# Language-agnostic Relation Extraction from Wikipedia Abstracts for Knowledge Graph Extension

Nicolas Heist and Heiko Paulheim

*University of Mannheim, Data and Web Science Group, B6 26, 68159 Mannheim, Germany*

**Abstract.** Large-scale knowledge graphs, such as DBpedia, Wikidata, or YAGO, can be enhanced by relation extraction from text, using the data in the knowledge graph as training data, i.e., using *distant supervision*. While most existing approaches use language-specific methods (usually for English), we present a language-agnostic approach that exploits background knowledge from the graph instead of language-specific techniques and builds machine learning models only from language-independent features. We demonstrate the extraction of relations from Wikipedia abstracts, using the twelve largest language editions of Wikipedia. From those, we can extract 1.6M new relations in DBpedia at a level of precision of 95%, using a RandomForest classifier trained only on language-independent features. We furthermore investigate the similarity of models for different languages and show an exemplary geographical breakdown of the information extracted.

Keywords: Relation Extraction, Knowledge Graph Extension, Wikipedia, DBpedia

## 1. Introduction

Large-scale knowledge graphs, like DBpedia [18], Freebase [3], Wikidata [33], or YAGO [19], are usually built using heuristic extraction methods, e.g., from Wikipedia infoboxes, by exploiting crowd-sourcing processes, or both. These approaches can help creating large-scale public cross-domain knowledge graphs, but are prone both to errors as well as incompleteness.

Therefore, over the last years, various methods for refining those knowledge graphs have been proposed [24]. For filling missing relations (e.g., the missing birthplace of a person), *relation extraction* methods are proposed. Those can be applied to fill in relations for entities derived from Wikipedia pages without or with only sparsely filled infoboxes.

Most methods for relation extraction work on text and thus usually have at least one component which is explicitly specific for the language at hand (e.g., stemming, POS tagging, dependency parsing), like, e.g., [9,28,38], or implicitly exploits some characteristics of that language [2]. Thus, adapting those methods to work with texts in different natural languages is usally not a straight forward process.

In this paper, we propose a language-agnostic approach. Instead of knowledge about the language, we take background knowledge from the DBpedia knowledge graph into account. With that, we try to discover certain patterns in how Wikipedia abstracts are written. For example, in many cases, any genre mentioned in the abstract about a band is usually a genre of that band, the first city mentioned in an abstract about a person is that person's birthplace, and so on. In that case, the linguistic assumptions that we make about a language at hand are quite minimal. In fact, we only assume that for each language edition of Wikipedia, there are certain ways to structure an abstract of a given type of entity, in terms of what aspect is mentioned where (e.g., the birth place is the first place mentioned when talking about a person). Thus, the approach can be considered as language-independent (see [2] for an in-depth discussion).

The choice for Wikipedia abstracts as a corpus mitigates one of the common sources of errors in the re-

lation extraction process, i.e., the entity linking. Since Wikipedia articles can be unambiguously related to an instance in a knowledge base, and Wikipedia page links contained in Wikipedia abstracts are mostly free from noise, the corpus at hand can be directly exploited for relation extraction without the need for an upfront potentially noisy entity linking step.

By applying the exact same pipeline without any modifications to the twelve largest languages of Wikipedia, which encompass languages from different language families, we demonstrate that such patterns can be extracted from Wikipedia abstracts in arbitrary languages. We show that it is possible to extract valuable information by combining the information extracted from different languages, and investigate whether such patterns are similar across languages.

The rest of this paper is structured as follows. In section 2, we review related work. We introduce our approach in section 3, and discuss various experiments in section 4. We conclude with a summary and an outlook on future work.

## 2. Related Work

Various approaches have been proposed for relation extraction from text, in particular from Wikipedia. In this paper, we particularly deal with *closed* relation extraction, i.e., extracting new instantiations for relations that are defined a priori (by considering the schema of the knowledge graph at hand, or the set of relations contained therein).

Using the categorization introduced in [24], the approach proposed in this paper is an *external* one, as it uses Wikipedia as an external resource in addition to the knowledge graph itself. While internal approaches for relation prediction in knowledge graphs exist as well, using, e.g., association rule mining [13,14], tensor factorization [15,29], or graph embeddings [41], we restrict ourselves to comparing the proposed approach to other external approaches.

Most of the approaches in the literature make more or less heavy use of language-specific techniques. Distant supervision is proposed by [21] as a means to relation extraction for Freebase from Wikipedia texts. The approach uses a mixture of lexical and syntactic features, where the latter are highly language-specific. A similar approach is proposed for DBpedia in [1]. Like the Freebase-centric approach, it uses quite a few language-specific techniques, such as POS tagging and lemmatization. While those two approaches use Wiki-

pedia as a corpus, [12] compare that corpus to a corpus of news texts, showing that the usage of Wikipedia leads to higher quality results.

[22] introduce an approach for mining relations from Wikipedia articles which exploits similarities of dependency trees for extracting new relation instances. In [37], the similarity of dependency trees is also exploited for clustering pairs of concepts with similar dependency trees. The construction of those dependency trees is highly language specific, and consequently, both approaches are evaluated on the English Wikipedia only.

An approach closely related to the one discussed in this paper is *iPopulator* [17], which uses Conditional Random Fields to extract patterns for infobox values in Wikipedia abstracts. Similarly, *Kylin* [36] uses Conditional Random Fields to extract relations from Wikipedia articles and general Web pages. Similarly to the approach proposed in this paper, *PORE* [34] uses information on neighboring entities in a sentence to train a support vector machine classifier for the extraction of four different relations. Although those approaches could be transferred to multi-lingual settings, the papers only report results for English language texts.

In the recent past, some approaches based on deep learning have been proposed which are reported to or would in theory also work on multi-lingual text [23,32,39,40]. They have the advantages that (a) they can compensate for shortcomings in the entity linking step when using arbitrary text and (b) that explicit linguistic feature engineering is replaced by implicit feature construction in deep neural networks. In contrast to those works, we work with a specific set of texts, i.e., Wikipedia abstracts. Here, we can assume that the entity linking is mostly free from noise (albeit not complete), and directly exploit knowledge from the knowledge graph at hand, i.e., in our case, DBpedia.

In contrast to most of those works, the approach discussed in this paper works on Wikipedia abstracts in *arbitrary* languages, which we demonstrate in an evaluation using the twelve largest language editions of Wikipedia. To the best of our knowledge, while most of the approaches discussed above are only evaluated on one or at maximum two languages, this is the work to be evaluated on a larger variety of languages.

## 3. Approach

Our aim is to identify and exploit typical patterns in Wikipedia abstracts. As a running example, we use the

*genre* relation which may hold between a music artist and a music genre. Fig. 1 depicts this example with both an English and a French Wikipedia abstract. As our aim is to mine relations for the canonical DBpedia, extracted from the (largest) English language Wikipedia, we inspect all links in the abstract which have a corresponding entity in the main DBpedia knowledge base created from the English Wikipedia.[1] For other languages, we take one intermediate step via the interlanguage links in Wikipedia, which are extracted as a part of DBpedia [18].

