Entity Resolution in Texts Using Statistical Learning and Ontologies

Tadej Štajner, Dunja Mladenić

Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia
{tadej.stajner, dunja.mladenic}@ijs.si

Abstract. Ambiguities, which are inherently present in natural languages represent a challenge of determining the actual identities of entities mentioned in a document (e.g., *Paris* can refer to a city in France but it can also refer to a small city in Texas, USA or to a 1984 film directed by Wim Wenders having title *Paris, Texas*). Disambiguation is a problem that can be successfully solved by entity resolution methods.
This paper studies various methods for estimating relatedness between entities, used in collective entity resolution. We define a unified entity resolution approach, capable of using implicit as well as explicit relatedness for collectively identifying in-text entities. As a relatedness measure, we propose a method, which expresses relatedness using the heterogeneous relations of a domain ontology. We also experiment with other relatedness measures, such as using statistical learning of co-occurrences of two entities or using content similarity between them. Evaluation on real data shows that the new methods for relatedness estimation give good results.

Keywords: Entity resolution, text mining, semantic annotation, ontology mapping

1 Introduction

Integration and sharing of data across different data sources is the basis for an intelligent and efficient access to multiple heterogeneous resources. Since a lot of knowledge is present in plain text rather than a more explicit format, an interesting subset of this challenge is integrating texts with structured and semi-structured resources, such as ontologies. This is especially interesting in the context of Open Linked Data, where the main motivation is to have cross-dataset mappings across as many datasets as possible. However, textual datasets have to be treated differently in some ways. This involves dealing with natural language ambiguities in names of entities. We formulate this as an entity resolution problem, where we are trying to choose the correct corresponding entities from the ontology for the entities mentioned in text.

Our goal is to explore possible improvements of entity resolution quality by using ontologies in different ways along with statistical knowledge. To achieve this, we

experiment with using different kinds of available data that could help in improving in-text entity resolution quality. Since entities, which are related, tend to appear together in documents more often, we explore the possibilities of expressing relatedness in different ways, such as similarities of entities' descriptions, the entity graph topology and entity co-occurrence information.

For example, in the case when we have a document where there are two unknown entities referred to by the names "Elvis" and "Memphis". The first is a common personal name and the second one the name of several locations. We would like to use this relatedness information between those two entities to help in resolving "Elvis" as a well-known singer and "Memphis" as a city in Tennessee, where the identified singer lived.

A long-term goal of this work is to improve the quality of in-text entity resolution using existing ontologies and mappings between them. In other words, we would like to be able to bootstrap existing knowledge with the intention of obtaining new knowledge.

2 Related work

Machine learning methods are successfully being used in text mining and analysis of documents [1]. Problems, analogous to entity resolution appear in many different areas. The theoretical foundations of entity resolution are defined in the theory of record linkage [2]. Related challenges can also be found in database integration [3,4], object identification [5], duplicate detection [6] and word sense disambiguation [7,8].

When observing our problem statement from a natural language processing perspective, we can describe our approach as disambiguation using background knowledge, which is a pattern, often found in literature [10,11,12]. For the purposes of this paper we use the ontology as background knowledge represented as a graph of entities, identified with URIs, described with attributes and interconnected with different relationships. Such models can be easily constructed from RDF data [13], which is general enough to describe other domains, such as entity-relational and class models [14]. We also require that we are aware of possible phrases that represent possible labels[1] of entities. As we will show in subsequent sections, we can also benefit from having descriptions[2] of entities, which can be used beneficially for entity resolution via vector space model similarity [11,15,16,17].

There also exist methods which use relational information for disambiguation, [18], which estimates relevance with a PageRank score over candidate meanings. A collective approach using Markov logic is shown in [19]. Since different relation types have different meaning, [20] suggests an adaptive method of determining relational significance.

When solving the entity resolution problem, the usual approach involves performing graph clustering over the entity graph using a certain similarity criterion

[1] http://www.w3.org/2000/01/rdf-schema#label
[2] http://www.w3.org/2000/01/rdf-schema#comment

[9]. In context of relational data, it is a combination of attribute similarity and relational similarity. However, such approaches are more often found in structured data, whereas our approach attempts to use these techniques on linking unstructured text with semi-structured data. Also, when using ontologies as a sense inventory, relationships between entities are heterogeneous. The proposed novel method for determining relatedness in collective entity resolution is based on using relational entity resolution. A distinction in entity resolution approaches can be made in regard to the entity resolution independence assumptions:

- Pair-wise resolution - decisions are being done independently for each mention of an entity in the document
- Collective resolution - decisions do not assume independence of resolution decisions, enabling us to use relatedness data in the subsequent decisions.

