# Ontology Mapping Neural Network: An Approach to Learning and Inferring Correspondences among Ontologies

Yefei Peng[1]*, Paul Munro[1], and Ming Mao[2]

[1] University of Pittsburgh, Pittsburgh PA 15206, USA,
yep3@pitt.edu, pmunro@pitt.edu,
[2] SAP Labs, Palo Alto CA 94304, USA,
ming.mao@sap.com,

**Abstract.** An ontology mapping neural network (OMNN) is proposed in order to learn and infer correspondences among ontologies. It extends the Identical Elements Neural Network (IENN)'s ability to represent and map complex relationships. The learning dynamics of simultaneous (interlaced) training of similar tasks interact at the shared connections of the networks. The output of one network in response to a stimulus to another network can be interpreted as an analogical mapping. In a similar fashion, the networks can be explicitly trained to map specific items in one domain to specific items in another domain. Representation layer helps the network learn relationship mapping with direct training method.

OMNN is applied to several OAEI benchmark test cases to test its performance on ontology mapping. Results show that OMNN approach is competitive to the top performing systems that participated in OAEI 2009.

**Keywords:** neural network, ontology mapping, analogy, learning

## 1 Introduction

Ontology mapping is important to the emerging Semantic Web. The pervasive usage of agents and web services is a characteristic of the Semantic Web. However agents might use different protocols that are independently designed. That means when agents meet they have little chance to understand each other without an"interpreter". Ontology mapping is "a necessary precondition to establish interoperability between agents or services using different ontologies." [2]

The Ontology Mapping Neural Network (OMNN) extends the Identical Elements Neural Network(IENN)'s [3, 4, 1, 5] ability to represent and map complex relationships. The network can learn high-level features common to different tasks, and use them to infer correspondence between the tasks. The learning dynamics of simultaneous (interlaced) training of similar tasks interact at the

* The author is working at Google now. Email: yefeip@google.com

shared connections of the networks. The output of one network in response to a stimulus to another network can be interpreted as an analogical mapping. In a similar fashion, the networks can be explicitly trained to map specific items in one domain to specific items in another domain.

## 2 Network Architecture

The network architecture is shown in Figure 1. $A_{in}$ and $B_{in}$ are input subvectors for nodes from ontology A and ontology B respectively. They share one representation layer $AB_r$. $RA_{in}$ represents relationships from graph A; $RB_{in}$ represents relationships from graph B. They share one representation layer $R_r$.

In this network, each to-be-mapped node in graph is represented by a single active unit in input layers ($A_{in}$, $B_{in}$) and output layers ($A_{out}$, $B_{out}$ ). For relationships representation in input layer ($RA_{in}$, $RB_{in}$), each relationship is represented by a single active unit.

The network shown in Figure 1 has multiple sub networks shown in the following list.

1. $Net_{AAA}$ : $\{A_{in}\text{-}AB_r\text{-}X_{AB};\ RA_{in}\text{-}R_{RA}\text{-}X_R\ \}\text{-}H_1\text{-}W\text{-}H_2\text{-}V_A\text{-}A_{out}$;
2. $Net_{BBB}$ : $\{B_{in}\text{-}AB_r\text{-}X_{AB};\ RB_{in}\text{-}R_{RB}\text{-}X_R\ \}\text{-}H_1\text{-}W\text{-}H_2\text{-}V_B\text{-}B_{out}$;
3. $Net_{AAB}$ : $\{A_{in}\text{-}AB_r\text{-}X_{AB};\ RA_{in}\text{-}R_{RA}\text{-}X_R\ \}\text{-}H_1\text{-}W\text{-}H_2\text{-}V_B\text{-}B_{out}$;
4. $Net_{BBA}$ : $\{B_{in}\text{-}AB_r\text{-}X_{AB};\ RB_{in}\text{-}R_{RB}\text{-}X_R\ \}\text{-}H_1\text{-}W\text{-}H_2\text{-}V_A\text{-}A_{out}$;

An explicit cross training method is proposed to train the correspondence of two relationships by directly making their representations more similar. Only a portion of the neural network is involved in this cross training method: the input subvectors and representation layer. For example, we want to train the relationship correspondence of $< R_1, R_2 >$, where $R_1$ belongs to ontology A and $R_2$ belongs to ontology B. $R_1$ will be presented at $RA_{in}$. The output at $R_r$ will be recorded, which we will name as $RR_1$. Then $R_2$ is presented at $RB_{in}$. $RR_1$ will be treated as target value at $R_r$ for $R_2$. Weights $RU_B$ will be modified so that $R_1$ and $R_2$ have more similar representation at $R_r$ with standard back propagation method. Then $< R_1, R_2 >$ will be trained so that weight $RU_A$ will be modified to make $R_1$'s representation at $R_r$ similar to that of $R_2$. The sub networks involved in this training method will be named as $RNet_{AB}$ and $RNet_{BA}$.

Network is initialized by setting the weights to small random values from a uniform distribution. The network is trained with two vertical training tasks ($Net_{AAA}$ and $Net_{BBB}$), two cross training tasks ($Net_{AAB}$ and $Net_{BBA}$), and two explicit training tasks ($RNet_{AB}$ and $RNet_{BA}$ ).

## 3 Results

Selected OAEI [3] benchmark tests are used to evaluate OMNN approach. All test cases share the same reference ontology, while the test ontology is different.

