Ranking Entities in a Large Semantic Network

Michael Schuhmacher, Simone Paolo Ponzetto

Research Group Data and Web Science, University of Mannheim
{michael,simone}@informatik.uni-mannheim.de

Abstract. We present two knowledge-rich methods for ranking entities in a semantic network. Our approach relies on the DBpedia knowledge base for acquiring fine-grained information about entities and their semantic relations. Experiments on a benchmarking dataset show the viability of our approach.

1 Introduction

Entity ranking [2] is the task of ordering a given set of entities on the basis of their relevance with respect to a reference entity. As an example, "Apple Inc." can have different degrees of association with other entities, ranging from highly related ones ("Steve Jobs") to mildly ("NeXT") or marginally relevant ones ("Ford Motor Company") – see Figure 1. Entity ranking can be produced automatically by computing the degree of semantic relatedness between the reference entity, and each of the other entities of interest. Much work in the field of Natural Language Processing has focused on knowledge-rich approaches to semantic relatedness [5]. However, almost all approaches using knowledge resources rely on the hierarchical structure of a taxonomy, typically WordNet, as opposed to full-fledged semantic networks – like, for instance, DBpedia [1] – containing fine-grained, explicit semantic relations, and whose taxonomic backbone represents only a fraction of the semantic information they encode.

2 Knowledge-based entity ranking

We study two knowledge-rich methods to rank entities in a semantic network.

Path-based method. We compute relatedness directly on the basis of the cheapest path between two entities. This leverages information from the knowledge base by means of a weighting method that takes into account the explicit semantic relations found within the resource (see [4] for details):

1) We build from the set of input entities a labeled, directed graph containing the entities themselves and all intermediate entities and relations in the knowledge base.
2) We weight graph edges by edge cost, where weights capture the degree of associativity between the source and target nodes. We use the information-theoretic measures of [4] to capture different degrees of associations between entities in the semantic network on the basis of their specificity.

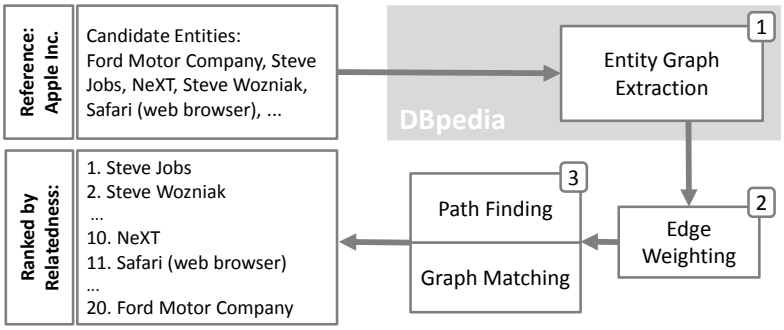

Fig. 1. Entity ranking workflow.

3) We compute semantic distances between entity pairs – i.e., the reference entity and each of the entities of interest – as the minimum path cost between them in our weighted graph. Finally, we rank the entity pairs increasingly by semantic distance.

Graph-matching method. The approach described so far relies on entities being connected by meaningful semantic relations in the reference resource. However, this requirement could be too strict for some entities, namely those for which very few or no informative semantic relations exist in the underlying ontology (e.g., technical terms like "Oxygen fluoride" or "Manifold"). Even more problematic, this method cannot be applied to entities that are not found in the knowledge base – e.g., how much related is "Simone Ponzetto" with "I-SEMANTICS"? For this reason, we explore a second, alternative approach:

1) We represent each entity using the set of entities linked within the abstract of the corresponding (English) Wikipedia article. For example, "Manifold" links to "Topological Space", "Lemniscate", "Klein bottle", and so on.
2) For each set of entities, we build a weighted semantic graph following the previously described graph construction method, in order to identify the subgraph of DBpedia covered by each definition.
3) Given that each entity is now represented as a subgraph, we view computing relatedness as a graph comparison problem, and compute relatedness using a Graph Edit Distance based measure, which finds the optimal matching between two entity-based graphs using the Hungarian method [3].

Our hunch here is to use a knowledge-rich text similarity method applied to the entities' textual descriptions in order to overcome the limited availability of semantic relations for some entities in the knowledge base. Crucially, this method enables knowledge-rich entity ranking even for those entities which are not in the knowledge base, provided they can be associated with a semantified textual description. To this end, we build upon the work from [4] who present a structure-based method to compute semantic similarity between documents, here applied to compute entity relatedness instead.

	Path-based				Graph-based			
	baseline	jointIC	combIC	IC+PMI	baseline	jointIC	combIC	IC+PMI
Hollywood Celebr.	0.639	0.541	**0.690**	0.661	0.439	**0.506**	0.417	0.401
IT Companies	0.559	0.636	**0.644**	0.583	0.355	**0.446**	0.298	0.278
Television Series	0.529	0.595	**0.643**	0.602	0.302	**0.473**	0.300	0.280
Video Games	0.451	**0.562**	0.532	0.484	**0.552**	0.519	0.434	0.424
Chuck Norris	0.458	0.409	**0.558**	0.506	0.448	**0.544**	0.425	0.291
All	0.541	0.575	**0.624**	0.579	0.414	**0.489**	0.365	0.343

Table 1. Performance on the entity ranking KORE dataset.