### 3.1. Overall Approach

For 395 relations that can hold between entities, the ontology underlying the DBpedia knowledge graph[2] defines an explicit domain and range, i.e., the types of objects that are allowed in the subject and object position of this relation.[3] Each Wikipedia page also maps to an entity in the DBpedia knowledge graph, some of which are typed. We consider a pair of a Wikipedia page $p_0$ and a Wikipedia page $p_1$ linked from the abstract of $p_0$ as a *candidate* for a relation $R$ if the corresponding DBpedia entities $e_0$ and $e_1$ have types that are equal to the domain and range of $R$. In that case, $R(e_0, e_1)$ is considered a candidate axiom to be included in the DBpedia knowledge graph. In the example in Fig. 1, given that the *genre* relation holds between musical artists and genres, and the involved entities are of the types, one candidate each is generated from both the English and the French DBpedia.[4]

We expect that candidates contain a lot of false positives. For example, for the *birthplace* relation holding between a person and a city, all cities linked from the person's web page would be considered candidates. However, cities may be referred to for various different reasons in an abstract about a person (e.g., they may be their death place, the city of their alma mater,

etc.). Thus, we require additional evidence to decide whether a candidate actually represents a valid instantiation of a relation.

For taking that decision, we train a *machine learning model*. For each abstract of a page for which a given relation is present in the knowledge base, we use the *partial completeness assumption* [10] or *local closed world assumption* [7], i.e., we consider the relation to be complete. Hence, all candidates for the relation created from the abstract which are contained in the knowledge base are considered as *positive* training examples, all those which are not contained are considered as *negative* training examples. In the example in Fig. 1, *Industrial Rock* would be considered a positive example for the relation *genre*, whereas the genre *Rock*, if it were linked in the abstract, would be considered a negative example, since it is not linked as a genre in the DBpedia knowledge graph.

### 3.2. Feature Engineering

For training a classifier, both positive and negative examples need to be described by *features*. Table 2 sums up the features used by the classifiers proposed in this paper.

We use features related to the actual candidates found in the abstract (i.e., entities whose type matches the range of the relation at hand), i.e., the total number of candidates in the abstract (F00) and the candidate's sentence (F01), the position of the candidate w.r.t. all other candidates in the abstract (F02) and the candidate's sentence (F03), as well as the position of the candidate's sentence in the abstract (F07). The same is done for all entities, be it candidates or not (F04,F05,F06). Since all of those measures yield positive integers, they are normalized to $(0, 1]$ by using their inverse.

Further features taken into account are the existence of a back link from the candidate's page to the abstract's page (F08), and the vector of all the candidate's types in the DBpedia ontology (FXX). Table 1 depicts the translated feature table for the French Wikipedia abstract depicted in Fig. 1. In this example, there are five candidates (i.e., entities of type `dbo:MusicGenre`, three of which are also contained in the DBpedia knowledge graph (i.e., they serve as true positives).

For the creation of those features which are dependent on the types, the types are taken from the canonical (i.e., English) DBpedia, using the interlanguage

---

[1]For this work, we use the 2014 version of DBpedia, which was the most recent release available at the time the experiments were conducted. This version is available at `http://oldwiki.dbpedia.org/Downloads2014`. All statements made in this paper about the size etc. of DBpedia correspond to that version.

[2]`http://dbpedia.org/services-resources/ontology`

[3]Note that the underlying OWL ontology distinguishes *object properties* that hold between entities, and *datatype properties* that hold between an entity and a literal value. Here, we only regard the former case.

[4]Prefixes used in this paper: `dbr=http://dbpedia.org/`, `dbf=http://fr.dbpedia.org/`, `dbo=http://dbpedia.org/ontology/`

Nine Inch Nails (abbreviated NIN; stylized as NIИ) is an American <u>industrial rock</u> band, founded in 1988 by <u>Trent Reznor</u> in <u>Cleveland</u>, <u>Ohio</u>. As its main <u>producer</u>, singer, songwriter, and instrumentalist, Reznor is the only official member of the group and remains solely responsible for its direction. Nine Inch Nails' music straddles a wide range of genres. After recording a new album, Reznor usually assembles a <u>live band</u> to perform with him. The touring band features a revolving lineup that often rearranges songs to fit a live setting. On stage, Nine Inch Nails often employs visual elements to accompany performances, which frequently include light shows.

*English Wikipedia article*

*DBpedia*

*Interlanguage Links*

Nine Inch Nails (NIN ou parfois stylisé NIИ) ~~est un groupe de metal industriel~~ américain fondé en 1988 par <u>Trent Reznor</u> à ~~Cleveland~~. Étant le principal <u>producteur</u>, chanteur, compositeur et instrumentiste, Reznor représente l'unique membre officiel de Nine Inch Nails et reste le seul responsable de sa direction. Principalement metal industriel, la musique de Nine Inch Nails propose un assemblage éclectique de genres comme le <u>rock alternatif</u>, l'<u>ambient</u>, l'<u>electronica</u> et la <u>synthpop</u>.
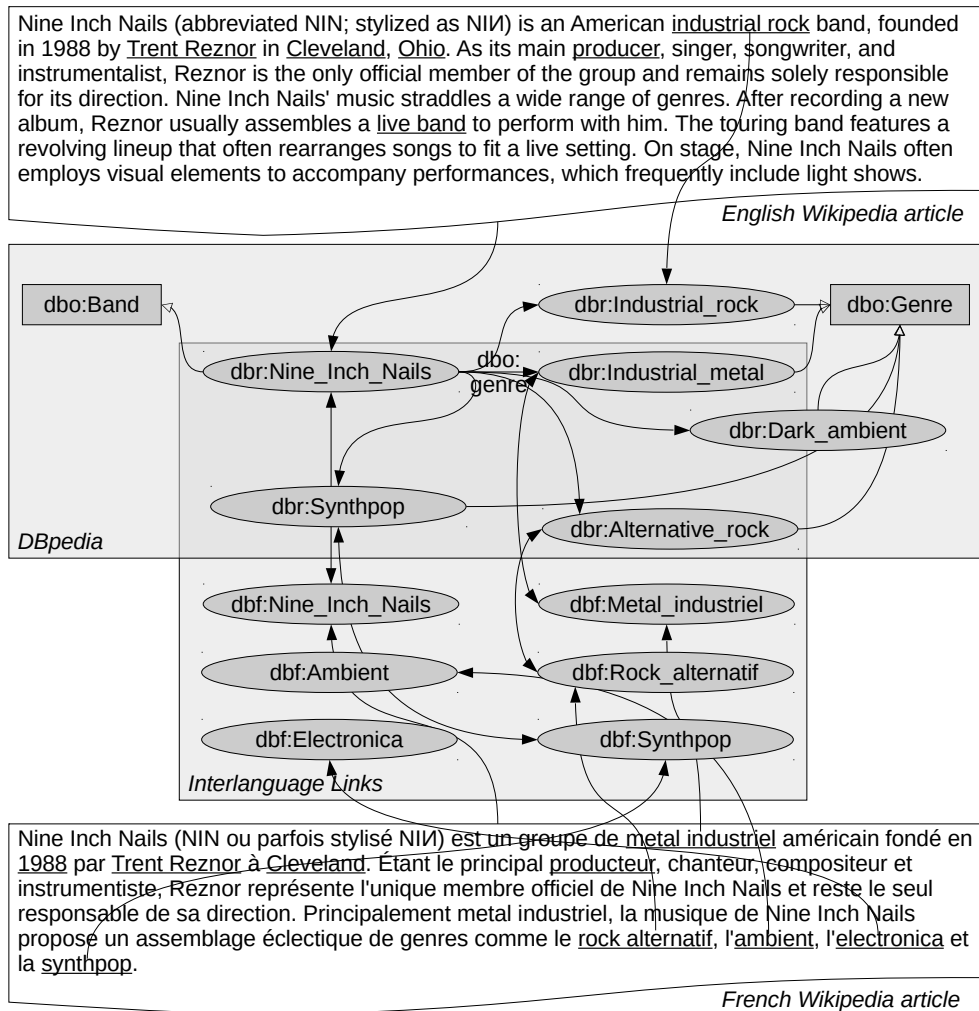
*French Wikipedia article*

