ARTICLE IN PRESS

Web Semantics: Science, Services and Agents on the World Wide Web ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Web Semantics: Science, Services and Agents on the World Wide Web

journal homepage: www.elsevier.com/locate/websem

Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery

Tomáš Kliegr *

Department of Information and Knowledge Engineering, Faculty of Informatics and Statistics, University of Economics, nám. W Churchilla 4, 13067, Prague, Czech Republic
Multimedia and Vision Research Group, Queen Mary, University of London, 327 Mile End Road, London E1 4NS, United Kingdom

ARTICLE INFO

Article history:
Received 1 October 2014
Accepted 4 November 2014
Available online xxxx

Keywords:
DBpedia
Hearst patterns
Hypernym
Linked data
YAGO
Wikipedia
Type inference

ABSTRACT

The Linked Hypernyms Dataset (LHD) provides entities described by Dutch, English and German Wikipedia articles with types in the DBpedia namespace. The types are extracted from the first sentences of Wikipedia articles using Hearst pattern matching over part-of-speech annotated text and disambiguated to DBpedia concepts. The dataset covers 1.3 million RDF type triples from English Wikipedia, out of which 1 million RDF type triples were found not to overlap with DBpedia, and 0.4 million with YAGO2s. There are about 770 thousand German and 650 thousand Dutch Wikipedia entities assigned a novel type, which exceeds the number of entities in the localized DBpedia for the respective language. RDF type triples from the German dataset have been incorporated to the German DBpedia. Quality assessment was performed altogether based on 16.500 human ratings and annotations. For the English dataset, the average accuracy is 0.86, for German 0.77 and for Dutch 0.88. The accuracy of raw plain text hypernyms exceeds 0.90 for all languages. The LHD release described and evaluated in this article targets DBpedia 3.8, LHD version for the DBpedia 3.9 containing approximately 4.5 million RDF type triples is also available.

1. Introduction

The Linked Hypernyms Dataset (LHD) provides entities described by Dutch, English and German Wikipedia articles with types taken from the DBpedia namespace. The types are derived from the free-text content of Wikipedia articles, rather than from the semistructured data, infoboxes and article categories, used to populate DBpedia [1] and YAGO [2]. The dataset contains only one type per entity, but the type has stable and predictable granularity. These favorable properties are due to the fact that the types are sourced from the first sentences of Wikipedia articles, which are carefully crafted by the Wikipedia editors to contain the most important information.

To illustrate the LHD generation process, consider the first sentence of the Wikipedia article entitled "Karel Čapek": *Karel Čapek (...) was a Czech writer of the early* 20th *century best known for his science fiction, including his novel War with the Newts and the play R.U.R. that introduced the word robot.* This text is first processed with a part of speech (POS) tagger. Consequently, using a *JAPE grammar*, a regular expressions language referencing the underlying text as well as the assigned POS tags, the hypernym "writer" is extracted. This hypernym is then disambiguated to a DBpedia Ontology class `dbo:Writer`. The resulting entry in LHD is the RDF type triple[1]:

```
dbp:Karel_Čapek rdf:type dbo:Writer .
```

The LHD dataset was subject to extensive evaluation, which confirms the following hypotheses:

- high quality types for DBpedia entities can be extracted from the first sentences of Wikipedia articles,
- resulting set of types provides a substantial complement to types obtained by the analysis of Wikipedia infoboxes and categories.

This dataset can thus be used to "fill the gaps" in DBpedia and YAGO, the two largest semantic knowledge bases derived

* Correspondence to: Department of Information and Knowledge Engineering, Faculty of Informatics and Statistics, University of Economics, nám. W Churchilla 4, 13067, Prague, Czech Republic.
E-mail address: tomas.kliegr@vse.cz.

[1] `dbo:` standing for http://dbpedia.org/ontology/ and `dbp:` for http://dbpedia.org/resource/.

http://dx.doi.org/10.1016/j.websem.2014.11.001
1570-8268/

from the semistructured information in Wikipedia. To illustrate the individual types of complementarity, consider the following examples.

- LHD can provide a more *specific* type than DBpedia or YAGO. This is typically the case for less prolific entities, for which the semistructured information in Wikipedia is limited. The most specific type provided by DBpedia or YAGO for the "HMS Prince Albert (1984)" entity is `dbo:Ship`, while LHD assigns the type `dbp:Warship` (as a subclass of `dbo:Ship`).
- LHD can provide a more *precise* type. An asteroid named "1840 Hus" is assigned type `dbo:Asteroid` in LHD, while DBpedia assigns it the imprecise type `dbo:Planet` (asteroid is not a subclass of planet).
- LHD is in some cases the only knowledge base providing *any type information*. For example, for asteroid "1994 Shane", neither DBpedia nor YAGO provide a type, while LHD does.
- LHD helps to choose the primary most specific type for an entity. DBpedia assigns Karel Čapek, a famous Czech writer, `dbo:Person` as the most specific DBpedia type, YAGO assigns `yago:CzechScience FictionWriters`, but also several other less commonly sought for types, such as `yago:PeopleFromTrutnov District`. Following the choice of Wikipedia editors for the first article's sentence, LHD assigns a single type: `dbo:Writer`. This can help to identify `yago:CzechScienceFictionWriters` as the primary most specific type for Karel Čapek (as opposed to `yago:PeopleFromTrutnovDistrict`).

The last bullet point shows that even if the LHD provided type is less specific than the type provided in YAGO or DBpedia, it may not be completely redundant. The LHD dataset for German, English and Dutch is provided under a free license. Additionally, this paper along with the complementary resources[2] describes the LHD design process in detail sufficient to allow for generation of the dataset also for other language versions of Wikipedia.

This paper is organized as follows. Section 2 gives a survey of related work. Section 3 describes the text-mining algorithm, Section 4 the procedure for disambiguating the hypernyms with a DBpedia URI and the resulting datasets. Section 5 describes the alignment of the linked hypernyms with DBpedia and YAGO2s ontologies. Human evaluation of accuracy is presented in Section 7. The following two sections discuss LHD extensions. Section 8 presents LHD 2.0 draft, which uses statistical type inference to increase the number of types mapped to the DBpedia Ontology. Steps required to extend LHD to other languages are covered in Section 9. The conclusion in Section 10 summarizes the key statistics, gives dataset license, availability and discusses possible applications, including a named entity recognition system based on the Linked Hypernyms Dataset.

2. Related work

The use of methods from computational linguistics on extraction of machine-readable knowledge from electronic dictionary-like resources has long been studied (cf. Wilks et al. [3]) with research specifically on extraction of hyponymy–hypernymy relation from lexical resources using patterns dating back to at least 1984 [4]. The hypernym discovery approach proposed here is based on the application of a special type of hand-crafted lexico-syntactic patterns often referred to as Hearst patterns [5]. The prototypical Hearst pattern goes along the sentence frame H0:

"An L_0 is a (kind of) L_1" (H0).

Hearst patterns were so far used primarily on large text corpora with the intent to discover all word-hypernym pairs in a collection. The extracted pairs can serve e.g. for taxonomy induction [6,7] or ontology learning [8]. This effort was undermined by the relatively poor performance of syntactic patterns in the task of extracting *all* candidate hypernym/hyponym word pairs from a *generic* corpus. The recall–precision graph for the seminal hypernym classifier introduced by [6] indicates precision 0.85 at recall 0.10 and precision 0.25 at recall of 0.30.

Utilization of hypernyms discovered from textual content of *Wikipedia articles* was investigated in a number of works. Strube and Ponzetto [9] built a large scale taxonomy from relation candidates extracted from English Wikipedia categories. One of the sources of evidence for a relation being classified as a *subsumption* or not is obtained by applying Hearst patterns (and corresponding anti-patterns) on Wikipedia and the Tipster corpus. The result of the classification was determined based on whether a majority of the matches are accounted for the patterns or the anti-patterns. Detection of hypernyms in the free text of Wikipedia articles was used as one of the methods to classify relation candidates extracted from the categories and as such had only a marginal influence on the overall results (0.04 precision improvement).

