Disclose High Quality Structured Data with Airpedia

Alessio Palmero Aprosio *, Marco Fossati, and Claudio Giuliano
Fondazione Bruno Kessler, Via Sommarive 18, 38123 Trento, Italy
E-mail: {aprosio,fossati,giuliano}@fbk.eu

Abstract. The advent of Wikipedia as the best digital representation of cross-domain knowledge is now a reality. The DBpedia project aims at converting Wikipedia content into structured intelligence through the Linked Open Data paradigm and currently holds a vital role in the growth of the Web of Data as a multilingual interlinking kernel. However, its main classification system relies on a time-consuming manual procedure that aims at mapping Wikipedia infobox data to a common ontology. We present Airpedia, a tool that automatically generates class and property mappings for any DBpedia language chapter. We support our findings with the deployment of the Swedish, Ukrainian, and Esperanto DBpedias. Evaluations demonstrate that Airpedia is not only comparable to humans in terms of precision, but also provides a recall leap, with special regard to entities lacking infoboxes.

Keywords: Wikipedia, Data Quality, Kernel Methods, Supervised Learning, Linked Open Data

1. Introduction

A major portion of the World Wide Web content is nowadays represented as unstructured data, namely documents. Understanding its meaning is a complex task for machines and still relies on subjective human interpretations. The Web of Data envisions its evolution as a repository of machine-readable structured data. This would enable an automatic and unambiguous content analysis and its direct delivery to end users. The idea has not only engaged a long strand of research, but has also been absorbed by the biggest web industry players. Companies such as Google, Facebook and Microsoft, have already adopted large-scale semantics-driven systems, namely Google's *Knowledge Graph*,[1] Facebook's *Graph Search*,[2] and Microsoft's *Satori*.[3] Moreover, the World Wide Web Consortium, together with the Linked Data[4] initiative, has provided a standardized technology stack to publish freely accessible interconnected datasets.

On the other hand, Wikipedia is the result of a crowdsourced effort and stands for the best digital approximation of encyclopedic human knowledge. Its data has been growingly drawing both research and industry interests, and has driven the creation of several knowledge bases, the most prominent being BabelNet [27], DBpedia [20], Freebase [1], YAGO [18], WikiData [39], and WikiNet [25]. In particular, DBpedia[5] acts as the central component of the growing Linked Data cloud and benefits from a steadily increasing multilingual community of users and developers. Its stakeholders range from journalists [16] to governmental institutions [3], all the way to digital libraries [17]. The core contribution of DBpedia is to automatically extract structured data from unstructured (e.g., article abstracts) or semi-structured Wikipedia content, typically *infoboxes*.[6]

An infobox is a set of *attribute/value* pairs which serves as a richly informative summary and is typically rendered as a box on the top-right part of a Wikipedia article. It is crucial to highlight that each Wikipedia

[*] Corresponding author. E-mail: aprosio@fbk.eu.
[1] `http://bit.ly/goog-kg`
[2] `http://bit.ly/facebook-gs`
[3] `http://bit.ly/ms-satori`
[4] `http://linkeddata.org`
[5] `http://dbpedia.org`
[6] `http://bit.ly/wp-infobox`

chapter is maintained by different communities, with different guidelines that may critically affect the content of infoboxes. For instance, the Italian edition has the infobox `Carica_pubblica` (counterpart of the English `Officeholder` one), which only contains domain-specific *attributes*, while generic ones usually appear in `Bio`. Moreover, *values* are generally not constrained. Hence, in some editions there can be a single attribute `born` containing both place and date of birth, while other languages decide to split this piece of information into different attributes. Such data heterogeneity issues are alleviated by DBpedia through a *mapping* procedure that connects infobox attributes to the DBpedia ontology,[7] which is maintained via a collaborative paradigm similar to Freebase. Infoboxes can be perceived as the expression of a specific language and culture, while the ontology is the representation of the whole world of all Wikipedia chapters, thus acting as a multilingual glue to homogenize the data.

However, the current mapping paradigm weakens the data consumption capabilities, since it is affected by three major drawbacks, namely (a) the need for a heavy manual effort to create mappings, (b) its language-dependent nature (for each new language, a new set of mappings is required), and (c) the lack of synchronization whenever changes in the Wikipedia infoboxes occur. Furthermore, a clear problem of coverage has been recently pointed out in [34,33,35,10]. For instance, although the English Wikipedia contains over 4.2 million pages, DBpedia has only classified less than 2.2 million pages into its ontology. One of the major reasons is that a significant amount of Wikipedia articles do not contain an infobox, which is a valuable piece of information to infer the type of an article. This results in a large number of untyped entities, thus restraining the exploitation of the knowledge base.

