Types of Property Pairs and Alignment on Linked Data – A Preliminary Analysis

Kno.e.sis Center, Wright State University, Dayton, OH USA
Kalpa Gunaratna, Krishnaprasad Thirunarayan, and Amit Sheth

1. Overview

Linked Open Data has encouraged dataset publication on the Web. **Data interoperability** and **integration** are **challenging** in this vast data space. These issues should be addressed at different levels, and property alignment is one such level. We focus on **object-type property alignment** and their relative distribution in LOD.

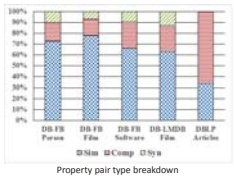
We propose a novel technique using **property extensions** to match **equivalent properties** between linked datasets and it outperforms existing syntactic and semantic techniques.

Challenges

- *Properties capture meaning in RDF triples and hence are complex.*
- *Name heterogeneities exist in LOD.*
- *Property alignment requires complex algorithms that go beyond mere name analysis of the properties.*

4. Analysis

Property distribution



Property pair type breakdown

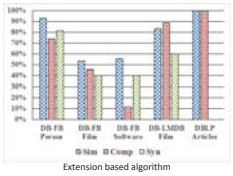
Exact naming of property pairs

Datasets

- 5000 instance samples from DBpedia, Freebase, LinkedMDB, DBLP RKB Explorer, and DBLP L3S.
- They belong to Person, Film, Software, and Articles.
- Sim, Comp, and Syn stand for simple, complex, and synonymous property pairs respectively.

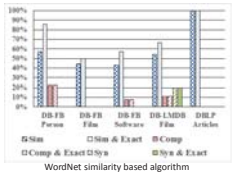
- Majority of the property pairs are simple followed by complex and synonymous.
- We expect to achieve higher accuracy using simple matching techniques as many of the pairs are simple.
- Also, many do not have exactly the same string in the property names.

Property alignment



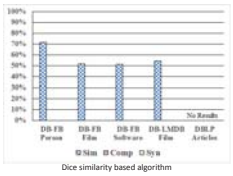
Extension based algorithm

Analyses the frequency of subject-object pairs matched in property extensions to decide on the alignment.



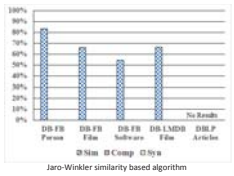
WordNet similarity based algorithm

Calculates WordNet similarity on property names for the alignment. It is calculated using normalized values of LCH, RES, HSO, JCN, LESK, PATH, WUP, and LIN measures.



Dice similarity based algorithm

Uses Dice's string similarity measure on property names for the alignment.



Jaro-Winkler similarity based algorithm

Uses Jaro-Winkler string similarity measure on property names for the alignment.

- Syntactic or dictionary based approaches failed to produce good results, even though majority of the pairs are simple.
- It shows that properties are harder to process for alignment using string manipulation techniques.
- Extension based analysis captures the meaning of properties for better alignment and outperformed others.

2. Extension based approach

Equivalent properties have exactly the same property extensions (OWL definition). In practice, high overlap decides equivalence, which we call ***Statistical Equivalence***‡.

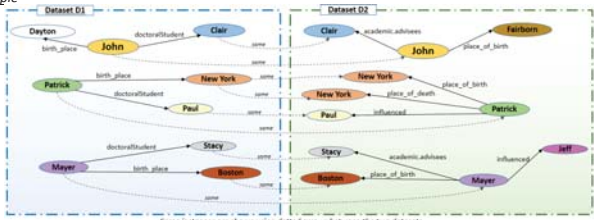
Match count $\mu(P_1,P_2) = |\{S_1P_1O_1 \in D_1 \mid \exists\, S_2P_2O_2 \in D_2 \wedge S_1 \overset{same*}{\longleftrightarrow} S_2 \wedge O_1 \overset{same*}{\longleftrightarrow} O_2\}|$

Co-appearance count $\lambda(P_1,P_2) = |\{S_1P_1O_1 \in D_1 \mid \exists\, S_2P_2O_2 \in D_2 \wedge S_1 \overset{same*}{\longleftrightarrow} S_2\}|$

* is used to denote a link path for the relationship

***Statistically Equivalent* property pair (P_1,P_2) has:** $\mu(P1,P2)/\lambda(P_1,P_2) \geq \alpha$ where, $\mu(P1,P2) \geq k$, and $0 < \alpha \leq 1, k > 1$

Example



Same instances are shown using dotted arrows between the two datasets.