To the best of my knowledge, [10] were first to implement a system that extracts a hypernym for the Wikipedia *article subject* with high precision from *the first sentence* of the article text with the help of Part of Speech (POS) tagger. The discovered hypernyms were used as features in a Conditional-Random-Fields-based named entity tagger yielding again only a moderate improvement in accuracy.

HypernymFinder [11] is an algorithm that searches a hypernym for a specific noun phrase. It identifies a number of candidates by searching for occurrences of Hearst patterns featuring the query hyponym and then uses the frequency of the matches to determine the best hypernyms. The Hearst patterns were matched against a large 117 million web page corpus. The authors record an improvement over the results reported earlier by [6] for lexicosyntactic patterns with baseline precision at 0.90 and recall at 0.11.

The 2007 paper [10] laid foundations to the use of Hearst patterns over Wikipedia that is called *Targeted Hypernym Discovery* task (THD) in this paper. To get hypernym for a particular entity, THD applies Hearst patterns on a document describing the entity. In earlier work using English Wikipedia, we obtained accuracy of 87% when extracting hypernyms from articles describing named entities [12]. To the extent of my knowledge, this 2008 paper presented the first evaluation of the quality of hypernyms discovered from Wikipedia. Similar results for extracting hypernyms from articles describing people in German Wikipedia were later reported by [13] (also refer to Section 7).

Contrasted to HypernymFinder, which uses a set of randomly selected noun phrases as query hyponyms, the set of query hyponyms in THD is limited to Wikipedia article titles. With this constraint, the first Hearst pattern match in the first sentence of the respective article yields hypernyms with higher precision and substantially higher recall of 0.94 and 0.88 respectively for English Wikipedia (cf. Section 7.1). Note that the results for THD, HypernymFinder [11], and the algorithm of Snow et al. [6] cannot be directly mutually compared, since the latter evaluates precision and recall over candidate hypernym/hyponym word pairs (the input is a large corpus), while HypernymFinder is concerned with whether or not a good hypernym for a given noun phrase can be retrieved (the input is again a large corpus), and eventually THD evaluates whether a good hypernym for Wikipedia article subject can be retrieved (the input is that article's first sentence).

Tipalo [14] is the most closely related system to the workflow used to generate LHD. Similarly to approach presented in this paper, Tipalo covers the complete process of generating types for

[2] http://ner.vse.cz/datasets/linkedhypernyms.

Please cite this article in press as: T. Kliegr, Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery, Web Semantics: Science, Services and Agents on the World Wide Web (2014), http://dx.doi.org/10.1016/j.websem.2014.11.001

Table 1
Tipalo output for the "Kanai Anzen" entity. Retrieved using on-line service at http://wit.istc.cnr.it/stlab-tools/tipalo/ on 23/09/14.

Subject	Predicate	Object
dbpedia:Kanai_Anzen	rdf:type	domain:Omamorous
dbpedia:Kanai_Anzen	rdf:type	domain:Religion
dbpedia:Kanai_Anzen	rdf:type	domain:JapaneseAmulet
domain:JapaneseAmulet	rdfs:subClassOf	domain:Amulet
dbpedia:Amulet	owl:equivalentClass	dbpedia:Amulet

DBpedia entities from the free text of Wikipedia articles. However, while LHD generation process uses THD to extract the hypernym directly from the POS-tagged first sentence, the extraction process in Tipalo is more complex. The algorithm starts with identifying the first sentence in the abstract which contains the definition of the entity. In case a coreference is detected, a concatenation of two sentences from the article abstract is returned. The resulting natural language fragment is deep parsed for entity definitions using the FRED tool [15] for ontology learning. FRED uses methods based on frame semantics for deriving RDF and OWL representations of natural language sentences.

The result of analyzing the entity definition is maximum one type for THD, while Tipalo may output multiple types. If there are multiple candidate hypernyms in the definition, Tipalo uses all of them. Also, if a hypernym is composed of a multi-word noun phrase Tipalo outputs multiple types formed by gradually stripping the modifiers (cf. example below).

To illustrate the differences, consider the Wikipedia page "Kanai Anzen". Using the first sentence of the Wikipedia entry: *Kanai Anzen is a type of omamori, or Japanese amulet of the Shinto religion.*, THD outputs just the head noun of the first candidate hypernym[3] ("omamori"). Tipalo result for this Wikipedia page is presented in Table 1. Tipalo outputs four types ("JapaneseAmulet", "Amulet", "Religion" and "Omamorous"). Similarly to steps subsequent to the THD execution, Tipalo detects whether the entity is a class or instance and correspondingly selects the relation (`rdfs:subClassOf` or `rdf:type`) with which the entity will be linked to the assigned types. Another interesting aspect common to both systems is their use of DBpedia resources as classes.

In this specific example, the results of both tools are comparable and somewhat complementary: LHD provides a more precise DBpedia mapping (omamori is a type of Japanese amulet), while Tipalo output contains supplemental taxonomic information (JapaneseAmulet as a subclass of Amulet). While in LHD all types are represented with DBpedia concepts, Tipalo also outputs concepts in the FRED namespace.[4]

Tipalo uses a context-based disambiguation algorithm which links the concepts to WordNet synsets. Consequently, OntoWordnet 2012, an OWL version of WordNet, is used to align the synsets with types from the Dolce Ultra Lite Plus[5] (DULplus) and the Dolce Zero (D0)[6] ontologies. The latter being an ontology defined by the authors which generalizes a number of DULplus classes in OntoWordnet. In contrast, LHD aims at providing types suitable for DBpedia and YAGO enrichment. To this end, the types assigned to entities are from the DBpedia namespace, preferably DBpedia Ontology classes.

To illustrate the differences in ontology mapping results, consider the types returned for "Lupercal" (an example listed on the Tipalo homepage). Tipalo assigns type `dbp:Cave`, which is mapped via the `owl:equivalentClass` to `wn30:synset-cave-noun-1` and is marked as a subclass of `d0:Location`.[7] In contrast, LHD assigns this entity with `dbo:Cave`, a class from the DBpedia ontology.

As could be seen, there are multiple dissimilarities between the LHD generation process and Tipalo both on the algorithmic and conceptual level. The scale of the resources is also different. Tipalo is demonstrated with a proof of concept ontology constructed from analyzing 800 randomly selected English Wikipedia pages and evaluated on 100 articles. However, its online demo service is able to process any Wikipedia article. LHD was generated for three complete Wikipedia languages and is supplemented by evaluation performed on two orders of magnitude larger scale. A limited comparison with Tipalo in terms of hypernym extraction results is covered in Section 7.

The Linked Hypernyms Dataset described in this paper is a comprehensive attempt to extract types for DBpedia entities from the free text of Wikipedia articles. The dataset is generated using adaptations of previously published algorithms, approaches and systems: Hearst patterns are used to extract hypernyms from the plain text, Wikipedia search for disambiguation, and string-based ontology matching techniques for alignment with DBpedia and YAGO ontologies.

By providing results not only for the English Wikipedia, but also for the entire Dutch and German Wikipedias, it is demonstrated that the presented approach can effectively be extended to other languages. The retrieval of new types for entities from the free-text can provide a complementary information to other recent DBpedia enrichment efforts [16,17], which derive new types either from data already in the Linked Open Data (LOD) cloud (as in [16]), or from the semistructured information (cross-language links in [17]).

3. Targeted hypernym discovery

The Targeted Hypernym Discovery implementation used to perform the linguistic analysis for Linked Hypernyms Dataset is an extended and reworked version of the algorithm presented in [12]. The precision and recall of the grammars was improved. Also, the workflow was changed to support multilingual setting and grammars for German and Dutch were added. The largest conceptual deviation from the original algorithm as well as from the prototypical H0 pattern is that the occurrence of the subject (*L0*) is not checked. According to empirical observation this change increases recall with negligible effect on precision.[8]

The schematic grammar used is

"* is a (kind of) L_1" (H1).

where * denotes a (possibly empty) sequence of any tokens. This modification increased recall. Restricting the extraction to the first match in the article's first sentence helped to improve the precision. The grammars were manually developed using a set of 600 randomly selected articles per language.