The core contributions of this paper can be resumed as follows.

1. We present the first public release of *Airpedia*, a tool that tackles the aforementioned issues through a completely data-driven methodology and ultimately leads to the automatic construction of new DBpedia language chapters from scratch.
2. We illustrate three case studies that support the effectiveness of both the approach and the tool. More specifically, we deploy:
   - three previously missing DBpedia chapters, namely *Swedish*, *Ukrainian*, and *Esperanto*;
   - class and property mappings for each of them;
   - type coverage extension for articles lacking infoboxes.

The remainder of this paper is structured as follows. Section 2 reviews the state of the art. We illustrate the core approach and summarize the results we collected in Sections 3, 4 and 5. Section 6 describes the tool, how to use it and where to find its download and documentation links. We present our case studies in Section 7, before drawing our conclusions in Section 8.

2. Related work

The recently proposed automatic methods for type inference may be considered as bootstrapping resources to extract classes from Wikipedia data. [34] leverages a data-driven statistical strategy to assess and repair class declaration inconsistencies. [30] uses extracted datatypes to train a named entity recognizer. [19,11] identify the most appropriate class of a Wikipedia article by parsing its page abstract using natural language processing tools and resources. In this context, only English Wikipedia is considered, therefore this classifier cannot be easily adapted to other languages. Similarly, [35] only considers the English DBpedia and therefore does not take advantages from inter-language links. In addition, there is some manual effort to classify biographies (using tokens from categories[8]), that leads to very good results, but is not automatically portable to other languages; again linguistic tools are used to extract the definition from the first sentence. The approach presented in [12] classifies people over an excerpt of the WordNet ontology, using kernel functions that implicitly map entities, represented by aggregating all contexts in which they occur, into a latent semantic space derived from Wikipedia. This approach queries online the name of the entity to collect contextual information. [13] proposes an unsupervised method based on lexical entailment, consisting in assigning an entity to the category whose lexicalization can be replaced with its occurrences in a corpus preserving the meaning. [9] provides a deep analysis of several tools, either conceived specifically for knowledge extraction on the Semantic Web, or adaptable to it, or even acting as aggregators of extracted data from other tools.

[7] `http://bit.ly/dbp-ontology`

[8] For example, articles that belong to `Living people` category or categories ending with *births* or *deaths* were classified as `Person`

On the other hand, large-scale knowledge bases are experiencing a steadily growing commitment of both research and industry communities. A plethora of resources have been released in recent years. We report below a summary of the most influential examples, which all attempt to extract structured data from Wikipedia, although with different aims. BabelNet [27] is a multilingual lexico-semantic network, which recently moved towards a Linked Data compliant representation [4]. BabelNet emanates from the lexical databases community, with WordNet [5] being the most mature approach. DBpedia [20] leads current efforts based on the automatic extraction of unstructured and semi-structured content from all the Wikipedia language chapters. It serves as the kernel of the Linked Data cloud, gathering a huge amount of research efforts in the Web of Data and Natural Language Processing. The underlying framework is strengthened by a vibrant open source community of users and developers. Freebase [1] is the result of a crowdsourced effort, bearing a fine-grained schema thanks to its contributors. MENTA [2] is a massive lexical knowledge base, with data coming from 271 languages. WiBi [6] attempts to build a merged taxonomy by taking into account Wikipedia knowledge encoded both at the category and at the page layers. WikiNet [25,26] is built on top of heuristics formulated upon the analysis of Wikipedia content to deliver a multilingual semantic network. YAGO [18,38] provides a linkage facility between Wikipedia categories and WordNet terms.

3. Class mappings creation

Given a Wikipedia template classified as infobox, the first part of the workflow would try to map it, when possible, to a DBpedia ontology class. Using the same example described in the introduction, our goal is to map the `Carica_pubblica` infobox in the Italian Wikipedia to the `OfficeHolder` class in the DBpedia ontology. The presented approach is instance-based and exploits the already mapped Wikipedia pages (instances) 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 can be used to tune the tradeoff between precision and recall. For example, consider the Polish template `Wojna_infobox` ("War" in English), which is employed in 3,770 pages in the Polish Wikipedia. By leveraging cross-language links, we notice that 2,842 have a counterpart in one of the six pivot languages, and 2,716 are classified in one of the corresponding DBpedia chapters. In particular, 2,697 pages are classified as `MilitaryConflict`. Since 2,697 corresponds to 99% of the classified pages, we can assume that `MilitaryConflict` is the class that best approximates the possible mapping of the `Wojna_infobox` template. In this case, the Polish word "Wojna" means "War", clearly a synonym of `MilitaryConflict`.

