Automatic Identifying Entity Type in Linked Data

Qingliang Miao, Ruiyu Fang, Shuangyong Song, Zhongguang Zheng, Lu Fang, Yao Meng, Jun Sun
Fujitsu R&D Center Co., Ltd.
Chaoyang District, Beijing P. R. China 100027
{qingliang.miao,fangruiyu,shuangyong.song,zhengzhg,fanglu,mengyao,sunjun}@cn.fujitsu.com

Abstract

Type information is an important component of linked data. Unfortunately, many linked datasets lack of type information, which obstructs linked data applications such as question answering and recommendation. In this paper, we study how to automatically identify entity type information from Chinese linked data and present a novel approach by integrating classification and entity linking techniques. In particular, entity type information is inferred from internal clues such as entity's abstract, infobox and subject using classifiers. Moreover, external evidence is obtained from other knowledge bases using entity linking techniques. To evaluate the effectiveness of the approach, we conduct preliminary experiments on a real-world linked dataset from Zhishi.me [1]. Experimental results indicate that our approach is effective in identifying entity types.

1 Introduction

An increasing number of linked datasets is published on the Web. At present, there have been more than 200 datasets in the LOD cloud. Among these datasets, DBpedia (Bizer, C. *et al.*, 2009) and Yago (Suchanek, F.M. *et al.*, 2007) serve as hubs in LOD cloud. As the first effort of Chinese LOD, Zhishi.me (Niu, X. *et al.*, 2011) extracted RDF triples from three largest Chinese encyclopedia web sites i.e. Chinese Wikipedia, Baidu Baike[2] and Hudong Baike[3]. However, type information is incomplete or missing in these linked datasets. For example, more than 36% of type information is missing in DBpedia (Kenza Kellou-Menouer and Zoubida Kedad, 2012). Zhishi.me only uses the SKOS vocabulary to represent the category system and does not strictly define the "*rdf:type*" relation between instances and classes.

Type information is an important component of linked datasets. Knowing what a certain entity is, e.g., a person, organization, place, etc., is crucial for enabling a number of desirable applications such as query understanding (Tonon, A. *et al.*, 2013), question answering (Kalyanpur, A. *et al.*, 2011; Welty, C. *et al.*, 2012), recommendation (Lee, T. *et al.*, 2006; Hepp, M. 2008), and automatic linking (Aldo Gangemi *et al.*, 2012). Since it is often not feasible to manually assign types to all instances in a large linked data, automatic identifying type information is desirable. Furthermore, since open and crowd-sourced encyclopedia often contain noisy data, filtering out the incorrect type information is crucial as well (Heiko Paulheim and Christian Bizer, 2013).

Recently, more and more attention has been paid to extracting or mining type information from linked

[1] http://zhishi.me/

[2] http://baike.baidu.com/

[3] http://www.baike.com/

data. However, most of current techniques on obtaining type information are either language-dependent or inferring type information only from internal clues such as textual description of entity. Most existing work was mainly focused on mining entity type from internal clues, and missed out the point that the issue can be boosted by integrating external evidence. Our assumption is that given an entity e_1 without type information, if we can find an equivalent entity e_2 with type information, we can obtain the type information of e_1 directly.

In this paper, we investigate whether external evidence from other knowledge base could be helpful to entity type identification, and how to combine internal clues such as abstract, infobox and subject with external evidence. In particular, several learning features are extracted from entity abstract, infobox and subject, and then classifiers are trained to get entity type prediction models. Meanwhile, entity linking tools are utilized to link entities with external knowledge base e.g. DBpedia, where we can get type information. Finally, a voting mechanism is adopted to decide the final entity type. We have implemented our algorithms and present some experimental evaluation results to demonstrate the effectiveness of the approach.

The remainder of the paper is organized as follows. In the following section we review the existing literature on entity type identification. Then, we introduce the proposed approach in section 3. We conduct comparative experiments and present the results in section 4. At last, we conclude the paper with a summary of our work and give our future working directions.