The main features of the THD implementation used to generate the presented datasets include:

[3] As discussed in Section 3 this is the most reliable choice according to empirical observation.

[4] `http://www.ontologydesignpatterns.org/ont/fred/domain.owl#`.

[5] http://www.ontologydesignpatterns.org/ont/wn/dulplus.owl.

[6] `http://www.ontologydesignpatterns.org/ont/d0.owl`.

[7] `wn30syn: http://purl.org/vocabularies/princeton/wn30/instances/`.

[8] Validating whether the subject of the article's first sentence matches the article title is an unnecessary check, which sometimes causes false negative matches due to differences between the first sentence's subject and the article title. For example, the article entitled "ERAP" starts with: *Entreprise de recherches et d'activités pétrolières is a French petroleum company...*. Checking the occurrence of "ERAP" in the first sentence would result in no match.

Please cite this article in press as: T. Kliegr, Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery, Web Semantics: Science, Services and Agents on the World Wide Web (2014), http://dx.doi.org/10.1016/j.websem.2014.11.001

- **Only the first sentence of the article is processed**. More text (first paragraph, section) introduces noise according to empirical observation.
- **Only the first hypernym is extracted**.

> *Example.*
> Consider sentence: *Évelyne Lever is a contemporary French historian and writer.* The result of THD is one hypernym *historian*, the word *writer* is ignored. German articles are more likely to contain multiple hypernyms in the first sentence, while this is less common for English and Dutch.

- **Some Wikipedia article types are excluded**. Programmatically identifiable articles that do not describe a single entity are omitted. This applies to lists, disambiguation articles and redirects.
- **For multi-word hypernyms, the result is the last noun**.

> *Example.*
> Consider sentence: *Bukit Timah Railway Station was a railway station.* The THD result is "station", rather than "railway station". Extracting the complete multi-word sequence would yield a more specific hypernym in many cases, but a straightforward implementation would also negatively impact precision.

Multi-word hypernyms were left for future work.

- **Hypernym contained in the entity name or article title is ignored**.

> *Example.*
> While for a human it may be obvious that if something is named "Bukit Timah Railway Station" then it is a (railway) station, it follows from the nature of Hearst patterns that the hypernym in the entity name is ignored. Likewise, hypernyms contained in article title such as the word "novel" in "Hollywood (Vidal novel)" are ignored.

- **Common generic hypernyms that precede a more specific hypernym are skipped**.

> *Example.*
> Consider again the sentence: *Kanai Anzen is a type of omamori, or Japanese amulet of the Shinto religion.* THD skips the word "type" and returns the word "omamori". The list of these generic hypernyms is specified in the grammar for each language, and includes for example the "name of" expression, but also already relatively specific hypernyms such as species ("species of").

- **The result of THD is lemmatized**. In languages where hypernyms often appear in inflected forms lemmatization ensures that a base form is used as the hypernym.[9]

> *Example.*
> Consider sentence: *Die York University ist eine von drei Universitäten in Toronto.* With the first hypernym being Universitäten, the result of lemmatization is Universität, which is used as the plain text hypernym for this entry.

The set of Wikipedia article–hypernym pairs output by THD is referred to as the *"Plain Text" Hypernyms Dataset*.

4. Hypernym linking

The limitation of THD is that its output is a plain string, which is unusable in the Linked Data environment. As a first attempt to address the problem, the "most frequent sense" disambiguation is used.

This approach is based on a simple, yet according to experimental results [18], effective way of discovering links to DBpedia—the Wikipedia Search API.[10] Since there is an unanimous mapping between Wikipedia articles and DBpedia resources, the linking algorithm first searches for an article describing the hypernym in Wikipedia and then the URL of the first article hit is transformed to a DBpedia URI.

In the TAC English Entity Linking task [18], this approach had a close median performance among the 110 submissions with B^{3+} F1 measure on 2190 queries of 0.54–0.56 (depending on whether live Wikipedia or a Wikipedia mirror was used). The best system achieved B^{3+} F1 result of up to 0.75, the average B^{3+} F1 result was 0.56. Compared to other solutions, using Wikipedia search for disambiguation in the LHD generation process has several advantages. Wikipedia search is readily available for all Wikipedia languages, is fast, and implies no dependency on a third-party component.

4.1. Disambiguation

Wikipedia Search API uses a PageRank-like algorithm for determining the importance of the article in addition to the textual match with the query. Since the hypernyms tend to be general words with dominant most frequent sense, the most frequent sense assumption works well as experimentally demonstrated in Section 7.2.

It should be noted that the following possibility was investigated: using the hyperlinks that are sometimes placed on the hypernym in the source article. However, only a small fraction of articles contains such links, furthermore, the quality of these links seems to be lower than what can be obtained by the search-based mapping. Linked hypernyms are the output of the disambiguation process.

4.2. Data cleansing

The first step, applicable only to non-English DBpedia, is to use the DBpedia's interlanguage links to replace the linked hypernyms with their English counterparts.

The main cleansing step amounts to performing replacements and deletions according to manually specified rules. These rules were identified by manually checking several hundreds of the most frequent types assigned by THD.

Mapping rules are used to replace a particular linked hypernym. Mapping rules were introduced to tackle two types of problems:

- For some types the hypernym discovery makes systematic errors, typically due to POS tagger error or deficiency in the THD grammar.

[9] During LHD dataset generation, the lemma was used instead of the underlying string if it was made available by the tagger for the given language.

[10] http://www.mediawiki.org/wiki/Extension:Lucene-search.

Please cite this article in press as: T. Kliegr, Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery, Web Semantics: Science, Services and Agents on the World Wide Web (2014), http://dx.doi.org/10.1016/j.websem.2014.11.001

Table 2
Hypernyms and Linked Hypernyms Datasets—statistics and comparison with DBpedia and YAGO2s. The largest dataset for each language is listed in bold. The Wikipedia snapshots used to generate the datasets: December 1st, 2012 (German), October 11th, 2012 (Dutch), September 18th, 2012 (English).

Statistic	Dutch	English	German
Linked Hypernyms Dataset			
Wikipedia articles	1691k	5610k	2942k
–without redirect articles (is_page_redirect = 1 database field)	1505k	3299k	2252k
–without lists, images, etc. (identified from article name)	1422k	2590k	1930k
"Plain text" Hypernyms dataset	889k	1553k	937k
linked hypernyms (before data cleansing)	670k	1393k	836k
Linked Hypernyms Dataset—instances	**664k**	1305k	**825k**
Linked Hypernyms Dataset—classes	1k	4k	3k
Other datasets			
DBpedia 3.8—instances with type (instance_types_{lang}.nt)	11k	2351k	449k
YAGO2s—instances with type (yagoTypes.ttl)		**2886k**	

Example. A mapping rule tackling such issue is "`dbp:Roman` → `dbp:School`". The word "Roman" is an adjective that should never be marked as a hypernym. The reason is that the POS tagger incorrectly marks "Roman" as a noun if it appears in collocation "Roman catholic school" resulting in the THD grammar yielding "Roman" instead of "School". Since "Roman" is not output by THD virtually in any other case, the existence of the mapping rule increases recall without negatively impacting precision.
Based on this mapping rule, the following statement

```
dbp:Father_Hendricks rdf:type dbp:Roman .
```

is replaced by

```
dbp:Father_Hendricks rdf:type dbp:School .
```

- For some hypernyms, the hypernym linking algorithm produces an incorrect disambiguation.