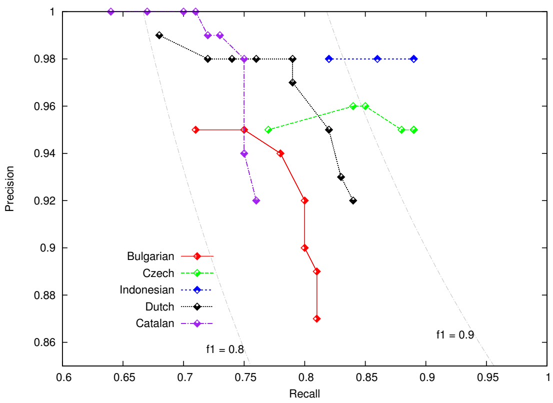


Fig. 2. Evaluation of the infobox mapping system.

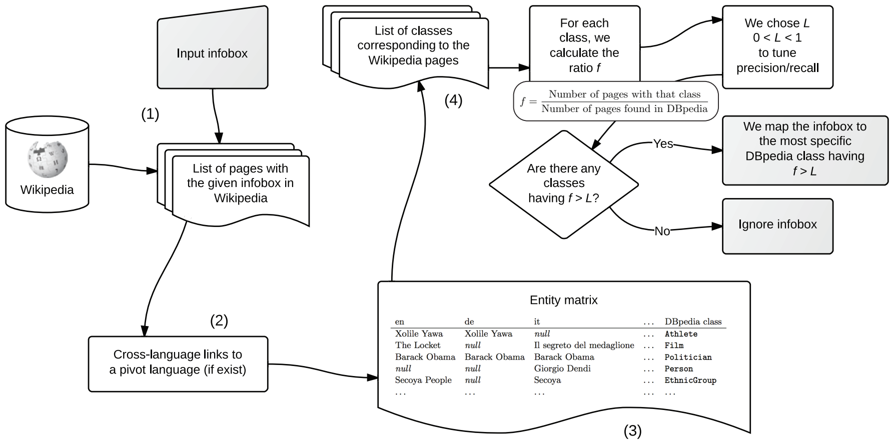
Consider now another example, involving a more ambiguous template, namely `Park_infobox` ("Park" in English). Although its translation does not give rise to ambiguity, the cross-language links bring to a different situation: the `Park` class has the majority, but its percentage is low (46%). Therefore, using a threshold $L = 0.5$, this solution is discarded and the superclass `Place` is assigned instead.

The mapping algorithm is detailed in [32] and is summarized in Figure 1.

3.1. 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 official DBpedia mapping website.[9] Specifically, the 5th April 2013 versions are used. Precision and recall values of the proposed method are calculated using these sets of mappings as gold standard. Figure 2 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 threshold L.

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


Input infobox
(1)
Wikipedia
List of pages with the given infobox in Wikipedia
(2)
Cross-language links to a pivot language (if exist)
List of classes corresponding to the Wikipedia pages
(4)
For each class, we calculate the ratio f
We chose L $0 < L < 1$ to tune precision/recall
$f = \frac{\text{Number of pages with that class}}{\text{Number of pages found in DBpedia}}$
Are there any classes having $f > L$?
Yes
No
We map the infobox to the most specific DBpedia class having $f > L$
Ignore infobox
Entity matrix
(3)

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**
...	...	...	...	...

Fig. 1. Workflow of the automatic infobox mapping system

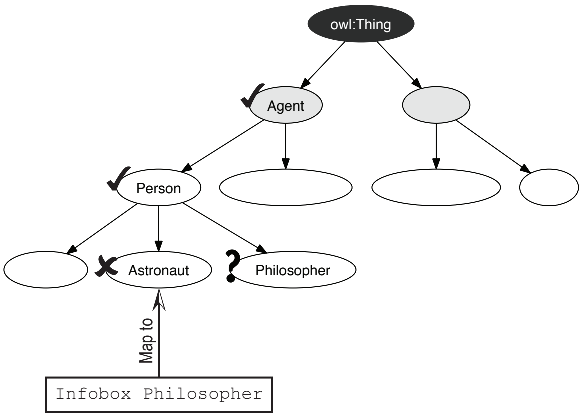


Fig. 3. Description of the evaluation.

