Automatic Mapping of Wikipedia Templates for Fast Deployment of Localised DBpedia datasets

Alessio Palmero Aprosio
Università degli Studi di Milano
Via Comelico, 39/41
20135 Milano, Italy
aprosio@fbk.eu

Claudio Giuliano
Fondazione Bruno Kessler
Via Sommarive, 18
38123 Trento, Italy
giuliano@fbk.eu

Alberto Lavelli
Fondazione Bruno Kessler
Via Sommarive, 18
38123 Trento, Italy
lavelli@fbk.eu

ABSTRACT

DBpedia is a Semantic Web resource that aims at representing Wikipedia in RDF triples. Due to the large and growing number of resources linked to it, DBpedia has become central for the Semantic Web community. The English version currently covers around 1.7M Wikipedia pages. However, the English Wikipedia contains almost 4M pages. This means that there is a substantial problem of coverage (even bigger in other languages). The coverage slowly increases thanks to the manual effort made by various local communities. This effort is aimed at manually mapping Wikipedia templates into DBpedia ontology classes and then run the open-source software provided by the DBpedia community to extract the triples. In this paper, we present an approach to automatically map templates and we release the resulting resource in 25 languages. We describe the used algorithm, starting from the existing mappings on other languages and extending them using the cross-lingual information available in Wikipedia. We evaluate our system on the mappings of a set of languages already included in DBpedia (but not used during the training phase), demonstrating that our approach can replicate the human mappings with high precision and recall, and producing an additional set of mappings not included in the original DBpedia.

Keywords

Ontology population, Semantic Web, Linked Open Data, DBpedia

1. INTRODUCTION

The demand of structured data collected from the Web is constantly growing and several knowledge bases (KBs) have been released to fit this need. Most of them have been populated using Wikipedia as primary data source. The collaborative encyclopedia represents a practical choice: it is freely available, big enough to cover a large part of human knowledge, and populated by about 100,000 active contributors. Therefore the information it contains represents a good approximation of what people need and wish to know. Some relevant examples of such KBs include FreeBase,[1] DBpedia,[2] and Yago,[3] created using different techniques based on crowd sourcing, machine learning algorithm or rule-base approaches. DBpedia, in particular, plays a central role in the development of the Semantic Web, as it is indeed one of the central interlinking hubs of the emerging Linked Open Data, and a large and growing number of resources are linked to it. Therefore we concentrate our efforts on its automatic expansion.

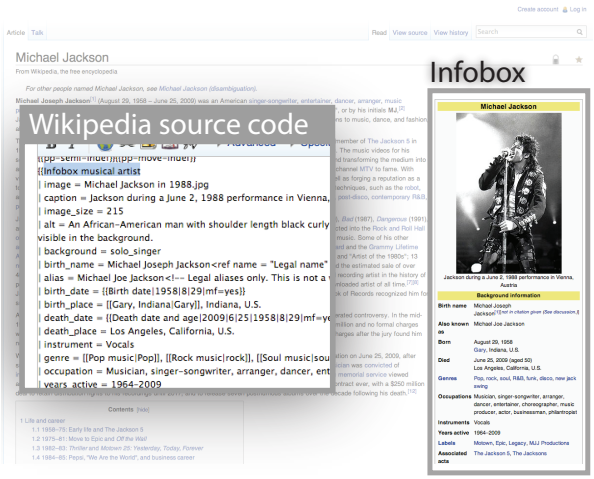

Figure 1: Example of Wikipedia infobox

The DBpedia community develops and maintains a KB available for download in RDF format. The KB is populated using a semi-automatic rule-based approach relying on Wikipedia *infoboxes*, a set of *subject-attribute-value* triples that represents a summary of some unifying aspect that the Wikipedia articles share. For instance, there is a specific infobox (`Persondata` in the English Wikipedia, and `Bio` in the Italian one) used for biographies, containing information such as *name*, *date of birth*, *nationality*, *activity*, etc. Figure 1 shows an example of infobox in Michael Jackson's Wikipedia article. The DBpedia community also releases an ontology, available for download in OWL format, or browseable on

[1] `http://www.freebase.com/`
[2] `http://dbpedia.org/About`
[3] `http://www.mpi-inf.mpg.de/yago-naga/yago/`

