MATCHING AND MERGING ANONYMOUS TERMS FROM WEB SOURCES

Kun Ji[1], Shanshan Wang[2] and Lauri Carlson[3]

[1,2,3]Department of Modern Languages, University of Helsinki, Helsinki, Finland

ABSTRACT

This paper describes a workflow of simplifying and matching special language terms in RDF generated from trawling term candidates from Web terminology sites with TermFactory, a Semantic Web framework for professional terminology. Term candidates from such sources need to be matched and eventually merged with resources already in TermFactory. While merging anonymous data, it is important not to lose track of provenance. For coding provenance in RDF, TF uses a minor but apparently novel variant of RDF reification. In addition, TF implements a toolkit of methods for dealing with graphs containing anonymous (blank) nodes.

KEYWORDS

RDF, provenance, anonymous/blank nodes, LSP, professional terminology work

1. INTRODUCTION

Collaborative human-to-human dictionary work based on crowdsourcing has produced success stories like Wiktionary, but Web 3.0 tools have as yet made little impact in the everyday business of professional terminologists. Professional terminoloy work in the Austrian (ISO/TC 37) tradition [1] starts with concept analysis in a given subject field and proceeds from there to standardization and/or description of the concepts and terms designating them. It serves but is distinct from translation related terminology work which consists of finding multilingual equivalents for terms occurring in text.

TermFactory [2] (TF) is a Semantic Web (SW) framework for multilingual professional terminology management. It is mainly aimed to bring Semantic Web services to collaborative professional terminology work. It has been used to manage various multilingual, multi-domain terminology databases converted into RDF, including the Finnish-English WordNet and a six-language version of ICD-10 [3]. The long term goal is to help produce terminology of sufficient explicitness and quality to serve automatic localization and high quality translation [4].

For professional terminologists, Semantic Web resources and tools should be particularly apt for harvesting and sifting of terminological raw data. We call “half open” terminology sources sites which are not covered by an already existing linked data entry point, but are accessible on the web through individual requests. To allow RDF access to non-RDF third party websites on the fly, TF provides a facility for plugging in website specific converters of page content to RDF. This facility was described in [5] and tested on a sample of well-known terminology sites.

Traditional dictionary and term tools generally support term search by designation, either exact match or implicit or explicitly defined fuzzy match according to rules that vary per tool. Beyond

DOI : 10.5121/ijwest.2014.5304

the most established sciences, concepts have no globally agreed identifiers. Some tools truncate or lemmatize term components (words).

Some tools allow disambiguating the query by subject field. However, subject field is not always narrow enough for sense disambiguation. Sense disambiguation may call for finer semantic relations, such as hypernymy, only provided by some lexicographic sources like Wiktionary or WordNet. Merging information from many different types of sites provides valuable data for terminologists in term standardization and harmonization, but the process itself is complicated by the anonymity of the data.

Looking at the samples in [5], it is clear that many term databases (e.g. among those included in EUTermBank) duplicate the same data. Professional terminology demands source indications, which are largely missing in the examined data. In order to improve on this and advance the state of the art in web based terminology work, TermFactory provides tools to take care of provenance and reduce duplication in the data. Some of the duplication is obvious enough to remove by automatic means, leaving subtler cases to human experts. This paper presents a SW workflow for preparing data trawled from the Web [5] to produce better term candidates for examination by professional terminologists.

2. BLANK NODES IN RDF

TF represents results of the terminology trawls in [5] as RDF graphs that abound in anonymous resources, represented as blank nodes. Blank nodes have been a sore issue in RDF from the start. Chen et al. [5] list pros and cons of blank nodes. Blanks

+ code collections (lists)
+ code provenance (reification)
+ code non-binary relations and structures
+ replace URIs when not known, needed or wanted
- cause clutter with SPARQL queries.
- cause clutter and broken links in merging RDF graphs
- complicate linking in Linked Data

[6] suggest three ways to alleviate the problems.

1. Use RDF reasoning to remove redundant blanks (lean graphs)
2. Use OWL reasoning (ids, keys, identities) to uniquely identify blanks
3. Generate (predictable) URIs for blank nodes.