Since collective entity resolution can take relatedness between entities into account, we experiment with the following definitions of relatedness:

- **Content similarity as a relatedness measure** can be used in situations where only available data is in form of attributes and textual descriptions and no explicit relationships between entities, as shown in [22].
- **Entity co-occurrences as a relatedness measure** are useful in situations where we can obtain a corpus of documents, annotated with resolved in-text entities, which can be used as a training set for a supervised approach to entity resolution. In general, co-occurrences are a common source of training data for information retrieval problems, analogous to entity resolution. Use cases that apply this technique can be found in [23], who uses it for protein identification and [24], who successfully resolves geographical locations. Utilization of entity co-occurrences for identifying synonyms in a unsupervised approach, which is analogous to entity resolution, can be seen in [25]. Co-occurrences have also been used to construct a generative model [8] for entity resolution.
- **Explicit relationships as a relatedness measure:** relationships between entities are the most explicit form of relatedness. However, not all relationships have the same significance. This paper proposes one such possible approach to heterogeneous relational entity resolution which bases relational significance on the frequency of the relation appearing in the ontology with regard to entity types. This measure was suggested in [21] as one of the suggested methods of determining a minimal informative subgraph of a graph. Since this problem as well as multi-relational entity resolution both use the notion of relational significance, this paper will explore the possibilities of using this measure as a means of quantifying relatedness between entities.

3 Entity resolution from text

3.1 Treating disambiguation as entity resolution

For representing the text as a collection of entities, the necessary first step is to identify potential entities in the text. However, since the entity resolution algorithm can benefit from better information on the in-text entity, we added a named entity extraction step. For this purpose we used the Stanford Named Entity Recognizer [26]. Before using our background knowledge base, we can still perform a part of co-reference resolution with the identified entities, such as canonicalization, partial name consolidation and acronym consolidation. Simple de-duplication of extracted entities also helps to reduce the search space when performing collective entity resolution. Once we have a basic understanding of which distinct entities we are trying to resolve, we can search our ontology for possible candidates that could match the named entities. We then perform a series of decisions, where entities from the document are matched with the most relevant ontology entity based on some relevance criteria. This is then repeated as long as there are unmatched entities in the documents or none of the remaining candidates fulfill the minimum criteria for matching.

3.2 Pair-wise entity matching

When matching an entity from the document to a candidate entity, we employ some heuristics to evaluate the confidence of their match. One such heuristic is description similarity. Note that since this scenario has no *a priori* matches of document entities with ontology entities, we have no use for relational information.

When a single entity has multiple documents, as shown in example in Fig.2, our task is to evaluate each candidate and finally match the document entity with the top candidate. Description similarity is defined as the cosine similarity of TF-IDF vectors of descriptions that represent the given entities. Since one of the entities is a document entity, its descriptions is the document text itself. We then resolve each entity in the document to its most similar candidate among the candidates from the ontology.

3.3 Collective resolution with relatedness

While leaving behind assumptions of independence, we can then benefit from using information on relatedness between entities. Collective candidate selection is performed with the following sequence of steps, adapted from the relational clustering algorithm [9] and adapted from the general dataset reconciliation domain to a text-ontology alignment scenario.

Required: document entities, candidate entities;
Initialize priority queue *q,* list *selected_matches*;

For each potential pair between document entity *f* and candidate entity *e*:
    Insert *(pairwise_relevance(f,e), f, e)* into *q;*
While *q* is not empty:
    Pop *(relevance$_{f,e}$, f, e)* from *q*;
        Add *(relevance$_{f,e}$, f, e)* to *selected_matches*;
    For each *entry* in *q* containing *f*:
        Remove *entry* from *q*;
For each *entry* in q,:
    Update *collective_relevance(e$_{entry}$, f$_{entry}$, selected_matches)*;
Return *selected_matches*

Fig. 1. Collective in-text entity resolution algorithm

Fig. 1 describes the adapted entity resolution algorithm. The high-level operation is the same for all of the described approaches.

$$relevance_{collective}(f, e, S) = relevance_{pairwise}(f, e) + \lambda \cdot \frac{\sum_{e_s \in S} related(e, e_s)}{|\{e_s \in S;\ related(e, e_s) \neq 0\}|}$$

Fig. 2. Collective relevance estimate as a combination of pair-wise and relational similarity

The three approaches differ only in the calculation details of the *relatedness* estimate, which is used in collective relevance calculation, as seen in Fig. 2. The following chapters will describe the respective relatedness estimation approaches.