DBpedia website.[4] The ontology has been manually created by the community and it is based on the most commonly used infoboxes within Wikipedia. Version 3.8 of the ontology covers a total of 359 classes, which form a subsumption hierarchy. It also includes more than 1,700 different properties. Crowd sourcing is used to map infoboxes and infobox attributes to the classes and properties of the DBpedia ontology, respectively. In particular, these mappings are collected and stored in an online repository.[5] Similarly, the ontology itself is modifiable by the users to be improved depending on the infoboxes that need to be mapped. A software tool is made available to extract the structured information contained in the infoboxes and to convert it to triples. By running this tool, every page in Wikipedia that contains an infobox mapped to a specific class is automatically added to such class. As the number of required mappings to cover all the infoboxes is extremely large, the mapping process follows an approach based on the frequency of the infoboxes and infobox attributes, most frequent items are mapped first. This guarantees a good coverage, as infoboxes are distributed according the Zipf's law [15]. Therefore, although the number of mappings may be small, a large number of articles can be added to the ontology. The resulting KB is made available as Linked Data,[6] and via DBpedia's main SPARQL endpoint.[7]

At the time of starting the experiments, the last available version of DBpedia, 3.8, covers around 4M entities in its English chapter, the same number of the articles included in the English Wikipedia. However, this apparently good result is due to the fact that, when a user-provided mapping is not available, each article in Wikipedia is by default mapped to the top-level class `owl:Thing`. In fact, only 1.7M pages are mapped to classes different from `owl:Thing`. In this paper, when we speak about coverage we will refer to these pages only.

At the early stages of the project, the construction of DBpedia was solely based on the English Wikipedia. Until 2011, the DBpedia dataset included data from non-English Wikipedia pages only if there existed an equivalent English page. However, since there are many pages in the non-English Wikipedia editions that do not have an equivalent English page, relying on English Wikipedia pages only had the negative effect that DBpedia did not contain data for these entities. More recently, other contributors around the world have joined the project to create localized and interconnected versions of the resource. The goal is to populate the same ontology used by the English project, taking articles from editions of Wikipedia in different languages. The DBpedia 3.7 release addressed this problem and provided 15 new localised editions of the dataset. These new localised versions of DBpedia required the same effort already used for the English one: infoboxes and properties in the localised Wikipedia are manually mapped to the corresponding classes in the ontology. Therefore, several research groups around the world started to join the project and take charge of this manual task. Furthermore, as the mappings have to follow syntactic guidelines due to the requested DBpedia language, each mapping can employ a community user for some minutes. If a corresponding class in the ontology does not exist yet, it can take longer. As each language may have hundreds of infoboxes, this effort can be heavy for the community.

At the time of writing, there are 16 different localized versions of DBpedia. The inclusion of more languages has widened the problem of coverage. As each edition of Wikipedia is managed by different groups of volunteers with different guidelines, the DBpedia leading idea of semi-automatically populating the ontology by mapping infoboxes to classes does not work properly in some cases. For example, the Italian chapter has only 50 mappings, but it covers more than 600K pages (out of around 1M articles in the corresponding Wikipedia), as some infoboxes cover highly populated classes, such as `Person` and `Place`. The French and Spanish chapters, differently, contain only 15K pages each, with 70 and 100 mappings, respectively. This shows the different policies adopted by the different Wikipedia communities have a strong impact on the coverage that have to be taken into account.

In this paper, we describe a resource obtained by automatically mapping Wikipedia infoboxes in 25 languages to the corresponding classes in the DBpedia ontology. These mappings can be used for the deployment of new chapters of DBpedia. To achieve this goal, we devised a three-step approach that exploits Wikipedia cross-language links in six pivot languages (English, Italian, German, Portuguese, Spanish, French) and uses the existing DBpedia mappings for these languages. The method is summarized as follows:

- First, the cross-language links are used to add Wikipedia articles not present in the DBpedia for one language but present in others. Through this first step, we increased the DBpedia coverage on Wikipedia articles by around 60% on the six languages considered in our experiments (Section 2). The cross-language link analysis in [6] shows that relative error of cross-lingual links in Wikipedia is less than 1%.
- Second, we extracted the list of templates from Wikipedia and classified them into two macro categories, used for recurring patterns, and infoboxes (Section 3).
- Third, we used a rule-based approach to map Wikipedia infoboxes, taken from versions of Wikipedia in different languages, to the most probable class in the DBpedia ontology (Section 4).

Evaluation has been performed on five languages (Bulgarian, Czech, Indonesian, Dutch and Catalan), already available in the DBpedia project: manually annotated Wikipedia infoboxes are used as test data for evaluation (Section 5). We show that our approach further increases the number of mappings with high accuracy and can be tuned to vary the tradeoff between precision and recall. This simple algorithm has never been used in real applications, while the results show that the approach is reliable and can save time in the mapping task.