In the TF use case presented here, term data from third party Web sources gets converted to RDF graphs with blank nodes, for one or more of the reasons listed as pluses in [5]'s list. Therefore the TF terminologist must face the minuses. We proceed now to explain how TF helps manage blank nodes so as to overcome the problems. We start with reification and continue with other uses of blank nodes.

3. PROVENANCE IN RDF

The founding philosophy of RDF may have been a realist one [7]: RDF graphs are true partial descriptions of the world out there. At least since [8], a relativist view reminiscent of Hintikka’s model set variety of modal logic [9] has won ground: RDF graphs present partial possible worlds as seen by this or that agent, not all true or consistent. To recover truth and trust, the context or provenance [10] of statements must be made explicit. As [11] note, in the Linked Data cloud, meta knowledge about the data becomes paramount. This is nothing new to terminologists. An indispensable requirement for professional terminology is source indications, which report provenance.

Note a small but significant difference between provenance and source. A provenance indication says 'a tells that p', a source indication says 'p as told by a'. Provenance is noncommittal about truth of p, source indication is veridical: p is a fact from a trusted source. The step from provenance to source makes a modal logic inference by the reflexive axiom T (a tells that p ergo p).

The multitude of proposals for handling provenance in RDF differ by whether they propose some extension to RDF triples or RDF graphs, or more conservatively, propose some use or interpretation of the existing devices. TF contributes a minor but apparently novel variant of the conservative type that has some attractions to us terminologists at least.

RDF statements are triples identified by subject, predicate, and object. Statements are not resources, so they cannot have properties. The original RDF standard way to identify a triple with a resource is statement reification. A RDF statement reification is a blank resource that identifies a triple by its key properties predicate, subject, and object. The reification can then carry provenance information, such as named graph (context).

```
:s :p :o .
```

is reified by:

```
_:b rdf:type rdf:Statement;
   rdf:subject :s ;
   rdf:predicate :p ;
   rdf:object :o ;
   meta:context <http://foo> .
```

Some early RDF adopters [12] did apply RDF reification as such to assign sources to triples. A drawback is that standard reification requires four additional triples for representing one statement per document as a resource. Also, it becomes cumbersome to write query patterns that concern provenance.

A variant of reification common in RDF modeling is property decomposition. Statement `s P o` is decomposed into two relations `s S p. p T o` (`S` inverse functional, `T` functional). A new (blank) resource `p`, uniquely determined by `s`, `P` and `o`, reifies the instance of `P` holding between `s` and `o`. This method is used in TermFactory for associating metadata properties to labels (reified as Designations) and to labeling relations (reified as Terms) [2]. Triple

```
:s rdfs:label "foo"
```

gets reified in TF as:

```
:s term:referentOf [ a term:Term ;
                   meta:source <http://foo> ;
                   term:hasDesignation [ a exp:Designation ;
                                       exp:form 'foo' ] .
```

Another simple (but to our knowledge novel) variant of reification is to replace triple object by a blank node having the original object as `rdf:value` and assign provenance and other meta properties to the blank:

```
:s :p [ rdf:value :o ; meta:context <http://foo> ] .
```

Although syntactically, the context looks like a property of the object, TermFactory semantics associates it to the whole triple. We call this variant of reification *value reification* and a model which represents quads in this way a *contexted model* [2].

A plus of value reification is that it places context information at the leaves of a subject-oriented RDF tree, which looks good in prettyprinted formats like TURTLE. A minus is that contexted models are not compatible as such with OWL. A datatype statement cannot be represented as a value reification in OWL, because a datatype property cannot have a blank node with properties as object. A workaround is to introduce object property counterparts to datatype properties in reified contexts.

A simple way to provide context to a triple is to add one more slot to make it a quadruple. Quads grouped as triples by shared context index in turn gives rise to the notion of named graph. These notions, reminiscent of modal logic [10], have made their way into the SPARQL standard as named graphs (cf. model sets) and datasets (cf. model systems).