3.3.1 Using semantic relations from the ontology

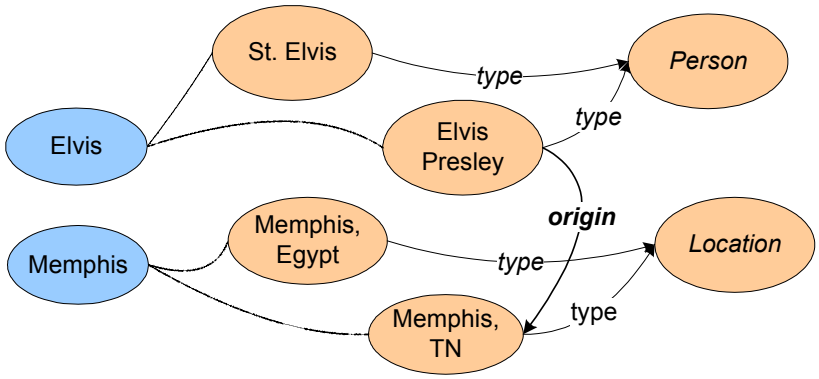

Fig. 3. Using different semantic relations as a relatedness measure

In Fig. 3, the blue nodes (Elvis and Memphis on the left) represent the document entities, whereas all the other nodes (colored pink) represent entities from the ontology. In this case, the relatedness between the entities is expressed explicitly in

the form of RDF statements in the background knowledge - as shown in Figure 3. Consider the case where the subject »Elvis Presley in relation »Hometown« (as his »origin«) to the subject »Memphis, Tennessee«. For use in our resolution model, we interpret relations as links with a specified weight. If the relations in the ontology are only of a single type, they can all be treated equivalently. However, when dealing with heterogeneous ontologies, as is often the case, one has to estimate the importance of each link. For instance, if the ontology contained the RDF statement <*Elvis Presley, type, Person*>, this would not be too useful, since it would likely encompass every entity called "Elvis" since they are mostly of the type »Person«. On the other hand, the relation <*x, Hometown, Memphis_Tennessee*> is a strong indicator, because it covers a much smaller set of entities. This property is defined as selectivity, and its value can be used as a weighting of links in the graph. Determining the selectivity of the links is a problem, similar to finding the most informative subgraph in a given semantic graph, described in [21]. The authors wanted to find the smallest subgraph, which would be sufficiently informative. For the purposes of determining subsets of the links they have developed a few metric to estimate the selectivity. One of the proposed metrics, which is also suitable for our domain, is Instance Participation Selectivity, which stipulates that the selectivity of the assertion <*s, p, o*> is inversely proportional the number of statements RDF which correspond to the <*type (s), p, type (o)* > where the predicate "*type(x)*" is defined as the relation of *rdf:type* of the entity. Let *π(type(s),p,type(o))* be the set of all statements in the domain ontology, where type of subject is *type(s)*, the predicate is *p* and type of object is *type(o)*.

$$IPS(s,p,o) = \frac{1}{|\pi(type(s),p,type(o))|}$$

To balance the estimate values for our use case, this paper modifies the equation slightly to:

$$IPS_{log}(s,p,o) = \frac{1}{\log\left(1 + |\pi(type(s),p,type(o))|\right)}$$

The consequence is that the link type <*Person, Origin, Area*> is less selective than <*Person, Origin, City*>, which is also what we want to model. This approach therefore enables us to quantify the relatedness of a pair of entities based on ontology data. The direct relatedness score is then calculated as:

$$relatedness_{direct}(e_i,e_j) = \frac{\sum_{<e_i,p,e_j>\in KB} IPS(e_i,p,e_j)}{|<e_i,p,e_j>\in KB|}$$

However, when considering actual relatedness, we also take into account not only direct relations, but also indirect ones – the relations to entities that are in the common neighborhood. We define this as:

$$Nbr(e) = \{f; relatedness_{direct}(e,f) \neq 0\}$$

$$Nbr(e_i,e_j) = Nbr(e_i) \cup Nbr(e_j)$$

We define indirect relatedness as an average of paths between both entities:

$$relatedness_{indirect}(e_i, e_j) = \frac{\sum_{f \in Nbr(e_i, e_j)} relatedness_{direct}(e_i, e_j)}{|Nbr(e_i, e_j)|}$$