These curves confirm the differences between the various versions of Wikipedia: in some cases the precision is high (e.g, 1 in Catalan), while in other languages it does not exceed .95. The different shapes of the curves reflect the heterogeneous structure of infoboxes, as the policies on infoboxes change from language to language [28].

The evaluation is performed as proposed by [22] for a similar hierarchical categorisation task. Figure 3 shows an example of the evaluation. The system tries to classify the infobox `Philosopher` and maps 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 a true positive. The obtained results demonstrate the reliability of our system. Once generated, the mappings can be checked by a human and can be used to start a new chapter of DBpedia or to extend an existing one, saving time in the mapping task.

4. Property mappings creation

In addition to class mappings, the DBpedia mappings also involves properties. For instance, the Italian infobox `Carica_pubblica` is not only useful to assign the `OfficeHolder` DBpedia ontology class for pages that include the infobox, but also carries a lot of additional information, such as the office held, the order, the party, and so on. These pieces of information are stored in the infobox *attributes*, which can be mapped to the DBpedia ontology properties, similarly to what happens to infoboxes and ontology classes (cf. previous Section).

Given an infobox I and an attribute A_I contained in I, the Airpedia tool maps the pair (I, A_I) to a DBpedia ontology property R. The approach exploits the redundancy of Wikipedia across different language editions, assuming that, if values of a given infobox attribute are similar to values of a given DBpedia property, then the attribute can be mapped to the property. In fact, a piece of information expressed in a particular page of Wikipedia is often repeated in different chapters. For instance, `Barack Obama`'s page in the English Wikipedia contains an infobox with his birth date, birth place, etc. The same kind of data is often included in the infoboxes of the corresponding pages in other Wikipedia editions. Thus, we leverage such redundancy to perform the alignment between Wikipedia attributes and DBpedia properties.

The main challenge relies in the comparison between data collected from DBpedia and attribute values stored in Wikipedia infoboxes. This is due to the fact that DBpedia is strongly typed, while Wikipedia does not have an explicit type system. Attribute values often contain a mixture of dates, numbers, and text, represented, formatted, and approximated in different ways depending on the Wikipedia edition and on the human editors. To tackle these issues, a function (detailed in [33]) is defined using a set of heuristics for numbers and dates, in order to extract structured information for each attribute value.

Existing versions of DBpedia are required to train the system. Specifically, we test the English, German, French, Spanish, and Portuguese editions. Given a target language l, the procedure extracts the mappings between DBpedia properties and infobox attributes in such language. l can be included in the set of languages chosen as training data. However, the experiments ignore l, since the main purpose is to build mappings for those Wikipedia chapters for which the corresponding DBpedia does not yet exist (such as Swedish, Ukrainian and Esperanto, cf. Section 7).

4.1. Mapping extraction

The algorithm used to determine whether an attribute A_I contained in the infobox I in Wikipedia can be mapped to a given property R in DBpedia is detailed in [33]. To find the mappings, we compute the pairwise similarities between the elements in the set of all the possible attributes A_I and the elements in the set of all the possible properties R. The candidates are represented as pairs (A_I, R), the pairs with the highest similarity $S(A_I, R)$ are considered correct mappings. The similarity is an average result calculated using instance-based similarities between the values of property R in different DBpedia editions and the values of the attribute A_I in different Wikipedia pages in the target language.

4.2. Evaluation

Experiments have been carried on Italian, using existing DBpedia editions in five languages (English, Spanish, Portuguese, German, and French) as training data. To perform the evaluation, three annotators created a gold standard by manually annotating 15 infoboxes (for a total of 100 different attributes), randomly extracted from the first 100 most frequent infoboxes in the Italian Wikipedia. The gold standard is available online on the Airpedia website.[10]

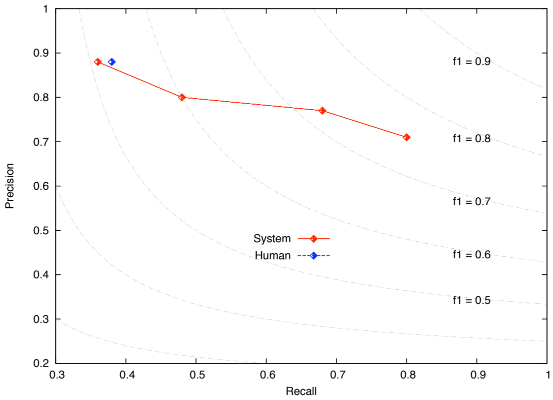


Fig. 4. Precision/recall curve of our system compared with the DBpedia original manual mapping in Italian. From left to right, λ value is 0.9, 0.7, 0.5, and 0.3.