Fig. 1. Approach illustrated with extraction from English (above) and French (below) Wikipedia article.

Table 1

Example Feature Representation

| Instance | F00 | F01 | F02 | F03 | F04 | F05 | F06 | F07 | F08 | dbo:MusicGenre | dbo:Place | dbo:Band | ... | Correct |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Industrial_Metal | 0.2 | 1.0 | 1.0 | 1.0 | 0.25 | 1.0 | 1.0 | 1.0 | 1.0 | true | false | false | ... | true |
| Alternative_Rock | 0.2 | 0.2 | 0.5 | 1.0 | 0.25 | 0.2 | 1.0 | 0.3 | 1.0 | true | false | false | ... | true |
| Ambient_music | 0.2 | 0.2 | 0.3 | 0.5 | 0.25 | 0.1 | 0.5 | 0.3 | 0.0 | true | false | false | ... | false |
| Electronica | 0.2 | 0.2 | 0.25 | 0.25 | 0.3 | 0.1 | 0.3 | 0.3 | 0.0 | true | false | false | ... | false |
| Synthpop | 0.2 | 0.2 | 0.2 | 0.25 | 0.25 | 0.1 | 0.25 | 0.3 | 0.0 | true | false | false | ... | true |

links between the language specific chapters, as indicated in Fig. 1.

With the help of a feature vector representation, it is possible to learn fine grained classification models, such as *The first three genres mentioned in the first or second sentence of a band abstract are genres of that band.*

### 3.3. Machine Learning Algorithms

Initially, we experimented with a set of seven classification algorithms, i.e., Naive Bayes, k-NN, Decision Tree [31], RIPPER [5], Random Forest (RF) [4], Neural Networks [16] and Support Vector Machines (SVM) [6]. For all those classifiers, we used the im-

Table 2
List of features used by the classifier

| ID | Name | Range |
|----|------|-------|
| F00 | NumberOfCandidates | (0,1] |
| F01 | CandidatesInSentence | (0,1] |
| F02 | CandidatePosition | (0,1] |
| F03 | CandidatePositionInSentence | (0,1] |
| F04 | EntitiesInSentence | (0,1] |
| F05 | EntityPosition | (0,1] |
| F06 | EntityPositionInSentence | (0,1] |
| F07 | SentencePosition | (0,1] |
| F08 | BackLink | T / F |
| FXX | InstanceTypes | T / F |

plementation in *RapidMiner*[5], and, for the preliminary evaluation, all classifiers were used in their standard setup.

For those five classifiers, we used samples of size 50,000 from the ten most frequent relations in DBpedia, the corresponding English language abstracts, and performed an experiment in ten-fold cross validation. The results are depicted in table 3. We can observe that the best results in terms of F-measure are achieved by Random Forests, which has been selected as the classifier to use in the subsequent experiments.

Furthermore, we compared the machine learning approach to four simple baselines using the same setup:

**Baseline 1** The first entity with a matching type is classified as a positive relation, all others as negative.

**Baseline 2** All entities with a matching type are classified as positive relations.

**Baseline 3** The first entity with a matching ingoing edge is classified as a positive relation. For example, when trying to extract relations for `dbo:birthPlace`, the first entity which already has one ingoing edge of type `dbo:birth-Place` would be classified as positive.

**Baseline 4** All entities with a matching ingoing edge are classified as positive relations.

The results of the baseline evaluations are depicted in table 4. We can observe that in terms of F-measure, they are outperformed by RandomForest. Although the margin seems small, the baseline approaches usually have a high recall, but low precision. In fact, none of them reaches a precision above 0.5, which means that

by applying such approaches, at least half of the relations inserted into a knowledge graph would be noise.

## 4. Experiments

We conducted different experiments to validate the approach. First, we analyzed the performance of the relation extraction using a RandomForest classifier on the English DBpedia only. Here, we follow a two-fold approach: for once, we use a cross-validated silver standard evaluation, where we evaluate how well existing relations can be predicted for instances already present in DBpedia. Since such a silver-standard evaluation can introduce certain biases, we additionally validate the findings on a subset of the extracted relations in a manual retrospective evaluation.[6]

In a second set of experiments, we analyze the extraction of relations on the twelve largest language editions of Wikipedia, which at the same time are those with more than 1M articles, i.e., English, German, Spanish, French, Italian, Dutch, Polish, Russian, Cebuano, Swedish, Vietnamese, and Waray.[7,8] Note that this selection of languages does not only contain Indo-European, but also two Austroasiatic and an Austronesian language.

In addition, we conduct further analyses. First, we investigate differences of the relations extracted for different languages with respect to topic and locality. For the latter, the hypothesis is that information extracted, e.g., for places from German abstracts is about places in German speaking countries. Finally, with RIPPER being a symbolic learner, we also inspect the models as such to find out whether there are universal patterns for relations that occur in different languages.

### 4.1. Pre-Study on English Abstracts

In a first set of experiments, we analyzed the performance of our method on English abstracts only. Since we aim at augmenting the DBpedia knowledge graph at a reasonable level of precision, our aim was to

---

[5]http://www.rapidminer.com/

[6]See [24] for a discussion of the advantages and drawbacks of such evaluation methods.

[7]According to http://wikistats.wmflabs.org/display.php?t=wp, as of December 2015.