2 Related Work

In the field of entity type inference, there are two dominant methods, namely, content-based (*Aldo Gangemi et al., 2012; Tianxing Wu et al., 2014*) and link-based methods (*Andrea Giovanni Nuzzolese et al., 2012; Heiko Paulheim and Christian Bizer, 2013*). Next we will introduce these methods respectively.

Content-based methods usually utilize entity descriptions such as abstract, infobox and properties to identify entity types. Several learning features are extracted from textual data and classification or clustering models are trained to predict entity types. For example, Aldo Gangemi et al., first extracted definitions from Wikipedia pages, used a natural language deep parser FRED to produce a logical RDF representation of definition sentences, and then select types and type-relations from the RDF graph based on graph patterns. Finally, a word sense disambiguation engine is used to identify the types of an entity and their taxonomical relations (*Aldo Gangemi et al., 2012*). Tianxing Wu et al., also mined type information from abstracts, infobox and categories of article pages in Chinese encyclopedia Web sites. They presented an attribute propagation algorithm to generate attributes for categories and a graph-based random walk method to infer instance types from categories of entities (*Tianxing Wu et al., 2014*). Man Zhu et al., transformed type assertion detection into multiclass classification of pairs of type assertions, and adopted Adaboost as the meta classifier with C4.5 as the base classifier (*Man Zhu et. al., 2014*). Kenza Kellou-Menouer and Zoubida Kedad utilized a density-based clustering algorithm to discovery types in RDF datasets. They first adopted Jaccard similarity to measure the closeness between two entities. In particular, they calculated the similarity between two given entities by considering their respective sets of both incoming and outgoing properties. Then entities are grouped according to their similarity (*Kenza Kellou-Menouer and Zoubida Kedad, 2015*).

Link-based methods can also be used in entity type assignment. For example, Heiko Paulheim and Christian Bizer proposed a heuristic link-based type inference mechanism. They used each link from and to an instance as an indicator for the resource's type. For each link, they use the statistical distribution of types in the subject and object position of the property for predicting the instance's types (*Heiko Paulheim and Christian Bizer, 2013*). Andrea Giovanni Nuzzolese et al., utilized k-Nearest Neighbor algorithm for classifying DBpedia entities based on the wikilinks (*Andrea Giovanni Nuzzolese et al., 2012*).

In this paper, we integrate content-based methods and external evidence to identify entity type. We views type identification as classification issue, and adopt classifiers to train type prediction models. Meanwhile, entity linking tools are adopted to link entities with external knowledge base, where we can get type information. Finally, we use a weighted voting approach to obtain the entity type.

3 The Approach

In this section, we will introduce the architecture of the system as shown in figure 1. The inputs of the system are entity data as illustrated in figure 2, the outputs are entity types. In particular the system consists of two parallel parts: (1) classification module; (2) entity linking module;

In classification module, we first extract entity definition from its abstract. And then, we extract several learning features from its definition, infobox, and subject. We choose several classification models to train the entity type prediction model.

In entity linking module, we first construct profile for each entity, and then entity linking tool (Qingliang Miao *et al.*, 2015) is used as a bridge to get entity type information from other linked data i.e. DBpedia. Finally, a voting mechanism is used to get the final answer. In particular, if these two models' results are different, we use entity linking based results as the final answer.

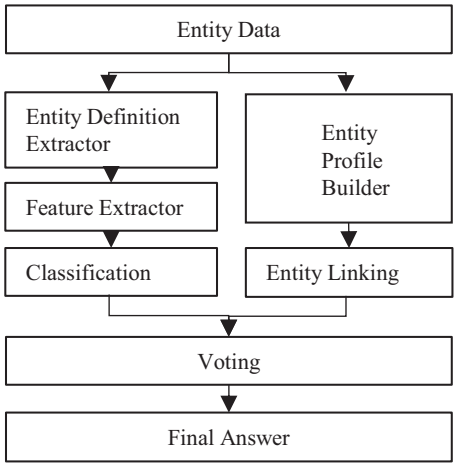

Figure 1: The workflow of the approach.

3.1 Classification based Model

In this section, we mainly introduce learning features and feature selection method.

