_class: Time, label: has_time
PP – range_class: Silk_Product, label: has_output</td><td>0.747</td></tr>
</table>

As for description matching, seven concepts are matched. Those concepts were also aligned with other matching, thus it can be additional plus score to weight up the confidence score. By focusing on confidence score, we found that 87 concepts from the 101 matched equivalent concepts are assigned with over 100 confidence score points especially the concepts with description matched. Unfortunately, the confidence score cannot be measured systemically, but ontology developers has no complain against the score and satisfy with the given confidence score.

6 Conclusion

In this paper, we present a new method to capture synonym concepts from several ontologies. The framework exploits information within ontological concepts including a label of a concept, properties of a concept and a concept's description. The aforementioned information is treated as features for considering similarity. Once the score of each feature is calculated, those scores are used for making decision to align a pair of concepts. Not only alignment of equivalent concepts is implemented in this work, but the confidence score is also calculated to distinguish the guaranteed pair from ambiguous pairs. From testing the framework against manual pair alignment, the system shows the potential to work equivalently to human selection. Moreover, there are some captured concepts which can be considered similar concepts that were overlooked by manual selection.

To improve the performance, we plan to add more features from concept's information such as hierarchical structure of the concept and other relevant ontological details. We also plan to test the system with large scale ontologies to approve its speed and robustness.

7 References

[1] D. Fensel, F. van Harmelen, I. Horrocks, D. L. McGuinness, and P. F. Patel-Schneider, “OIL: an ontology infrastructure for the Semantic Web,” *IEEE Intell. Syst.*, vol. 16, no. 2, pp. 38–45, Mar. 2001.

[2] S. Aitken, R. Korf, B. Webber, and J. Bard, “COBrA: a bio-ontology editor,” *Bioinformatics*, 2005.

[3] A. Bernstein and E. Kaufmann, “GINO–a guided input natural language ontology editor,” *Semant. Web-ISWC 2006*, 2006.

[4] V. Haarslev and R. Möller, “Racer: A Core Inference Engine for the Semantic Web.,” *EON*, 2003.

[5] H. Tan and P. Lambrix, “SAMBO Results for the Ontology Alignment Evaluation Initiative 2007.,” *OM*, pp. 2–9, 2007.

[6] W. Hu and Y. Qu, “Falcon-AO: A practical ontology matching system,” *Web Semant. Sci. Serv. Agents World Wide Web*, vol. 6, no. 3, pp. 237–239, Sep. 2008.

[7] and E. M. Nagy, Miklos, Maria Vargas-Vera, “DSSim-ontology mapping with uncertainty Conference Item,” 2006.

[8] J. Li, J. Tang, Y. Li, and Q. Luo, “Rimom: A dynamic multistrategy ontology alignment framework,” *Knowl. Data Eng. …*, vol. 21, no. 8, pp. 1218–1232, 2009.

[9] Y. Jean-Mary, “Asmov: Results for oaei 2010,” *Ontol. Matching 126*, 2010.
[10]M. Seddiqui and M. Aono, “Anchor-flood: results for OAEI 2009,” in *Proceedings of the ISWC 2009 Workshop on Ontology Matching*, 2009.
[11]P. Shvaiko and J. Euzenat, “Ontology matching: state of the art and future challenges,” *Knowl. Data Eng. IEEE Trans. 25*, vol. 1, no. X, pp. 158–176, 2013.
[12]P. Luekhong, T. Ruangrajitpakorn, T. Supnithi, and R. Sukhahuta, “Pooja : Similarity-based Bilingual Word Alignment Framework for SMT,” in *Proceedings of the 10th International Symposium on Natural Language Processing, Phuket, Thailand*, 2013.
[13]G. Miller, “WordNet: a lexical database for English,” *Commun. ACM*, 1995.
[14]“WordNet Search - 3.1.” [Online]. Available: http://wordnetweb.princeton.edu/perl/webwn. [Accessed: 28-Jul-2014].
[15]K. Kozaki, Y. Kitamura, M. Ikeda, and R. Mizoguchi, “Hozo: an environment for building/using ontologies based on a fundamental consideration of ‘Role’ and ‘Relationship,’” *... Manag. Ontol. ...*, 2002.
[16]N. Noy, M. Crubézy, and R. Fergerson, “Protege-2000: an open-source ontology-development and knowledge-acquisition environment,” in *AMIA Annu Symp Proc*, 2003, p. 2003.