The Journal of Web Science, 2018, 4: 20–33

Multi-Cultural Interlinking of Web Taxonomies with ACROSS

Natalia Boldyrev[1], Marc Spaniol[2] and Gerhard Weikum[1]

[1] *Max Planck Institute for Informatics, Saarland Informatics Campus, Germany, {natalia/weikum}@mpi-inf.mpg.de*

[2] *Université de Caen Normandie, Caen, France, marc.spaniol@unicaen.fr*

ABSTRACT

The Web hosts a huge variety of multi-cultural taxonomies. They encompass product catalogs of e-commerce, general-purpose knowledge bases and numerous domain-specific category systems. The enormous heterogeneity of those sources is a challenging aspect when multiple taxonomies have to be interlinked. In this paper we introduce the ACROSS system to support the alignment of independently created Web taxonomies. Each taxonomy is shaped by its unique culture, which is three-fold: categorization criteria of the taxonomy, language, and socio-economic background. For mapping categories between different taxonomies, ACROSS harnesses instance-level features as well as distant supervision from an intermediate source like multiple Wikipedia editions. ACROSS includes a reasoning step, which is based on combinatorial optimization. In order to reduce the run time of the reasoning procedure without sacrificing quality, we study two models of user involvement. Our experiments with heterogeneous taxonomies for different domains demonstrate the viability of our approach and improvement over state-of-the-art baselines.

ISSN 2332-4031; DOI 10.1561/106.00000012

1 Introduction

1.1 Motivation and Problem

The availability of knowledge bases (KBs) on the Web has impacted the way recommendation and analytic applications process enterprise, Web and social media content. Those knowledge collections range from commercial endeavors such as Google Knowledge Graph (Singhal, 2012), centered around Freebase (Bollacker *et al.*, 2008), to academic projects like DBpedia (Auer *et al.*, 2007), Yago (Suchanek *et al.*, 2007), NELL (Carlson *et al.*, 2010), BabelNet (Navigli and Ponzetto, 2012), and more. Semantic types or topics are a vital component of the KB's. They are usually organized in a tree or a DAG. However, those taxonomies of topics are extremely diverse reflecting specific orientations to a domain, market and application.

For example, the Freebase taxonomy organizes everything belonging to books under the `/book` domain: types like `Books`, `Publishers` and others. On the other hand, Yago distinguishes between organizations and creative works and places the types `Novels` and `Publishing Organizations` in different branches of its upper-level ontology. Freebase-style nodes provide, thus, a better asset to users in exploring a KB, whereas Yago types with semantically rigorous instances serve a different purpose – boosting reasoning in programmed applications.

In a wider sense, KB's also include social, academic or enterprise collections such as Web site directories like dmoz.org, various product catalogs of online-stores as those of Amazon, digital libraries such as the US Library of Congress or the German National Library, Wikipedia editions, specialized online communities on health issues, music, etc. Although their contents are often more document- or user-post-oriented than entity-centric, their taxonomies are essential for navigation and search.

On first glance, this wide variety of taxonomies, with overlapping topics or types, presents a *curse of heterogeneity*. However, there is an enormous *asset of cultural diversity* at the users' disposal. Thinking of this diversity as a call for full-fledged data integration or top-down standardization of the taxonomies across all knowledge collections is infeasible approach. Instead, what we set out for in this paper is to finding *alignments* that allow users to navigate across the boundaries of knowledge repository and explore different taxonomies together, while living with existing diversity. For instance, book lovers might be interested in finding out which books like-minded people are associating with their favorite topic in a different language of Amazon's online shop or on a social tagging site such as Shelfari. However, making a transition between taxonomies is not a trivial task when crossing market or culture borders. As an example, the category `Kinder- & Jugendliteratur` on amazon.de (en. Children & Youth Literature) has two relevant counterparts on amazon.com – `Children's Books` and `Teen & Young Adult`. Resolving semantic equivalence of these categories is challenging both for syntactic-based and structure-based alignment approaches (the amazon.de category is a leaf, whereas the amazon.com categories are not).

1.2 Approach and Contribution

In this paper we focus on multiple knowledge taxonomies that are culture-specific such as product catalogs of Amazon in different languages. The fact, that "43% of Europeans never purchase online products and services in languages other than their

own... "[1] illustrates the demand in overcoming linguistic borders. However, the language of a taxonomy is only one of the cultural facets. The purpose of a taxonomy and its market orientation motivate particular choices of categorization criteria and instances. As an example, consider the book categorization on amazon.com and the Dewey Decimal Classification, which are incomparable.

To clarify our usage of the term *culture* in this paper, we focus on the following *cultural aspects of a knowledge taxonomy*:

- categorization criteria,
- language, and
- socio-economic background of the taxonomy.

The existing methods in ontology alignment and data integration rely on sufficient overlap of instances or sufficient similarity in the structures, which serve as anchors for computing alignments. If two taxonomies differ in at least one of the points listed above, the overlap on the instance or structure level cannot be guaranteed. These cases require sophisticated treatment.

We present ACROSS (short for ACCuRate alignment of multi-cutural taxOnomy SystemS). Specifically, for a given type or topic of knowledge base, we compute a ranked list of its semantically most related nodes in a freely selectable target taxonomy.

Alignment tasks of this kind have been addressed in the prior literature in two major areas. Catalog integration considers either instance-to-category (Agrawal and Srikant, 2001) or category-to-category (Bouquet *et al.*, 2003; Ichise *et al.*, 2003) mapping use cases. The latter approaches rely on lexical and domain knowledge to resolve the semantics of the category or on the items shared by two catalogs to induce the similarity of a pair of categories. ACROSS does not require reconciling seed instances between two input taxonomies, and is also applicable when taxonomies for same domain have low or no overlap on the instance level.

Work on ontology alignment (Suchanek *et al.*, 2011; Udrea *et al.*, 2007) focuses on joint schema and instances matching over different ontologies. This is pretty much a full-fledged data integration task, and quite different from both its input characteristics and output requirements from our task. Ontology alignment usually produces a one-to-one mapping for instances, classes and relations. Moreover, it utilizes ontology-specific features as attributes or information about domains and ranges. Closest to our work is research on aligning different Wikipedia editions (in different languages) in terms of infoboxes (Nguyen *et al.*, 2011) and categories (Göbölös-Szabó *et al.*, 2012).

Alignment tasks also arise in computational linguistics: mappings between language resources like WordNet (Fellbaum, 1998), PropBank (Palmer *et al.*, 2005), VerbNet (Kingsbury and Palmer, 2002), ReVerb (Fader *et al.*, 2011), Patty (Nakashole *et al.*, 2012), ReNoun (Yahya *et al.*, 2014), etc. These are exclusively geared for linguistic repositories of words, multi-word phrases, and their synonymy sets. It is totally different from dealing with type or topic taxonomies.

The core component of ACROSS is **semantic enrichment**, which is produced by mapping onto an intermediate taxonomy. This allows comparing two categories from different sources over the common space of semantic labels, if the categories do not share entities and are in different languages.

By computing similarities over the semantic labels, pairwise correspondences between categories of two taxonomies can be found. This can be seen as a basic algorithm to interlinking. However, this approach produces larger candidate sets, and a user needs a considerable effort in order to choose to which counterpart to navigate.

ACROSS includes a **constraint-aware reasoning step** to ensure linking to the most semantically related nodes while respecting two types of constraints. A hierarchy-preserving rule disallows that a descendant of a node i in one taxonomy is mapped to an ancestor of i's counterpart in the other taxonomy. Another rule ensures the coherence of the counterpart candidate sets by filtering out non-correlating candidates. The above constraints are expressed as an integer programming model, which can be solved with off-the-shelf tools like Gurobi[2].

These constraints are similar in spirit to a mapping repair or alignment debugging (Solimando *et al.*, 2014; Euzenat and Shvaiko, 2013), which use different formalism to describe the set of alignments violating a constraint, e.g. first order logic. ACROSS, in contrast, uses weighted predicates which make the constraint-aware reasoning more flexible.

Integer programming is known to be NP-hard in general and the exact reasoning over complex taxonomies can be a very time-consuming part. In order to bring the run times down while performing exact reasoning, we study two seeding strategies.