3 Experiments

Experimental setting. We use the KORE entity ranking dataset [2], consisting of 21 different reference entities from four different domains. Relatedness assessments were obtained using a crowd-sourcing approach. We evaluate using Spearman's rank correlation (ρ) and DBpedia 3.8 as knowledge base.

Results and discussion. The results in Table 1 indicate that weighing paths based on their information content (as introduced in [4]) consistently outperforms a baseline approach that simply computes entity relatedness as a function of distance in the network. In the case of the path-based approach, the best weighting schema is combIC, which achieves an average increase of 15.5% (statistically significant for each task at $p \leq .001$ level with paired t-test).

The graph-matching approach always performs lower than the cheapest path based method. Error analysis revealed that this is due to the fact that, although the Wikipedia abstracts from which entity graphs are built provide us with an enriched context, they also introduce noise deriving from generic entities – especially in the case of popular (and hence, highly hyperlinked) entities. For instance, in the abstract for "Apple Inc." we found hyperlinks to "Coca-Cola" and "Fortune 500". While a context-based approach could still help with those poorly connected entities, we opt here for evaluation on benchmarking data (i.e., KORE) and leave further experimental analysis for future work.

Path-based method with top-K paths. Our path-based method achieves competitive performance – when compared against [2], our methods achieves a performance only slightly lower than their original proposal ($\rho = 0.673$), while outperforming all its approximations ($\rho = 0.621$ and 0.425). However, our approach relies only on the single cheapest path connecting two entities. Consequently, we analyze the impact of taking multiple paths between a pair of entities, and aggregating evidence by averaging their costs to compute the final relatedness score. We show the results in Figure 2. For all three weighting schemes the performance of our method monotonically decreases with the number of top-k paths used for computing relatedness. The best results are obtained for $k = 1$, namely the cheapest path only, thus indicating that robust performance on this task relies on finding specific, highly informative paths – and thus meaningful semantic relations – between entities. Again, the best results are obtained using the combIC weighting, which outperforms all other measures for any k.

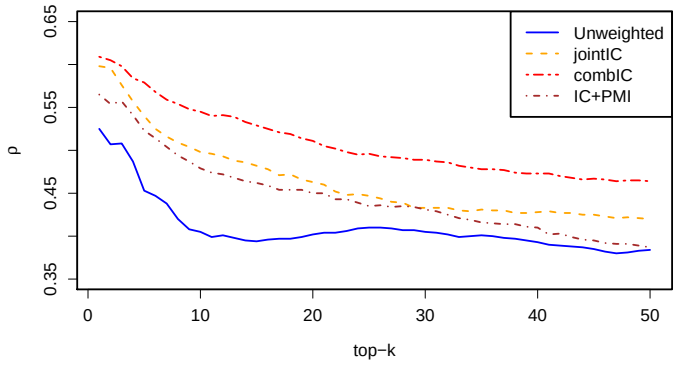

Fig. 2. Results using top-k average path costs.

Path-based method with different knowledge base. We next perform a diachronic evaluation by evaluating the path-based method using the latest DBpedia Version (3.9), which contains more entities (+6.2%) and semantic relations (+23.9%)[1]. Results for all three weighting approaches show minimal variations (ρ =0.592, 0.620 and 0.580 for jointIC, combIC and IC+PMI, respectively), as opposed to the unweighted baseline, which, in contrast decreases by more than 6 points (−11.3%). Manual inspection revealed that the increased amount of new relations causes the unweighted approach more often to choose noisy, i.e., low-informative paths. In contrast, thanks to our weighting, we are able to maintain a stable performance, regardless of the continuous growth of the network.

4 Conclusions

We presented a knowledge-rich approach to entity ranking. Results indicate that fine-grained semantic information from a wide-coverage knowledge base can be effectively used for this task when combined with robust weighting and path search techniques. Future work will explore multilinguality, and exploit relatedness scores of unknown entities for knowledge base population.

References

1. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ive, Z.: Dbpedia: A nucleus for a web of open data. In: ISWC+ASWC. pp. 722–735 (2007)
2. Hoffart, J., Seufert, S., Nguyen, D.B., Theobald, M., Weikum, G.: KORE: Keyphrase overlap relatedness for entity disambiguation. In: CIKM. pp. 545–554 (2012)
3. Riesen, K., Bunke, H.: Approximate graph edit distance computation by means of bipartite graph matching. Image Vision and Computing 27(7), 950–959 (2009)
4. Schuhmacher, M., Ponzetto, S.P.: Knowledge-based graph document modeling. In: WSDM. pp. 543–552 (2014)
5. Zhang, Z., Gentile, A.L., Ciravegna, F.: Recent advances in methods of lexical semantic relatedness – a survey. Natural Language Engineering 1(1), 1–69 (2012)

[1] `http://wiki.dbpedia.org/Datasets39/DatasetStatistics`