We compute the final semantic relatedness score as a linear combination of direct and indirect relatedness:

$$relatedness_{ontology}(e_i, e_j) = \lambda_1 relatedness_{direct}(e_i, e_j) + \lambda_2 relatedness_{indirect}(e_i, e_j)$$

3.3.2 Using content similarity

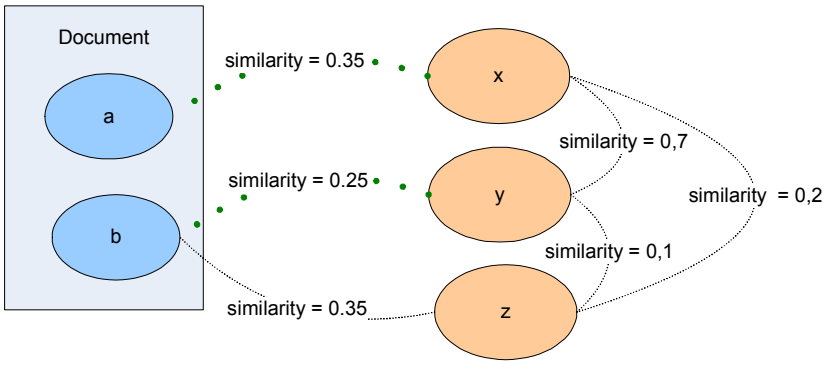

Fig. 4. Using content similarity as a relatedness measure (the green dotted lines represent selected entities)

In some situations, we do not have explicit relations between entities. If the entities have descriptive attributes, we use them to estimate relatedness with comparing their content similarity, as illustrated in Fig. 24. One advantage of such approach is that we do not require any more data than with pair-wise resolution, which adds to the flexibility of this method. This approach was first explored in [22] and is formulated as:

$$relatedness_{content}(e_i, e_j) = sim_{content}(e_i, e_j)$$

3.3.3 Using co-occurrences

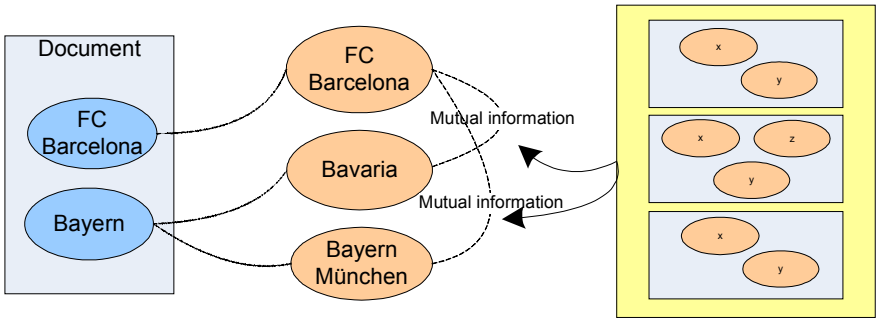

Fig. 5. Using mutual information form entity co-occurrences as relatedness

We can also represent relatedness between entities as co-occurrences, as shown in Fig. 3.Data on the co-occurrences of two events are successfully used in information retrieval problems such as cross-language information retrieval [29] and determining the importance of words [30,31], which is a problem related to entity resolution. Intuition for the use of the co-occurrences is, the more often that the two events occur together more frequently than by chance, the more likely is that they are related. This principle was also demonstrated in [11] with a collective generative model. In our domain, we can model relatedness with point-wise mutual information [32] of two entities occurring in the same document.

$$relatedness_{co-occurences}(e_i, e_j) \;=\; SI(e_i, e_j) = log\ \frac{p(e_i, e_j)}{p(e_i)p(e_j)}$$

$$8(y)$$

Since this procedure requires supervised statistical learning, its output quality depends on the quality and coverage of the training corpus.

3.4 Combining methods

Since each relevance estimation method produces its own relevance score, it would make sense to have means of combining them. This can be done with expressing the relatedness function as a linear combination of all relatedness estimation functions.

$$\begin{aligned} relatedness(e, f) \\ &= \lambda_1 relatedness_{content}(e, f) \\ &+ \lambda_2 relatedness_{co-occurences}(e, f) \\ &+ \lambda_3 relatedness_{ontology}(e, f) \end{aligned}$$

Fig. 6. Combining relatedness estimations

The lambda parameters in Fig. 6. are experimentally obtained using a hill-climbing approach by maximizing the average $F_{0.2}$ score for the test set.