In Linked Data, entities are usually descripted using Resource Description Framework (RDF)[4]. Each entity in Linked Data space is identified by a unique HTTP dereferenceable Uniform Resource Identifier (URI) and the relations of resources are described with simple subject-predicate-object triples. Figure 2 shows an example of entity "首尔/Seoul". The task of this research is to identify the type of the entity using existing information as illustrated in figure 2.

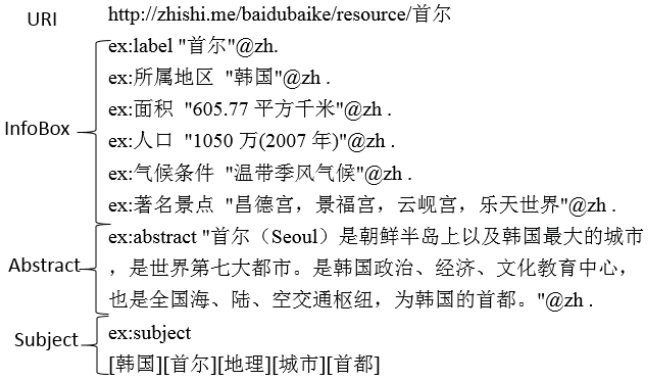

Figure 2: Linked Data example of entity "首尔/Seoul"

Pattern feature

Typically, the definition of an entity is in the first *k* sentences of its abstract. Inspired by (Aldo Gangemi *et al.*, 2012), we use a set of heuristics based on lexico-syntactic patterns to extract entity definition. The pattern features are derived from entity definition text in the form of "[entity] is/belongs [t_i] [$word_1 \ldots word_n$]", where t_i is the type keyword of entity type *i* and *n* is the distance between the key word t_i and the sentence's end. Table 1 shows some examples of the patterns.

Entity type	Patterns
Insect	<是.+虫>,<is.+insect>, <属于.+纲><belongs to.+species>
University	<是.+大学>,<是.+高校> <is.+university/college>
Game	<是.+游戏>,<is.+game>
City	<是.+城市>,<一座.+城市> <is.+city>
Scene	<是.+景点>,<是.+胜地>, <is.+attraction/scenic>

Table 1: Example of pattern features

Table 2 shows top 5 type keywords of each entity type. The type keyword set is obtained from encyclopedia and Chinese corpus and we will detail the process in next section. The type keywords are selected from keyword set manually.

The feature vector based on pattern is Q_i, where *N* is the number of entity type. If the first *k*

[4] http://www.w3.org/RDF/

sentences x in abstract contain the patterns in Table 1, we set the value δ, otherwise the value is 0. In our experiment, we set δ =1.0 empirically. For example, the definition of "首尔/Seoul" we extracted from abstracts is "首尔（Seoul）是朝鲜半岛上以及韩国最大的城市". And the feature value for type "city" is δ and 0 for the other types.

$$Q_i = \begin{cases} \delta \ , if \ \ f(x,t_i) = 1; \\ 0, \ if \ \ f(x,t_i) = 0; \end{cases} \ i \in \{1,2,...N\}$$

Keyword feature

Besides pattern features described above, we use keywords features as well. To ensure high coverage and quality of keywords for each type, we use rule base method and statistic based method to mine related keywords. For rule based method, we first collect entity description page with type information from three Chinese encyclopedia. Through analyzing description page, we extract 4 types of contents to construct keyword set, "*Title*", "*Alias*", "*Category*", and "*Related Entity*".

- Title: The titles in Chinese encyclopedia are used as labels for the corresponding entities directly.
- Alias: The alias in Chinese encyclopedia is used to represent the same entity. For example, [北京|北平|京师].
- Category: Categories describe the subjects of a given entity.
- Related Entities: In Chinese encyclopedia there are related entities of a given entity. For example, related entities of "大学(university)" are "北京大学(Peking university)", "清华大学 (Tsinghua University)

For statistic based method, we use word2vec model to obtain word vectors based on Chinese corpus and obtain similar word lists for each entity type. The final keyword list is obtained by a voting method. Table 2 shows the top 5 keywords for each type.