Example. The `dbp:Body` carries the "physical body of an individual" meaning, while it appears almost exclusively in the "group of people" sense. This is corrected by mapping rule: "`dbp:Body` → `dbp:Organisation`".
Based on this mapping rule, the following statement

```
dbp:National_Executive_Committee rdf:type dbp:Body.
```

is replaced by

```
dbp:National_Executive_Committee rdf:type
dbp:Organization .
```

Deletion rules were introduced to remove all entities with a "black-listed" hypernym. Again, there were two reasons to blacklist a hypernym:

- The linked hypernym is too ambiguous with little information value. Example: `dbp:Utility` or `dbp:Family`.
- The linked hypernym cannot be disambiguated to a single concept that would hold for the majority of its instances.

Example.
Consider `dbp:Agent`, which either denotes an organization or a chemical compound. Since none of the senses is strongly dominating, a deletion rule for statements with this concept as a hypernym was introduced.
Based on this mapping rule, the following statements were deleted (among others):

```
dbp:Metoclopramide rdf:type dbp:Agent.
dbp:US_Airline_Pilots_Association rdf:type dbp:Agent.
```

In the current release, these rules have global validity, i.e. it is not possible to specify a context in which they apply.

The resulting *Linked Hypernyms Dataset* is published using the N-Triples notation [19]. The *"Plain text" Hypernyms Dataset* is made available in one article–hypernym tuple per line format. A separate file is downloadable for each language. The number of records in the Linked Hypernyms Dataset is about 10%–20% (depending on the language—ref. to Table 2) smaller than for the "Plain text" Hypernyms Dataset, which is in part caused by the application of the deletion rules.

5. DBpedia and YAGO alignment

The results of hypernym linking, described in the previous section, are DBpedia URIs that are not well connected to the LOD cloud. The linked hypernyms are URIs from the (http://dbpedia.org/resource/) namespace (`dbp:` prefix), which is used in DBpedia to identify entities. Each DBpedia resource can be mapped to a Wikipedia article using the following naming scheme:
`http://dbpedia.org/resource/`*`Name`* corresponds to `http://en.wikipedia.org/wiki/`*`Name`* (similarly for other languages). While there are other knowledge bases that use entities from the `dbp:` namespace as types (cf. Tipalo in Section 2), it is preferred to use as types concepts from the DBpedia Ontology. These concepts reside in the http://dbpedia.org/ontology/ namespace (`dbo:` prefix).

This section describes the alignment of the LHD types from the `dbp:` namespace to the DBpedia ontology (version 3.8 containing 359 classes). This ontology is particularly suitable for two reasons: it facilitates the use of the Linked Hypernyms Dataset for DBpedia enrichment, and the fact that many concepts in the ontology have names of one or a few word length simplifies the alignment process, since the THD generated linked-hypernyms are concepts with a short name consisting mostly of one word. For DBpedia ontology alignment, a conservative string-based approach is adopted, which requires complete match with the class name. Complementary set of mappings was generated using a substring match with a follow-up manual verification.

In the second step alignment with the version 2s of the YAGO ontology [2] was performed. YAGO2s does not only contain complementary facts to DBpedia, but with 450.000 concepts in the taxonomy it provides much wider possibilities for matching with the linked hypernyms than the DBpedia Ontology. Again, a simple string-based ontology alignment algorithm was used. The substantially higher number of classes in YAGO resulted in a higher number of mappings. For this reason, the manual verification of the approximate mappings was not performed. It should be noted that this has no effect on the quality of the dataset, since the YAGO mapping was performed only to identify the RDF type triples which are novel w.r.t. to DBPedia *and* YAGO and to gather the corresponding statistics. Types from the YAGO ontology are not used in LHD.

Please cite this article in press as: T. Kliegr, Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery, Web Semantics: Science, Services and Agents on the World Wide Web (2014), http://dx.doi.org/10.1016/j.websem.2014.11.001

5.1. Alignment with the DBpedia ontology

The alignment with DBpedia is performed using the "exact match" algorithm in order to ensure the highest reliability. For each RDF type triple in LHD, the algorithm tries to find a DBpedia Ontology class for the object (the hypernym) based on a complete textual match. If such a match is successful, the object of the statement is replaced by the DBpedia Ontology class.

Example.
The output of the disambiguation phase is the following statement:

```
dbp:Karel_Čapek rdf:type dbp:Writer .
```

Since for "Writer" there is a class in DBpedia Ontology, this statement is replaced with:

```
dbp:Karel_Čapek rdf:type dbo:Writer .
```

The new statement is better interconnected in the LOD cloud.

If no concept with a fully matching name[11] is found, an approximate match is attempted in order to improve the interconnectedness.

Approximate matching returns the DBpedia Ontology concept which ends with the linked hypernym as substring. In case of multiple matches, the one with longest match is selected. Arbitrary selection is made in case of a tie. The result of this process is a set of candidate subclass relations between linked hypernyms and the DBpedia ontology concepts. Since there are only 359 classes in the DBpedia 3.8 ontology, there were 600 mapping candidates for English,[12] it was possible to perform manual verification. Based on the result, the type was either marked as confirmed, a mapping/deletion rule was created, or no action taken indicating that the mapping is incorrect. After the manual processing of the results, the algorithm was re-executed excluding the confirmed mappings.

Example.
Some of the mappings reviewed included:
1) '`dbp:Township` → `dbo:Ship`',
2) '`dbp:Warship` → `dbo:Ship`',
3) '`dbp:Planets` → `dbo:Planet`',
4) '`dbp:Bicyclist` → `dbo:Cyclist`'.
Except for the first mapping, all were confirmed.

It should be emphasized that all mappings identified based on approximate matching are serialized as extra RDF type triples, preserving the original statements.

Example.
For the "HMS Prince Albert (1984)" entity mentioned earlier, LHD contains both the original specific type, a DBpedia resource, and a universal mapping of this type to its superclass in the DBpedia Ontology:

```
dbp:HMS_Prince_Albert_(1864) rdf:type dbp:Warship
dbp:Warship rdfs:subClassOf dbo:Ship
```

The results of this phase are:

- replacements in LHD in case of an exact match,
- mapping file for confirmed approximate matches,
- mapping file with unconfirmed approximate matches.

[11] The stemmed substring after the last "/" in the URI, and `rdfs:label` are considered as concept name.

[12] It follows from the type of the matching algorithm employed that the space of mapping candidates is restricted to linked hypernyms that have one of the classes from the DBpedia Ontology as a substring (excluding exact match).

5.2. Alignment with the YAGO ontology

While the primary goal of the DBpedia Ontology alignment is to use the better connected concepts from the DBpedia Ontology namespace instead of DBpedia resources as linked hypernyms, the purpose of YAGO alignment is to detect facts (RDF type triples) in the Linked Hypernyms Dataset that are confirmed by YAGO2s.

Overlap with YAGO2s[13] was checked only for a portion of entity-hypernym tuples with high confidence, which passed the novelty check against DBpedia. These are three partitions commonly denoted in Table 3 as *DBpedia Enrichment Dataset*. Each entity in the dataset was assigned to one of the four categories (listed in the order of priority):

- **YAGO No Type**, entity is not assigned any YAGO2s type,
- **YAGO Exact**, a perfect match between the linked hypernym and YAGO2s type assigned to the entity was found,
- **YAGO Approximate**, a YAGO2s type assigned to the entity containing the linked hypernym as a substring was found,
- **YAGO No Match**, none of the above applies.

To perform the comparison, a transitive closure of YAGO2s ontology types was used. The number of RDF type triples falling into the individual partitions is reported in Table 4.

Example.
Consider statement:

```
dbp:H._R._Cox rdf:type dbp:Bacteriologist .
```

The DBpedia Ontology 3.8 does not contain a class for bacteriologist, which places this statement (after other preconditions discussed in section 6.5 have been tested) to the *DBpedia Enrichment Dataset* partition *Not mapped/New*. YAGO assigns this entity multiple classes,[a] but none of these or their superclasses have "bacteriologist" as a substring. This places the statement into the *YAGO No Match* partition of *Not mapped/New* in Table 4.