This paper is an extended version of our conference paper (Boldyrev *et al.*, 2016), which focused on:

- defining and modeling the alignment problem for multi-cultural knowledge taxonomies,
- utilizing a taxonomy mediation source for category assignment of culture-independent semantic labels, and
- developing an effective algorithm for computing alignments based on the semantic labels, using integer optimization.

In addition to the contributions of our preliminary work (Boldyrev *et al.*, 2016), this manuscript addresses the following:

- studying different seeding strategies for bringing the runtimes down for exact reasoning with two types of constraints, without sacrificing the quality of the alignment;
- a comprehensive experimental study with user assessments for alignments between a variety of KB pairs:

[1] http://www.lr-coordination.eu/multilingual-europe, retrieved on 28.02.2017.

[2] http://www.gurobi.com/

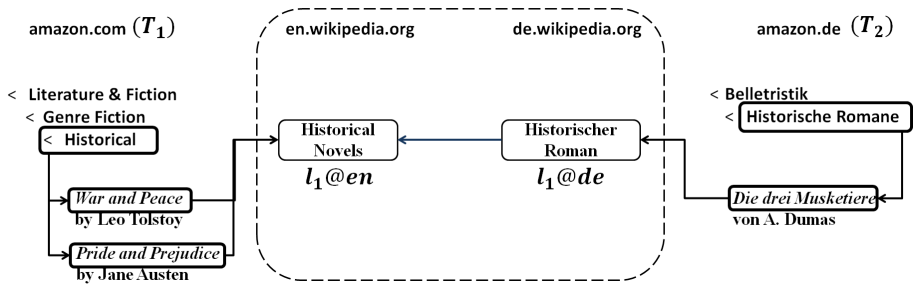


Figure 1: Example alignment of categories `Historical` from amazon.com and `Historische Romane`(Historical Novels) from amazon.de. Wikipedia serves as a mediator for obtaining labels.

- analyzing linkings produced by ACROSS with respect to concepts with high and low spelling differences. We demonstrate that ACROSS is able to cover more cases, where relying on syntactic similarity or translation fails;
- performing sensitivity study of linking with respect to the taxonomic levels. ACROSS outperforms the baseline solutions, producing linkings for categories on all taxonomic levels;
- demonstrating that the proposed seeding strategies drastically reduce the run times of the reasoning step when dealing with complex taxonomies.

2 Computational Model

Definition 1. A *knowledge taxonomy* T is a directed acyclic graph (DAG) with nodes of two types: instances and categories. There are two types of edges. Subcategory-of edges are defined over category nodes, and instances are connected to categories with instance-of edges.

Our goal is to align two taxonomies T_1 and T_2, for a wide variety of choices for T_1 and T_2. More specifically the goal is to compute, for each category i of T_1 a ranked list of most suitable counterparts $j_1, j_2, \ldots$ in T_2.

Our methods harness Wikipedia as an intermediate knowledge taxonomy, as different Wikipedia editions offer pages from a variety of languages. We can associate a category node i from a given taxonomy T with a set of Wikipedia pages, using simple mapping heuristics onto Wikipedia, either based on the instances of i or based on the surface name of i (see Section 3). For this mapping, we choose the Wikipedia edition that corresponds to T's language, as illustrated in Figure 1. `Historical Novels` and `Historischer Roman` are two labels obtained from Wikipedia. We canonicalize the labels towards one of the Wikipedia editions by following the inter-language links.

Note, that instead of Wikipedia any Wikipedia-like source can be taken. We discuss using alternative intermediate source in Section 6 in more detail.

3 Basic Methods

Given two taxonomies T_1 and T_2, we compute the alignment in three major steps:

1. Compute semantic labels for all nodes i and j of T_1 and T_2, respectively, via mappings to an intermediate Wikipedia edition by finding relevant Wikipedia pages for
   a. the titles of i and j
   b. the instances of i and j

   The titles of the relevant Wikipedia pages are considered as semantic labels. To perform matching onto Wikipedia pages, we rely on the Wikipedia search API. Contrary to using lexical rule-based matching strategies, we do not depend on the language in which the matching is carried out. Based on the overlap of the semantic labels, the instances-based and name-based alignments are produced.
2. Generate candidate mappings between T_1 and T_2 by combining instances-based and name-based mappings.
3. Consider additional constraints on the alignments and use combinatorial optimization methods to identify good alignments among the candidate ones.

Steps 1 and 2 can be viewed already as complete albeit very basic alignment algorithm. Step 3 is our main contribution and discussed further in Section 4 The rest of this section gives details of Steps 1 and 2.

3.1 *Name-based Semantic Labels (Step 1.a)*

The name-based rule finds relevant Wikipedia pages (i.e., semantic labels) for category i of taxonomy T using the title of i.

Definition 2. Let L_i be a set of title-based semantic labels for category $i \in T_1$ and L_2 the analogous set for category $j \in$

T_2. Then *name-based similarity* $n\text{-}sim(i,j)$ between i and j is defined as Jaccard similarity between L_1 and L_2:

$$n\text{-}sim(i,j) = \frac{L_i \cap L_j}{L_i \cup L_j} \quad (1)$$

In this scenario, search results and their socially curated inter-language links are used as a "smart translation" of the Wikipedia community. If two category titles are not exact translation of each other, the alignment between them still can be restored. As an example, a category from medical department of amazon.de `Blutzuckermessgeräte`(en. Glucometers) can be matched with the English target `Blood Glucose Monitors` without being a literal translation and without involving expensive synonym resolution procedure.

3.2 Instance-based Semantic Labels (Step 1.b)

In contrast to the name-based procedure, we pose instance names from a taxonomy to the Wikipedia search API and retrieve a list of relevant pages per instance. Wikipedia search results serve as *semantic labels* for the instances and, transitively, for the categories in T_1 and T_2. In the case of two taxonomies T_1 and T_2 originated in different languages, search results are canonicalized to one of the both languages. We achieve this by following the inter-language links in Wikipedia.

Contrary to the name-based rule, the same semantic label can be assigned to a category through many instances. In Figure 1, two instances of the same category return `Historical Novels` in the search. A natural way of modelling this situation is expressing each category i in the taxonomy T as a frequency vector over the set of semantic labels. A frequency vector captures the weight of a semantic label in a category, as well as its specificity - distribution over all categories in the source. This is similar to the $tfidf$ measure for terms in a document collection.

Let $V_i = \{v_{i,1}, v_{i,2}...\}$ be the frequency vector of semantic labels for category i. Each component $v_{i,l}$, describing label l, is computed as:

$$v_{i,l} = lf(l,i) \cdot icf(l,T) \quad (2)$$

with $lf(l,i)$ being the label frequency in category i and $icf(l,T)$ being the inverse category frequency in source T.

$$icf(l,T) = \log \frac{C}{C'} \quad (3)$$

where C is the total number of categories in T and C' the number of categories containing l.

Due to following inter-language links, categories from T_1 and T_2 are mapped to the same space of semantic labels.

Definition 3. The *instance-based similarity* of two categories $i \in T_1$ and $j \in T_2$ is defined as cosine similarity over their frequency vectors of semantic labels:

$$i\text{-}sim(i,j) = \frac{\sum_{l=1}^{n} v_{i,l} \cdot v_{j,l}}{\sqrt{\sum_{l=1}^{n} v_{i,l}^2} \cdot \sqrt{\sum_{l=1}^{n} v_{j,l}^2}} \quad (4)$$

where n is the total number of semantic labels.

In contrasts to the Definition 2, semantic labels contribute to categories with different weights. Since set-based similarity measures are not able to deal with weighted items, we have chosen cosine similarity as one of the standard approaches.

Using Wikipedia search accounts for linguistic complexity, niche- and market-specific instances. Drug names are a good illustration, as they are usually not shared across countries. Consider two categories - `Pain Relievers` from a U.S.-based retailer and `Schmerzmittel` (en.: Pain Relievers) from a Germany-based one. `Aleve` is a product in `Pain Relievers` and `Dolormin` is a product in `Schmerzmittel`. In this representation, each category contains a disjoint set of products. Through mapping to Wikipedia pages, both drug names are lifted to the semantic label `Naproxen`. This lifting allows "crossing" the market borders and making a transition between categories `Pain Relievers` and `Schmerzmittel`.