Entity type	Keywords
Insect	{昆虫,虫,物种,天敌,害虫}/{insect, species, predators, pets }
University	{大学,高校,校园,学院,分校}/{university, college, campus, branch}
Game	{游戏,电脑游戏,电子游戏,网络游戏,在线游戏}/{games, computer games, web game, online games }
City	{首都,大都市,城市,省会,城区}/{capital, metropolis, cities, provincial capital, urban}
Scene	{景点,名胜,旅游,景区,风景}/{attractions, scenic, tourism, scenic, scenery}
Politician	{政治家,政界,活动家,外交家,政客}/{ politician, activists, diplomats }
Song	{歌曲,歌词,演唱,歌名,曲目}/{song, lyrics, singing, song title, track }
Novel	{小说,短篇小说,科幻小说,武侠小说,传记}/{novel, short story, science fiction, martial arts novel, biography }
Cartoon	{动画,漫画,动画片,动画制作,电视}/{attractions, scenic, tourism, scenic, scenery}
Actor	{演员,导演,编剧,主演,剧情}/{actor, director, screenwriter, starring, drama}

Table 2: Top 5 keywords for each type

Infobox features

Since different entity types have different properties. For example, person has birthday and organization has locations. We extract property names from infobox and use them as infobox features. For example, in figure 1, property features of entity "首尔/Seoul" is "所属地区/region", "面积/area", "人口/population", "气候条件/climatic condition", "著名景点/famous scenery".

Subject features

Besides infobox features, we collect entity subject information from zhishi.me. Subject information contains many domain-specific terms, which are indicator of entity types. Table 3 shows some example of subject features. In this study, all these above features are binary features.

Entity	Subject features
颐和园/Summer Palace	{公园,景点,旅游}/{park, attraction, tourism}
静冈市/Shizuoka City	{日本,城市}/{japan, city}
面包超人/Anpanman	{动画片,萌}/{cartoon, cute}

Table 3: Example of subject features

Feature selection

The learning features are all extracted empirically, therefore, effective feature selection is necessary. We design a feature selection scheme as below: we take 'maximum probability of a feature representing a category' as the indicator of the effectiveness of features, and remove features whose effectiveness is smaller than a threshold T. In our experiment, we set T=0.85 empirically based on the development set. The changing curve of F-measure and threshold T is shown in figure 3.

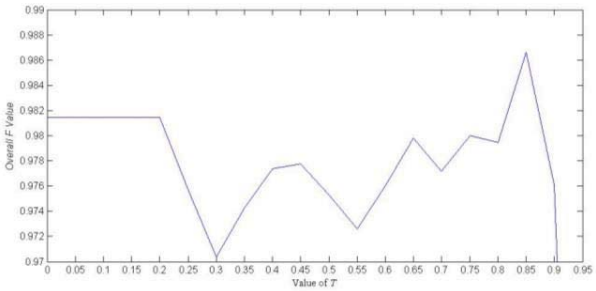

Figure 3: F-measure changes with threshold T.

3.2 Entity linking based Model

To use type information of DBpedia, we use entity linking tool to link entities with Chinese DBpedia. Since entities in Chinese DBpedia lack of "*rdf:type*" property, we use following steps to get type information.

Using "sameAs" relation

Since many entities in English DBpedia have "*rdf:type*" property, we can use "owl:sameAs" relation to obtain type information of Chinese DBpedia entities. For example, <*zhishi.me:伊斯兰堡*> is linked with <*zh.dbpedia:伊斯兰堡*> that is same as English DBpedia entity: <*en.dbpedia:Islamabad*>, and the type of <*en.dbpedia:Islamabad*> is <*dbo:City*>. Therefore, the type of <*zhishi.me:伊斯兰堡*> is city. Figure 4 illustrates the process.

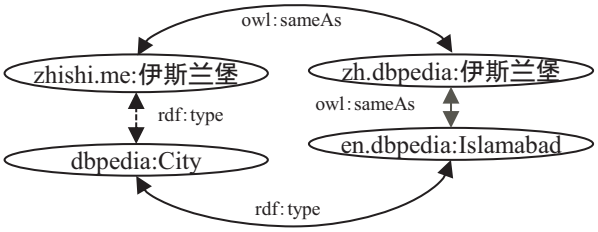

Figure 4: Example of "sameAs" relation.