[a] `wikicategory_American_microbiologists`, `wikicategory_Indiana_State_University_alumni`

6. Partitions of the dataset

LHD is divided into several partitions according to the ontology alignment results and redundancy of RDF type triples with respect to DBpedia 3.8 Ontology and the DBpedia 3.8 instance file, which contains statements assigning DBpedia instances to DBpedia Ontology classes. The individual partitions are described in the remainder of this section. Table 3 gives the essential statistics on each partition.

6.1. Mapped/classes

This partition contains statements, where the entity (the subject) is found to be used as a hypernym (object) in another LHD statement. The entity does not have any DBpedia Ontology type assigned in the DBpedia instance file.

Example.

```
dbp:Llama rdfs:subClassOf dbp:Camelid .
```

It should be noted that compared to partition "Notmapped /Spurious Entity" (Section 6.7), there is no contradicting evidence for `dbp:Llama` to be a class. As a result, this partition uses the `rdfs:subClassOf` relation.

[13] The latest release as of submission.

Please cite this article in press as: T. Kliegr, Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery, Web Semantics: Science, Services and Agents on the World Wide Web (2014), http://dx.doi.org/10.1016/j.websem.2014.11.001

Table 3
LHD subdatasets.

Dataset	Mapped classes	Mapped existing	Notmapped probable overlap	Mapped new—no overlap	Notmapped new	Mapped new—no type	Notmapped spurious entity	Notmapped spurious hypernym
				DBpedia enrichment dataset				
Relation	Subclass	Type	type	type	type	Type	Type	Type
Entries (EN)	4043	217,416	5330	126,032	736,293	198,040	1149	20,850
Entries (DE)	2854	50,539	622	58,765	586,419	125,013	59	3,692
Entries (NL)	1304	15,392	235	16,884	563,485	67,990	0	57
Accuracy (EN)				0.82	0.83	0.94		

Table 4
Partitions of the DBpedia Enrichment Dataset (English) according to overlap with YAGO2s. The accuracy of plain text hypernyms is marked with †, the accuracy of linked hypernyms with ‡.

Partition according to DBpedia alignment result	Subpartitions according to YAGO Ontology alignment result									
	No type		Exact		No match		Approx.		All	
	size	acc	size	acc	size	acc	size	acc	size	acc
Mapped/New—No Overlap	9,699	0.98†	59,365	1.00†	35,775	0.95†	21,193	NA	126,032	0.97†
		0.91‡		0.99‡		0.90‡				0.82‡
Not mapped/New	150,333	0.89†	199,916	1.00†	295,217	0.93†	90,827	NA	736,293	0.93†
		0.81‡		0.86‡		0.77‡				0.83‡
Mapped/New—No Type	38,258	0.95†	74,503	1.00†	72,745	0.98†	12,534	NA	198,040	0.97†
		0.87‡		0.95‡		0.94‡				0.94‡
all	198,290	0.91†	333,784	1.00†	403,737	0.95†	NA	NA	1,060,365	0.94†
		0.83‡		0.90‡		0.90‡				0.85‡

Most, but not all, of the statements have type from the dbp namespace.

6.2. Mapped/existing

This partition contains statements, where the entity was not found to be used as a hypernym in another LHD statement. The entity does have a DBpedia Ontology type assigned in the DBpedia 3.8 instance file. The type assigned by LHD was successfully mapped to a DBpedia Ontology class. Consequently, it was found out that the same statement already exists in the DBpedia instance file.

Example.
`dbp:Czech_Republic rdf:type dbo:Country .`
Identical statement to the above LHD triple is already contained in the DBpedia instance file.

6.3. Notmapped/probable overlap

This partition contains statements, where the entity was not found to be used as hypernym in another LHD statement. The entity does have a DBpedia Ontology type assigned in DBpedia instance file. The type assigned by LHD was *not* mapped to a DBpedia Ontology class, however, it was found out that a similar statement already exists in the DBpedia instance file.

Example.
`dbp:Boston_Cyberarts_Festival rdf:type dbp:Festival .`
The DBpedia 3.8 ontology does not contain a class that would have "festival" as a substring, therefore the mapping failed and the type is represented with a DBpedia resource. However, the instance `dbp:Boston_Cyberarts_Festival` is assigned type `schema.org/Festival` in the DBpedia 3.8 instance file. Since there is a textual match between concept names of the LHD and Schema.org types, this triple is classified as a probable overlap.

All statements have type from the dbp namespace.

6.4. Mapped/new—no overlap

This partition contains statements, where the entity was not found to be used as hypernym in another LHD statement. The type assigned by LHD was mapped to a DBpedia Ontology class, however, it was found out that while the DBpedia 3.8 instance file assigns at least one DBpedia Ontology type to this entity, none of the assigned types matches the LHD type.

Example.
`dbp:Karel_Čapek rdf:type dbo:Writer .`
The `dbp:Karel_Čapek` entity has already multiple types in the DBpedia 3.8 instance file, with the most specific type being `dbo:Person`. The type assigned by LHD is new with respect to this list.

It should be noted that this partition contains also statements, whose type can be mapped to the DBpedia Ontology via the approximate mappings (cf. Section 5.1).

Example.
`dbp:HMS_Prince_Albert_(1864) rdf:type dbp:Warship .`

About 89% of the statements in the English dataset have type from the dbo namespace and the rest from the dbp namespace (these are mapped via the approximate mappings).

6.5. Not mapped/new

This partition contains statements, where the entity was not found to be used as hypernym in another LHD statement. The type assigned by LHD was not mapped to a DBpedia Ontology class.

Example.
`dbp:H._R._Cox rdf:type dbp:Bacteriologist .`

This partition contains typically statements with a specific type that is not covered by the DBpedia Ontology. All statements have type from the dbp namespace.

Please cite this article in press as: T. Kliegr, Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery, Web Semantics: Science, Services and Agents on the World Wide Web (2014), http://dx.doi.org/10.1016/j.websem.2014.11.001

6.6. Mapped/new—no type

The entity was not found to be used as hypernym in another LHD statement. The type assigned by LHD is mapped to a DBpedia Ontology class. The entity is not assigned any DBpedia Ontology type in the DBpedia 3.8 instance file. As a consequence, the type assigned by LHD must be new.

Example.

```
dbp:Vostok_programme rdf:type dbo:Project .
```

The `dbp:Vostok_programme` entity does not have any entry in the DBpedia 3.8 instance file.

About 93% of the statements in the English dataset have type from the `dbo` namespace and the rest from the `dbp` namespace (these are mapped via the approximate mappings).

6.7. Notmapped/spurious entity

This partition contains statements, where the entity (the subject) is found to be used as a hypernym (object) in another LHD statement and at the same time the entity has a DBpedia Ontology type assigned in the DBpedia 3.8 instance file.

Example.

```
dbp:Coffee rdf:type dbp:Beverage .
```

The subject is used as a hypernym (class) because it is used in LHD statements such as:

```
dbp:Organic_coffee rdf:type dbp:Coffee .
```

At the same time DBpedia contains statements that use `dbp:Coffee` as an instance:

```
dbp:Coffee rdf:type dbo:Food .
```

This contradicting evidence places the statement into the spurious category.

While using the same concept both as instance and class is possible through the OWL 2 *punning* construct, the purpose of this and the following LHD partitions is to isolate such possibly dubious statements for further validation.

6.8. Notmapped/spurious hypernym

The hypernym is used as an instance in a statement in the DBpedia 3.8 instance file.

Example.

```
dbp:Aspartate_transaminase rdf:type dbp:Phosphate .
```

The `dbp:Phosphate` concept is already assigned a type in the DBpedia instance file:

```
dbp:Phosphate rdf:type dbp:ChemicalCompound .
```

The fact that `dbp:Phosphate` is used as an instance in DBpedia renders suspicious the extracted LHD statements, which use it as a class.