3.3 Candidate Alignments (Step 2)

The second step merges mapping produced by instance- and name-based rules. Alignment weight $w(i,l)$ between categories $i \in T_1$ and $j \in T_2$ is a linear combination of two weights:

$$w(i,l) = \alpha \cdot i\text{-}sim(i,j) + (1-\alpha) \cdot n\text{-}sim(i,j) \quad (5)$$

For a source $i \in T_2$, the found candidate targets $j_1, j_2...$ are ranked according to their weights. Parameter α controls which of the two semantification rules is more emphasized. In our experiments, we used $\alpha = 0.5$. We experimented with alpha values 0.3, 0.5 and 0.7. Our manual inspection revealed that ACROSS with alpha set to 0.5 performed best. Automatic adjustment of alpha is a subject of further research.

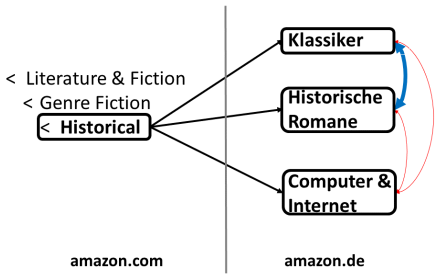
Combining the two rules induces benefits in at least two aspects. First, we reduce the problem of *sparsity*. This occurs, when a category has a long or rare title and the name-based rule fails to generate a mapping, the instances-based mapping still produces an alignment. Second, we apply community knowledge in order to resolve textual *ambiguities*. We achieve this by incorporating the weights coming from the instances-based mapping serve as a context for ranking categories with ambiguous names. For example, both book categories `Fiction by Country/Germany` and `Travel/Germany` have the same category name, but can be clearly disambiguated while looking at their instances.

4 Advanced Alignment Methods

The basic alignment described in the previous section maps each source category $i \in T_1$ to a set of candidate targets $j_1, j_2... \in T_2$ in isolation. That is, it considers neither the parent-child relations between the candidate targets, nor the correlation between the candidates. The methods introduced in this section are aimed at joined alignment between a pair of taxonomies.

For each pair of categories $i \in T_1$ and $j \in T_2$, which share at least one semantic label, we create a binary variable $A_{i,j}$. $A_{i,j}$ is set to 1 if categories i and j are aligned in the current solution. Otherwise, it is 0.

(a) Parent-child constraint violation.



(b) Anti-correlation constraint violation. Counterpart `Computer & Internet` correlates negatively with the rest of the linkings (`Klassiker` and `Historische Romane`).

Figure 2: Examples of constraint violations.

4.1 Alignment based on Integer Linear Programming

Objective Function

The primary goal is to find an alignment with the maximal weight. Linking between a pair of categories i and j, from T_1 and T_2 respectively, is weighted as in Formula 5. When considering all candidate linkings between T_1 and T_2, the objective is:

$$\max \sum_{\substack{i \in T_1, \\ j \in T_2}} w(i,j) \cdot A_{i,j} \tag{6}$$

It is obvious, that by setting all $A_{i,j} = 1$, the function reaches its maximal weight. This, however, can lead to inconsistent alignments. In the rest of this subsection we describe two types of inconsistencies and introduce constraints to counter them.

Constraints

Parent-Child Constraint (PCH)
Taxonomies organize their categories in hierarchies. When mapping different source categories to a target taxonomy, we could arrive at a situation where a parent-child relationship in the source taxonomy is reversed in the mapping to the target taxonomy. Figure 2a shows an example. We view such a situation as a violation of a parent-child constraint. We consider two cases:

a). A source category i is linked to targets j and k, and j is a (transitive) parent of k. For example, `Literature & Fiction` from amazon.com might be linked both to `Belletristik` and `Belletristik/Historische Romane`. Dropping the latter target category makes the candidate list more concise.

b). There is a pair of crossing links - a parent-child pair from the source taxonomy is linked to a child-parent pair in the target taxonomy. In this case, only one of the two linkages should be kept. In the example of Figure 2a, aligning the pair of categories (`Historical`, `Historische Romane`) should exclude the noisy pair (`Short Stories`, `Belletristik`) from a feasible solution.

We introduce a set of linear constraints in order to exclude the alignments violating the hierarchy relation. Expression 7 blocks linking category i both to j and j's parent. Thus, it tackles the violation of type a.

$$\begin{aligned} &\forall i \in T_1, j, k \in T_2 : \text{if } j \text{ is more general than } k \\ &\implies A_{i,j} + A_{i,k} \leq 1 \end{aligned} \tag{7}$$

The analogous constraint is added for a category $j \in T_2$ and a pair of categories $i, u \in T_1$, where i is more general than u.

In order to resolve the violation of type b, at most one linking from a pair of crossing links might enter a feasible solution.

$$\begin{aligned} &\forall i, u \in T_1, j, k \in T_2 : \text{if } i \text{ is more general than } u \text{ and} \\ &k \text{ is more general than } j \implies A_{i,j} + A_{u,k} \leq 1 \end{aligned} \tag{8}$$

Anti-Correlation Hard Constraint (ACH)
This set of constraints addresses another desirable property of taxonomy alignments. When mapping a source category i to multiple target categories $j_1, j_2 \ldots$, we expect the target categories to be semantically coherent. Figure 2b illustrates a situation where this is violated. Candidate target `Computer & Internet`, which is obviously a wrong match, is negatively correlated with the other two candidate targets. Dropping it makes the candidate list more coherent.

We formalize this intuition by computing the instance-based correlation between candidate targets. When two targets are negatively correlated, only one of them should be kept. This is specified by the following constraints:

$$\begin{aligned} A_{i,j} + A_{i,k} &\leq 1 \text{ if } corr(j,k) \leq 0 \\ A_{i,j} + A_{u,j} &\leq 1 \text{ if } corr(i,u) \leq 0 \end{aligned} \tag{9}$$

where $corr(x,y)$ is the Pearson's correlation coefficient between the instance vectors of the categories x and y:

$$corr(x,y) = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n \sum x_i^2 - (\sum x_i)^2} \cdot \sqrt{n \sum y_i^2 - (\sum y_i)^2}} \tag{10}$$

x_i expresses the number of occurrences of instance i in the category x to capture multiple occurrences of an instance in a category. Entries of y have analogous meaning.

Anti-Correlation Soft Constraint (ACS)
Forcing all candidate targets to be positively correlated may be too aggressive. Instead, we can relax the anti-correlation constraint and define a "soft" variant of it via a penalty or reward term in the objective function of the combinatorial optimization.

For a source category i and candidate targets j_1, j_2... the reward is the pairwise correlation between all target categories. We denote this by $corr_{T_2}$:

$$corr_{T_2/T_1} = \sum_{i \in T_1} \sum_{j \in T_2} \sum_{k \in T_2} corr(j,k) \cdot A_{i,j} \cdot A_{i,k} \quad (11)$$

In other words, $corr_{T_2/T_1}$ expresses the degree of coherence within the taxonomy T_2 when matching the classes of T_1 to the classes of T_2.

Analogously, we define the reward for pairwise correlation of the source categories that would be aligned with the same target. We denote this as $corr_{T_1/T_2}$. Note that negative correlations between category pairs in either the targets in T_2 or the sources in T_1 automatically reduces the value of the sum and thus results in a penalty.

Now, we extend the objective function, beyond merely maximizing the alignment weight, by maximizing the sum of the alignment weight and the two reward terms. The objective function of this model thus becomes:

$$\max[\sum_{\substack{i \in T_1, \\ j \in T_2}} w(i,j) \cdot A_{i,j} + corr_{T_1/T_2} + corr_{T_2/T_1}] \quad (12)$$

Note that the reward terms contain a product of decision variables. Since A variables are binary, one can easily convert this model into a linear model with linear constraints by introducing a new binary variable for each pair $(A_{i,j}, A_{i,k})$. It increases the dimensionality of the model, but makes it more expressive. Most of the state-of-the-art solvers like Gurobi are capable to deal with quadratic constraints and/or objective terms and aim to tighten the model formulation by, for example, presolving it and applying cutting planes algorithms[3].