Using redirect relation

In some cases, we can use redirect relation to obtain the type. Figure 5 shows an example. <*zhishi.me:青岛*> is same as <*zh.dbpedia:青岛*> and <*zh.dbpedia:青岛*>is redirected from <*zh.dbpedia:青岛市*>, and <*zh.dbpedia:青岛市*> is same as <*en.dbpedia:Qingdao*> whose type is city.

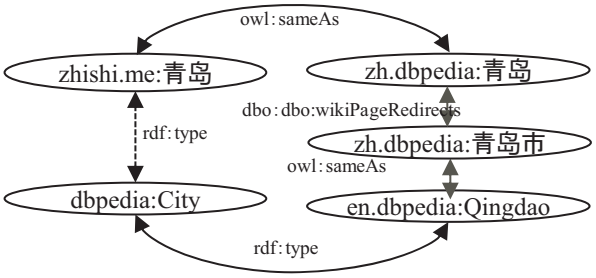

Figure 5. Example of "redirect" relation.

Using category information

Besides "sameAs" and "redirection" relation, we use entity category information to infer type information as well. Category information in DBPedia is usually a strong indicator for entity type. For example, person usually has category information "People_from_Beijing" or "People_born_1960s". Therefore, we can infer an entity's type from category. In particular, we use a simple method that match category information e.g. "People" with DBpedia ontology class.

Type mapping

Since the DBPedia Ontology (dbo) is different from type information in Zhishi.me, we have to map dbo with entity type in Zhishi.me. In particular, given a dbo type, we use a type mapping table shown in table 4 to find the corresponding type in Zhishi.me. We use entity linking tools to link Zhishi.me training data with DBPedia, and obtain the type mapping relation.

For example, if entity e_1 in Zhishi.me with type "Politician" is linked with e_2 in DBPedia with type "Governor", we can obtain a mapping relation between "Politician" and "Governor".

Type in Zhishi.me	Type in DBPedia
Insect	dbo:Insect
University	dbo:University
Game	dbo:ViedoGame
Politician	dbo:Politician;dbo:OfficeHolder dbo:Governor;dbo:Ambassador dbo:Chancellor
City	dbo:City;dbo:Capital;dbo:Town dbo:Settlement
Song	dbo:Song
Novel	dbo:Novel
Scene	dbo:NaturalPlace;dbo:Mountain dbo:Canal;dbo:Park
Cartoon	dbo:Cartoon;dbo:Comic dbo:TelevisionShow;dbo:Film
Actor	dbo:Actor;dbo:Artist

Table 4: Type mapping table

4 Experiment

In order to evaluate the effectiveness of the proposed approach, we conduct our experiments by using test data from JIST15 type identification challenge[5]. The data includes 1397 entities with type information and 500 unlabeled entities that are used as test data. There are 10 classes including insect, university, game, politician, city, song, novel, scene, cartoon and actor. The statistics of the data is shown in Table 5.

Entity Type	# training data	# testing data
Insect	124	41
University	157	42
Game	143	59
Politician	134	43
City	139	59
Song	139	59
Novel	150	51
Scene	130	60
Cartoon	134	38
Actor	147	48

[5] ttp://www.jist2015.org/index.php?m=list&a=index&id=48&skip=50

Table 5: The statistics of the test data

Precision, Recall and F-measure are used as the evaluation metric. All of them are defined as follows where a_i is the number of URLs that are actually in label i and also predicted in label i, b_i is the number of URLs that are predicted in label i, c_i is the number of URLs that are actually in label i.

$$precision = \sum_{i=1}^{n} \frac{a_i}{b_i}$$

$$recall = \sum_{i=1}^{n} \frac{a_i}{c_i}$$

$$f-measure = \frac{2\,precision * recall}{precision + recall}$$

In experiment, we first evaluate the performance using internal information only, namely classification based method. And then we evaluate whether external knowledge is useful to improve type identification performance. We also compare with our method with state of the art method (Tianxing Wu. *et al.*, 2014).