[8]The datasets of the extracted relations for all languages can be found online at http://dws.informatik.uni-mannheim.de/en/research/language-agnostic-relation-extraction-from-wikipedia-abstracts

Table 3

Pre-Study results on five machine learning algorithms and the ten most frequent relations. The table depicts precision, recall, and F-measure in ten fold cross validation.

| Relation | Naive Bayes | | | Random Forest | | | RIPPER | | | Neural Net | | | SVM | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | P | R | F | P | R | F | P | R | F | P | R | F | P | R | F |
| dbo:birthPlace | .69 | .65 | .67 | .69 | .76 | .72 | .72 | .73 | .72 | .61 | .75 | .67 | .72 | .74 | .73 |
| dbo:family | .55 | .93 | .69 | .87 | .83 | .85 | .85 | .83 | .84 | .77 | .83 | .80 | .87 | .83 | .85 |
| dbo:deathPlace | .42 | .30 | .35 | .51 | .30 | .38 | .64 | .18 | .28 | .61 | .19 | .29 | .66 | .20 | .31 |
| dbo:producer | .35 | .55 | .43 | .35 | .14 | .20 | .47 | .04 | .07 | .23 | .10 | .14 | .48 | .05 | .09 |
| dbo:writer | .55 | .61 | .58 | .62 | .55 | .58 | .64 | .54 | .59 | .52 | .51 | .51 | .67 | .53 | .59 |
| dbo:subsequentWork | .11 | .21 | .14 | .35 | .10 | .16 | .42 | .02 | .04 | .21 | .07 | .11 | .61 | .06 | .11 |
| dbo:previousWork | .18 | .43 | .25 | .39 | .18 | .25 | .59 | .05 | .09 | .57 | .08 | .14 | .60 | .10 | .17 |
| dbo:artist | .94 | .94 | .94 | .94 | .95 | .94 | .95 | .96 | .95 | .95 | .86 | .90 | .95 | .89 | .92 |
| dbo:nationality | .76 | .90 | .82 | .76 | .92 | .83 | .77 | .91 | .83 | .72 | .81 | .76 | .77 | .92 | .84 |
| dbo:formerTeam | .79 | .74 | .76 | .85 | .88 | .86 | .85 | .88 | .86 | .82 | .77 | .79 | .85 | .89 | .87 |
| Average | .53 | .63 | .56 | .63 | .56 | .58 | .69 | .51 | .53 | .60 | .50 | .51 | .72 | .52 | .55 |

Table 4

Pre-Study results on the four baselines and the ten most frequent relations. The table depicts precision, recall, and F-measure in ten fold cross validation.

| Relation | Baseline 1 | | | Baseline 2 | | | Baseline 3 | | | Baseline 4 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | P | R | F | P | R | F | P | R | F | P | R | F |
| dbo:birthPlace | .47 | .99 | .64 | .46 | 1.00 | .63 | .49 | .98 | .66 | .48 | .99 | .65 |
| dbo:family | .18 | .85 | .30 | .17 | 1.00 | .29 | .87 | .84 | .86 | .86 | 1.00 | .92 |
| dbo:deathPlace | .28 | .97 | .43 | .27 | 1.00 | .43 | .30 | .93 | .46 | .30 | .96 | .46 |
| dbo:producer | .17 | .93 | .29 | .15 | 1.00 | .26 | .33 | .80 | .47 | .32 | .87 | .46 |
| dbo:writer | .41 | .69 | .52 | .19 | 1.00 | .32 | .56 | .59 | .58 | .45 | .86 | .59 |
| dbo:subsequentWork | .02 | 1.00 | .04 | .02 | 1.00 | .04 | .02 | .19 | .03 | .02 | .19 | .03 |
| dbo:previousWork | .04 | 1.00 | .08 | .04 | 1.00 | .08 | .04 | .20 | .06 | .03 | .20 | .06 |
| dbo:artist | .31 | .99 | .47 | .27 | 1.00 | .42 | .57 | .87 | .69 | .53 | .87 | .66 |
| dbo:nationality | .73 | .96 | .83 | .64 | 1.00 | .78 | .74 | .96 | .84 | .64 | 1.00 | .78 |
| dbo:formerTeam | .35 | .72 | .47 | .40 | 1.00 | .57 | .78 | .70 | .74 | .81 | .98 | .89 |
| Average | .30 | .91 | .41 | .26 | 1.00 | .38 | .47 | .71 | .54 | .44 | .79 | .55 |

learn models which reach a precision of at least 95%, i.e., that add statements with no more than 5% noise to the knowledge graph. Out of the 395 relations under inspection, the RandomForest classifier could learn models with a precision of 95% or higher for 99 relations. For the 99 models that RF could extract with a minimum precision of 95%, the macro (micro) average recall and precision are 31.5% (30.6%) and 98.2% (95.7%), respectively.

By applying the 99 models to all candidates, a total of 998,993 new relation instances could be extracted, which corresponds to roughly 5% of all candidates. Fig. 2 depicts the 20 relations for which most instances are extracted.

For validating the precision and recall scores computed on the existing relation instances, we sampled
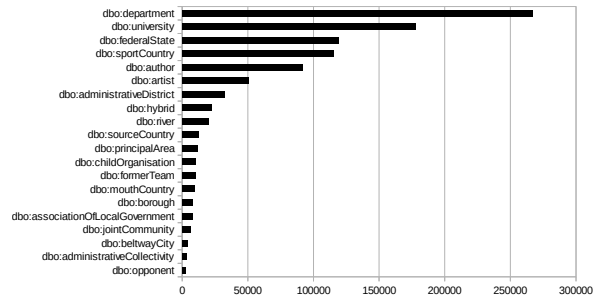


Fig. 2. 20 most frequent relations extracted from English abstracts

each 200 *newly* generated from five relations (i.e., 1,000 in total) and validated them manually. For the selection of entities, we aimed at a wider coverage of common topics (geographic entities, people, books,

Table 5

Results of the manual verification of precision and recall scores computed on the existing relation instances. $R_e$ and $P_e$ denotes the recall and precision of the models computed on the existing relation instances, while $R_m$ and $P_m$ denotes those verified by manual computation.

| Relation | $R_e$ | $P_e$ | $R_m$ | $P_m$ |
|---|---|---|---|---|
| dbo:musicalBand | 96.2 | 95.1 | 87.9 | 96.7 |
| dbo:author | 68.2 | 95.2 | 53.4 | 91.85 |
| dbo:department | 64.5 | 99.5 | 53.5 | 93.65 |
| dbo:sourceCountry | 98.9 | 98 | 98.85 | 97.8 |
| dbo:saint | 41.2 | 100 | 53.25 | 95.5 |

Table 6

Size of the 12 largest language editions of Wikipedia, and percentage of articles linked to English.

| Language | # Entities | % links to English |
|---|---|---|
| English | 4,192,414 | 100.00 |
| Swedish | 2,351,544 | 17.60 |
| German | 1,889,351 | 42.21 |
| Dutch | 1,848,249 | 32.98 |
| French | 1,708,934 | 51.48 |
| Cebuano | 1,662,301 | 5.67 |
| Russian | 1,277,074 | 42.61 |
| Waray | 1,259,540 | 12.77 |
| Italian | 1,243,586 | 55.69 |
| Spanish | 1,181,096 | 54.72 |
| Polish | 1,149,530 | 53.70 |
| Vietnamese | 1,141,845 | 28.68 |

music works), as well as relations which can be validated fairly well without the need of any specific domain knowledge (e.g., about proteins or chemical compounds).