Figure 4 shows the precision/recall curve, using different parameters in the mapping algorithm [33]. The results show that the coverage of the baseline (Human) is around 38% with a precision of around 88%. Our system is able to achieve comparable results in term of precision (87%), but it leads to a significant improvement in recall maintaining acceptable precision. Specifically, by exploiting existing mappings, we can cover up to 70% of the attributes with a precision around 80%.

5. Type coverage extension

In the last step of the workflow, our tool attempts to map a Wikipedia page to the corresponding class, in case the infobox is missing or the previous steps could not find a candidate. When the page is already classified in DBpedia, the tool tries to look for more specific classes. For instance, at the time of writing this paper, the Wikipedia page describing the writer `John Gay`[11] is classified as `Artist` using the original DBpedia mappings. Airpedia classifies the same entity as `Writer` (subclass of `Artist`).

The system is able to guess a type for pages without infobox by exploiting Wikipedia cross-language links to assign pages to a DBpedia class. For each entity described in Wikipedia, the tool looks for every page in every edition of the encyclopedia. In other words, the entity `Barack Obama` appears across multiple

[10] `http://bit.ly/air-gs`
[11] `https://en.wikipedia.org/wiki/John_Gay`

pages in several Wikipedia language chapters. In our approach, we first extract information (features) from each of these pages, and then we train a classifier with the Wikipedia pages appearing in any DBpedia mapping-based types dataset.[12] Finally, we exploit the resulting model to classify any Wikipedia page in any language.

Both the algorithm and the evaluation are detailed in [31].

5.1. Kernels for Entity Classification

The strategy adopted by kernel methods [37,36] consists of splitting the learning problem in two parts. First they embed the input data in a suitable feature space, and then use a linear algorithm (e.g., the perceptron) to discover nonlinear patterns in the input space. Typically, the mapping is performed implicitly by a so-called *kernel function*. The kernel function is an inner product, which can intuitively be considered as a similarity measure between the input data, that depends on the specific data type and domain. A typical similarity function is the inner product between feature vectors. Characterizing the similarity of the inputs plays a crucial role in determining the success or failure of the learning algorithm, and it is one of the central questions in the field of machine learning.

In practice, we combine five different kernel functions that calculate the pairwise similarity between entities using their corresponding Wikipedia articles as source of information. They are the only domain-specific elements of the proposed classification system, while the learning algorithm is a general purpose component. Many classifiers can be used with kernels, and we opt for k-nearest neighbor (k-nn).

The five basic kernels used are:

Bag-of-templates K_T**.** A count of how many occurrences of templates their corresponding Wikipedia articles in a specific language share.

Bag-of-categories K_C**.** Wikipedia categories are intended to group together articles on similar subjects and have proven useful in text classification [40], ontology learning [24], and ontology population [38].

Bag-of-sections K_S**.** Wikipedia articles are structured in several sections that might provide relevant cues for classification. For example, biographical articles typically include sections like *Early life*, *Career*, and *Personal life*; while articles referring to cities usually include sections like *Places of interest*, *Demographic evolution*, and *Administration*.

Table 1

Results of the most frequent class baseline (MF), the basic kernels (see Section 5.1) and the composite kernel K, using $z = 10$

	MF	K_T	K_C	K_S	K_W	K_L	K
Prec.	0.35	0.97	0.90	0.94	0.81	0.84	**0.91**
Rec.	0.38	0.31	0.40	0.16	0.22	0.41	**0.48**
F_1	0.31	0.47	0.55	0.27	0.34	0.55	**0.63**

Bag-of-words K_W**.** The use of infoboxes, categories, and sections ensures highly accurate classification, however it produces extremely sparse feature spaces that compromises the recall. To overcome this problem, content words of the text article are also exploited as additional sources of information.

Latent semantic analysis K_L**.** Since the bag-of-words representation does not deal well with lexical variability, latent semantic kernels are employed and an effective semantic vector space model using (unlabeled) external knowledge [37] is defined. It has been shown that semantic information is fundamental for improving the accuracy and reducing the amount of training data in many natural language tasks, including fine-grained classification of named entities [7,12], question classification [21], text categorization [14], and word sense disambiguation [15]. For each language, we derive the proximity matrix Π from the 200,000 most visited Wikipedia articles. After removing terms that occur less than 5 times, the resulting dictionaries contain about 300,000 terms. We use the SVDLIBC package[13] to compute the singular value decomposition (SVD), truncated to 100 dimensions.

