OTMapOnto: Optimal Transport-based Ontology Matching

Yuan An, Alex Kalinowski, and Jane Greenberg

Metadata Research Center, College of Computing and Informatics, Drexel University,
Philadelphia, PA 19104, USA
{ya45,ajk437,jg3243}@drexel.edu

Abstract. This paper describes *OTMapOnto*, an optimal transport-based ontology matching system. It leverages the techniques developed in computational optimal transport to match terms across different ontologies. The system starts with converting ontology elements into embedding vectors which can incorporate linguistic, structural, and logical information. It then applies optimal transport to the problem of moving masses from the source embedding space to the target embedding space. The solution to the optimal transport problem consists a shape-based Wasserstein distance and a coupling matrix between the embeddings of the source and target ontologies. The coupling matrix gives rise to a set of candidate matchings which can be refined through further process. The current version of the OTMapOnto system makes use of pre-trained word embeddings, such as fasttext and BioWordVec, for embedding the labels of ontology elements. We report that optimal transport is a promising solution to discovering ontology matching with higher recall when provided with good representations of ontologies[1].

Keywords: ontology matching · optimal transport · ontology embedding.

1 Presentation of the System

1.1 State, purpose, general statement

OTMapOnto is an ontology matching system that applies optimal transport to ontology embeddings for discovering matchings. The system starts with representing an ontology using a set of numerical vectors/embeddings each of which can encode the semantic and structural information of an ontology element. If both the source and target ontologies are embedded in the same vector space, we can calculate the geometric distances between pairs of source and target ontology elements. The distances can be utilized for deriving potential matchings between ontologies. For example, Kolyvakis et al. [9] report an approach that first represents ontology elements in numerical embeddings. The approach then

---

applies the Stable Marriage algorithm to deriving candidate matchings based on the cosine similarity between embedding vectors. Other embedding-based supervised [14] and distantly supervised [5] methods have been proposed for matching ontologies. *OTMapOnto* differs, with an unsupervised method that formulates the *matching problem* as a *transformation problem* from a source ontology to a target ontology. Taking the two ontologies as two whole systems of knowledge encoded in numerical vectors, *OTMapOnto* first finds an optimal way to transport masses from the source ontology to the target ontology. It then derives candidate matchings between individual elements from the coupling matrix of the optimal transport, for instance, by a following Nearest Neighbor or Stable Marriage algorithm. This paper presents the first version of the system that only makes use of linguistic information for embedding ontology elements.

1.2   Specific techniques used

Figure 1 illustrates the architecture of the *OTMapOnto* system. In this paper, we focus on two main components: *ontology embeddings* and *optimal transport*.
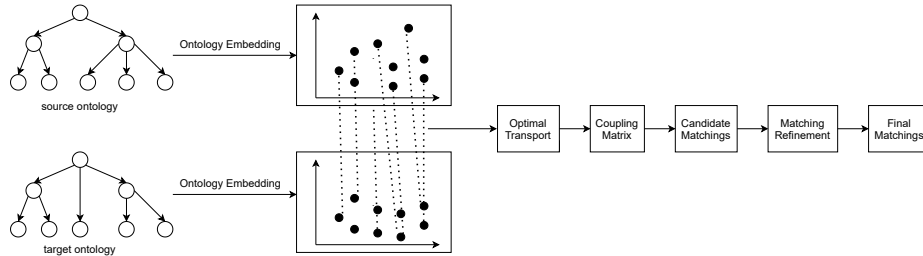

Fig. 1. The Architecture of the *OTMapOnto* System

Ontology Embeddings. We consider an ontology as a 7-tuple

$$\mathcal{O} = \{\mathcal{C}, \mathcal{R}, \mathcal{G}, \mathcal{L}, \mathcal{D}, \mathcal{S}, \mathcal{P}\},$$

where $\mathcal{C} = \{C_1, C_2, ..., C_n, e_1, e_2, ..., e_m\}$ is a set of concepts C_i and entities e_i, $\mathcal{R} = \{R_1, R_2, ..., R_r\}$ is a set of relations corresponding `subClassOf`, Object and Datatype properties, $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ is the graph structure with a set of vertices $\mathcal{V} = \{V_1, V_2, ..., V_{n+m}\}$ and edges $\mathcal{E} = \{E_1, E_2, ..., E_r\}$, $\mathcal{L} = \{L_1, L_2, ..., L_k\}$ is a set of logical formulas, $\mathcal{D} = \{D_1, D_2, ..., D_d\}$ is a set of textual descriptions, $\mathcal{S} = \{S_1, S_2, ..., S_s\}$ is a set of distant evidence sentences, and $\mathcal{P}$ is a set of literal values associated with the Datatype properties.