The results are depicted in table 5. It can be observed that the precision values obtained in cross-validation are rather realistic (i.e., the deviation from the estimate is 3% on average), while the recall values are less reliable (with a deviation of 9% on average). The first observation is crucial, as it allows to create new relations for the knowledge graph at a reasonable level of precision, i.e., the amount of noise introduced is strictly controlled.

### 4.2. Cross-Lingual Relation Extraction

In the next experiment, we used the RandomForests classifier to extract models for relations for the top 12 languages, as depicted in Table 6.

As a first result, we look at the number of relations for which models can be extracted at 95% precision.
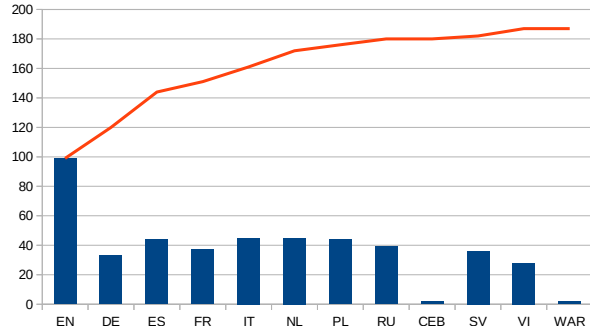


Fig. 3. Number of relations extracted at 95% precision in the top 12 languages. The bars show the number of relations that could be extracted for the given language, the line depicts the accumulated number of relations for the top N languages.
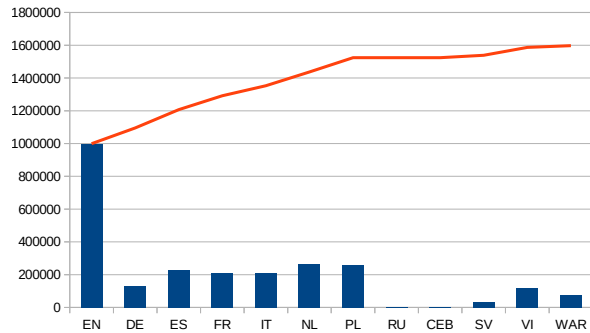


Fig. 4. Number of statements extracted at 95% precision in the top 12 languages. The bars show the number of statements that could be extracted for the given language, the line depicts the accumulated number of statements for the top N languages.

While it is possible to extract models for 99 relations at that level of precision for English, that number almost doubles to 187 when using the top twelve languages, as depicted in Fig. 3. These results show that it is possible to create high precision models for relations in other languages for which this is not possible in English.

When extracting new statements (i.e., instantiations of the relations) using those models, our goal is to extract those statements in the canonical DBpedia knowledge base, as depicted in Fig. 1. The number of extracted statements per language, as well as cumulated statements, is depicted in Fig. 4.

At first glance, it is obvious that, although a decent number of models can be learned for most languages, the number of statements extracted are on average an order of magnitude smaller than the number of statements that are extracted for English. However, the additional number of extracted relations is considerable: while for English only, there is roughly 1M relations, 1.6M relations can be extracted from the top 12 lan-

guages, which is an increase of about 60% when stepping from an English-only to a multi-lingual extraction. The graphs in Figures 3 and 4 also show that the results stabilize after using the seven largest language editions, i.e., we do not expect any significant benefits from adding more languages with smaller Wikipedias to the setup.

As can be observed in Fig. 4, the number of extracted statements is particularly low for Russian and Cebuano. For the latter, Fig. 3 already showed that only a small number of high quality models can be learned, mostly due to the low number of inter-language links to English, as depicted in table 6. For the former, the number of high quality models that can be learned is larger, but the models are mostly unproductive, since they are learned for rather exotic relations. In particular, for the top 5 relations in Fig. 2, no model is learned for Russian.

It is evident that the number of extracted statements is not proportional to the relative size of the respective Wikipedia, as depicted in table 6. For example, although the Swedish Wikipedia is more than half the size of the English one, the number of extracted statements from Swedish is by a factor of 28 lower than those extracted from English. At first glance, this may be counter intuitive.

The reason for the number of statements extracted from languages other than English is that we only generate candidates if both the article at hand and the entity linked from that article's abstract have a counterpart in the canonical English DBpedia. However, as can be seen from table 6, those links to counterparts are rather scarce. For the example of Swedish, the probability of an entity being linked to the English Wikipedia is only $0.176$. Thus, the probability for a candidate that both the subject and object are linked to the English Wikipedia is $0.176 \times 0.176 = 0.031$. This is pretty exactly the ratio of statements extracted from Swedish to statements extracted from English ($0.036$). In fact, the number of extracted statements per language and the squared number of links between the respective language edition and the English Wikipedia have a Pearson correlation coefficient of $0.95$. This shows that the low number of statements is mainly an effect of missing inter-language links in Wikipedia, rather than a shortcoming of the approach as such.[9]

### 4.3. Topical and Geographical Analysis by Language

To further analyze the extracted statements, we look at the topical and geographical coverage for the *additional* statements (i.e., statements that are not yet contained in DBpedia) that are extracted for the twelve languages at hand. First, we depict the most frequent relations and subject classes for the statements. The results are depicted in Fig. 5 and 6. It can be observed that the majority of statements is related to geographical entities and their relations. The Russian set is an exception, since the topic of most extracted relations musical works, in contrast to geographic entities, as for the other languages. Furthermore, the English set has the largest fraction of person related facts.

We assume that the coverage of Wikipedia in different languages is, to a certain extent, biased towards places, persons, etc. from countries in which the respective language is spoken.[10] Thus, we expect that, e.g., for relations extracted about places, we will observe that the distribution of countries to which entities are related differs for the various language editions.