[4] `http://mappings.dbpedia.org/server/ontology/classes/`
[5] The repository is publicly available at `http://mappings.dbpedia.org`
[6] `http://wiki.dbpedia.org/Downloads`
[7] `http://dbpedia.org/sparql`

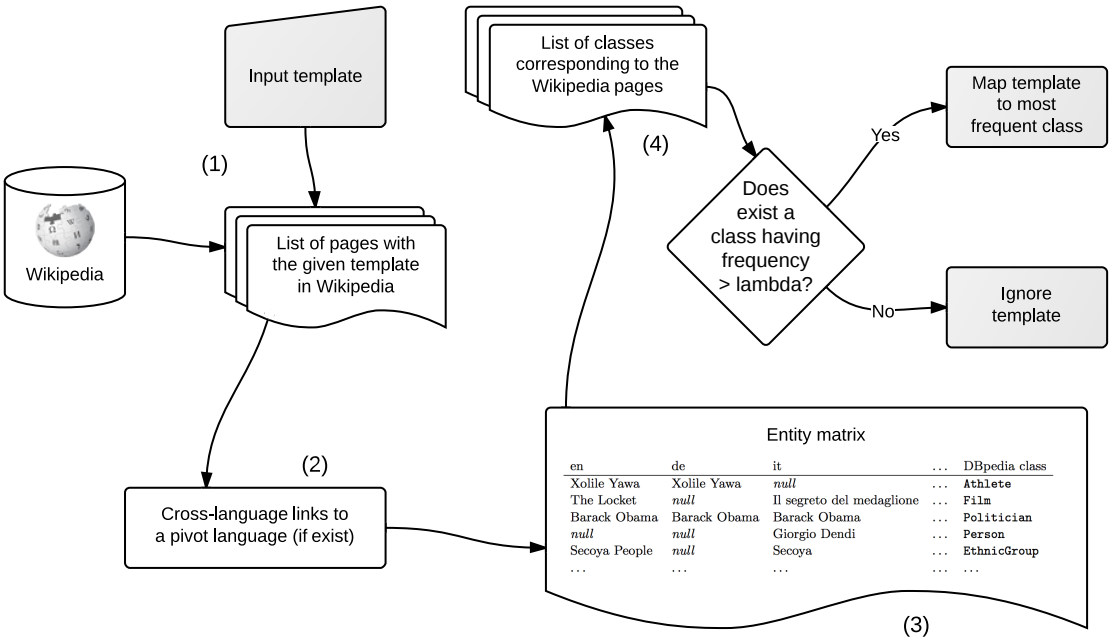

Figure 2: Workflow of the system.

2. WIKIPEDIA ENTITY REPRESENTATION

As said in Section 1, the English and Italian Wikipedia have an infobox for biographies (`PersonData` and `Bio`, respectively), while Spanish and French do not. DBpedia stores the cross-language information, but it is not used to map the infoboxes. For example, Clint Eastwood is classified as `Actor` in the French DBpedia and as `Person` in the Italian one. We deal with this problem, trying to classify pages in all languages to the most specific class.

This process exploits existing handcrafted mappings in six languages (English, Italian, German, Portuguese, Spanish, French).

Let $\mathcal{L}$ be the set of languages available in Wikipedia, we first build a matrix E where the i-th row represents an entity e_i and j-th column refers to the corresponding language $l_j \in \mathcal{L}$. The cross-language links are used to automatically align on the same row all Wikipedia articles that describe the same entity. The element $E_{i,j}$ of this matrix is *null* if a Wikipedia article describing the entity e_i does not exist in l_j. An entity in our system is therefore represented as a row of the matrix, where each j-th element is a Wikipedia article in language l_j. In our experiments, the entity matrix E is built starting from the six pivot languages listed above. Figure 3 shows a portion of the entity matrix.

First, we need to assign a single class in the DBpedia ontology to each entity of the matrix. As said, we will use the classes already annotated by the DBpedia community. Using the DBpedia annotation tool, annotators can assign a unique class to each infobox. However, this is not necessarily true if we consider more than one language. For example, the British Library is classified as a `Library` (subset of `Building`) in the English DBpedia, and as an `EducationalInstitution` (subset of `Organisation`) in the German DBpedia. We deal with these cases filtering the classes from different DBpedias as follows.