Various methods have been proposed for representing individual components in an ontology as embeddings. For example, translational-based methods [4, 19, 10, 12] and graph neural networks (GNN) [8] encode an ontology based on

its graph structure. Text-enhanced methods for ontology embedddings [11, 16, 2, 17, 18] encode lexical words of ontology elements. Logic-aware methods [15, 7] incorporate logical constraints into ontology embeddings. For the current *OTMapOnto* system, we apply pre-trained language models, such as fasttext [3] and BioWordVec [20], to embedding the labels of the set of ontology concepts, Object and Datatype properties, $\mathcal{O} = \{C_1, C_2, ..., C_n, R_1, R_2, ..., R_r\}$, as a set of numerical vectors $\mathbf{X}_{\mathcal{O}} = \{\mathbf{x}_i \in \mathbb{R}^d, i = 1..n + r\}$.

Specifically, for each element in $\mathcal{O}$, the system first normalizes the element's label via a sequence of standard text processing steps. The normalized label is split into individual words which in turn are fed into a pre-trained language model to obtain their corresponding word embeddings. The embedding of the element is the average of the embeddings of the individual words in the normalized label.

Optimal Transport. Given two sets of embeddings $\mathbf{X} = \{\mathbf{x}_i \in \mathbb{R}^d, i = 1..n\}$ and $\mathbf{Y} = \{\mathbf{y}_j \in \mathbb{R}^d, j = 1..m\}$, where each embedding is represented as a vector $\mathbf{x}_i$ or $\mathbf{y}_j \in \mathbb{R}^d$. Let $\mu = \sum_{i=1}^{n} p(\mathbf{x}_i)\delta_{\mathbf{x}_i}$ and $\nu = \sum_{j=1}^{m} q(\mathbf{y}_j)\delta_{\mathbf{y}_j}$ be two probability distributions defined on the two sets $\mathbf{X}$ and $\mathbf{Y}$, respectively, with $\delta_{\mathbf{x}}$ as the Dirac at the point $\mathbf{x}$. $p(\mathbf{x}_i)$ and $q(\mathbf{y}_j)$ are probability weights associated with each set. Usually, we consider uniform weights, e.g., $p(\mathbf{x}_i) = \frac{1}{n}$, for $i = 1..n$, and $q(\mathbf{y}_j) = \frac{1}{m}$, for $j = 1..m$. However, if additional information is provided, $p(\mathbf{x}_i)$ and $q(\mathbf{y}_j)$ can incorporate the information as non-uniform distributions. Optimal transport (OT) defines a distance between two distributions analogous to an optimal plan for mass transportation. Specifically, let $\mathbf{C} = [c(\mathbf{x}_i, \mathbf{y}_j)]_{i,j}$ be a cost matrix with $c(\mathbf{x}_i, \mathbf{y}_j)$ measuring a ground distance between the individual embeddings $\mathbf{x}_i$ and $\mathbf{y}_j$. Let $\mathbf{T} = [T(\mathbf{x}_i, \mathbf{y}_j)]_{i,j}$ be a matrix of a transport plan (or *couplings*) with $T(\mathbf{x}_i, \mathbf{y}_j)$ specifying how much mass will be transported from point $\mathbf{x}_i$ to point $\mathbf{y}_j$. Let $\Pi(\mu, \nu)$ be the set of all feasible transport plans defined as: $\Pi(\mu, \nu) \stackrel{\text{def}}{=} \{\mathbf{T} \in \mathbb{R}_+^{n \times m} | \mathbf{T}\mathbb{1}_n = \mu, \mathbf{T}^\top \mathbb{1}_m = \nu\}$, where $\mathbb{1}_n$ and $\mathbb{1}_m$ are all one vectors, $\mathbf{T}\mathbb{1}_n = \mu$ and $\mathbf{T}^\top \mathbb{1}_m = \nu$ are marginal constraints on feasible plans. The <u>Optimal Transport</u> problem is to find the map $\mathbf{T} : \mathbf{X} \to \mathbf{Y}$, where

$$\mathbf{T} = \operatorname*{argmin}_{\mathbf{T} \in \Pi(\mu, \nu)} \sum_{i=1}^{n} \sum_{j=1}^{m} c(\mathbf{x}_i, \mathbf{y}_j) \cdot T(\mathbf{x}_i, \mathbf{y}_j), \text{s.t.}, \mathbf{T}\mathbb{1}_n = \mu, \mathbf{T}^\top \mathbb{1}_m = \nu \quad (1)$$