4 Data

Our assumption is that the ontology consists of knowledge database that contains enough data to be able to perform the following tasks. First, it should be able to refer to each entity with multiple aliases to facilitate candidate retrieval., Second, it should be able to provide enough additional entity features, which we can use to compare those entities to each other and to article anchors that we attempt to link to. Following these requirements, we chose to use a part of DBpedia, as described in [34] for the facts that it provides both description and attribute data from Wikipedia, as well as references to other ontologies that describe other aspects of the same real-world objects. For the purpose of having rich heterogeneous relational data, we also used the Yago ontology, defined [35], which maps Wikipedia concepts to corresponding WordNet classes. Since a direct mapping from Yago to DBpedia exists, merging the two together is trivial. However, both ontologies are much broader than what our approach requires – we currently only use information on aliases, textual descriptions, *rdf:type* attributes and Yago categories of entities.

5 Evaluation

5.1 Methodology

For determining the quality of the methods we have used precision and recall, measured at a certain level of confidence in the suggested entities for a given article. We then compared the suggested entities for those articles with manually identified entities of those articles.

Precision and recall are balanced with a relevance score threshold, selecting only those entities whose relevance score is above this threshold. This serves as a useful balancing tool, since in many examples the entity cannot be correctly resolved because they do not even exist in the domain ontology. In those cases, even the best candidate has a relatively low score.

We report the final results the value of F_α, which is the weighted harmonic average of precision and recall. Namely, in some applications we want to rate precision higher than recall, as false positives are much less desired than false negatives. Therefore, we provide results for two α values, one with equally weighted precision and recall ($\alpha=1$) and one that weights precision higher than recall ($\alpha=0.2$).

We perform evaluation using the New York Times article corpus [33], using 39953 articles from January 2007 to April 2007 as training data for construction of TF/IDF weighted vectors. The articles were then processed with an implementation of the described algorithm. For evaluating the performance of different approaches we manually selected and evaluated 945 entity resolution decisions from 79 articles as either correct or incorrect. Those articles were then used as a test set on which we based our quality estimation. Since the methods of pair-wise content comparison, collective content comparison and collective relational comparison are unsupervised, they do not require any pre-labeled articles as training data. On the other hand, using

co-occurrences as a relatedness measure requires training data for statistical learning. For this purpose, we take the remainder of the articles that we did not use as a test set and process them with the collective relational comparison method. Since we wish to maximize the training data quality with our best effort, we use only entities whose relevance estimate is above a certain threshold. We used the same threshold which gives us 95% precision and 45% recall on our test data. The collective relational comparison is used because it gives the highest quality output for this purpose. We experimentally determined the parameters for the methods to maximize the $F_{0.2}$. These values depend on a specific ontology and text corpus, so they are not necessarily universally applicable.

5.2 Results

Method	**Relatedness**	**max $F_{1.0}$**	**max $F_{0.2}$**
Pair-wise		0.749	0.772
Collective	Content similarity	0.750	0.789
Collective	Co-occurrences	0.721	0.747
Collective	Relations	0.728	0.789
Collective	Combined	0.741	**0.799**

Table 1. F-scores of respective methods

Results show that additional information does indeed show improvement in $F_{0.2,}$ as can be seen in Table 1. However, on higher recall (on values over 0.55), collective methods show a tendency for having performance barely similar to the baseline method of pair-wise resolution. This is evident in relatively low $F_{1.0}$ scores. The reason for this behavior is that because collective resolution depends on earlier decisions when deciding on an entity candidate, it is sensitive to the case of misjudging an early decision within a document. However, this high precision at low recall comes at the expense of precision at high recall. In that case, it is merely comparable to that of the baseline method of pair-wise entity resolution. This is also the cause of the small differences we see in the $F_{1.0}$ score.

Method	Relatedness	Precision at max $F_{0.2}$	Recall at max $F_{0.2}$
Pair-wise		0.784	0.717
Collective	Content similarity	0.836	0.616
Collective	Co-occurrences	0.818	0.522
Collective	Relations	0.868	0.541
Collective	Combined	0.882	0.543

Table 2. Precision and recall at max $F_{0.2}$

Further observations in Table 2. confirm that while precision successfully increases for the max $F_{0.2}$ scenario, there is something to be desired regarding recall at that point.