Adding the context index takes RDF from classical to intensional logic. The list of use cases for named graphs in theW3C working draft [13] for “RDF spaces” in fact reads like the table of contents of a modal logic reader. The truth definition for triples in RDF spaces is a copy of truth at a possible world (context) in intensional logic. The RDF spaces draft is on the brink of pushing RDF from a subset of classical logic to modal logic. It would not be a surprise if in the long run, RDF and SPARQL both natively represent quads as well as triples.

TF uses contexted models to represent quads in RDF. The TF query engine translates between quads (datasets) and contexted RDF documents (models). Contexted models provide a way to combine contexted data across models without going beyond the current state of W3C standards. To maintain provenance data when querying multiple sources, the TF query engine converts contexted models from cross-model queries to datasets (quads).

With contexted models, TF does what it can to promote quads to first class RDF citizens within current standard RDF. The result of a TF SPARQL query can already be returned as a dataset, for provenance information can be coded in a query result graph using value reification. Here is an example dataset `CONSTRUCT` query which returns a dataset back as it was.

```
CONSTRUCT { ?s ?p [ rdf:value ?o ; meta:context ?g ] }
WHERE { GRAPH ?g { ?s ?p ?o } }
```

The syntax would be clearer if GRAPH were allowed in the `CONSTRUCT` part of the query, but that is not in the standards yet. But the resulting contexted model of the query engine can be converted back to a dataset and printed in quad format in TF.

4. A WORKFLOW FOR MATCHING TERM CANDIDATES

A TF term trawl from the Web results in a list of triples culled from various sources. The provenance of each triple is coded using value reification.

The raw data is concept oriented like the original sources, i.e. forms a listing of concepts and terms associated to them. Figure 1 is a sample from a trawl of en *fishery* in EU TermBank and TermWiki.com described in [4] with provenance indications included.

```
[ meta:source     [ rdf:value    "Wielojęzyczny Tezaurus GEMET; terminy
polskie - Instytut Ochrony Środowiska, Warszawa; terminy angielskie -
Collection: General Multilingual Environmental Thesaurus. The 2001
version of GEMET of the European Environment Agency in Copenhagen" ;
                   meta:context  <http://tfs.cc/eutb/>
                  ] ;
  meta:subjectField  [ rdf:value    "environmental policy" ;
                   meta:context  <http://tfs.cc/eutb/>
                  ] ;
  sign:hasGloss        [  rdf:value       "\"The  industry  of  catching,
processing and selling fish. (Source: CED)\""@en ;
                   meta:context  <http://tfs.cc/eutb/>
                  ] ;
  sign:labeledBy    [ rdf:value    "fishery"@en ;
                   meta:context  <http://tfs.cc/wiktionary/>
                  ] ;
  sign:labeledBy    [ rdf:value    "Fischerei"@de ;
                   meta:context  <http://tfs.cc/eutb/>
                  ]
] .

[ meta:subjectField  [ rdf:value    "Language" ;
                   meta:context  <http://tfs.cc.termwiki/>
                  ] ;
  sign:hasGloss        [  rdf:value       "The  business  or  practice  of
catching fish;                                       fishing."@en ;
                   meta:context  <http://tfs.cc/termwiki/>
                  ] ;
  sign:labeledBy    [ rdf:value    "fishery"@en ;
                   meta:context  <http://tfs.cc/wiktionary/>
                  ] ;
  sign:labeledBy    [ rdf:value    "Fischerei"@de ;
                  meta:context  <http://tfs.cc/termwiki/>
                  ] ;
] .
```

Figure1. Raw result of trawl

To evaluate the possibility of merging concepts coming from different sources, the terminologist can run SPARQL queries on the data to tabulate the shared and different language designations (labels) between the entries. A sample is shown in Table 1 below:

Table 1. Query_Results (6 answer/s limit 1000)

?r1	?g1	?r2	?g2	?v	?type
_:b0	"fishing"@en	_:b1	"seafood company"@en	"fi kalastus; kalatalous / kalastusyritys"	"diff"
_:b0	"fishing"@en	_:b1	"seafood company"@en	"риболов"@bg	"left"
_:b0	"fishing"@en	_:b1	"seafood company"@en	"kalastus; kalatalous"@fi	"left"
_:b0	"fishing"@en	_:b1	"seafood company"@en	"kalastusyritys"@fi	"right"