5 Seeding Strategies

All the proposed methods are instances of an integer linear programming, which is known to be NP-hard in general. One way of dealing with large optimization instances is to solve a relaxation of the model. However, if we target the exact solution of the original problem, other approaches have to be studied. Consider an example of a model with two constraints:

$$A_{i,j} + A_{i,k} \leq 1 \quad (13)$$
$$A_{i,j} + A_{l,m} \leq 1 \quad (14)$$

The variables in the model are closely coupled by being combined in mutual exclusion constraints. Fixing a variable to value 1 propagates the computation of other variables in the model in a cascading manner. We propose to incorporate a small number of truth linkings into the reasoning model, guiding the solver towards the optimal solution.

[3] http://www.gurobi.com/resources/getting-started/mip-basics

Definition 4. We define a pair of perfectly matching categories i and j of two different taxonomies to be a seed. That is, the corresponding variable $A_{i,j}$ is set to 1 by a human annotator.

For example, the pair of categories (`Historische Romane`, `Genre Fiction/ Historical`) from the German and the English Amazon match perfectly.

In previous research the problem of providing a small number of seeds without sacrificing the performance of a classifier has been studied in the scope of semi-supervised learning Chapelle *et al.* (2010). In the context of the label propagation framework, seeds are nodes for which correct labels are provided. Lin and Cohen Lin and Cohen (2010) study the impact of selecting seeds based on network properties. The observation is, that "central" (or authoritative) nodes likely spread their influence in the network, so that annotating them will significantly improve the quality of a classifier.

In our study, we address not only the effectiveness of the seed categories with respect to the reasoning procedure, but also the amount of user involvement needed to find a matching counterpart. Our first observation is that some linkings are easier to detect for a human. On the other hand, seed categories can be scored by their impact in the model and the most influential ones be presented to a human annotator for labeling. Subsections 5.1 and 5.2 describe these two strategies.

5.1 Tree-based Seeding

When browsing through the product categories of online shops, one notices that some labeling decisions can be made instantly. Fig. 3 presents categories of two Amazon health departments (Germany- and US-based).

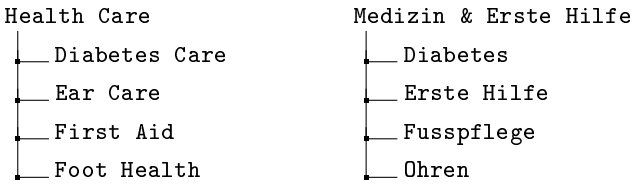


Figure 3: Examples of top level categories in Amazon's health departments (Germany- and US-based)

The matching categories can be detected by literal translation of category titles, and producing these alignments is not laborious. Generally, we assume that the top-level nodes are easier to annotate than the nodes deep in the taxonomy. This suggests the following strategy.

All categories in the source taxonomy are sorted according to their depth in descending order and the top k nodes are presented to a user for labeling. The ties are broken at random. This seeding rule has its limitations when the top-level categories of both taxonomies are orthogonal. In practice, one might go to the highest level at which a human annotator can make alignment decisions.

5.2 Impact-based Seeding

Despite the simplicity of labeling, following the depth-based strategy may have only small impact on run time.

Assume, all seeded categories appear only in one mutual exclusion constraint each. Therefore, by fixing k seed categories, we resolve at most k constraints. However, there might be categories participating in many constraints. The extended influence of these variables make them better seeds with respect to the the optimization model. Detecting the most influential seeds is the idea behind the impact-based seeding strategy. For labeling purposes, the top k categories scored by impact are presented to a human annotator.

Definition 5. The *impact-based score* of a source category i is calculated as the number of times the variables related to i participate in constraints.

In Inequality 13, both variables connect source category i with targets. For this constraint, $impact(i) = 2$. In Inequality 14 variables describe connections for two sources, i and l. Here $impact(i) = 1$ and $impact(l) = 1$. The total impact score for a source variable is summed up over all constraints in the model. By seeding the feedback on category i (e.g., $A_{i,j}$=1) both $A_{i,k}$ and $A_{l,m}$ get fixed to zero. In contrast, when fixing the ground truth for category l ($A_{l,m} = 1$), only $A_{i,j}$ is resolved to zero.

6 Experimental Evaluation

In order to evaluate the alignment quality of our methods, we performed experiments with different taxonomies and human judges for assessment.

6.1 Experimental Setup

We experimented with taxonomies covering three domains: health, books and software. Our experiments are based on data retrieved from amazon.com and amazon.de[4]. Amazon.com is the US-centric Web site of Amazon, while amazon.de represents its German "counterpart". Despite being part of the same enterprise, category names and category system are independently maintained and, thus, different.

In addition to the before mentioned alignments "within" Amazon, we add two additional data sets for the book domain: a well curated library catalog from the German National Library, dnb.de, based on the Dewey Decimal Classification (DDC). As for contrasting, we incorporate the social tagging community shelfari.com, which is based on a community-created taxonomy. Thus, the taxonomies are very different in nature. First, they have different *curation levels*, ranging from manually curated up to social tagging. Second, they are culture specific based on their different *origin*. Third, they differ in their *sizes* varying from a broad 10,000 categories (shelfari.com) to focused 150 categories (amazon.de, health branch). Table 1 summarizes data set properties.

[4]Health domain: "Health Care" and "Medizin & Erste Hilfe"; books domain: "Books" and "Bücher"; software domain: "Software" in both stores

We now describe how the intermediate taxonomies were used. We consider each instance or category title as a query and retrieve relevant Wikipedia pages using its API. We perform both - title and text search. The top k retrieved results become semantic labels (in the experiments we set $k = 5$). From our manual inspection, we observe that setting k larger blows up the set of semantic labels, which are in many cases noisy. When aligning two taxonomies in different languages, labels of the source language are converted to the target language by following inter-language link. If there is no inter-language link for a search result, this Wikipedia page is disregarded.

Three judges participated in manual evaluation of the generated alignments. Each taxonomy pair was evaluated by two of them on a random sample of 100 categories. Alignment output of each method was annotated as *matching* or *wrong*. The annotators were instructed to mark as *matching* all relevant counterparts. I.e., both categories `Classical Hellenic Poetry and Drama` and `Hellenic Literatures` are considered to be matching for `Drama/Greek and Roman`. Cohen's kappa of the inter-annotator agreement is 0.69, which is considered to be fairly good Landis and Koch (1977).

6.2 Methods

We have the following models under comparison:

1. baseline (Section 3.3),
2. ACROSS with enabled constraint-aware reasoning,
3. ACROSS with seeding,
4. The *WikiMatch* (Hertling and Paulheim, 2012) approach makes a look-up in Wikipedia to align two input taxonomies. For a given category title as input query, it retrieves the results from the Wikipedia search engine. The similarity between two categories is expressed as the Jaccard similarity over the Wikipedia articles returned for each category. Following the inter-language links provided by Wikipedia allows WikiMatch to compare two data sources from different languages.
5. The *S-Match* (Giunchiglia *et al.*, 2004) method reconstructs logical formulas for each category in the taxonomy. For example, category `History/Europe` is converted to the logical formula `History AND Europe`. A correspondence between two categories is found by comparing their logical formulas. We run *S-Match* with the "Structure Preserving Semantic Matching" option, which respects structural properties such as matching leaves only with leaves and internal nodes only with internal nodes.

6.3 Measures

We introduce the quality measures by which we compare the effectiveness of different alignment methods.

Let S be the set of source categories in the sample set for assessments. For a category $i \in S$, let $C(i)$ be the ranked list of target categories that are generated by some method.

Table 1: Properties of the used taxonomies.

Source	Domain	Categories	Instances	Market	Source	Domain	Categories	Instances	Market
amazon.de (Health)	Health	150	116,000	German	amazon.de (Books)	Books	8,293	962,000	German
amazon.com (Health)	Health	198	435,000	US	amazon.com (Books)	Books	5,846	1,754,000	US
amazon.de (Software)	Software	701	125,000	German	dnb.de	Books	910	1,720,000	German
amazon.com (Software)	Software	281	100,000	US	shelfari.com	Books	12,803	1,173,000	US

Table 2: Examples of trivial and non-trivial alignments.