7. Evaluation

This section presents experimental results that demonstrate the coverage as well as the quality of the datasets. Evaluation of the hypernym discovery algorithm is covered in Section 7.1 and of the disambiguation algorithm in Section 7.2. The assessment of the final Linked Hypernyms Dataset is reported in three subsections. Section 7.3 introduces the evaluation methodology and presents the overall accuracy. Accuracy of the entity-linked hypernym pairs novel w.r.t. existing knowledge bases is examined in Section 7.4 and the accuracy of the rediscovered (redundant) pairs in Section 7.5.

7.1. Hypernym discovery

The quality of the hypernym discovery was evaluated on three manually tagged corpora (English, German, Dutch) with the GATE framework (http://gate.ac.uk).

Using the random article functionality from the Wikipedia search API, 500 articles for each language were selected. Corpus containing the articles' first sentences was created for each of the languages. The first sentences were extracted automatically using a customized GATE Regex Sentence Splitter plugin with negligible error. Lists, disambiguation articles and redirects were skipped along with empty articles or articles with failed first sentence extraction.

For the English corpus, the first appearance of a hypernym in each of the documents was independently annotated by three annotators with the help of the Google Translate service. The annotators were students with good command of English, elementary German and no knowledge of Dutch. The groundtruth was established by the consensus of two annotators. For German and Dutch, all documents were annotated by two annotators, when there was no consensus, an annotation by the third annotator was provided. To compare with previous work [13], a focused dataset consisting of documents describing people was manually created from the German dataset. It should be noted that the documents used for evaluation were unseen during the grammar development phase.

The GATE Corpus Quality Assurance tool was used to compute precision and recall of the computer generated annotations with human ground-truth. The results are summarized in Table 5. For computing the metrics, partially correct (overlapping) annotations were considered as incorrect. It can be seen that the results are quite consistent, with precision exceeding 0.90 for all languages. The best results were obtained for the German person subset, with precision 0.98 and recall 0.95. This is on par with the 0.97 precision and 0.94 recall reported for lexico-syntactic patterns and the Syntactic–Semantic Tagger respectively, the best performing algorithms in [13]. A statistic significance test was not performed due to differences in annotation methodology: while [13] annotated all hypernyms in the input text, in experiments presented here only the first specific hypernym was annotated.[14] The results are almost identical to those obtained by the Tipalo algorithm [14] for the type selection subtask. This evaluation was performed on 100 English Wikipedia articles with 0.93 precision and 0.90 recall.

7.2. Disambiguation algorithm

Correctly identifying a hypernym is an essential step for linking the source entity to DBpedia. The credibility of the most frequent sense assumption made by the linking algorithm was evaluated on a set of 466 hypernym–document pairs. These were all groundtruth hypernyms in the English dataset introduced in Section 7.1.[15] The hypernyms were issued as queries to the Wikipedia search observing whether the first hit matches the semantics of the hypernym in the context of the original article.

Three raters have evaluated the results of this experiment. The consensus was determined based on a majority vote. The percentage of ratings in each category is presented in Table 6.

[14] Consider sentence: "Rhabditida is an order of free-living, zooparasitic and phytoparasitic microbivorous nematodes (roundworms)". The THD assigned hypernym "order" was considered incorrect, as the annotators agreed on "nematodes". Both "order" and "nematodes" are, however, valid hypernyms for Rhabditida.

[15] For 34 documents the groundtruth was "no hypernym".

Please cite this article in press as: T. Kliegr, Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery, Web Semantics: Science, Services and Agents on the World Wide Web (2014), http://dx.doi.org/10.1016/j.websem.2014.11.001

Table 5
Hypernym discovery results. In column labels, A refers to the human annotation, and B to computer-generated result.

Language	Docs	Docs with groundtruth	Match	Only A	Only B	Partially correct	Precision	Recall	F1.0
English	500	500	411	55	24	0	0.94	0.88	0.91
German	497	488	409	45	23	2	0.94	0.90	0.92
German-person	225	223	205	10	4	1	0.98	0.95	0.96
Dutch	500	495	428	45	34	3	0.92	0.90	0.91

Table 6
Evaluation of the disambiguation algorithm (consensus rating).

Language	Total docs	Docs with hypernym	Docs with consensus	Precise	Imprecise	Disambiguation page	Incorrect
English	500	466	464	69.4%	7.1%	21.1%	2.4%

Table 7
Inter-rater agreement (English), κ refers to Cohen's Kappa for two raters, and Agreem. to the number of matching ratings divided by the number of all ratings.

Metric	ann1 vs ann2		ann1 vs agr		ann2 vs agr	
	plain	linked	plain	linked	plain	linked
κ	0.702	0.667	**0.930**	**0.925**	0.767	0.743
Agreem.	0.973	0.925	**0.993**	**0.981**	0.980	0.944

Table 8
Overall accuracy.

	ann1		ann2		Agreement	
Dataset	Plain	Linked	Plain	Linked	Plain	Linked
Dutch	**0.93**	**0.88**	NA	NA	NA	NA
English	0.95	0.85	0.96	0.90	**0.95**	**0.86**
German	**0.95**	**0.77**	NA	NA	NA	NA

The results indicate that with only 2.4% incorrect type assignments the hypernym linking algorithm does not make many outright errors. However, 21% of articles is mapped to an ambiguate type (a disambiguation page), selecting a correct specific sense would thus be a valuable direction for future work.

7.3. Overall accuracy

This integrating experiment focused on evaluating the accuracy of entity-linked hypernym tuples in the Linked Hypernyms Dataset. In contrast to the setup of the separate evaluation of the disambiguation algorithm reported in Section 7.2, the input are the RDF type triples that have been subject to the data cleansing and DBpedia alignment. Also, the evaluation guidelines required the rater to assess the correctness of the triples also when the type (linked hypernym) is a disambiguation page. If any of the listed senses covers the entity, the linked hypernym is correct, otherwise it is marked as incorrect.

The sample size of 1000 allowed to report all results with the lower and upper limits of the 95% confidence interval within approximately 2%–3% from the average accuracy on the sample. The 2% span was also used to evaluate the type relation in YAGO2 [2]. For English, all entities were judged by two raters (students), when there was no consensus, judgments of the third rater (expert ontologist) were requested. The groundtruth was established by the consensus of two raters. The degree of agreement among the raters is provided by Table 7.

The results indicate almost perfect match between the judgments provided by rater 1 and the consensus judgments. For German and Dutch, the results are only based on the judgments of the best performing rater 1.

For each entity-linked hypernym pair the task was to assess whether the linked hypernym is a correct type for the entity. For linked hypernym pointing to a DBpedia ontology class, this was determined based on the description of the class, for DBpedia resources, based on the content of the associated Wikipedia page.

As a supplementary task, the rater(s) also assessed the correctness of the plain text hypernym.

The overall accuracy of the Linked Hypernyms Dataset as evaluated on 1000 randomly drawn entities per language is reported in Table 8. A direct comparison with Tipalo has not been attempted, since it uses a different reference type system (DULplus). The accuracy on the English dataset can be, however, compared with the YAGO2 ontology: the accuracy of linked hypernyms (*linked* in Table 8) is at 0.86 lower than the average accuracy of the type relation (0.98) reported for YAGO [2]. It should be noted that the accuracy of the plain text hypernyms (*plain* in Table 8) is in the range of 0.93–0.95 for all three languages. This shows that the error is mainly introduced by the disambiguation algorithm.

The following Sections 7.4 and 7.5 present additional evaluations on 11,350 entities from individual subsets of the English Linked Hypernyms Dataset using the same methodology, but only with one rater. The use of only one rater is justified by the high agreement with the inter-rater consensus in the English overall accuracy evaluation.

It should be noted that in the evaluations, the mappings to ontology classes resulting from approximate matching were not considered. This applies both to the evaluation of the overall accuracy as well as to the evaluation on the individual subsets performed in the following Sections 7.4 and 7.5. Also, the comparison of the results with YAGO2s in this section is only indicative, due to variations in the rating setup.

7.4. Accuracy of the DBpedia enrichment dataset

This experiment focused on evaluating the accuracy of statements that were found to be novel with respect to a) DBpedia, and b) DBpedia and YAGO2s.