To validate this hypothesis, we determine the country to which a statement is related as follows: given a statement *s* in the form

```
s p o .
```

we determine the set of pairs $P_s := < r, c >$ of relations and countries that fulfill

```
s r c .
c a dbo:Country .
```

and

```
o r c .
c a dbo:Country .
```

For all statements $S$ extracted from a language, we sum up the relative number of relations of a country to each statement, i.e., we determine the weight of a country $C$ as

$$w(C) := \sum_{s=1}^{|S|} \frac{|\{< r, c > \in P_s | c = C\}|}{|P_s|} \tag{1}$$

Figure 7 depicts the distributions for the countries. We can observe that while in most cases, facts about

---

Table 7

Most frequently extracted relations across the top 12 languages.

| Relation | No. of languages |
|----------|------------------|
| department | 11 |
| musicalBand | 10 |
| family | 9 |
| president | 9 |
| derivative | 8 |
| veneratedIn | 7 |
| majorIsland | 7 |
| operator | 7 |
| guest | 6 |
| crosses | 6 |

Table 8

Cross-language model similarity for different relations, averaged across the 12 most frequently extracted relations and all language pairs.

| Relation | Similarity |
|----------|------------|
| crosses | 0.286 |
| department | 0.320 |
| derivative | 0.217 |
| family | 0.139 |
| majorIsland | 0.000 |
| musicalBand | 0.741 |
| operator | 0.292 |

US related entities are the majority, only for Polish, entities related to Poland are the most frequent. For Swedish, German, French, Cebuano and Italian, the countries with the largest population speaking those languages (i.e., Sweden, Germany, France, Philippines, and Italy, respectively), are at the second position. For Spanish, Spain is at the second position, despite Mexico and Colombia (rank 11 and 6, respectively) having a larger population. For the other languages, a language-specific effect is not observable: for Dutch, the Netherlands are at rank 8, for Vietnamese, Vietnam is at rank 34, for Waray, the Philippines are at rank 7. For Russian, Russia is on rank 23, preceded by Soviet Union (sic!, rank 15) and Belarus (rank 22).

The results show that despite the dominance of US-related entities, there is a fairly large variety in the geographical coverage of the information extracted. This supports the finding that adding information extracted from multiple Wikipedia language editions helps broadening the coverage of entities.

### 4.4. Analysis of Cross-Lingual Patterns

For the analysis, we look at the relations which are extracted in *most* languages. Those relations are depicted in table 7.

In order to analyze how similar the relations are, we stepped from Random Forests to a more symbolic learner, i.e., the rule learning algorithm *RIPPER*, which had delivered good results in the pre-study as well (see section 3.3). RIPPER learns only disjunctive rules, where each condition is a simple comparison of an attribute and an atomic value.

By design, RIPPER tries to learn a model for the smallest class, and it does so by using the separate-and-conquer algorithm, trying to cover as many positive ex-

amples with each rule as possible [5]. Since there are usually less candidates which are actual instantiations of a relation than candidates which are not, the positive class in most case is also the smallest class. This means that the *first rule* learned by RIPPER covers the majority of positive examples in the language, i.e., it represents the *dominant pattern* of the relation.

To measure the similarity of two models, we use the overlap of conditions in the first rule of the different languages. We compute the average probability of a condition to be contained in the first rule of another language model as well and use that average probability as a similarity measure between the RIPPER rule models learned for different languages.

From the twelve relations depicted in table 7, RIPPER was able to extract a model for more than one language in seven cases. The similarities of the models are depicted in table 8.

It can be observed that the model similarity strongly varies. For the *majorIsland* relations, only two models were learned, one for Spanish, one for French, and they are completely different from one another. On the other hand, for the *musicalBand* relation, the first rule is completely identical for six out of ten models learned. This rule is

```
    PositionOfSentence <= 0.5
&& !BackLink
=> false
```

Here, a rule for the negative case (i.e., a candidate not being a valid statement) has been learned. The rule thus means that for a candidate (i.e,. a band being linked from an abstract about a music single) being a valid statement, it has to be contained in the first sentence[11], or a backlink from the object's

---

[11]Note that the `PositionOfSentence` feature is used in its inverse, as described in section 3.2.
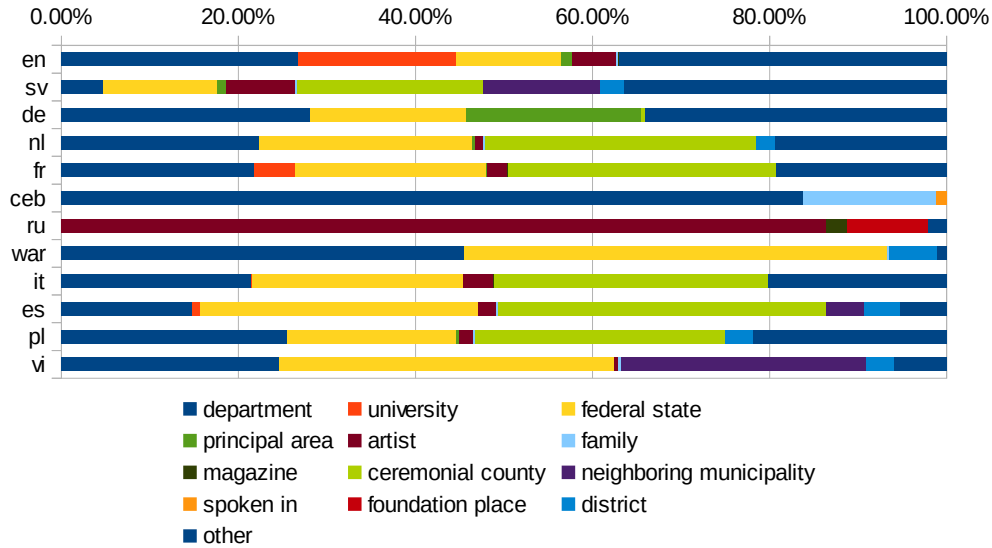
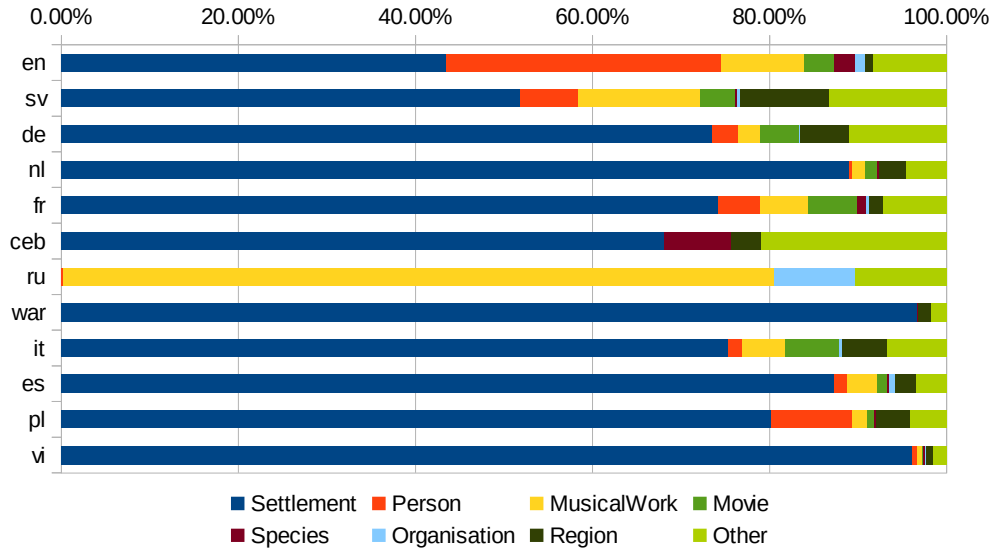Fig. 5. Distribution of relations in the different language extractions



Fig. 6. Distribution of subject types in the different language extractions

to the subject's article has to be present. Since this rule is learned for many languages, it appears to be a language-independent pattern.