In this experiment, we have compared with four classification algorithms, Naïve Bayes, Bayes Net, Random Forest and Support Vector Machine. Figure 6 shows experiment results, from which we can see *F-measure* is relative high in classification method, and Random Forest algorithm performs best among four classifiers and F-measure is above 0.98.This results indicate the learning features are very predictive for this task.

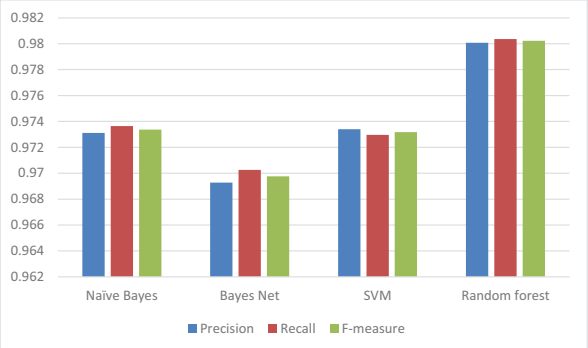

Figure 6: Experiment results on precision.

To evaluate whether external evidence derived from other knowledge base is helpful, we have built and compared two kinds of type identification methods, one with utilizing entity linking techniques and the other without.

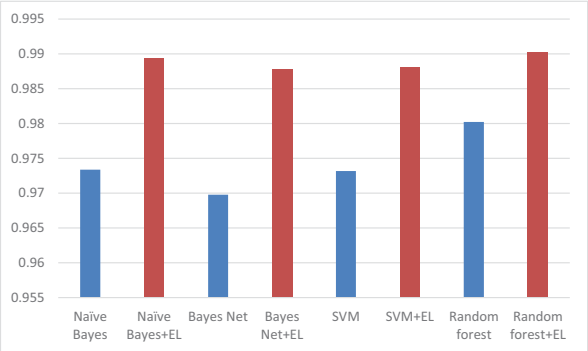

Figure 7: Experiment results of models with and without entity linking on f-measure.

Figure 7 shows the comparing results of type identification models with and without entity linking. From Figure 7 we can see that when incorporating entity linking results, the average *F-measure* can be improved by 1.5%. The improvement of *F-measure* is likely attributable to the external knowledge base. The improvement is not as much as expected. Through carefully analyze the results, we find two reasons. First, entity linking tools only link 40% entity in testing data. Second, most derived type from external knowledge base is consistent with classification results

In order to validate whether the improvement is significant, we adopt pair-wise t-tests on *F-measure*. For all t-tests, p-values are all less than 0.01, therefore the improvement is significant. We confirm that the improvement of *F-measure* is due to incorporating external evidence and we believe that it will achieve better results if we incorporate enough and high quality external evidence.

From the above analysis, it is evident that entity linking results can be incorporated as knowledge to improve the performance of entity type identification.

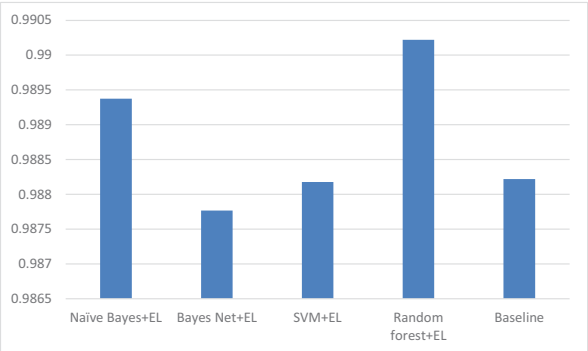

Figure 8: Experiment results of models with entity linking and baseline on f-measure

We also use state of the art method (*Tianxing Wu et al., 2014*) as baseline and conduct experiment to compare our method with the baseline. Figure 8 shows the experiment results. From figure 8 we can see our best performance (Random forest with entity linking) outperform state of the art method by 1.1%.