The map $\mathbf{T}$ also gives rise to a distance measure between the two distributions called <u>Wasserstein distance</u>:

$$W(\mu, \nu) \stackrel{\text{def}}{=} \min_{\mathbf{T} \in \Pi(\mu, \nu)} \langle \mathbf{C}, \mathbf{T} \rangle = \min_{\mathbf{T} \in \Pi(\mu, \nu)} \sum_{i=1}^{n} \sum_{j=1}^{m} c(\mathbf{x}_i, \mathbf{y}_j) \cdot T(\mathbf{x}_i, \mathbf{y}_j) \quad (2)$$

The objective is a linear programming problem where the time complexity $O(N^3)$ is prohibitively large for large N. The common speedup is to replace the objective with an entropy regularized objective function. As a result, the problem can be solved efficiently using Sinkhorn iterations [6].

4   Yuan An, Alex Kalinowski, and Jane Greenberg

Driving and Refining Ontology Matchings. Both the source and target ontologies are embedded through the same pre-trained language model. For the ground distance $\mathbf{C} = [c(\mathbf{x}_i, \mathbf{y}_j)]_{i,j}$, where $i = 1..n, j = 1..m$, we experimented with following two approaches:

- Using Euclidean distances between the embeddings of pairs of source and target ontology elements.
- Using Wasserstein distances between the immediate neighborhoods of source and target ontology elements. The immediate neighborhood of an element includes parents and children in the `subClassOf` hierarchy, domain and range elements in direct relations, and the synsets in WordNet.

The solution, $\mathbf{T} = [T(\mathbf{x}_i, \mathbf{y}_j)]_{i,j}$, $i = 1..n, j = 1..m$, to the optimal transport problem provides the most efficient way to transform the entire source ontology to the entire target ontology. Obviously, not every coupling in $\mathbf{T} = [T(\mathbf{x}_i, \mathbf{y}_j)]_{i,j}$ corresponds to an ontology matching. We derive a set of candidate matchings as follows:

- **Mutual Nearest Neighbor (MNN):** for a $\mathbf{x}_p \in \{\mathbf{x}_i \in \mathbb{R}^d, i = 1..n\}$, find $\mathbf{y}_q \in \{\mathbf{y}_j \in \mathbb{R}^d, j = 1..m\}$, such that, $T(\mathbf{x}_p, \mathbf{y}_q) = \max\{T(\mathbf{x}_p, \mathbf{y}_j), j = 1..m\}$ and $T(\mathbf{x}_p, \mathbf{y}_q) = \max\{T(\mathbf{x}_i, \mathbf{y}_q), i = 1..n\}$.
- **Top-K Targets (TopK):** for a $\mathbf{x}_p \in \{\mathbf{x}_i \in \mathbb{R}^d, i = 1..n\}$, find k targets $\{\mathbf{y}_{q_1}, \mathbf{y}_{q_2}, .., \mathbf{y}_{q_k}\} \subset \{\mathbf{y}_j \in \mathbb{R}^d, j = 1..m\}$, such that, $T(\mathbf{x}_p, \mathbf{y}_{q_z}) \geq \max\{T(\mathbf{x}_p, \mathbf{y}_j), j \neq q_1..q_k\}$, for $z = 1..k$.

The set of candidate matchings will go through a sequence of refinement steps including exact label string checking, synonym verification, and context distance measurement.

1.3 Link to the system and parameters file

https://github.com/anyuanay/otmaponto_django

1.4 Link to the set of provided alignments

https://github.com/anyuanay/otmaponto_django/tree/master/results

2 Results

2.1 Anatomy

For this track, we applied two different pre-trained models, BioWordVec [20] and fasttext [3] . The BioWordVec was trained on 28 millions PubMed articles and 2 millions MIMIC III Clinical notes. By a combination of optimal transport and MNN, we achieved a 87% precision and 85% recall. By retrieving the Top10 targets, the recall was 93%, while the precision is very low. However, the size of the BioWordVec is very large (26G). It is infeasible to submit the system for evaluation. Instead, we created a running Web service with pre-loaded fasttext model (6G). Our own evaluation results is 64% precision an 81% recall.