- If the entity belongs to more than one ontology class and these classes have one or more ancestor class in common, then the most specific common class is used. For example, *Barack Obama* is `OfficeHolder` in the French DBpedia and `President` in the Spanish one. These two classes are both subclass of `Person`, so we only consider this class.
- If the more-than-one-class problem involves classes connected in a chain of subclass of relations, we consider the most specific class. For instance, the famous singer *Michael Jackson* is classified as a `Person` in the Italian and German DBpedia, an `Artist` in the English DBpedia and a `MusicalArtist` in the Spanish DBpedia. The most specific class is the last one, so the entity *Michael Jackson* is considered as a `MusicalArtist`.
- Finally, when an entity is classified using two classes not having a common ancestor, that entity is left as

en	de	it	...	DBpedia class
Xolile Yawa	Xolile Yawa	*null*	...	`Athlete`
The Locket	*null*	Il segreto del medaglione	...	`Film`
Barack Obama	Barack Obama	Barack Obama	...	`Politician`
null	*null*	Giorgio Dendi	...	`Person`
Secoya People	*null*	Secoya	...	`EthnicGroup`
...	...	...	...	...

Figure 3: A portion of the entity matrix

	EN	DE	IT	PT	FR	ES
Wikipedia templates	391,780	49,744	104,044	43,385	148,200	24,946
Wikipedia infoboxes	21,250	1,525	2,238	2,469	2,587	552
Wikipedia article pages	3,932,148	1,325,792	924,544	740,585	1,251,585	953,848
DBpedia mapped pages	1,716,555	205,903	607,842	226,497	15,463	15,987
DBpedia mapped pages after CL	1,902,585	482,747	652,395	430,603	518,874	419,168

Table 1: Statistics taken from different chapters of Wikipedia and DBpedia: number of templates in Wikipedia, number of filtered templates in Wikipedia, number of pages in Wikipedia, in DBpedia, and in DBpedia after using Wikipedia cross-language links.

`Unknown`. Articles about battle tanks are examples of this kind, as different DBpedias classify them both as `Weapon` and `MeanOfTransportation`.

Table 1 shows statistics for each language, about this enriched DBpedia.

3. EXTRACTING TEMPLATES

A template is a special Wikipedia page created to be included in other pages. Templates usually contain patterns that might need to show up on any number of articles. They are commonly used for boilerplate messages, standard warnings or notices, infoboxes, navigational boxes and similar purposes. We can divide them into two broad categories:

Infoboxes are fixed-format tables designed to be added to the top right-hand corner of articles to consistently present a summary of some unifying aspect that the articles share. The DBpedia project uses this category of templates: most of them are manually linked to a particular class in the DBpedia ontology.

Macro templates are used to give graphic coherence to the same element in different pages. For instance, templates of this class are used for country flags, dates, portals, and so on. This class is not useful for our purpose, therefore we will ignore them.

Wikipedia does not provide a simple way to assign templates to the correct category. Thus we implement a simple rule-based hand-crafted classifier based on the following heuristics:

1. If a template is an infobox, it is included only once in the page, so for each template we count the total number of occurrences and the total number of pages in which it appears. The ratio between these two values must be 1. There are some rare cases (like the `Bio` template in Italian) in which an infobox can be included twice, so we relaxed this constraint and considered 1.5 as a good ratio.

2. Templates can be represented using their parameters, that can be a single value or `key/value` pairs. Infoboxes use only the latter format, so we removed the others.

3. Finally, infoboxes are usually written one key/value pair per line, for readability purpose; we only keep templates in which the difference between the number of lines and the number of pairs is greater than or equal to zero.

In this way, we remove more than 90% of templates, obtaining few infoboxes for each page (on average 1.27 and 1.18 templates per page in English and Italian, respectively). Statistics about the extraction are shown in Table 1.

To assess the correctness of our approach, we compared the list of extracted infoboxes and the list of the mapped templates in DBpedia: the first set strictly contains the second one, therefore we can estimate that our heuristic rules are a good approximation of the required set. A similar approach for template filtering has been also used in [1].

4. TEMPLATE MAPPING

Given a Wikipedia template classified as infobox in the previous step (Section 3), our goal is to map it, when possible, to a DBpedia ontology class. To this aim, we use the matrix built in Section 2 as information source to find the mapping. The approach is instance based: we exploit the Wikipedia pages (instances) already mapped with a DBpedia class and their cross language links to the pages that contain the template to be mapped. A simple method based on the frequencies of the resulting classes allows us to tune the tradeoff between precision and recall. The mapping algorithm (Figure 2) is implemented as follows.

1. Given as input an infobox t taken from a version of Wikipedia in a specific language l (where l is not a pivot language), we collect all the pages P_l that include t. See Figure 4, parts (A) and (B).

2. We use the cross-language links contained in P_l to retrieve the pages P_E in the matrix E, for which we know the DBpedia classes (Section 2). See Figure 4, part (C).

3. From the matrix E, we collect the DBpedia classes C of the pages P_E and count their number of occurrences f_c ($c \in C$). See Figure 4, part (D). We also collect and count the occurrences of the parent classes in C. For example, if the classes `Person` and `Organisation` occur 5 and 3 times, respectively, we also consider the class `Agent` with frequency 8, as the latter class is the common ancestor of the first two.