Having defined all the basic kernels, representing different characteristics of entity descriptions, we finally define the composite kernel as

$$K(e_1, e_2) = \sum_{n=1} \frac{K_n(e_1, e_2)}{\sqrt{K_n(e_1, e_2) K_n(e_1, e_2)}}, \qquad (1)$$

where K_n is a valid basic kernel. The individual kernels are normalized. This plays an important role in allowing us to integrate information from heterogeneous feature spaces.

Table 1 reports the results of the most frequent class baseline, the basic kernels (K_T, K_C, K_S, K_W, and

[12] http://wiki.dbpedia.org/Datasets#h434-10

[13] http://tedlab.mit.edu/~dr/svdlibc/

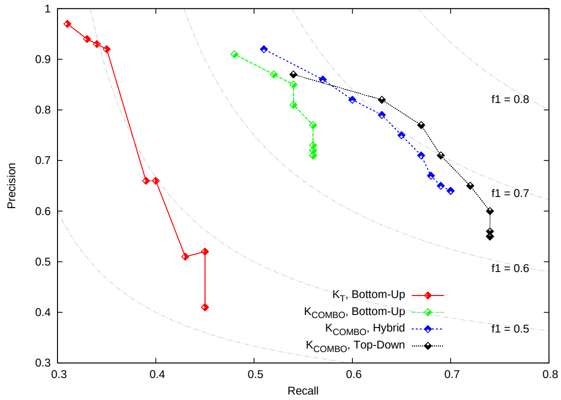


Fig. 5. Precision/recall curve of the page classification system using four different setups and different thresholds in the k-nn configuration.

K_L), and the composite kernel K. The experimental results show that the composite kernel K significantly outperforms the basic kernels. To assess the statistical significance between the obtained results (p-value $=$ 0.05), approximate randomization [29] is used.

5.2. *Evaluation*

Experiments are carried out on a set of Wikipedia articles that contains 500 randomly extracted entities not already present in DBpedia in any language. The dataset is split in development set (100 entities) and test set (400 entities). All entities have been manually annotated by the authors, with the most specific available class in the version 3.8 of the DBpedia ontology. 50 more entities have been annotated by three different annotators, resulting in an inter-agreement of 78% (Fleiss' kappa measure [8]). An additional `Unknown` class has been introduced to annotate those entities that cannot be assigned to any class in the ontology. When an entity is assigned to a class, it is also implicitly assigned to all its super-classes. For instance, classifying *Michael Jackson* as a `MusicalArtist`, the system implicitly classifies him also as `Artist`, `Person` and `Agent`.

The evaluation is performed using the same hierarchical model as in Section 3.1.

Figure 5 shows the precision/recall curves obtained using four different setups (for additional information, see [31]).

6. The tool

Airpedia[14] includes a set of tools[15] for processing Wikipedia XML dumps, DBpedia RDF datasets and finally for creating new mappings for a target DBpedia chapter. More specifically, the tools implement the entire workflow described in the previous sections.

The core Airpedia module is written in Java. It contains multiple runnable main classes that perform the following tasks:

1. Wikipedia dumps download and processing;
2. class and properties mappings extraction;
3. pages classification.

An additional set of bash scripts help the user speed up the whole procedure.

The package is released open-source under the Apache License 2.0.[16] Exhaustive documentation can be found in the wiki section of the project,[17] including an additional part, showing how to upload mappings to the DBpedia website (username and password needed) and how to create the RDF triples using the DBpedia Extraction Framework. The documentation describes also all the preprocessing steps needed to implement the whole workflow (for instance, downloading the Wikipedia dumps and the existing DBpedia mappings, extracting the dumps, identifying the infoboxes, and so on).

7. Some experiments on missing DBpedias

To assess the approach detailed in this paper, we generated three new DBpedia chapters from scratch, namely Swedish, Ukrainian and Esperanto.[18]

We applied the Airpedia tools to extract mappings and classified pages starting from the Wikipedia XML dumps and we uploaded the mappings to the DBpedia mapping website. We also published the datasets in the Airpedia website[19] and made a SPARQL endpoint available for queries.

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

[15] `https://bitbucket.org/fbk/airpedia/`

[16] `http://www.apache.org/licenses/`

[17] https://bitbucket.org/fbk/airpedia/wiki/Home

[18] Esperanto is a constructed language spoken by about 2M people in the world. The Esperanto Wikipedia is – with its 200K+ articles – one of the biggest Wikipedia for which the corresponding DBpedia chapter has not been created yet. It is ranked 34th in the list of Wikipedias sorted by the number of articles, and it is constantly updated by a community of almost 100K users.

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

Table 2

Statistics on the Swedish DBpedia.