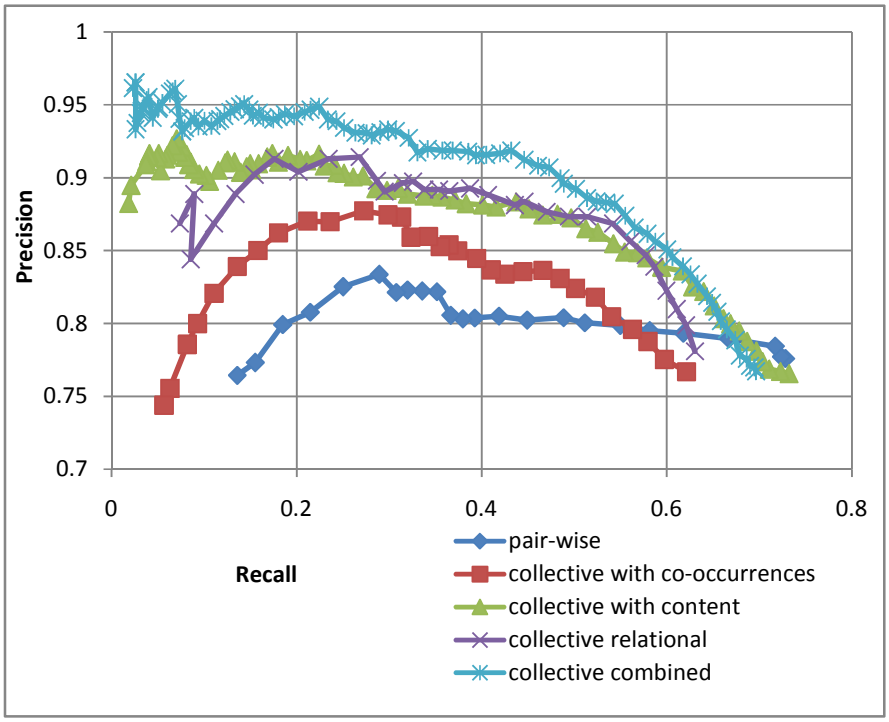

Fig. 7. Comparison of entity resolution quality of different relatedness heuristics

Fig. 7 shows that additional precision can in fact be obtained by performing relational entity resolution and this even applies to scenarios where we do not have homogenous relations between entities.

Method	Relatedness	Recall at 80% prec.	Recall at 90% prec.
Pair-wise		0.51	/
Collective	Content similarity	0.66	0.28
Collective	Co-occurrences	0.55	/
Collective	Relations	0.61	0.27
Collective	Combined	0.65	0.48

Table 3. Recall at two levels of high precision: 80% and 90% precision.

As is confirmed in Table 3., all collective methods have an advantage over the baseline when looking at recall at 90% precision. Here we can also demonstrate improvement with combining all of the aforementioned methods, which yields the

best overall result. We can also state that we have reached our goal of operating with higher precision, which would not be possible at all with simple pair-wise resolution.

When observing all the collective methods that we discuss in this paper, the best performing ones were collective resolution with content similarity and collective resolution with relational weighing. However, we are not able to confidently prove whether any of them is significantly better than the other. On the other hand, both outperformed the statistical learning method of counting entity co-occurrences. The cause of the lower performance of collective resolution with co-occurrences as relatedness is most likely the choice of the training set. Since it was not feasible to manually construct a training set of sufficient size, we decide to automatically construct a training corpus with the best performing method without using prior co-occurrences. For this purpose, we used collective resolution with combined multi-relational and content similarity. We selected only those entities, whose estimate was greater than the threshold that yielded 48% recall at 90% precision on the test set. The resulting performance is between the baseline and the performance of the training set for the greater part of the curve. This method of collective resolution with co-occurrences also exhibits a significant drop in precision at higher recall values. However, we can still conclude that even this method performs favorably to the baseline at higher thresholds. The best performance is obtained with collective combined method that is outperforming the other tested methods in the part with high precision and high recall. In the best performing range of recall between 0.3 and 0.5 this combined method is the only one that achieves precision over 0.9.

One of the causes for this sort of behavior is that some documents tend to discuss unrelated entities. Furthermore, in longer texts, the entities, mentioned at the beginning of the document are not necessarily related to the ones on the other parts of the document, which suggests that we should experiment with taking the document paragraph structure into account.