[3] http://oaei.ontologymatching.org/

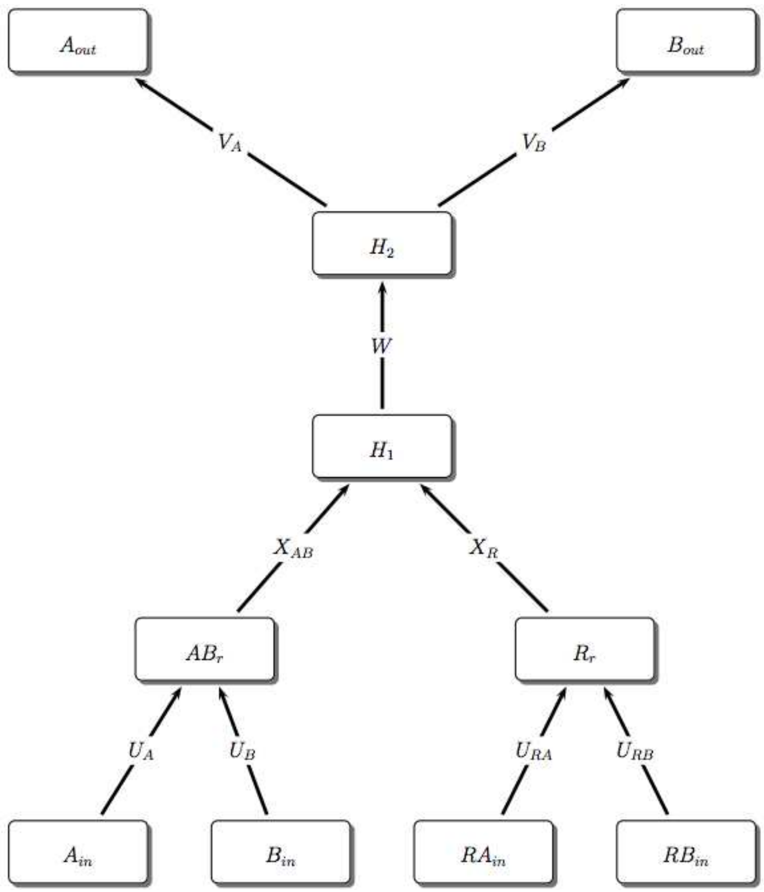


**Fig. 1.** Proposed network architecture

The reference ontology contains 33 named classes, 24 object properties, 40 data properties, 56 named individuals and 20 anonymous individuals. In the OMNN approach, classes are treated as items; object properties and data properties are treated as relationships; lastly individuals are not used.

Texture information is used to generate high confident mappings which are then used as cross-training data in OMNN. However OMNN does not focus on how well texture information is used.

In order to compare with other approaches that heavily use texture information, 19 test cases with limited texture information are selected to be used in our experiments. They are test case 249, 257, 258, 265, 259, 266 and their sub-cases.

To get a meaningful comparison, the Wilcox test is performed to compare OMNN with the other 12 systems participated in OAEI 2009 on precision, recall and f-measure. The result is shown in Table 1. It shows that OMNN has better F-measure than 9 of the 12 systems, OMNN's recall is significantly better than 10 of the systems. It should be noted that $p$-value$< 0.05$ means there is significant difference between two systems compared, then detailed data is visited to reveal which is one is better than the other.

**Table 1.** $p$-value from Wilcox test for 19 benchmark test cases. The green color means that OMNN is significantly better than the system; red color means the system is significantly better than OMNN; yellow means no significant difference. Significance is defined as $p$-value$< 0.05$.

| System | Precision | Recall | F-Measure |
|---|---|---|---|
| aflood | 0.000 | 0.570 | 0.182 |
| AgrMaker | 0.014 | 0.000 | 0.000 |
| aroma | 0.420 | 0.000 | 0.000 |
| ASMOV | 0.000 | 0.046 | 0.679 |
| DSSim | 0.027 | 0.000 | 0.000 |
| GeRoMe | 0.042 | 0.000 | 0.000 |
| kosimap | 0.008 | 0.000 | 0.000 |
| Lily | 0.000 | 0.306 | 0.000 |
| MapPSO | 0.000 | 0.000 | 0.000 |
| RiMOM | 0.136 | 0.002 | 0.032 |
| SOBOM | 0.811 | 0.000 | 0.000 |
| TaxoMap | 0.011 | 0.000 | 0.000 |

.

## References


1. Bao, J., Munro, P.W.: Structural mapping with identical elements neural network. In: Proceedings of the International Joint Conference on Neural Networks - IJCNN 2006. pp. 870–874. IEEE (2006)
2. Ehrig, M.: Ontology Alignment: Bridging the Semantic Gap (Semantic Web and Beyond). Springer-Verlag New York, Inc., Secaucus, NJ, USA (2006)
3. Munro, P.: Shared network resources and shared task properties. In: Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society (1996)
4. Munro, P., Bao, J.: A connectionist implementation of identical elements. In: In Proceedings of the Twenty Seventh Ann. Conf. Cognitive Science Society Proceedings. Lawerence Erlbaum: Mahwah NJ (2005)
5. Munro, P.W.: Learning structurally analogous tasks. In: Kurková, V., Neruda, R., Koutník, J. (eds.) Artificial Neural Networks - ICANN 2008, 18th International Conference. Lecture Notes in Computer Science, vol. 5164, pp. 406–412. Springer: Berlin/Heidelberg (2008)