## 5. Conclusion and Outlook

Adding new relations to existing knowledge graphs is an important task in adding value to those knowledge graphs. In this paper, we have introduced an approach that adds relations to DBpedia using abstracts in Wikipedia. Unlike other works in that area, the approach presented in this paper uses background knowledge from DBpedia, but does not rely on any language-specific techniques, such as POS tagging, stemming, or dependency parsing. Thus, it can be applied to Wikipedia abstracts in any language.

While we have worked with DBpedia only in this paper, the approach can be applied to other cross-domain knowledge graphs, such as YAGO or Wikidata, as well, since they also link to DBpedia. Furthermore, for a significant portion of Semantic Web datasets, links to DBpedia exist as well [27], so that the
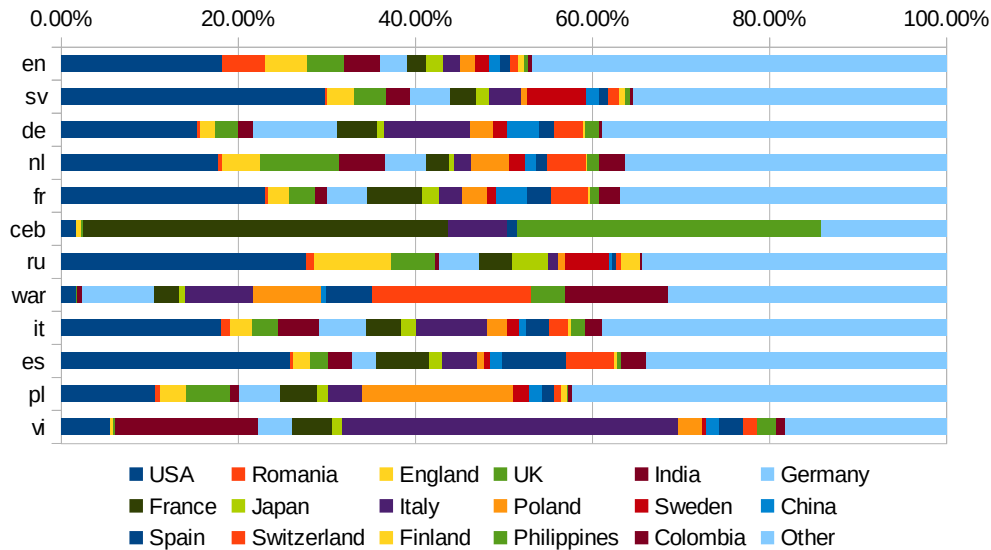
Fig. 7. Distribution of locality in the different language extractions

approach can be applied even to such domain-specific datasets.

The experimental results show that the approach can add a significant amount of new relations to DBpedia. By extending the set of abstracts from English to the most common languages, the coverage both of relations for which high quality models can be learned, as well as of instantiation of those relations, significantly increases. Furthermore, we can observe that for certain relations, the models learned are rather universal across languages.

The finding for some relations, the models are rather universal across different languages, gives way to interesting refinements of the approach. Since the learning algorithms sometimes fail to learn a high quality model for one language, it would be interesting to reuse a model which performed good on other languages in those cases, i.e., to decouple the learning and the application of the models. With such a decoupled approach, even more instantiations of relations can be potentially extracted, although the effect on precision would need to be examined with great care. Furthermore, designing ensemble solutions that collect evidence from different languages could be designed, which would allow for lowering the strict precision threshold of 95% applied in this paper (i.e., several lower precision models agreeing on the same extracted relations could be a similarly or even more valuable evidence as one high-precision model).

In our experiments, we have only concentrated on relations between entities so far. However, a significant fraction of statements in DBpedia and other knowledge graphs also have literals as objects. That said, it should be possible to extend the framework to such statements as well. Although numbers, years, and dates are usually not linked to other entities, they are quite easy to detect using, e.g., regular expressions or specific taggers such as *HeidelTime* [30]. With such a detection step in place, it would also be possible to learn rules for datatype properties, such as: *the first date in an abstract about a person is that person's birthdate*, etc.

Furthermore, our focus so far has been on adding missing relations. A different, yet related problem is the detection of wrong relations [24,25,26]. Here, we could use our approach to gather *evidence* for relations in different language editions of Wikipedia. Relations for which there is little evidence could then be discarded (similar to DeFacto [11]). While for adding knowledge, we have tuned our models towards *precision*, such an approach, however, would require a tuning towards *recall*. In addition, since there are also quite a few errors in numerical literals in DBpedia [8,35], an extension such as the one described above could also help detecting such issues.

So far, we have worked on one genre of text, i.e., abstracts of encyclopedic articles. However, we are confident that this approach can be applied to other genres of articles as well, as long as those follow typical structures. Examples include, but are not limited to: extracting relations from movie, music, and book reviews, from short biographies, or from product descriptions. All those are texts that are not strictly struc-

tured, but expose certain patterns. While for the Wikipedia abstracts covered in this paper, links to the DBpedia knowledge graph are implicitly given, other text corpora would require entity linking using tools such as DBpedia Spotlight [20].

In summary, we have shown that Wikipedia abstracts are a valuable source of knowledge for extending knowledge graphs such as DBpedia. Those abstracts expose patterns which can be captured by language-independent features, thus allowing for the design of language-agnostic systems for relation extraction from such abstracts.

# References

[1] Alessio Palmero Aprosio, Claudio Giuliano, and Alberto Lavelli. Extending the Coverage of DBpedia Properties using Distant Supervision over Wikipedia. In *NLP&DBpedia*, volume 1064 of *CEUR Workshop Proceedings*, 2013.

[2] Emily M Bender. Linguistically naïve!= language independent: why nlp needs linguistic typology. In *Proceedings of the EACL 2009 Workshop on the Interaction between Linguistics and Computational Linguistics: Virtuous, Vicious or Vacuous?*, pages 26–32. Association for Computational Linguistics, 2009.

[3] 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, New York, 2008. ACM.

[4] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[5] William W. Cohen. Fast effective rule induction. In *Machine Learning, Proceedings of the Twelfth International Conference on Machine Learning, Tahoe City, California, USA, July 9-12, 1995*, pages 115–123, 1995.

[6] Nello Cristianini and John Shawe-Taylor. *An Introduction to Support Vector Machines and Other Kernel-based Learning Methods*. Cambridge University Press, 2010.

[7] Xin Luna Dong, K Murphy, E Gabrilovich, G Heitz, W Horn, N Lao, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge Vault: A Web-scale approach to probabilistic knowledge fusion. In *20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 601–610, 2014.

[8] Daniel Fleischhacker, Heiko Paulheim, Volha Bryl, Johanna Völker, and Christian Bizer. Detecting Errors in Numerical Linked Data Using Cross-Checked Outlier Detection. In *The Semantic Web–ISWC 2014*, volume 8796 of *LNCS*, pages 357–372. Springer, Switzerland, 2014.