Use-case	Trivial	Non-Trivial
amazon.de ↔ amazon.com (Health)	Alternative Medizin ↔ Alternative Medicine Erste Hilfe ↔ First Aid	Diabetes/Injektionsspritzen & -kanülen ↔ Insulin Injectors Schlafen & Beruhigung ↔ Sleep & Snoring
amazon.de ↔ amazon.com (Software)	Betriebssysteme ↔ Operating Systems Sprachen ↔ Languages	Homebanking & Money Management ↔ Budgeting Aktien & Börse ↔ Investment Tools
dnb.de ↔ shelfari.com (Books)	Sozialwissenschaften ↔ Social Sciences Ethik ↔ Ethics & Morality	Der politische Prozess ↔ Political Theory Bildhauerkunst, Keramik, Metallkunst ↔ Sculpture
shelfari.com ↔ amazon.com (Books)	Speech Processing ↔ Speech Processing Science & Math ↔ Science	Latin America ↔ Argentina Mountain Biking ↔ Cycling

Categories in $C(i)$ are ranked by the alignment weight (see Formula 5) in decreasing order.

Since i is linked to a ranked list of target candidates, we consider standard information retrieval measures provided by TREC evaluation script[5].

1. **Mean Reciprocal Rank (MRR).** We are interested in at which position in the ranked list of output categories we see the first match. Let i be a source category and r the rank of a match. Then, the reciprocal rank is $RR(i) = \frac{1}{r}$ If no match exists, then $RR(i) = 0$.

   For a sample of $|S|$ source categories, the MRR value is defined as:

$$MRR = \frac{1}{|S|} \cdot \sum_{i \in S} RR(i) \tag{15}$$

2. **Mean Average Precision**(MAP) captures the accumulated precision over all ranked target categories at different recall levels:

$$MAP = \frac{1}{|S|} \sum_{i \in S} \frac{1}{S_i} \sum_{k \in S_i} precision(C(i,k)) \tag{16}$$

   where $|S|$ is the sample size, S_i is the set of correct counterparts for source i and $C(i,k)$ is the ranked list of targets for i with cut-off rank k. We report on MAP with cut-off at rank 5.

   Note, that if a method did not return a matching candidate $j \in S_i$, precision value is taken to be zero.

3. **Success@1** measures the portion of sources for which a correct counterpart was produced at rank 1:

$$success@1 = \frac{1}{|S|} \cdot \sum_{i \in S} precision(C(i,1)) \tag{17}$$

4. **Utility** is an unnormalized set utility measure, expressing how noisy is the list of retrieved documents. It rewards the method with α points for finding a correct match and penalizes with β points for retrieving an irrelevant counterpart.

   For a source i, the utility of its counterpart list $C(i)$ is:

$$utility(C) = \alpha \cdot \text{No. of relevant counterparts} - \beta \cdot \text{No. of non-relevant counterparts} \tag{18}$$

   The final utility score for a method is computed as average utility over S. In our experiments, we set $\alpha = \beta = 1$.

5. **Coverage** expresses the number of source categories which were aligned with at least one matching counterpart.

[5] http://trec.nist.gov/trec_eval

6.4 Setup

All the methods under consideration rely on the similarity between the category titles. This can bias them towards producing "trivial matches" - nearly word-by-word translations. Therefore, we separate annotated examples into two groups - trivial and non-trivial matches and run the evaluation separately. A source category is said to have a trivial match if there is an unambiguous counterpart which can be detected by translation.

Using the Yandex machine translation service[6], we cast all the titles of German taxonomies into English. The tokens of the titles are lemmatized and sorted, such that the matching between `Benjamin Franklin (president)` and `Presidents: Franklin, Benjamin` can be restored. If such a counterpart can not be found, we claim the source being a non-trivial case. Some examples of non-trivial and trivial alignment cases are given in Table 2.

Note that, category `Travel Guides/Europe` from the books department belongs to the non-trivial case as well. Although counterparts can be found by simple translation, they are ambiguous: `Cooking by Continent/Europe`, `Religion/Europe` or `Traveling/Europe`. Such categories belong to the non-trivial cases, since there is a need for disambiguation procedure.

6.5 Results

The experimental results for different taxonomy pairs are given in Table 3. Results cover the full range of alignments of taxonomies with respect to size, curation level and origin. Plots on Figure 4 summarize the percentage of source categories which could be covered by at least one counterpart depending on method and category's depth in the taxonomy.

Across all the experiments, we observe that for non-trivial cases the performance of all methods degrades whereas trivial alignments can be restored by any method with fairly high MAP@5, MRR and success@1 values.

We now discuss our findings on the strengths and weaknesses of each method separately.

WikiMatch outputs high quality alignments in terms of MRR, success@1 and utility for almost all of the use-cases. However, when considering two taxonomies with dissimilar categorization criteria and category titles, WikiMatch does not ensure high coverage for cross-lingual scenarios for non-trivial cases (for example, alignment between amazon.com ↔ amazon.de on Health). The absence of a reasoning or an alignment repair step leads WikiMatch to incoherent counterparts for category `Religion` in amazon.de (Books) (cf. Table 4, second row), where all the counterparts are aligned with the highest weight (1.0). The authors of WikiMatch discuss this limitation as well. WikiMatch also experiences difficulties when aligning categorization schemes with different naming criteria, e.g. shelfari.com ↔ dnb.de. The semantic relatedness between categories `Drama/Greek and Roman` and `Classical Hellenic Poetry and Drama` could not be resolved. This leads to lower MAP@5 values for the non-trivial cases.

We run the *S-Match* software on the amazon.com ↔ shelfari.com use case only, since it is not capable to deal with multi-lingual input taxonomies. S-Match performs well on the sources which have a target with similar tree path and category names along this path, therefore the correct counterpart `History/Europe` is taken and the wrong candidate `Travel/Europe` is eliminated. Slight modifications in wording or tree path decrease recall by filtering out candidates. Dealing with language varieties implies involving additional resources such as WordNet. S-Match ensures non-zero coverage for all levels in the shelfari taxonomy, however only 6% of the leaf categories got matched with a counterpart.

It is worth mentioning that, both, S-Match and WikiMatch, do not consider instances of the categories while constructing an alignment.

Baseline. Our baseline solution reaches fairly high MAP and MRR values (up to 0.72 of MRR for dnb.de → shelfari.com for non-trivial cases). Since we rely on the instances for inferring the semantics of a category, our basic alignment procedure ensures better coverage. For all the use-cases, all the taxonomic levels could be provided with counterparts, covering more than 90% of the second-level categories in the health, software and books (shelfari.com → amazon.com) experiments. For non-trivial cases in the health domain experiments, ACROSS outperforms WikiMatch by producing correct alignments for 62 source categories versus 8.

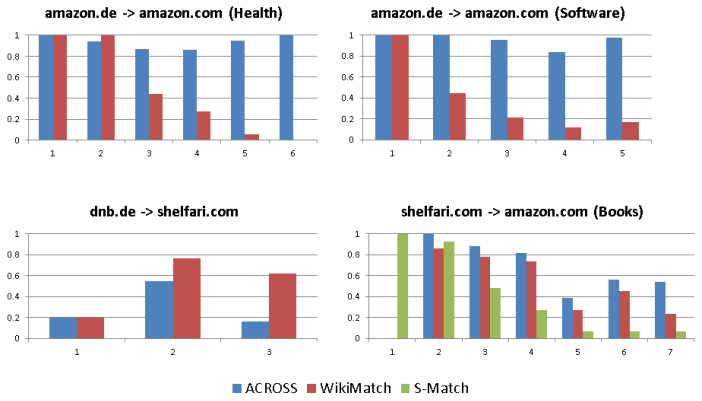


Figure 4: Percentage of source categories per taxonomy level covered by different methods - comparing baseline ACROSS matching with WikiMatch and S-Match.

Plots on Figure 4 illustrate that baseline ACROSS covers categories on all levels in the taxonomy. In the experiments on Health and Software domain, it produced counterparts for more than 80% of categories on depth 3-6. In the shelfari.com ↔ amazon.com experiment, both baseline ACROSS and WikiMatch failed to produce the correct linking between roots "All Books"(shelfari.com) and "Books"(amazon.com). The results of related pages returned from Wikipedia search API for queries "All Books" and "Books" are dissimilar, which leads to almost zero Jaccard coefficient. On the instances level, categories contain representative books from all child categories. Therefore,

[6] https://tech.yandex.com/translate/. Our choice of the translation tool was motivated by the volume of data one can translate using free service. For Yandex it is 10,000,000 characters/month (as compared to 2,000,000 characters/month for Microsoft Translate).

Table 3: Results for trivial and non-trivial cases.