4. t is then mapped to most specific and most frequent class (max f_c). See Figure 4, part (E). A parameter λ ($0 \leqslant \lambda \leqslant 1$) is used to filter c such that $f_c \leqslant \lambda$.

   If the percentage of pages mapped to c exceeds λ, then we map the infobox t to c, otherwise we climb one level up and recalculate the percentage until it exceeds λ. If we reach the root of the ontology taxonomy without any class percentage exceeding λ, then the system abstains, the infobox is discarded and no class is mapped to it. The parameter λ can be used to tune the tradeoff between precision and recall: the higher λ, the higher precision and the lower recall; the lower λ, the higher recall and the lower precision. See Section 5 for further details.

Consider, for example, the Polish template `Wojna_infobox` ("War" in English). It is included in 3,770 pages in the Polish Wikipedia. By using cross-language links, we found that 2,842 of them are present in one of the six pivot languages and 2,716 are classified in one of the corresponding DBpedia.

Table 2(a) shows the classes mapped from the pages in this set of entities. We can see that 2,697 pages are classified as `MilitaryConflict`. Since 2,697 out of a total of 2,716 classified pages corresponds to 99%, we can assume that this is the class that best approximate the possible mapping of the `Wojna_infobox` template. In particular, in this case the Polish word "Wojna" means "War", clearly a synonym of `MilitaryConflict`.

Let us consider another example, involving a more ambiguous template, `Park_infobox` ("Park" in English). Although its translation does not give rise to ambiguity, the cross-language links bring to the situation shown in Table 2(b). In this case, the `Park` class surely has the majority, but its percentage is low (46%), therefore using a parameter $\lambda = 0.5$ this solution is discarded and `Place` is given instead.

5. EXPERIMENTS AND EVALUATION

Experiments have been carried out on 5 languages (Bulgarian, Czech, Indonesian, Dutch, and Catalan) for which manually mapped infoboxes can be downloaded from the DBpedia official mapping website.[8] Specifically, we used the version made available on April 5, 2013.

Precision and recall values are calculated using these sets of mappings as gold standard.

Figure 5 shows the precision/recall curves, the grey dashed lines join points with the same F_1, showing that F_1 values range from 0.8 and 0.9. The different precision/recall points are obtained by varying the parameter λ.

These curves confirm the differences between the various versions of Wikipedia: in some cases the precision is high (in Catalan we reach 100%), while in other languages it does not exceed 95%. Also, the precision/recall curves differ in shape and direction when our algorithm is applied to different languages. These differences reflect the different structure of infoboxes in the Wikipedia editions, as the policies on infoboxes change from language to language. [10]

The evaluation is performed as proposed by [7] for a similar hierarchical categorisation task. Figure 6 shows an example of the evaluation. The system tries to classify the infobox `Philosopher` and map it to the ontology class `Astronaut`, while the correct classification is `Philosopher`. The missing class (question mark) counts as a false negative, the wrong class (cross) counts as a false positive, and the correct classes (ticks) count as true positives.

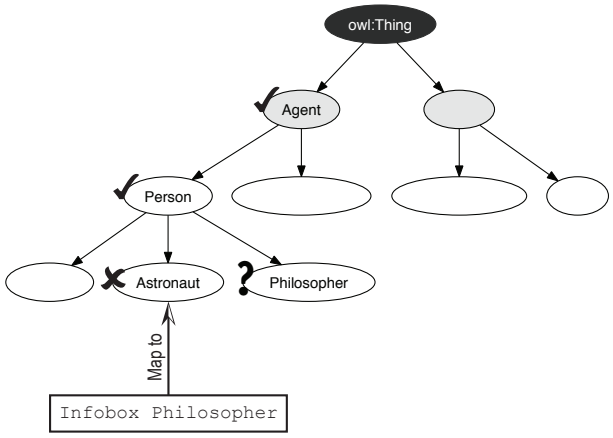

Figure 6: Description of the evaluation.

6. THE RESOURCE

We mapped templates to DBpedia classes for 25 languages.[9] For each language, we map only the templates that appear more than 50 times in the corresponding Wikipedia and released 3 different mappings corresponding to 3 distinct λ parameters, 0.1, 0.9, 0.5, corresponding to the maximum recall, precision, and F_1, respectively. These values are based on the evaluation performed on the five languages considered in Section 5.

Table 3 shows the number of mappings extracted for each language. The first 11 languages in the Table have already

[8] `http://mappings.dbpedia.org/`

[9] The complete resource is available at `http://www.airpedia.org/`.