[9] Katrin Fundel, Robert Küffner, and Ralf Zimmer. RelexâĂŤrelation extraction using dependency parse trees. *Bioinformatics*, 23(3):365–371, 2007.

[10] Luis Antonio Galárraga, Christina Teflioudi, Katja Hose, and Fabian Suchanek. AMIE: association rule mining under incomplete evidence in ontological knowledge bases. In *Proceedings of the 22nd international conference on World Wide Web*, pages 413–422, 2013.

[11] Daniel Gerber, Diego Esteves, Jens Lehmann, Lorenz Bühmann, Ricardo Usbeck, Axel-Cyrille Ngonga Ngomo, and René Speck. Defacto – temporal and multilingual deep fact validation. *Web Semantics: Science, Services and Agents on the World Wide Web*, 35(2), 2015.

[12] Daniel Gerber and A-C Ngonga Ngomo. Bootstrapping the Linked Data web. In *Workshop on Web Scale Knowledge Extraction*, 2011.

[13] Jiseong Kim, Eun-Kyung Kim, Yousung Won, Sangha Nam, and Key-Sun Choi. The Association Rule Mining System for Acquiring Knowledge of DBpedia from Wikipedia Categories. In *Workshop on NLP and DBpedia*, 2015. https://nlpdbpedia2015.files.wordpress.com/2015/08/nlpdbpedia_2015_submission_13.pdf.

[14] Christian Kolthoff and Arnab Dutta. Semantic Relation Composition in Large Scale Knowledge Bases. In *3rd Workshop on Linked Data for Information Extraction*, volume 1467 of *CEUR Workshop Proceedings*, pages 34–47, 2015. http://ceur-ws.org/Vol-1467/.

[15] Denis Krompaß, Stephan Baier, and Volker Tresp. Type-Constrained Representation Learning in Knowledge Graphs. In *International Semantic Web Conference*, volume 9366 of *LNCS*, pages 640–655, Switzerland, 2015. Springer. http://dx.doi.org/10.1007/978-3-319-25007-6_37.

[16] Miroslav Kubat. Neural networks: a comprehensive foundation by simon haykin, macmillan, 1994, ISBN 0-02-352781-7. *Knowledge Eng. Review*, 13(4):409–412, 1999.

[17] Dustin Lange, Christoph Böhm, and Felix Naumann. Extracting structured information from Wikipedia articles to populate infoboxes. In *Proceedings of the 19th ACM Conference on Information and Knowledge Management (CIKM)*, pages 1661–1664, New York, 2010. ACM.

[18] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. DBpedia – A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semantic Web Journal*, 6(2), 2013.

[19] Farzaneh Mahdisoltani, Joanna Biega, and Fabian M. Suchanek. YAGO3: A Knowledge Base from Multilingual Wikipedias. In *Conference on Innovative Data Systems Research*, 2015.

[20] Pablo N Mendes, Max Jakob, Andrés García-Silva, and Christian Bizer. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, 2011.

[21] 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*, pages 1003–1011, Stroudsburg, PA, 2009. Association for Computational Linguistics.

[22] Dat PT Nguyen, Yutaka Matsuo, and Mitsuru Ishizuka. Relation extraction from wikipedia using subtree mining. In *Proceedings of the National Conference on Artificial Intelligence*, volume 22, page 1414. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007.

[23] Thien Huu Nguyen and Ralph Grishman. Relation extraction: Perspective from convolutional neural networks. In *Proceedings of NAACL-HLT*, pages 39–48, 2015.

[24] Heiko Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, 2016. to appear.

[25] Heiko Paulheim and Christian Bizer. Improving the Quality of Linked Data Using Statistical Distributions. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 10(2):63–86, 2014.

[26] Heiko Paulheim and Aldo Gangemi. Serving DBpedia with DOLCE–More than Just Adding a Cherry on Top. In *International Semantic Web Conference*, volume 9366 of *LNCS*, International, 2015. Springer.

[27] Max Schmachtenberg, Christian Bizer, and Heiko Paulheim. Adoption of the Linked Data Best Practices in Different Topical Domains. In *International Semantic Web Conference*, 2014.

[28] Alexander Schutz and Paul Buitelaar. Relext: A tool for relation extraction from text in ontology extension. In *The Semantic Web–ISWC 2005*, pages 593–606. Springer, 2005.

[29] Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. Reasoning With Neural Tensor Networks for Knowledge Base Completion. In *Advances in Neural Information Processing Systems 26 (NIPS 2013)*, pages 926–934. Curran Associates, Inc., Newry, 2013. `http://papers.nips.cc/paper/5028-reasonin`.

[30] Jannik Strötgen and Michael Gertz. Heideltime: High quality rule-based extraction and normalization of temporal expressions. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 321–324. Association for Computational Linguistics, 2010.

[31] Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Addison-Wesley, 2005.

[32] Patrick Verga, David Belanger, Emma Strubell, Benjamin Roth, and Andrew McCallum. Multilingual relation extraction using compositional universal schema. *arXiv preprint arXiv:1511.06396*, 2015.

[33] Denny Vrandečić and Markus Krötzsch. Wikidata: a Free Collaborative Knowledge Base. *Communications of the ACM*, 57(10):78–85, 2014.

[34] Gang Wang, Yong Yu, and Haiping Zhu. Pore: Positive-only relation extraction from wikipedia text. In *Proceedings of the 6th International The Semantic Web and 2Nd Asian Conference on Asian Semantic Web Conference*, ISWC'07/ASWC'07, pages 580–594, Berlin, Heidelberg, 2007. Springer-Verlag.

[35] Dominik Wienand and Heiko Paulheim. Detecting Incorrect Numerical Data in DBpedia. In *The Semantic Web: Trends and Challenges*, volume 8465 of *LNCS*, pages 504–518. Springer, International, 2014.

[36] Fei Wu, Raphael Hoffmann, and Daniel S Weld. Information extraction from Wikipedia: Moving down the long tail. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 731–739, New York, 2008. ACM.

[37] Yulan Yan, Naoaki Okazaki, Yutaka Matsuo, Zhenglu Yang, and Mitsuru Ishizuka. Unsupervised relation extraction by mining wikipedia texts using information from the web. 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 1021–1029, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[38] Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. Kernel methods for relation extraction. *The Journal of Machine Learning Research*, 3:1083–1106, 2003.

[39] Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. Distant supervision for relation extraction via piecewise convolutional neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP), Lisbon, Portugal*, pages 17–21, 2015.

[40] Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, Jun Zhao, et al. Relation classification via convolutional deep neural network. In *COLING*, pages 2335–2344, 2014.

[41] Yu Zhao, Sheng Gao, Patrick Gallinari, and Jun Guo. Knowledge base completion by learning pairwise-interaction differentiated embeddings. *Data Mining and Knowledge Discovery*, 29(5):1486–1504, 2015. `http://dx.doi.org/10.1007/s10618-015-0430-1`.