As the DBpedia only baseline, all three parts of the *DBpedia Enrichment Dataset* are used: *"Mapped/New—No Overlap"*, *"Not Mapped/New"*, and *"Mapped/New—No Type"*. Each of these was further partitioned to four subsets according to YAGO2s overlap (see Table 4). For measuring the accuracy of entity-linked hypernym pairs novel w.r.t. YAGO2s, the partitions of the *DBpedia Enrichment Dataset* with either no YAGO2s type assigned or with no match against YAGO2s are used. Nine evaluations were performed, each on a random sample of 831–1000 entities from the respective dataset.

The results of the evaluation are provided in Table 3. The best performing dataset is – surprisingly – dataset *Mapped / New—No Type* which contains entities with no type assigned by DBpedia. While type extraction from the semistructured information used to populate the DBpedia type relation presumably failed for these 198,040 entities, THD provides a type with accuracy of 0.94. The weighted average accuracy for the *DBpedia Enrichment* dataset containing 1,060,365 entities is 0.85.

The total number of RDF type triples novel with respect to DBpedia and simultaneously with YAGO2s (*YAGO Enrichment* dataset)

Please cite this article in press as: T. Kliegr, Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery, Web Semantics: Science, Services and Agents on the World Wide Web (2014), http://dx.doi.org/10.1016/j.websem.2014.11.001

amounts to 602,027 (*YAGO No Type + YAGO No Match* partitions in Table 4). For the hardest subset, where neither DBpedia nor YAGO2s assign any type,[16] the accuracy is 0.87.

7.5. Accuracy of statements confirmed by YAGO

The subject of evaluation are subsets of the DBpedia Enrichment Datasets containing entities for which the linked hypernym does not match any DBpedia assigned type, but there is an exact match with a YAGO2s type. The number of entities in these subsets is 333,784, the average accuracy is 0.91. Three evaluations were performed, each on a random sample of 878–1000 entities from the respective dataset. The results for all three subsets are reported in bold in Table 4.

Interestingly, the *YAGO Exact Match* partition of *Mapped / New—No Overlap* exhibits accuracy of 0.994. For the entities in this dataset[17] the type is assigned with higher accuracy than is the 0.9768 average accuracy for the type relation reported for the YAGO ontology [2] (chi-square test with $p < 0.05$).

This nearly 2% improvement over YAGO indicates that the free-text modality can be successfully combined with the semistructured information in Wikipedia to obtain nearly 100% correct results. The second, and perhaps more important use for the rediscovered RDF type triples is the identification of *the most common type* as seen by the author(s) of the corresponding Wikipedia entry.

8. Extending coverage—LHD 2.0

Even after the ontology alignment, most RDF type statements in LHD have a DBpedia resource as a type, rather than a class from the DBpedia Ontology.

Increasing the number of entities aligned to the DBpedia Ontology is a subject of ongoing work. Alignment of the types for which the simple string matching solution failed to provide a mapping was attempted with state-of-the-art ontology alignment algorithms in [20]. Experiments were performed with LogMapLt, YAM++ and Falcon, all tools with a success record in the Ontology Alignment Evaluation Initiative.[18]

Best results were eventually obtained with a statistical type inference algorithm proposed specifically for this problem. Using this algorithm, the draft version 2.0 of LHD [20] maps more than 95% of entities in the English dataset to DBpedia Ontology classes. For German and Dutch the number of entities with a type from the `dbo` namespace is also increased significantly. It should be noted that this increase in coverage comes at a cost of reduced precision. LHD 2.0 draft is thus an extension, rather than a replacement for the version of the dataset presented in this paper.

Example.
The following statements from the "notmapped" partitions (cf. Sections 6.5 and 6.3):

```
dbp:H._R._Cox rdf:type dbp:Bacteriologist .
dbp:Boston_Cyberarts_Festival rdf:type dbp:Festival .
```

are supplemented in LHD 2.0 draft with:

```
dbp:H._R._Cox rdf:type dbo:Scientist .
dbp:Boston_Cyberarts_Festival rdf:type dbo:MusicFestival.
```

[16] Note that part of the discrepancy in entity coverage between the Linked Hypernyms Dataset, DBpedia and YAGO2s is due to Wikipedia snapshots used to populate the datasets being from different timepoints.

[17] Out of the total 59,365 entries, entities for evaluation were sampled from the 50,274 entities with type from the `dbo` namespace (entities with approximate mappings were excluded).

[18] http://oaei.ontologymatching.org/.

9. Extending LHD to other languages

Extending LHD to another language requires the availability of a part-of-speech tagger and a manually devised JAPE grammar adjusted to the tagset of the selected tagger as well as to the language.

The first precondition is fulfilled for most languages with many speakers. POS taggers for French, Italian and Russian, languages currently uncovered by LHD, are all available within the TreeTagger framework. For other languages there are third-party taggers that can be integrated. Next, manually devising a JAPE grammar requires some effort, first on creating a development set of articles with tagged hypernyms, and subsequently on tweaking the grammar to provide the optimum balance between precision and recall.

A viable option, which could lead to a fully automated solution, is generating a labeled set of articles by annotating as hypernyms noun phrases that match any of the types assigned in DBpedia, and subsequently using this set to train a hypernym tagger, e.g. as proposed in [13]. The hypernyms output by the tagger could be used in the same way as hypernyms identified by the hand-crafted JAPE grammars, leaving the rest of the LHD generation framework unaffected.

The LHD Generation framework has been made available under an open source license. The published framework differs in the workflow presented in this article in that it performs hypernym extraction from the article abstracts included in the DBpedia RDF n-triples dump (instead of the Wikipedia dump).

10. Conclusion

This paper introduced the Linked Hypernyms Dataset containing 2.8 million RDF type triples. Since the types were obtained from the free text of Wikipedia articles, the dataset is to a large extent complementary to DBpedia and YAGO ontologies, which are populated particularly based on the semistructured information—infoboxes and article categories.

The Linked Hypernyms Dataset generation framework adapts previously published algorithms and approaches, which were proposed for extracting hypernyms from electronic dictionaries and encyclopedic resources, and applies them on large scale on English, Dutch and German Wikipedias.

Using three annotators and 500 articles per language, the F1 measure for hypernym discovery was found to exceed 0.90 for all languages. The best results were obtained for the German person subset, with precision 0.98 and recall 0.95.

The disambiguation algorithm, which is used to link the hypernyms to DBpedia resources, was evaluated on 466 English article–hypernym pairs. This experiment pointed at the fact that while there was only 2.4% incorrect type assignments, 21% of the linked hypernyms are disambiguation entities (articles). Selecting the correct specific sense would be an interesting area of future work.

The third integrating evaluation assessed the cumulative performance of the entire pipeline generating the Linked Hypernyms Dataset: hypernym discovery, disambiguation, data cleansing and DBpedia ontology alignment. The human evaluation was reported separately for the entire English, German and Dutch datasets. The English dataset was subject to further analysis, with evaluation results reported for its twelve interesting partitions. Compared to existing work on DBpedia enrichment or hypernym learning (e.g. [13,14,16]), an order-of-magnitude more human judgments were elicited to assess the quality of the dataset.

Some of the results are as follows: The accuracy for the 1 million RDF type triples novel with respect to DBpedia is 0.85% ± 2%, out of these the highest accuracy (0.94) is for the subset of 198,040

Please cite this article in press as: T. Kliegr, Linked hypernyms: Enriching DBpedia with Targeted Hypernym Discovery, Web Semantics: Science, Services and Agents on the World Wide Web (2014), http://dx.doi.org/10.1016/j.websem.2014.11.001

entities, which have no DBpedia type. With accuracy 0.87, the Linked Hypernyms Dataset provides a new type for 38,000 entities that had previously no YAGO2s or DBpedia Ontology type.

There are about 770 thousand novel RDF type triples for the German dataset, and 650 thousand for the Dutch dataset. The number of these RDF type triples exceeds the number of entities in the localized DBpedia 3.8 for the respective language. Version of the YAGO2s ontology for localized Wikipedias is not provided.