Generated Candidate Matching Property Lists – Matches selected are in Boldface

[D1:doctoralStudent] [**[D2:academic.advisees, 2:2]** , [D2:influenced, 1:2]] ← matching list for doctoralStudent

[D1:birth_place] [**[D2:place_of_birth, 2:3]** , [D2:place_of_death, 1:1]] ← matching list for birth_place

Property Pair (matched)	MatchCount	Co-appearanceCount
[D1:doctoralStudent, D2:academic advisees]	2	2
[D1:birth_place, D2:place_of_birth]	2	3

Selection using $\alpha = 0.5$ and $k = 2$

3. Property pair types

Two orthogonal ways.

(1) ***On the basis of semantics***

- (i) **Equivalent properties**
- (ii) **Property - sub property**

(2) ***On the basis of techniques and tools required for processing***

- (i) **Simple property pairs**: Have high syntactic similarity in the property names. Gleaned from Common suffix, prefix, different ordering, e.g., *birthPlace* vs. *placeOfBirth.*
- (ii) **Opaque property pairs**: Same meaning but different words.
  - (a) **Synonymous property pairs**: Property names are synonymous and intentional. External dictionary or lexical database can be used, e.g., *occupation* vs. *profession, city of birth* vs. *place of birth*
  - (b) **Complex property pairs**: Similarity cannot be determined using property names alone. Requires extra processing like domain and range, and extensions. These are ambiguous, e.g., *battle* vs. *participated in conflict, resting place* vs. *place of burial.*

4. Analysis

	Measure type	Dbpedia – Freebase (Person)	Dbpedia – Freebase (Film)	Dbpedia – Freebase (Software)	Dbpedia – LinkedMDB (Film)	DBLP_RKB – DBLP_L3S (Article)	Average
Extension Based Algorithm	Precision	**0.8758**	**0.9737**	0.6478	0.7560	**1.0000**	**0.8427**
	Recall	**0.8089***	**0.5138**	**0.4339**	**0.8157**	**1.0000**	**0.7145**
	F measure	**0.8410***	**0.6727**	**0.5197**	**0.7848**	**1.0000**	**0.7656**
WordNet Similarity	Precision	0.5200	0.8620	0.7619	0.8823	**1.0000**	0.8052
	Recall	0.4140*	0.3472	0.3018	0.3947	0.3333	0.3582
	F measure	0.4609*	0.4950	0.4324	0.5454	0.5000	0.4867
Dice Similarity	Precision	0.8064	0.9666	0.7659	**1.0000**	0.0000	0.7078
	Recall	0.4777*	0.4027	0.3396	0.3421	0.0000	0.3124
	F measure	0.6000*	0.5686	0.4705	0.5098	0.0000	0.4298
Jaro Similarity	Precision	0.6774	0.8809	**0.7755**	0.9411	0.0000	0.6550
	Recall	0.5350*	**0.5138**	0.3584	0.4210	0.0000	0.3656
	F measure	0.5978*	0.6491	0.4903	0.5818	0.0000	0.4638

Evaluation results of the four techniques used for the selected datasets of 5000 instances each. * marks estimated values because of the number of pairs to be manually evaluated. Boldface numbers represent the highest results for each dataset. Clearly, the extension based approach produces favorable results because it is able to capture semantics other than analyzing property names for alignment.

5. Conclusion

- We classify and study types of property pairs that exist in LOD considering techniques required for equivalent property alignment.
- Our analysis shows that the majority of the pairs are simple property pairs, but as string manipulation based simple techniques did not perform well, they should be extended to improve coverage.
- Property extensions could be analyzed to determine equivalent properties between two datasets effectively. It achieved higher precision and recall in every case showing its applicability to Linked Datasets.

‡ Gunaratna, K., Thirunarayan, K., Jain, P., Sheth, A., Wijeratne, S.: A statistical and schema independent approach to identify equivalent properties on linked data. In: 9th International Conference on Semantic Systems. ACM (2013)

Acknowledgement: Sponsored by NSF Award #IIS-1143717, "III: EAGER Expressive Scalable Querying over Linked Open Data."

More details at: http://bit.ly/13hIE6r