Method	TRIVIAL CASES					NON-TRIVIAL CASES				
	MAP@5	**MRR**	**Success@1**	**Utility**	**Relevant Matches**	**MAP@5**	**MRR**	**Success@1**	**Utility**	**Relevant Matches**
amazon.de → amazon.com (Health)										
WikiMatch	0.78	0.78	0.78	-1	11	0.26	0.40	0.38	-1.6	8
Baseline	0.76	0.84	0.78	-93.56	**22**	0.32	0.37	0.28	-84.00	**62**
ACROSS	0.75	0.86	0.86	0.17*	20	0.39*	0.44*	0.44*	**-0.22***	49
+ tree-based seeds	0.75	0.84	0.84	0.52*	16	0.38*	0.44*	0.43*	-0.25*	46
+ impact-based seeds	0.80*	0.89	0.89*	**0.63***	17	0.38*	0.45*	0.45*	-0.26*	47
ACROSS SOFT	**0.88**	**1.00**	**1.00**	0.61*	18	**0.43**	**0.49**	**0.49**	-0.68*	20
amazon.de → amazon.com (Software)										
WikiMatch	0.74	0.79	0.75	**1**	10	**0.26**	0.31	0.31	**-1.12**	10
Baseline	**0.75**	**0.84**	0.77	-25.5	**17**	**0.26**	0.36	0.27	-35.61	**68**
ACROSS	0.72	0.80	0.77	0.33*	15	**0.26**	0.37	0.32	-1.64*	44
+ tree-based seeds	0.68	0.78	0.75	0.31*	13	0.24	0.33	0.29	-1.64*	40
+ impact-based seeds	0.70	0.79	0.76	0.35*	14	0.25	0.34	0.30	-1.64*	40
ACROSS SOFT	0.73	0.83	**0.83**	-0.72*	15	**0.26**	**0.45**	**0.38**	-2.73	35
amazon.de → amazon.com (Books)										
WikiMatch	0.12	0.50	0.46	-1.30	7	0.14	0.30	0.25	-8.43	24
Baseline	0.14	0.54	0.46	-88.23	**12**	0.11	0.41	0.35	-46.69	**52**
ACROSS	0.16	0.60	0.60	**0.20***	6	**0.17**	0.36	0.33	**-0.87***	24
+ tree-based seeds	0.14	0.41	0.33	-7.41*	8	0.10	0.38	0.33	-3.56*	26
+ impact-based seeds	**0.57***	**0.73***	**0.71***	-1.85*	6	0.11	0.41	0.37	-3.13*	34
ACROSS SOFT	0.23	0.72	0.70	-1.60*	8	0.10	**0.49**	**0.47**	-0.94*	28
shelfari.com → amazon.com (Books)										
S-Match	**0.68**	**0.87**	**0.87**	**0.75**	7	0.47	**0.88**	**0.88**	**0.77**	24
WikiMatch	0.62	0.82	0.82	0.63	**38**	**0.58**	0.72	0.57	-1.04	**47**
Baseline	0.52	0.72	0.71	-1.17	37	0.48	0.61	0.44	-2.38	**47**
ACROSS	0.53	0.75	0.75	0.50	27	0.39	0.61	0.60	0.34	22
+ tree-based seeds	0.55	0.80	0.80	0.60	37	0.45	0.67	0.64	0.57	22
+ impact-based seeds	0.56	0.81	0.81	0.62	26	0.47	0.68	0.66	0.53	23
ACROSS SOFT	0.60	0.82	0.81	-0.40	34	0.52	0.73*	0.68*	-1.25	43
dnb.de → shelfari.com (Books)										
WikiMatch	0.19	0.54	0.66	0.41	**17**	0.12	0.38	0.26	-11.23	51
Baseline	0.19	0.55	0.40	0.3	**17**	**0.29**	0.72	0.64	-1.60	**67**
ACROSS	0.13	0.50	0.50	0.25	10	0.18	0.65	0.62	0.55*	49
+ tree-based seeds	**0.23**	**1.00***	**1.00***	**1.71**	14	0.26	**0.96***	**0.95***	**1.53***	46
+ impact-based seeds	0.19	0.82*	0.82*	1.23	14	0.24	0.85*	0.83*	1.10*	48
ACROSS SOFT	0.19	0.70*	0.65*	1.2	15	0.22	0.69	0.68	0.98*	50

* Improvement over the baseline is significant at 0.05 level (paired one-tailed t-test).

cosine similarity over instances sets was also below pruning threshold.

ACROSS with enabled constraint-aware reasoning increases the utility (purity) of the counterpart recommendations, reaching 0.55 for the dnb.de → shelfari.com non-trivial use case. It also provides users with more correct counterparts at rank one, outperforming the baseline by more 16% for amazon.de → amazon.com (health domain) case over non-trivial instances.

Table 5 illustrates how seeding affects run times in comparison to the ACROSS reasoning without seeds. The seed selection methods are described in Sections 5.1 and 5.2. For all settings, 10 pairs of matching categories (i.e., 10 variables) were provided as seeds with the following total number of variables:

1. amazon.de ↔ amazon.com (Health) - 11,638
2. amazon.de ↔ amazon.com (Software) - 13,993
3. amazon.de ↔ amazon.com (Books) - 49,647
4. shelfari.com ↔ amazon.com (Books) - 55,170
5. dnb.de ↔ shelfari.com (Books) - 1,506

Table 4: Anecdotic examples of found alignments.

Source Category	WikiMatch	ILP(best configuration)
Drama/Greek and Roman	–	Classical Hellenic Poetry and Drama Hellenic literatures
Biografien & Erinnerungen/ Religion (en.: Biographies & Memoirs/Religion)	Encyclopedias/Religion Humor & Entertainment/Religion Children's Books/Religions	Biographies & Memoirs/Luther, Martin
Crafts, Hobbies & Home/Scrapbooking	Home and Garden/Scrapbooking	Home and Garden/Scrapbooking
Alternative Medicine/Single Homeopathic Remedies	Homöopathische Einzelwirkstoffe (en.: Homeopathic Individual Active Substances)	Akupunktur (en.: Acupuncture) Alternative Medizin (en.: Alternative Medicine)

Table 5: Run time for solving ILP models.

Setting	amazon.de ↔ amazon.com (Health)	amazon.de ↔ amazon.com (Software)	dnb.de ↔ shelfari.com (Books)	shelfari.com ↔ amazon.com (Books)	amazon.de ↔ amazon.com (Books)
ACROSS	47.85	74,790.07	0.10	1,175.34	324,854.24
+ tree-based seeds	35.40	2,795.72	**0.05**	1,071.72	189,624.05
+ impact-based seeds	**25.95**	**1,648.67**	**0.05**	**874.93**	**35,810.88**

Incorporating only a small number of seeds drastically reduces the run times for complex cases, when reasoning has to be run over very noisy data or large taxonomies. For example, in the experiments over Software domain, ACROSS had to reason over 22 targets per source category on average, whereas for the shelfari.com → amazon.com only upon 8 targets per source on average. Following the impact-based seed selection strategy had the largest impact on bringing run times down. For the experiments over the Software domain, the run times were reduced by factor 45. In addition, the seeding step slightly improves linking quality, by raising MAP@5 to 0.8 for trivial alignments in the health domain and for utility in almost all experiments (see Table 3).

For all the experiments, ACROSS performed best in terms of MRR and success@1, when the anti-correlation constraint was softened. It is explained by Formula 9. When a target candidate got fixed, only a few other targets for the same source may enter the final solution. The *ACROSS SOFT* configuration penalizes a solution when non-correlating targets are assigned to a source, rather than aggressively filtering them out. For the health domain, combination of those two constraints improves MRR values up to 1.0.

7 Related Work

Data integration

Providing unified access to heterogeneous data sources is the overriding goal of the field of data integration (Doan *et al.*, 2012). A key task in data integration is to map between global (mediation) schemas and local schemas of the underlying sources. Similarly to the this situation, we align different sources from the same domain such as books. However, the notion of taxonomies that we consider here is quite different from database schemas. Moreover, the size and cultural diversity of our input taxonomies makes them unsuitable for the prevalent methods in schema mappings, which are either rule-based or use machine learning.