Number of pages in Wikipedia	1,935,569
Automatically created mappings for classes	228
Automatically created mappings for properties	1,136
Pages classified using the mappings (Section 3)	1,565,041
Additional classified pages (Section 5)	177,900
Total classified pages	1,742,941
Extracted properties	3,395,380
Website: `http://sv.dbpedia.org`	
SPARQL endpoint: `http://sv.dbpedia.org/sparql`	

Table 3

Statistics on the Ukrainian DBpedia.

Number of pages in Wikipedia	534,545
Automatically created mappings for classes	281
Automatically created mappings for properties	888
Pages classified using the mappings (Section 3)	305,779
Additional classified pages (Section 5)	75,749
Total classified pages	381,528
Extracted properties	822,220
Website: `http://uk.dbpedia.org`	
SPARQL endpoint: `http://uk.dbpedia.org/sparql`	

Table 4

Statistics on the Esperanto DBpedia.

Number of pages in Wikipedia	202,236
Automatically created mappings for classes	96
Automatically created mappings for properties	345
Pages classified using the mappings (Section 3)	83,320
Additional classified pages (Section 5)	52,128
Total classified pages	135,448
Extracted properties	330,798
Website: `http://eo.dbpedia.org`	
SPARQL endpoint: `http://eo.dbpedia.org/sparql`	

Tables 2, 3 and 4 show statistics on the deployed chapters.

8. Conclusions and future work

In this paper, we presented *Airpedia*, a tool for the construction of a DBpedia chapter from scratch. The workflow consists of three major phases:

1. Class mapping generation;
2. Properties mapping generation;
3. DBpedia ontology type coverage extension.

We evaluated our approach upon existing DBpedia editions and a manually annotated dataset of 400 Wikipedia articles. Tradeoff between precision and recall can be varied in all steps. Although the mapping generation procedure is not generally error-prone, the system can be used to speed it up, and serves as an initial step for the release of missing DBpedia chapters, or for the extension of existing ones. The quality of the obtained mappings may be further curated on account of a manual validation. We argue that such task is faster than the mapping one starting from scratch (i.e., from Wikipedia infoboxes).

There remains room for further improvements, especially with respect to the generation of properties mapping. For instance, the similarity function between values in different languages can be refined with a smarter normalization and a better recognition of typed entities (such as temporal expressions, units, and common abbreviations). In addition, the distant supervision paradigm [23] may be leveraged to extend the coverage over properties when the infobox is missing in the page or does not contain the required attribute.

Acknowledgements

The authors thank Fabio Bettani (Italian Esperanto Federation), Eva Blomqvist (Linköping University) and Kateryna Tymoshenko (University of Trento) for the precious help in translating the web page of the DBpedia chapters described in Section 7.

References

[1] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250. ACM, 2008.

[2] Gerard de Melo and Gerhard Weikum. MENTA: inducing multilingual taxonomies from Wikipedia. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1099–1108. ACM, 2010.

[3] Li Ding, Timothy Lebo, John S Erickson, Dominic DiFranzo, Gregory Todd Williams, Xian Li, James Michaelis, Alvaro Graves, Jin Guang Zheng, Zhenning Shangguan, et al. TWC LOGD: A portal for linked open government data ecosystems. *Web Semantics: Science, Services and Agents on the World Wide Web*, 9(3):325–333, 2011.

[4] Maud Ehrmann, Francesco Cecconi, Daniele Vannella, John McCrae, Philipp Cimiano, and Roberto Navigli. Representing Multilingual Data as Linked Data: the Case of BabelNet 2.0. In *Proc. of LREC*, 2014.

[5] Christiane Fellbaum. *Wordnet: An electronic lexical database*. MIT Press Cambridge, 1998.

[6] Tiziano Flati, Daniele Vannella, Tommaso Pasini, and Roberto Navigli. Two Is Bigger (and Better) Than One: the Wikipedia Bitaxonomy Project. In *Proc. of the 52nd Annual Meeting of the Association for Computational Linguistics*, 2014.

[7] Michael Fleischman and Eduard Hovy. Fine grained classification of named entities. In *Proceedings of the 19th International Conference on Computational Linguistics*, Taipei, Taiwan, 2002.

[8] Joseph L. Fleiss. Measuring Nominal Scale Agreement Among Many Raters. *Psychological Bulletin*, 76(5):378–382, 1971.

[9] Aldo Gangemi. A comparison of knowledge extraction tools for the semantic web. In *The Semantic Web: Semantics and Big Data*, volume 7882 of *Lecture Notes in Computer Science*, pages 351–366. Springer Berlin Heidelberg, 2013.