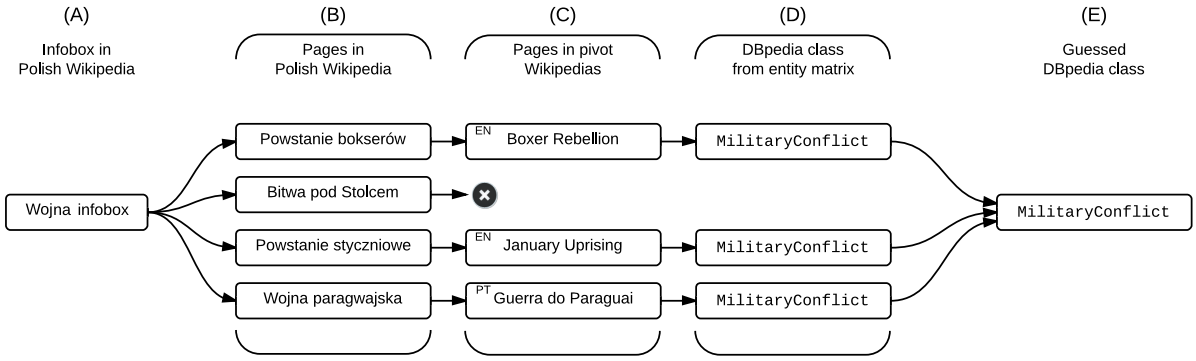

Figure 4: The algorithm applied to `Wojna infobox` in Polish.

`Wojna_infobox`	
Pages in Polish Wikipedia	3,770
Pages found using CLL	2,842
Pages classified in DBpedia	2,716
`MilitaryConflict` (2)	2,697
`Event` (1)	2,697
`Person` (2)	9
`Agent` (1)	10
`Place` (1)	8
`PopulatedPlace` (2)	4
...	...

(a)

`Park_infobox`	
Pages in Polish Wikipedia	321
Pages found using CLL	83
Pages classified in DBpedia	52
`Park` (3)	24
`ArchitecturalStructure` (2)	24
`Place` (1)	52
`PopulatedPlace` (2)	2
`ProtectedArea` (2)	2
`NaturalPlace` (2)	1
...	...

(b)

Table 2: The distribution of the pages having `Wojna_infobox` (a) and `Park_infobox` (b) on the six pivot languages (in brackets, the depth of the class in DBpedia ontology)

mappings in DBpedia (the second column shows the number of templates already mapped), while the remaining 14 languages do not have mappings. The last three columns show the number of extracted templates by our method, respectively, with $\lambda = 0.1$, $\lambda = 0.9$ and $\lambda = 0.5$. For example, we can quadruplicate the number of Catalan mappings with 100% precision. Notice that, even when the precision is not 100% and the process still needs human supervision, our approach can drastically reduce the time required, estimated in around 5 minutes per mapping if performed from scratch.[10]

7. RELATED WORK

The main reference for our work is the DBpedia project [2]. Started in 2007, it aims at building a large-scale knowledge base semi-automatically extracted from Wikipedia. The main problem here is that the Wikipedia infobox attribute names do not use the same vocabulary, and this results in multiple properties having the same meaning but different names and vice versa. In order to do the *mapping-based* extraction [3] organize the infobox templates into a hierarchy, thus creating the DBpedia ontology with infobox templates as classes. They manually construct a set of property and object extraction rules based on the infobox class. This classification is more consistent as compared to the one obtained by means of generic extraction, however it has smaller coverage. Nowadays, the ontology covers 359 classes which form a subsumption hierarchy and are described by 1,775 different properties. The English version is populated by around 1.7M Wikipedia pages, although the English Wikipedia contains almost 4M pages. Other languages suffer from an even lower coverage (see Table 1).

Differently, Yago [14], another similar project also started in 2007, aims at extracting and map entities from Wikipedia using categories (for fine-grained classes) and WordNet (for upper-level classes). Its coverage is higher, but it is monolingual and its ontology contains thousands of hundreds of classes: it may be difficult to use it in practical applications.

There are also other projects aiming to extract Wikipedia entity types boostrapping information contained in the categories. For example, [11] uses extracted datatypes to train a name entity recogniser, while [9] investigates Wikipedia categories and automatically cleans them.

[10]This is an average time evaluated during the mapping of the Italian DBpedia.