Catalog Integration

Our problem is similar in spirit to the task of integrating catalogs. Agrawal and Srikant (2001) find for each item in the source catalog an appropriate category in the master catalog. However, they make several strong assumptions: items are assigned only to the leaf categories, a document from the source is assigned to exactly one master category, there are common items in both catalogs through which the categorizations similarity is computed. In contrast to this work, we compute alignments on the level of categories and return a ranked list of counterparts. We utilize hierarchical relations as well and conduct experimental evaluations on product catalogs, which are of different nature as Internet directories of Web links.

Ichise *et al.* (2003) provide a framework for integrating two Internet directories by instance-based learning and determining mapping rules. Determining equivalence relations between two categories is based on a set of instances (Web links) common to both directories. In contrast, ACROSS can align two taxonomies even if they do not share instances via its semantification rules. Semantic coordination of hierarchical structures

is discussed by Bouquet *et al.* (2003). Category titles are converted into logical formulas taking into account lexical, domain and structural knowledge. In contrast to this, we infer the category's semantics from the instances it is populated with and do not depend on any word sense disambiguation.

Ontology Alignment

This field typically considers logically rigorous ontologies like OWL or RDF schemas along with the instances of classes and properties (Staab and Studer, 2013). There is a wealth of prior work on ontology alignment in this spirit; representatives and overviews include (Udrea *et al.*, 2007; Suchanek *et al.*, 2011; Euzenat and Shvaiko, 2013; Euzenat, 2014; Giunchiglia *et al.*, 2004). The Ontology Alignment Evaluation Initiative[7] provides test cases of two kinds: different ontologies translated into different languages and same ontologies translated into different languages. In our settings, we aim at bridging the gap between multicultural sources - similar to the test cases of the first sort. The difference to our work is that we focus on aligning taxonomies rather than full ontologies. Moreover, our inputs consist of 10,000's of categories, which is different in scale to the 1000's of schema elements that ontology alignment methods usually consider.

The WikiMatch system (Hertling and Paulheim, 2012) uses Wikipedia as an external source for aligning two ontologies. Concepts from the ontologies (i.e., classes, entities, or properties) are annotated with Wikipedia articles to which the concept names can be mapped. However, we do not rely solely on the surface name of a category while mapping to Wikipedia. We analyse the instances of the categories as well. This allows us to differentiate between book categories `Physics/Reference` and `Psychology/Reference`, whereas in WikiMatch both categories can not be properly disambiguated. Additionally, we introduce a set of constraints to filter out non-matching target candidates.

The WeSeE-Match tool (Paulheim, 2012) performs multilingual ontology alignment, based on computing string similarities of the translated titles. In contrast to this approach, we do not use any translation tools because of the culture-specific entity titles (e.g., book titles), inaccuracies in transliteration (e.g., for author names) and culture-specific categories. If a concept is aligned to several counterparts, WeSe-Match uses edit distance for reasoning the ranking. ACROSS uses more sophisticated resolution scheme relying on correclation between instances and catalog structures.

Wick *et al.* (2008) utilize conditional random fields to model the ontology interlinking. This approach incorporates first-order logic rules and allows finding multiple counterparts for a class. In contrast to this machine-learning model, our method does not need any training data.

Spaniol *et al.* (2013) describe a knowledge linking system for online statistics. This tool uses a mapping between two knowledge collections – eurostat Statistics Explained and Wikipedia – to generate links Our ACROSS systems extends this approach by considering sources of different cultural nature, and by devising a much more powerful alignment method based on combinatorial optimization and label propagation models.

Conservativity and consistency principles has been studied in previous works. Cleaning the basic alignment with ACROSS constraints is similar in spirit to the repair step of the following approaches, however without (directly) using matching weight and degree of correlation between categories. ALCOMO (Meilicke, 2011) is a library providing several alignment debugging procedures. It explicitly models finding a minimal repair to bring the alignment into consistent state. The cardinality of the filtered out matchings is out of the scope of this work. ACORSS focuses on producing the final alignment with the highest possible weight. In contrast to LogMap (Jiménez-Ruiz and Cuenca Grau, 2011), the filtered out alignments are not necessarily those of the minimal weight. YAM++ (Duchateau *et al.*, 2009) is a machine learning-based system, including a large variety of classifiers and relying on ALCOMO diagnosis library for inconsistency checking.

Incorporating background knowledge into the matching procedure has been addressed in (Aleksovski *et al.*, 2006; Sabou *et al.*, 2008). Utilizing either a set of domain-specific taxonomies or a set of ontologies from the Linked Open Data cloud was shown to drastically boost the performance of a matcher when the input data sources are highly heterogeneous and have low overlap on the lexical level. Both approaches target aligning two plain lists rather than ontologies or taxonomies. Sabou *et al.* (2008) emphasize utilizing a large number of possibly heterogeneous intermediate ontologies and combining annotations returned from these intermediate ontologies to produce alignments. Selecting the intermediate ontologies can also be automatic, rather than by laborious manual selection. In our settings, ACROSS obtains semantic labels from Wikipedia editions, which are fixed prior to the alignment procedure. In line with Sabou's approach, relevant concepts from the intermediate ontologies are retrieved by a search engine, rather than using a set of linguistic matching rules. We do not explicitly filter out noisy semantic labels and address this problem by entrusting the Wikipedia search engine and by using weighting schemes like TFIDF. Aleksovski *et al.* (2006) target detecting relations between concepts of two plain lists via relations of their anchors in the intermediate ontology. To find matching concepts in the intermediate ontology, simple lexical heuristics are used. Incorporating several Wikipedia editions when aligning multilingual taxonomies places ACROSS in line with the both approaches, which use an ensemble of intermediate ontologies. In line with both approaches, we use an ensemble of mediators (several Wikipedia editions). Nevertheless, ACROSS has a fundamental difference from these approaches. We focus on taxonomies, which are richer in structure than plain lists. This structure is exploited for filtering out noisy alignments.

Multilingual Data and Knowledge Alignment

The LAIKA system (Göbölös-Szabó *et al.*, 2012) aims to find missing links between categories across Wikipedia editions. It also utilizes link structures of several language-specific Wikipedias to enrich article-category and article-article links within a given Wikipedia edition. However, LAIKA needs input tax-

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

onomies where instances are already interlinked between two language- or culture-specific editions, which differs from our setting. Moreover, LAIKA is specifically geared for Wikipedia, whereas we consider taxonomies from a wide variety of non-Wikipedia sources.

Nguyen *et al.* (2011) describes a mapping Wikipedia infoboxes across different editions. Values of infobox attributes are represented as judiciously constructed feature vectors in the underlying Wikipedia. Following the cross-lingual interwiki links allows two attributes from different languages to be compared. In addition, link-structure similarity, correlation similarity, and infobox types are used to compute alignments between infobox fields. In contrast, our setting focuses on categories, which are disregarded in (Nguyen *et al.*, 2011). Moreover, we address a wide variety of taxonomies beyond Wikipedia.

Gracia *et al.* (2012) discuss challenges arising from multilingual data in Linked Open Data (Heath and Bizer, 2011). Our work is orthogonal to these issues: we focus on culture-specific category systems, not on RDF triples and entity linkage.

Spohr *et al.* (2011) describe an approach to multilingual and cross-lingual ontology matching. A set of structural and string similarity features is fed into SVM algorithm. We do not use any learning algorithm, respecting structural and textual similarities of aligned categories though.

8 Conclusions and Future Work

We presented the ACROSS system for aligning multi-cultural knowledge taxonomies. Our unique method maps all categories jointly and considers constraints to arrive at high-quality mappings, using integer linear programming. ACROSS incorporates a search-based semantification procedure in order to overcome language varieties without involving any language-dependent synonym resolution. Our comprehensive experiments show that ACROSS clearly outperforms a simpler baseline that considers only pairwise similarity in terms of semantic-label vectors. Additionally, we have studied two approaches to incorporate user feedback in order to limit the run times for our exact reasoning procedure.

As for future work, we will look into the joint alignment of categories and entities, especially for the challenging cases that involve long-tail entities which are not in Wikipedia.

References

Agrawal, R. and R. Srikant. 2001. "On Integrating Catalogs". In: *Proceedings of the Tenth International Conference on World Wide Web*. 603–612.