2.2   Conferences

For this track, we applied fasttext model. We achieved a precision of 23% and a recall of 73% which is higher than the recall of most of the systems in the OAEI2020 report. By just encoding the labels of elements, *OTMapOnto* has shown it was able to retrieve more matchings based on the optimal transport strategy. Improving the precision will be our future focus.

2.3   Material Sciences and Engineering

This is a new track without previous evaluation results. There are three test cases. For the first test case, *OTMapOnto* was able to achieve a precision of 23% and a recall of 39%. For the second case, *OTMapOnto* achieved a precision of 32% and a recall of 55%. For the third test case, *OTMapOnto* achieved a precision of 14% and a recall of 90%. Compared to the methods used in Engy's thesis [13], *OTMapOnto* could achieve better recall in most of the cases.

2.4   Large Biomedical Ontologies

For this track, *OTMapOnto* encountered out of memory errors. It was only able to perform matching on two small cases. For the FMA-SNOMED small, the precision was 38% and recall was 67%. For the FMA-NCI small, the precision was 45% and recall was 84%.

2.5   Disease and Phenotype

For this track, we evaluated *OTMapOnto* on two tasks: HP-MP 2017 and DOID-ORDO 2017 task. For the HP-MP 2017 task, *OTMapOnto* achieved a precision of 11% and a recall of 99.1%. For the DOID-ORDO 2017 task, *OTMapOnto* achieved a precision of 16% and a recall of 99.3%. The recalls are consistently better than the results in the OAEI2020 report.

2.6   Common Knowledge Graphs

For this track, the precision was 90% and the recall was 84%. No previous results are available. We look forward to the evaluation results in OAEI2021.

3   General Comments

3.1   Comments on the results (strengths and weaknesses)

The coupling matrix returned by the optimal transport solver contains mappings from all n source elements to all m target elements. The current *OTMapOnto* derives candidate matchings from the $n \times m$ matrix mainly through retrieving mutual nearest neighbors. On one hand, it is inevitably that the set of candidate matchings still contain many spurious ones. On the other hand, it is

expected that the process of optimal transport can discover the majority of accurate matchings. This phenomenon has been shown in several tasks, where the results have lower precision and higher recall compared to the previous results of other systems.

3.2   Discussions on the way to improve the proposed system

In this paper, we report a work demonstrating that just with the embeddings of the labels of ontology elements, the optimal transport-based method is promising for discovering more matchings to improve recall. In future work, we will develop a method for representing an ontology as a set of embedding vectors by integrating individual components including hyperbolic embeddings [1] for better representing tree structures in ontologies. We also aim to improve precision by developing a method to rank the TopK candidate matchings for each source element. The ranking method will be based on the structural and semantic context of the elements in each pair of candidate matching. The third improvement is to modify the distributions defined on the sets of source and target ontology embeddings for optimal transport. Instead of using uniform distributions for all points, we will consider a distribution reflecting the ground distances between the two sets. For example, if the ground distances from a source point to all target points are consistently large, the measure of the source point should be very small or nil in terms of mass transportation. Finally, for large scale ontologies, we need to break down the ontologies into smaller chunks for computing the optimal transport couplings. We will first partition the embeddings into clusters. Consequently, we will find candidate matchings from the pairs of clusters that have shorter Wasserstein distances.