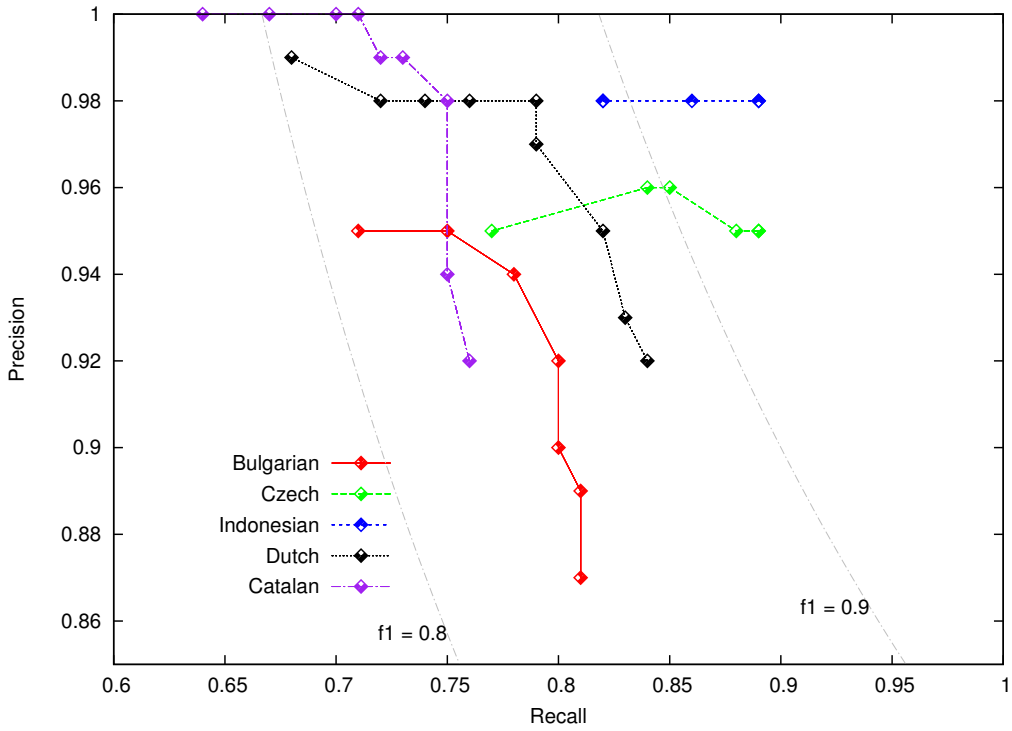

Figure 5: Precision/recall curves for five languages.

Language	DBpedia	$\lambda = 0.1$	$\lambda = 0.9$	$\lambda = 0.5$
Bulgarian	58	106	109	109
Catalan	49	191	198	197
Czech	66	131	137	135
Croatian	36	113	115	113
Hungarian	108	176	185	184
Indonesian	48	159	161	160
Dutch	99	362	372	371
Polish	340	216	228	226
Russian	30	340	354	352
Slovenian	160	88	91	91
Turkish	215	129	133	132
Belarusian	-	56	59	59
Danish	-	109	112	111
Estonian	-	51	52	51
Finnish	-	122	127	126
Icelandic	-	18	18	18
Lithuanian	-	112	113	113
Latvian	-	69	71	71
Norwegian	-	154	160	159
Romanian	-	134	138	136
Slovak	-	101	102	102
Albanian	-	27	28	28
Serbian	-	164	166	165
Swedish	-	175	182	182
Ukranian	-	236	247	242

Table 3: Infoboxes estracted and available as a resource.

The tool presented in [5], *Tipalo*, identifies the most appropriate class of a Wikipedia article by interpreting its page abstract using natural language processing tools and resources. In this context, only the English Wikipedia is considered, as this classifier cannot be easily adapted to other languages.

Similarly, [13] only considers the English DBpedia and therefore does not take advantage from inter-language links. In addition, there is some manual effort to classify biographies (using tokens from categories), that leads to very good results, but this approach is not automatically portable to other languages; again linguistic tools are used to extract the definition from the first sentence.

Finally, the Airpedia project [12] trains an instance-based supervised classifier using the entity matrix as training data. Features have been extracted from templates, categories and latent semantic analysis of the article text. The evaluation is performed on a manually annotated test set. The resulting resource is available through a SPARQL endpoint and a downloadable package on the Airpedia website.[11]

8. CONCLUSIONS AND FUTURE WORK

We have proposed a three-step approach that automatically maps templates in Wikipedia into DBpedia classes. We have first extended the population of DBpedia using cross-language links, then extracted the list of infoboxes from Wikipedia, and finally defined an approach that maps these infoboxes to the most probable class in the DBpedia ontology, starting from the existing mappings on six pivot languages. The experiments have been evaluated on the already present

[11] `http://www.airpedia.org`

mappings on five languages, showing high precision and recall. Tradeoff between precision and recall can be varied by means of a single parameter. The approach has been applied to 25 languages and the resulting resource is available at `http://www.airpedia.org/`. For 14 languages, we have created a set of mappings that can be used to start the corresponding chapters of DBpedia.