[10] Aldo Gangemi, Andrea Nuzzolese, Valentina Presutti, Francesco Draicchio, Alberto Musetti, and Paolo Ciancarini. Automatic typing of DBpedia entities. In *The Semantic Web – ISWC 2012*, volume 0 of *Lecture Notes in Computer Science*, pages 65–81. Springer Berlin / Heidelberg, 2012.

[11] Aldo Gangemi, Andrea Giovanni Nuzzolese, Valentina Presutti, Francesco Draicchio, Alberto Musetti, and Paolo Ciancarini. Automatic typing of DBpedia entities. In *International Semantic Web Conference (1)*, volume 7649 of *Lecture Notes in Computer Science*, pages 65–81. Springer, 2012.

[12] Claudio Giuliano. Fine-grained classification of named entities exploiting latent semantic kernels. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, CoNLL '09, pages 201–209, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[13] Claudio Giuliano and Alfio Gliozzo. Instance based lexical entailment for ontology population. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 248–256, Prague, Czech Republic, June 2007. Association for Computational Linguistics.

[14] Alfio Gliozzo and Carlo Strapparava. Domain kernels for text categorization. In *Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, pages 56–63, Ann Arbor, Michigan, June 2005.

[15] Alfio Massimiliano Gliozzo, Claudio Giuliano, and Carlo Strapparava. Domain kernels for word sense disambiguation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 403–410, Ann Arbor, Michigan, June 2005.

[16] Jonathan Gray, Lucy Chambers, and Liliana Bounegru. *The data journalism handbook.* " O'Reilly Media, Inc.", 2012.

[17] Bernhard Haslhofer, Elaheh Momeni, Manuel Gay, and Rainer Simon. Augmenting Europeana content with linked data resources. In *Proceedings of the 6th International Conference on Semantic Systems*, page 40. ACM, 2010.

[18] Johannes Hoffart, Fabian M Suchanek, Klaus Berberich, and Gerhard Weikum. Yago2: a spatially and temporally enhanced knowledge base from wikipedia. *Artificial Intelligence*, 194:28–61, 2013.

[19] Tomáš Kliegr and Ondřej Zamazal. Towards linked hypernyms dataset 2.0: Complementing dbpedia with hypernym discovery. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may 2014. European Language Resources Association (ELRA).

[20] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, et al. DBpedia – A large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web*, 2014.

[21] Xin Li and Dan Roth. Learning question classifiers: the role of semantic information. *Natural Language Engineering*, 12(3):229–249, 2005.

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

[23] 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.

[24] 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.

[25] Vivi Nastase and Michael Strube. Transforming Wikipedia into a large scale multilingual concept network. *Artificial Intelligence*, 194:62–85, 2013.

[26] Vivi Nastase, Michael Strube, Benjamin Boerschinger, Cäcilia Zirn, and Anas Elghafari. WikiNet: A Very Large Scale Multi-Lingual Concept Network. In *LREC*. Citeseer, 2010.

[27] Roberto Navigli and Simone Paolo Ponzetto. Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250, 2012.

[28] 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.

[29] Eric W. Noreen. *Computer-Intensive Methods for Testing Hypotheses : An Introduction.* Wiley-Interscience, 1989.

[30] 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.

[31] 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*, 2013.

[32] Alessio Palmero Aprosio, Claudio Giuliano, and Alberto Lavelli. Automatic Mapping of Wikipedia Templates for Fast Deployment of Localised DBpedia Datasets. In *Proceedings of the 13th International Conference on Knowledge Management and Knowledge Technologies*, 2013.

[33] Alessio Palmero Aprosio, Claudio Giuliano, and Alberto Lavelli. Towards an Automatic Creation of Localized Versions of DBpedia. In *Proceedings of the 12th International Semantic Web Conference*, 2013.

[34] Heiko Paulheim and Christian Bizer. Type inference on noisy rdf data. In *The Semantic Web–ISWC 2013*, pages 510–525. Springer, 2013.

[35] Aleksander 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.

[36] B. Schölkopf and A. Smola. *Learning with Kernels*. MIT Press, Cambridge, Massachusetts, 2002.

[37] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[38] 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.

[39] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledge base. *Communications of the ACM*, 57(10):78–85, 2014.

[40] Pu Wang, Jian Hu, Hua-Jun Zeng, and Zheng Chen. Using Wikipedia knowledge to improve text classification. *Knowledge and Information Systems*, 19:265–281, 2009. 10.1007/s10115-008-0152-4.