References

1. Alvarez-Melis, D., Mroueh, Y., Jaakkola, T.: Unsupervised hierarchy matching with optimal transport over hyperbolic spaces. In: Chiappa, S., Calandra, R. (eds.) Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 108, pp. 1606–1617. PMLR (26–28 Aug 2020)
2. An, B., Chen, B., Han, X., Sun, L.: Accurate text-enhanced knowledge graph representation learning. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers). pp. 745–755. Association for Computational Linguistics, New Orleans, Louisiana (Jun 2018). https://doi.org/10.18653/v1/N18-1068, https://www.aclweb.org/anthology/N18-1068
3. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Transactions of the Association for Computational Linguistics **5** (2017)
4. Bordes, A., Usunier, N., García-Durán, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: Burges, C.J.C., Bottou, L., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing

Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States. pp. 2787–2795 (2013), http://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data

5. Chen, J., Jiménez-Ruiz, E., Horrocks, I., Antonyrajah, D., Hadian, A., Lee, J.: Augmenting ontology alignment by semantic embedding and distant supervision. In: Verborgh, R., Hose, K., Paulheim, H., Champin, P.A., Maleshkova, M., Corcho, O., Ristoski, P., Alam, M. (eds.) The Semantic Web (ESWC). pp. 392–408. Springer International Publishing, Cham (2021)

6. Cuturi, M.: Sinkhorn Distances: Light speed Computation of Optimal Transport. Master's thesis, Red Hook, NY, USA (2013)

7. Demeester, T., Rocktäschel, T., Riedel, S.: Lifted rule injection for relation embeddings. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. pp. 1389–1399. Association for Computational Linguistics, Austin, Texas (Nov 2016). https://doi.org/10.18653/v1/D16-1146, https://www.aclweb.org/anthology/D16-1146

8. Hamilton, W.L.: Graph representation learning. Synthesis Lectures on Articial Intelligence and Machine Learning **14**(3), 1–159 (2020)

9. Kolyvakis, P., Kalousis, A., Smith, B., Kiritsis, D.: Biomedical ontology alignment: an approach based on representation learning. J Biomed Semant **9**(21) (2018)

10. Lin, Y., Liu, Z., Sun, M., Liu, Y., Zhu, X.: Learning entity and relation embeddings for knowledge graph completion. In: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence. p. 2181–2187. AAAI'15, AAAI Press (2015)

11. Lu, F., Cong, P., Huang, X.: Utilizing textual information in knowledge graph embedding: A survey of methods and applications. IEEE Access **8**, 92072–92088 (2020)

12. Ma, S., Ding, J., Jia, W., Wang, K., Guo, M.: Transt: Type-based multiple embedding representations for knowledge graph completion. In: Ceci, M., Hollmén, J., Todorovski, L., Vens, C., Dzeroski, S. (eds.) Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18-22, 2017, Proceedings, Part I. Lecture Notes in Computer Science, vol. 10534, pp. 717–733. Springer (2017). https://doi.org/10.1007/978-3-319-71249-9_43, https://doi.org/10.1007/978-3-319-71249-9_43

13. Nasr, E.: Evaluation of Automatic Ontology Matching for Materials Sciences and Engineering. Master's thesis, Albert Ludwig University of Freiburg, Germany (2020)

14. Nkisi-Orji, I., Wiratunga, N., Massie, S., Hui, K.Y., Heaven, R.: Ontology alignment based on word embedding and random forest classification. Machine Learning and Knowledge Discovery in Databases **11051I** (2019)

15. Rocktäschel, T., Singh, S., Riedel, S.: Injecting logical background knowledge into embeddings for relation extraction. In: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1119–1129. Association for Computational Linguistics, Denver, Colorado (May–Jun 2015). https://doi.org/10.3115/v1/N15-1118, https://www.aclweb.org/anthology/N15-1118

16. Veira, N., Keng, B., Padmanabhan, K., Veneris, A.: Unsupervised embedding enhancements of knowledge graphs using textual associations. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19. pp. 5218–5225. International Joint Conferences on Artificial Intelligence Organization (7 2019). https://doi.org/10.24963/ijcai.2019/725, https://doi.org/10.24963/ijcai.2019/725

17. Wang, Y., Zhang, H., Shi, G., Liu, Z., Zhou, Q.: A model of text-enhanced knowledge graph representation learning with mutual attention. IEEE Access **8**, 52895–52905 (2020). https://doi.org/10.1109/ACCESS.2020.2981212
18. Wang, Z., Li, J.Z.: Text-enhanced representation learning for knowledge graph. In: IJCAI (2016)
19. Wang, Z., Zhang, J., Feng, J., Chen, Z.: Knowledge graph embedding by translating on hyperplanes. In: AAAI (2014)
20. Zhang, Y., Chen, Q., Yang, Z., Lin, H., Lu, Z.: Biowordvec, improving biomedical word embeddings with subword information and mesh. Scientific Data **6**(52) (2019)