6 Conclusion

This paper proposes a framework for collective resolution of in-text entities on the basis of different notions of relatedness. As examples for this, we used three different relatedness estimation methods, each appropriate for a particular type of background knowledge. Among these methods we present and evaluate a novel method for determining relatedness based on commonness of ontological relations between two entity types and compare it to a supervised co-occurrence based approach and an approach using content similarity as relatedness. We confirm the previous related research that using collective resolution improves resolution precision and demonstrate this on various relatedness measures. Further improvement could be obtained by the use of machine learning on other segments of the problem, such as a means of determining the importance of relations rather than calculating their selectivity. A possible application in also in determining the significance of individual relevance estimates in the last step of calculation the total assessment.

The proposed solution capable of entity resolution from text is an important part of the knowledge extraction. The next level of this scenario would in addition to in-text entities, also identify the relations that occur between them. These newly identified relations between entities can be a basis for constructing new RDF statements, further building our ontology, thus closing the loop where we can use existing knowledge to obtain even more knowledge. This process brings new challenges, particularly in the field of selection of the appropriate statements on the basis of suitability for including them in the ontology, as discussed in [36]. Using this technology can also be useful for other purposes. Semantically expressed entities enable integration and interoperability with external data sources [37]. Also, visualization of the contents of the text in the format as described in [38] is also a use case for entity resolution.

On the other hand, our paper barely touches the possibilities that could be employed by using globally identified data approaches, opening way for better data integration, visualization and using annotated documents to enable semantic search. We expect that the proposed semantic article enrichment method to yield even more improvement on tasks that depend on the added semantic information, such as document summarization, triple extraction and recommendation systems. What all of those use cases have in common is dependence on a high quality output of the entity resolution phase.

References

1. Mladenić, D., Text Mining: Machine Learning on Documents," Encyclopedia of Data Warehousing and Mining, pp. 1109-1112, 2006.
2. Fellegi, I. and Sunter, A. "A theory for record linkage," Journal of the American Statistical Association, pp. 1183-1210, 1969.
3. Haas, L., Miller, R., Niswonger, B., Roth, M., Schwarz, P., and Wimmers, E., "Transforming heterogeneous data with database middleware: Beyond integration," IEEE Data Engineering Bulletin, vol. 22, no. 1, pp. 31-36,1999.
4. Winkler, W., "The state of record linkage and current research problems," Statistical Research Division, US Bureau of the Census, Wachington, DC, 1999.
5. Tejada, S., Knoblock, C., and Minton, S., "Learning object identification rules for information integration," Information Systems, vol. 26, no. 8, pp. 607-633, 2001.
6. Elmagarmid, A., Ipeirotis, P., and Verykios, V., Duplicate Record Detection: A Survey, 2006.
7. Yarowsky, D., Unsupervised word sense disambiguation rivaling supervised methods," in Proceedings of the 33rd annual meeting on Association for Computational Linguistics, pp. 189-196, Association for Computational Linguistics Morristown, NJ, USA, 1995.
8. Kalashnikov, D. and Mehrotra, S., A probabilistic model for entity disambiguation using relationships, in SIAM International Conference on Data Mining (SDM). Newport Beach, California, pp. 21-23, 2005.
9. Bhattacharya, I. and Getoor, L., Collective entity resolution in relational data, 2007