Aleksovski, Z., M. Klein, W. Ten Kate, and F. Van Harmelen. 2006. "Matching Unstructured Vocabularies Using a Background Ontology". In: *Proceedings of the Fifteenth International Conference on Knowledge Engineering and Knowledge Management*. 182–197.

Auer, S., C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak, and Z. Ives. 2007. "DBpedia: A Nucleus for a Web of Open Data". In: *Proceedings of the Sixth International Semantic Web Conference,* 722–735.

Boldyrev, N., M. Spaniol, and G. Weikum. 2016. "ACROSS: A Framework for Multi-cultural Interlinking of Web Taxonomies". In: *Proceedings of the Eighth International ACM Web Science Conference*. 127–136.

Bollacker, K., C. Evans, P. Paritosh, T. Sturge, and J. Taylor. 2008. "Freebase: a Collaboratively Created Graph Database for Structuring Human Knowledge". In: *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*. 1247–1250.

Bouquet, P., L. Serafini, and S. Zanobini. 2003. "Semantic coordination: a new approach and an application". In: *Proceedings of the Second International Semantic Web Conference*. 130–145.

Carlson, A., J. Betteridge, B. Kisiel, B. Settles, E. R. Hruschka Jr, and T. M. Mitchell. 2010. "Toward an Architecture for Never-Ending Language Learning". In: *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*. 1306–1313.

Chapelle, O., B. Schlkopf, and A. Zien. 2010. *Semi-Supervised Learning*. 1st. The MIT Press.

Doan, A., A. Halevy, and Z. Ives. 2012. *Principles of data integration*. Elsevier.

Duchateau, F., R. Coletta, Z. Bellahsene, and R. J. Miller. 2009. "(Not) yet another matcher". In: *Proceedings of the 18th ACM Conference on Information and Knowledge Management*. ACM. 1537–1540.

Euzenat, J. 2014. "First experiments in cultural alignment repair". In: *Proceedings of the Third International Workshop on Debugging Ontologies and Ontology Mappings*. 3–14.

Euzenat, J. and P. Shvaiko. 2013. *Ontology matching*. 2nd. Heidelberg: Springer-Verlag.

Fader, A., S. Soderland, and O. Etzioni. 2011. "Identifying relations for open information extraction". In: *Proceedings of the 2011 Conference on Empirical Methods on Natural Language Processing*. 1535–1545.

Fellbaum, C. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Giunchiglia, F., P. Shvaiko, and M. Yatskevich. 2004. "S-Match: an Algorithm and an Implementation of Semantic Matching". In: *Proceedings of the First European Semantic Web Symposium*. 61–75.

Göbölös-Szabó, J., N. Prytkova, M. Spaniol, and G. Weikum. 2012. "Cross-Lingual Data Quality for Knowledge Base Acceleration across Wikipedia Editions". In: *Proceedings of the Tenth International Workshop on Quality in Databases*. 1–7.

Gracia, J., E. Montiel-Ponsoda, P. Cimiano, A. Gómez-Pérez, P. Buitelaar, and J. McCrae. 2012. "Challenges for the Multilingual Web of Data". *Web Semantics: Science, Services and Agents on the World Wide Web*. 11: 63–71.

Heath, T. and C. Bizer. 2011. "Linked Data: Evolving the Web into a Global Data Space". *Synthesis Lectures on the Semantic Web*. 1(1): 1–136.

Hertling, S. and H. Paulheim. 2012. "WikiMatch: Using Wikipedia for Ontology Matching". In: *Proceedings of the Seventh International Workshop on Ontology Matching*. 37–48.

Ichise, R., H. Takeda, and S. Honiden. 2003. "Integrating Multiple Internet Directories by Instance-based Learning". In: *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*. Vol. 3. 22–28.

Jiménez-Ruiz, E. and B. Cuenca Grau. 2011. "LogMap: Logic-Based and Scalable Ontology Matching". In: *Proceedings of the Tenth International Semantic Web Conference*. 273–288.

Kingsbury, P. and M. Palmer. 2002. "From TreeBank to PropBank." In: *Proceedings of the Third International Conference on Language Resources and Evaluation*. 1989–1993.

Landis, J. R. and G. G. Koch. 1977. "The Measurement of Observer Agreement for Categorical Data". *Biometrics*. 33(1): 159–174.

Lin, F. and W. W. Cohen. 2010. "Semi-Supervised Classification of Network Data Using Very Few Labels". In: *Proceedings of the 2010 International Conference on Advances in Social Network Analysis and Mining*. 192–199.

Meilicke, C. 2011. "Alignment Incoherence in Ontology Matching. PhD Thesis, Universität Mannheim".

Nakashole, N., G. Weikum, and F. Suchanek. 2012. "PATTY: A taxonomy of relational patterns with semantic types". In: *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*. 1135–1145.

Navigli, R. and S. P. Ponzetto. 2012. "BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network". *Artificial Intelligence*. 193: 217–250.

Nguyen, T., V. Moreira, H. Nguyen, H. Nguyen, and J. Freire. 2011. "Multilingual Schema Matching for Wikipedia Infoboxes". *Proceedings of the Thirty-Eight International Conference on Very Large Databases*. 5(2): 133–144.

Palmer, M., D. Gildea, and P. Kingsbury. 2005. "The proposition bank: An annotated corpus of semantic roles". *Computational Linguistics*. 31(1): 71–106.

Paulheim, H. 2012. "WeSeE-Match results for OEAI 2012". In: *Proceedings of the 7th International Conference on Ontology Matching*. 213–219.

Sabou, M., M. d'Aquin, and E. Motta. 2008. "Exploring the Semantic Web as Background Knowledge for Ontology Matching". In: *Journal on Data Semantics XI*. Springer Berlin Heidelberg. 156–190.

Singhal, A. 2012. "Introducing the Knowledge Graph: Things, Not String. Official Blog of Google".

Solimando, A., E. Jiménez-Ruiz, and G. Guerrini. 2014. "Detecting and correcting conservativity principle violations in ontology-to-ontology mappings". In: *Proceedings of the Thirteenth International Semantic Web Conference*. Springer. 1–16.

Spaniol, M., N. Prytkova, and G. Weikum. 2013. "Knowledge Linking for Online Statistics". In: *Proceedings of the Fifty-Ninth World Statistics Congress of the International Statistical Institute*.

Spohr, D., L. Hollink, and P. Cimiano. 2011. "A Machine Learning Approach to Multilingual and Cross-Lingual Ontology Matching". In: *Proceedings of the Tenth International Semantic Web Conference*. 665–680.

Staab, S. and R. Studer. 2013. *Handbook on ontologies*. Springer Science & Business Media.

Suchanek, F. M., S. Abiteboul, and P. Senellart. 2011. "PARIS: Probabilistic Alignment of Relations, Instances, and Schema". *Proceedings of the Thirty-Eight International Conference on Very Large Databases*. 5(3): 157–168.

Suchanek, F. M., G. Kasneci, and G. Weikum. 2007. "YAGO: a Core of Semantic Knowledge". In: *Proceedings of the Sixteenth International World Wide Web Conference*. 697–706.

Udrea, O., L. Getoor, and R. J. Miller. 2007. "Leveraging Data and Structure in Ontology Integration". In: *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*. 449–460.

Wick, M. L., K. Rohanimanesh, A. McCallum, and A. Doan. 2008. "A Discriminative Approach to Ontology Mapping." In: *Proceedings of the 2008 International Workshop on New Trends in Information Integration*. 16–19.

Yahya, M., S. Whang, R. Gupta, and A. Y. Halevy. 2014. "ReNoun: Fact Extraction for Nominal Attributes." In: *Proceedings of the 2014 Conference on Empirical Methods In Natural Language Processing*. 325–335.

Natalia Boldyrev obtained her doctoral degree in October 2017 from Saarland University, Germany. She was doing her doctoral research at Max Planck Institue for Informatics in the area of alignment of heterogeneous knowledge repositories.

Marc Spaniol is a full professor at University of Caen Normandie, France. He is co-organizer of the Temporal Web Analytics Workshop (TempWeb) series. His research interests are in the area in the field of Web science, Web data quality, temporal Web analytics and knowledge evolution.

Gerhard Weikum is leading the department on databases and information systems at the Max Planck Institute for Informatics in Saarbrücken, Germany. His research spans transactional and distributed systems, self-tuning database systems, data and text integration, and the automatic construction of knowledge bases.