In addition to enriching DBpedia and YAGO2s with new types, it was demonstrated that the part of the Linked Hypernyms Dataset which overlaps with YAGO2s or DBpedia can be utilized to obtain a set of RDF type triples with nearly 100% accuracy.

There is a body of possible future extensions both on the linked data and linguistic levels. A certain limitation of the Linked Hypernyms Dataset is that a large number of linked hypernyms is not mapped to the DBpedia *ontology*. In the draft 2.0 version of the dataset, a statistical ontology alignment algorithm has been used to achieve a close to 100% coverage with DBpedia Ontology classes [20], however, at the cost of lower precision. Another viable direction of future work is investigation of the supplementary information obtainable from Targeted Hypernym Discovery. For example, according to empirical observation, the first sentence of the article gives several hints regarding temporal validity of the statements. For people, the past tense of the verb in the first sentence indicates that the person is deceased, while the object in the Hearst pattern preceded with limited vocabulary of words like "former" or "retired" hints at the hypernym (presumably vocation) not being temporarily valid.

The datasets are released under the Creative Commons license and are available for download from http://ner.vse.cz/datasets/linkedhypernyms. The raw data (human and computer generated hypernyms) used for the experimental evaluation, the annotation results, ratings and guidelines are also available. The LHD 1.3.8 release described and evaluated in this article targets DBpedia 3.8, version for DBpedia 3.9 containing 4.5 million RDF type triples is also available for download. Updated LHD generation framework for DBpedia 3.9 is available under an open source license. An example of an application which uses LHD to complement DBpedia and YAGO is a web-based entity recognition and classification system http://entityclassifier.eu [21]. The German LHD partition has been incorporated into the German DBpedia by the German DBpedia chapter to improve coverage with RDF Type triples.[19]

Acknowledgments

The author thanks the anonymous reviewers for their insightful comments. This work has been supported by the EU LinkedTV project (FP7-287911). The author would like to thank Milan Dojchinovski for insightful comments, and, in the first place, programming entityclassifier.eu, the first application using LHD.

References

[1] C. Bizer, J. Lehmann, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, S. Hellmann, DBpedia—a crystallization point for the web of data, Web Semant. 7 (2009) 154–165.
[2] J. Hoffart, F.M. Suchanek, K. Berberich, G. Weikum, YAGO2: A spatially and temporally enhanced knowledge base from Wikipedia, Artificial Intelligence 194 (2013) 28–61.
[3] Y.A. Wilks, B.M. Slator, L. Guthrie, Electric Words: Dictionaries, Computers, and Meanings, ACL-MIT Series in Natural Language Processing, The MIT Press, Cambridge, Mass., 1996.
[4] N. Calzolari, Detecting patterns in a lexical data base, in: Proceedings of the 10th International Conference on Computational Linguistics and 22Nd Annual Meeting on Association for Computational Linguistics, ACL'84, Association for Computational Linguistics, Stroudsburg, PA, USA, 1984, pp. 170–173. http://dx.doi.org/10.3115/980491.980527.
[5] M.A. Hearst, Automatic acquisition of hyponyms from large text corpora, in: Proceedings of the 14th Conference on Computational Linguistics, Vol. 2, COLING'92, ACL, Stroudsburg, PA, USA, 1992, pp. 539–545. http://dx.doi.org/10.3115/992133.992154.
[6] R. Snow, D. Jurafsky, A.Y. Ng, Learning syntactic patterns for automatic hypernym discovery, in: Advances in Neural Information Processing Systems, 17, MIT Press, Cambridge, MA, 2005, pp. 1297–1304.
[7] R. Snow, D. Jurafsky, A.Y. Ng, Semantic taxonomy induction from heterogenous evidence, in: Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics, ACL-44, Association for Computational Linguistics, Stroudsburg, PA, USA, 2006, pp. 801–808. http://dx.doi.org/10.3115/1220175.1220276.
[8] P. Cimiano, J. Völker, Text2onto: a framework for ontology learning and data-driven change discovery, in: Proceedings of the 10th International Conference on Natural Language Processing and Information Systems, NLDB'05, Springer-Verlag, Berlin, Heidelberg, 2005, pp. 227–238. http://dx.doi.org/10.1007/11428817_21.
[9] S.P. Ponzetto, M. Strube, Deriving a large scale taxonomy from wikipedia, in: Proceedings of the 22nd National Conference on Artificial Intelligence, in: AAAI'07, vol. 2, AAAI Press, 2007, pp. 1440–1445. URL: http://dl.acm.org/citation.cfm?id=1619797.1619876.
[10] J. Kazama, K. Torisawa, Exploiting Wikipedia as external knowledge for named entity recognition, in: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, EMNLP-CoNLL'07, 2007, pp. 698–707.
[11] A. Ritter, S. Soderland, O. Etzioni, What is this, anyway: Automatic hypernym discovery, in: Proceedings of AAAI-09 Spring Symposium On Learning, 2009, pp. 88–93.
[12] T. Kliegr, K. Chandramouli, J. Nemrava, V. Svátek, E. Izquierdo, Combining image captions and visual analysis for image concept classification, in: Proceedings of the 9th International Workshop on Multimedia Data Mining: Held in Conjunction With the ACM SIGKDD 2008, MDM'08, ACM, New York, NY, USA, 2008, pp. 8–17. http://dx.doi.org/10.1145/1509212.1509214.
[13] B. Litz, H. Langer, R. Malaka, Sequential supervised learning for hypernym discovery from Wikipedia, in: A. Fred, J.L.G. Dietz, K. Liu, J. Filipe (Eds.), Knowledge Discovery, Knowlege Engineering and Knowledge Management, in: Communications in Computer and Information Science, vol. 128, Springer-Verlag, Berlin, Heidelberg, 2011, pp. 68–80.
[14] A. Gangemi, A.G. Nuzzolese, V. Presutti, F. Draicchio, A. Musetti, P. Ciancarini, Automatic typing of DBpedia entities, in: P. Cudre-Mauroux, J. Heflin, E. Sirin, T. Tudorache, J. Euzenat, M. Hauswirth, J. X. Parreira, J. Hendler, G. Schreiber, A. Bernstein, E. Blomqvist (Eds.), The Semantic Web—ISWC 2012, in: Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2012, pp. 65–81. http://dx.doi.org/10.1007/978-3-642-35176-1_5.
[15] A. Gangemi, A comparison of knowledge extraction tools for the semantic web, in: ESWC, 2013, pp. 351–366.
[16] H. Paulheim, Browsing linked open data with auto complete, in: Proceedings of the Semantic Web Challenge co-located with ISWC2012, Univ., Mannheim, Boston, US, 2012.
[17] A.P. Aprosio, C. Giuliano, A. Lavelli, Automatic expansion of DBpedia exploiting Wikipedia cross-language information, in: The Semantic Web: Semantics and Big Data, 10th International Conference, ESWC 2013, Montpellier, France, May 26–30, 2013. Proceedings, 2013, pp. 397–411. http://dx.doi.org/10.1007/978-3-642-38288-8_27.
[18] M. Dojchinovski, T. Kliegr, I. Lašek, O. Zamazal, Wikipedia search as effective entity linking algorithm, in: Proceedings of the Sixth Text Analysis Conference, TAC'13, NIST, 2013.
[19] RDF Core working group, N-triples: W3C RDF Core WG internal working draft, 2001. http://www.w3.org/2001/sw/RDFCore/ntriples/.
[20] T. Kliegr, O. 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-2014), Reykjavik, Iceland, May 26–31, 2014, 2014, pp. 3517–3523. URL: http://www.lrec-conf.org/proceedings/lrec2014/summaries/703.html.
[21] M. Dojchinovski, T. Kliegr, Entityclassifier.eu: real-time classification of entities in text with Wikipedia, in: ECML'13, Springer, 2013, pp. 654–658.

19 http://de.dbpedia.org/node/30.