10. Schütze, H., Automatic word sense discrimination, Computational Linguistics, vol. 24, no. 1, pp. 97-123, 1998.
11. Bunescu, R. and Pasca, M., Using encyclopedic knowledge for named entity disambiguation, in Proceedings of the 11th Conference of the Euro pean Chapter of the Association for Computational Linguistics, pp. 3-7, 2006.
12. Cucerzan, S., Large-scale named entity disambiguation based on Wikipedia data, in Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pp. 708-716, 2007.
13. Klyne, G., Carroll, J., and McBride, B., Resource description framework (RDF): Concepts and abstract syntax, W3C recommendation, vol. 10, 2004.
14. Bizer, C., and Seaborne, A., D2RQ-treating non-RDF databases as virtual RDF graphs, in Proceedings of the 3rd International Semantic Web Conference (ISWC2004), 2004.
15. McCallum, A.: Information extraction: Distilling structured data from unstructured text," Queue, vol. 3, no. 9, pp. 48-57, 2005.
16. Lloyd, L., Bhagwan, V., Gruhl, D., and Tomkins, A.: Disambiguation of references to individuals, IBM Research Report, 2005.
17. Salton, G., Wong, A., and Yang, C.: A vector space model for automatic indexing, Communications of the ACM, vol. 18, no. 11, pp. 613-620, 1975.
18. Mihalcea, R.: Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling, in Proceedings of the conference on Human Language Technology and EMNLP, pp. 411-418, Association for Computational Linguistics Morristown, NJ, USA, 2005.
19. Singla, P. and Domingos, P.: Entity resolution with markov logic, in Proceedings of the Sixth IEEE International Conference on Data Mining, pp. 572-582, 2006.
20. Chen, Z., Kalashnikov, D., and Mehrotra, S.: Adaptive graphical approach to entity resolution, in Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries, pp. 204-213, ACM New York, NY, USA, 2007.
21. Ramakrishnan, C., Milnor, W. H., Perry, M., and Sheth, A. P.: Discovering informative connection subgraphs in multi-relational graphs," SIGKDD Explor. Newsl., vol. 7, no. 2, pp. 56-63, 2005.
22. Štajner, T.: From unstructured to linked data: entity extraction and disambiguation by collective similarity maximization, Identity and reference in web-base knowledge representation workshop, 2009
23. Li, X., Morie, P., and Roth, D., Semantic integration in text: From ambiguous names to identifiable entities, AI Magazine. Special Issue on Semantic Integration, vol. 26, no. 1, pp. 45-58, 2005.
24. Bunescu, R., Mooney, R., Ramani, A., and Marcotte, E.: Integrating co-occurrence statistics with information extraction for robust retrieval of protein interactions from Medline," in Proceedings of the BioNLP Workshop on Linking NLP Processing and Biology at HLTNAACL, vol. 6, pp. 49-56, 2006.
25. Overell, S., Magalhaes, J., and Ruger, S.: Place disambiguation with co-occurrence models, in CLEF 2006 Workshop, Working notes, 2006.

26. Yates A., and Etzioni, O.: Unsupervised resolution of objects and relations on the Web," in Proceedings of NAACL HLT, pp. 121-130, 2007.
27. Finkel, J., Grenager, T., and Manning, C.: Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling," Ann Arbor, vol. 100, 2005.
28. Cohen, W., Ravikumar. P., and Fienberg, S.: A comparison of string distance metrics for name-matching tasks, in Proceedings of the IJCAI-2003 Workshop on Information Integration on the Web (IIWeb-03), 2003.
29. Jang, M., Myaeng, S., and Park, S.: Using mutual information to resolve query translation ambiguities and query term weighting, in Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics, pp. 223-229, Association for Computational Linguistics Morristown, NJ, USA, 1999.
30. Church, K. and Hanks, P.: Word association norms, mutual information, and lexicography," Computational linguistics, vol. 16, no. 1, pp. 22-29,1990.
31. Li H., and Abe, N.: Word clustering and disambiguation based on cooccurrence data," in Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics-Volume 2, pp. 749-755, Association for Computational Linguistics Morristown, NJ, USA, 1998.
32. Manning, C. D. and Schütze, H.: Foundations of statistical natural language processing. Cambridge, MA, USA: MIT Press, 1999.
33. Sandhaus, E: The New York Times Annotated Corpus, 2008.40
34. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R. and Ives, Z.: Dbpedia: A nucleus for a web of open data," Lecture Notes in Computer Science, vol. 4825, p. 722, 2007.
35. Suchanek, F., Kasneci, G. and Weikum, G.: Yago: a core of semantic knowledge, in Proceedings of the 16th international conference on World Wide Web, pp. 697-706, ACM New York, NY, USA, 2007.
36. Suchanek, F. M., Sozio, M. and Weikum, G.: Sofie: a self-organizing framework for information extraction," in WWW '09: Proceedings of the 18th international conference on World wide web, (New York, NY, USA), pp. 631-640, ACM, 2009.
37. Decker, S., Melnik, S., Van Harmelen, F., Fensel, D., Klein, M., Broekstra, J., Erdmann, M. and Horrocks, I.: The semantic web: The roles of XML and RDF, IEEE Internet computing, vol. 4, no. 5, pp. 63-73, 2000.
38. Fortuna, B., Grobelnik, M., and Mladenić, D.: Visualization of text document corpus," Special Issue: Hot Topics in European Agent Research, vol. 29, pp. 497-502, 2005