DBpedia also maps entity properties, such as `BirthDate` and `birthPlace` for `Person`, `director` for `Film`, and so on. We are currently working to deal with this problem, using natural language processing tools to find the correct relation in the article text. This can be seen as a relation extraction task, and one of the most reliable approaches to tackle this problem (starting from a large available knowledge base) is *distant supervision* [8]. This paradigm has been successfully used for pattern extraction [16] and question answering [4].

9. REFERENCES

[1] Maik Anderka and Benno Stein. A breakdown of quality flaws in wikipedia. In *Proceedings of the 2nd Joint WICOW/AIRWeb Workshop on Web Quality*, WebQuality '12, pages 11–18, New York, NY, USA, 2012. ACM.

[2] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. DBpedia: a nucleus for a web of open data. In *Proceedings of the 6th international Semantic Web Conference and 2nd Asian Semantic Web Conference*, ISWC'07/ASWC'07, pages 722–735, Berlin, Heidelberg, 2007. Springer-Verlag.

[3] Christian Bizer, Jens Lehmann, Georgi Kobilarov, Sören Auer, Christian Becker, Richard Cyganiak, and Sebastian Hellmann. DBpedia - a crystallization point for the web of data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(3):154–165, September 2009.

[4] Elena Cabrio, Julien Cojan, Alessio Palmero Aprosio, Bernardo Magnini, Alberto Lavelli, and Fabien Gandon. QAKiS: an open domain QA system based on relational patterns. In Birte Glimm and David Huynh, editors, *International Semantic Web Conference (Posters & Demos)*, volume 914 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2012.

[5] Aldo Gangemi, Andrea Giovanni Nuzzolese, Valentina Presutti, Francesco Draicchio, Alberto Musetti, and Paolo Ciancarini. Automatic typing of DBpedia entities. In Philippe Cudré-Mauroux, Jeff Heflin, Evren Sirin, Tania Tudorache, Jérôme Euzenat, Manfred Hauswirth, Josiane Xavier Parreira, Jim Hendler, Guus Schreiber, Abraham Bernstein, and Eva Blomqvist, editors, *International Semantic Web Conference (1)*, volume 7649 of *Lecture Notes in Computer Science*, pages 65–81. Springer, 2012.

[6] Dimitris Kontokostas, Charalampos Bratsas, Sören Auer, Sebastian Hellmann, Ioannis Antoniou, and George Metakides. Internationalization of Linked Data: The case of the Greek DBpedia edition. *Web Semantics: Science, Services and Agents on the World Wide Web*, 15(0):51 – 61, 2012.

[7] I. Dan Melamed and Philip Resnik. Tagger evaluation given hierarchical tag sets. *Computers and the Humanities*, pages 79–84, 2000.

[8] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 1003–1011, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[9] Vivi Nastase and Michael Strube. Decoding Wikipedia categories for knowledge acquisition. In *Proceedings of the 23rd national conference on Artificial intelligence - Volume 2*, AAAI'08, pages 1219–1224. AAAI Press, 2008.

[10] Thanh Nguyen, Viviane Moreira, Huong Nguyen, Hoa Nguyen, and Juliana Freire. Multilingual schema matching for Wikipedia infoboxes. *Proc. VLDB Endow.*, 5(2):133–144, October 2011.

[11] Joel Nothman, James R. Curran, and Tara Murphy. Transforming Wikipedia into named entity training data. In *Proceedings of the Australasian Language Technology Workshop*, Hobart, Australia, 2008.

[12] Alessio Palmero Aprosio, Claudio Giuliano, and Alberto Lavelli. Automatic expansion of DBpedia exploiting Wikipedia cross-language information. In *Proceedings of the 10th Extended Semantic Web Conference*, ESWC 2013, 2013.

[13] A. Pohl. Classifying the Wikipedia articles into the OpenCyc taxonomy. In *Proceedings of the Web of Linked Entities Workshop in conjuction with the 11th International Semantic Web Conference*, 2012.

[14] Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 697–706, New York, NY, USA, 2007. ACM.

[15] Afroza Sultana, Quazi Mainul Hasan, Ashis Kumer Biswas, Soumyava Das, Habibur Rahman, Chris Ding, and Chengkai Li. Infobox suggestion for Wikipedia entities. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, CIKM '12, pages 2307–2310, New York, NY, USA, 2012. ACM.

[16] Fei Wu and Daniel S. Weld. Open information extraction using Wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 118–127, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.