5 Conclusion

In this paper, we study entity type information identification from Chinese linked data and present a novel approach by integrating classification and entity linking techniques. In particular, entity type information is inferred from internal clues using classifiers. Moreover, external evidence is obtained from other knowledge bases through entity linking techniques. Experimental results on real-world datasets show the learning features we selected are predictive. Moreover, results indicate external evidence derived by entity linking techniques is helpful to type identification as well. We believe that this study is just the first step in type identification and much more work needs to be done to further explore the issue. In our ongoing work, we plan to improve entity tools to find more equivalent entities in external knowledge base. We also plan to reduce the amount of training data, which is time consuming to obtain, by using entity linking results. For example, type information obtained by entity linking techniques could be used as training data directly. Another direction is to harvest external evidence from broader resources, e.g. text or web tables, not just from linked data or knowledge base. For instance, in sentence "...including cities such as Birmingham, Montgomery, Huntsville...", if we know the type information of "Birmingham", we can infer other entities' type as well. Similarly, if we know the type of an entity, the other entity types in the same column can also be obtained by reasoning. At last, we plan to study fine grained type identification.

References

Aldo Gangemi, Andrea Giovanni Nuzzolese, Valentina Presutti, Francesco Draicchio, Alberto Musetti, and Paolo Ciancarini. Automatic typing of DBpedia entities, In Proceedings of the 11th International Semantic Web Conference, 2012, pp. 65-81.

Andrea Giovanni Nuzzolese, Aldo Gangemi, Valentina Presutti and Paolo Ciancarini, Type inference through the analysis of Wikipedia links, In Proceedings of the LDOW2012, 2012.

Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia-a Crystallization Point for the Web of Data. Web Semantics: Science, Services and Agents on the World Wide Web 7(3), 2009, pp. 154-165.

Heiko Paulheim and Christian Bizer, Type Inference on Noisy RDF Data, In Proceedings of the 12th International Semantic Web Conference, 2013, pp. 510-525.

Hepp, M.: GoodRelations: An ontology for describing products and services offers on the web. In: EKAW 2008, Vol. 5268, 2008, pp. 329-346.

Kalyanpur, A., Murdock, J.W., Fan, J., Welty, C.: Leveraging Community-built Knowledge for Type Coercion in Question Answering. In Proceedings of the 10th International Semantic Web Conference, pp. 144-156.

Kenza Kellou-Menouer and Zoubida Kedad, Discovering Types in RDF Datasets, In Proceedings of the 12th Extended Semantic Web Conference, 2015, pp. 77-81.

Lee, T., Chun, J., Shim, J., Lee, S. G.: An Ontology-based Product Recommender System for B2B Marketplaces. International Journal of Electronic Commerce 11(2), 2006, pp. 125-155.

Man Zhu, Zhiqiang Gao, and Zhibin Quan, Noisy Type Assertion Detection in Semantic Datasets, In Proceedings of the 13th International Semantic Web Conference, 2014, pp. 373-388.

Niu, X., Sun, X., Wang, H., Rong, S., Qi, G., Yu, Y.: Zhishi.me - Weaving Chinese Linking Open Data. In Proceedings of the 10th International Semantic Web Conference, pp. 205-220.

Qingliang Miao, Yao Meng, Lu Fang, Fumihito Nishino and Nobuyuki Igata, Link Scientific Publications using Linked Data. In Proceedings of the 9th IEEE International Conference on Semantic Computing, 2015.

Suchanek, F.M., Kasneci, G., Weikum, G.: Yago: a Core of Semantic Knowledge. In Proceedings of the 16th International Conference on World Wide Web, 2007 pp. 697-706.

Tianxing Wu, Shaowei Ling, Guilin Qi, and Haofen Wang, Mining Type Information from Chinese Online Encyclopedias, In Proceedings of the 4th Joint International Conference, 2014, pp 213-229.

Tonon, A., Catasta, M., Demartini, G., Cudr´e-Mauroux, P., Aberer, K.: TRank: Ranking Entity Types Using the Web of Data. In Proceedings of the 12th International Semantic Web Conference, pp. 640-656

Welty, C., Murdock, J.W., Kalyanpur, A., Fan, J.: A Comparison of Hard Filters and Soft Evidence for Answer Typing in Watson. In Proceedings of the 11th International Semantic Web Conference, pp. 243-256.