?r1	**?g1**	**?r2**	**?g2**	**?v**	**?type**
_:b0	"fishing"@en	_:b1	"seafood company"@en	"fishery"@en	"same"
_:b0	"fishing"@en	_:b1	"seafood company"@en	"balıkçılık"@tr	"same"

The tabulation shows that the two Wiktionary senses of *fishery* 'fishing' and 'seafood company' have different designations in Finnish ("diff"), which is evidence against merging the senses into one concept. Based on such evidence, the TF terminologist can compose a related SPARQL query to merge duplicate entries.

The raw entries are concept oriented, as they are in the sources. To merge entries manually it helps to reorient the data lexicographically, as a listing from labels to concepts associated to them.

For this purpose, a TF terminologist can again use SPARQL queries. The following query inverts `meta:labeledBy` to the inverse property `meta:labels`.

```
CONSTRUCT {
        [ rdf:value ?r ; ?i [ rdf:value ?s ; meta:context ?g ] ] .
         ?s ?p [ rdf:value ?o ; meta:context ?g ]
}
WHERE {
     GRAPH ?g {
       ?s ?p ?o ; ?q ?r . FILTER (?p != ?q)
     } .
     ?q owl:inverseOf ?i .
}
```

The next step is to merge entries that share lemma (without losing triple provenance). The following query does the identification by adding `owl:sameAs` triples to the data.

```
CONSTRUCT {
        ?s ?p ?o . ?x owl:sameAs ?y
}
WHERE {
      { ?s ?p ?o } UNION
      {  ?x  sign:labels  ?a  ;  rdf:value  ?v  .  ?y  sign:labels  ?b  ;
rdf:value ?v }
}
```

Running these SPARQL scripts on blank data may generate *non-lean* RDF with duplicate and redundant triples. This is because SPARQL interprets blanks differently from RDF [6, 14]: SPARQL blanks are anonymous terms, while RDF blanks are existentially quantified variables. Making SPARQL semantics agree with RDF semantics would increase query complexity [15]. There are ways to write SPARQL queries to identify duplicate blanks, but such queries tend to be complex and obscure [14].

```
[ rdf:value    "fishery"@en ;
  sign:labels  [ rdf:value     _:b1 ;
               meta:context  <http://tfs.cc/wiktionary/>
             ] ;
  sign:labels  [ rdf:value     _:b2 ;
               meta:context  <http://tfs.cc/wiktionary/>
             ]
] .
[ rdf:value    "Fischerei"@de ;
  sign:labels  [ rdf:value     _:b0 ;
               meta:context  <http://tfs.cc/termwiki/>
             ] ;
  sign:labels  [ rdf:value
            [ meta:source
              [ rdf:value
          "Wielojęzyczny Tezaurus GEMET; terminy polskie - Instytut
Ochrony Środowiska, Warszawa; terminy angielskie - Collection:
General Multilingual Environmental Thesaurus. The 2001 version of
GEMET of the European Environment Agency in Copenhagen" ;
               meta:context  <http://tfs.cc/eutb/>
              ] ;
              meta:subjectField
               [ rdf:value    "environmental policy" ;
                meta:context  <http://tfs.cc/eutb/>
               ] ;
              sign:hasGloss
              [ rdf:value
          "\"The industry of catching, processing and selling fish.
(Source:
             CED)\""@en ;
               meta:context <http://tfs.cc/eutb/] .

_:b0   sign:hasGloss
       [  rdf:value      "The business or practice of catching fish;
fishing."@en ;
         meta:context  <http://tfs.cc/termwiki/>
       ] .
_:b1   sign:hasGloss  [ rdf:value    "fishing"@en ;
                     meta:context  <http://tfs.cc/wiktionary/>
                    ] .
_:b2   sign:hasGloss  [ rdf:value    "seafood company"@en ;
                     meta:context  <http://tfs.cc/wiktionary/>
                    ] .
```

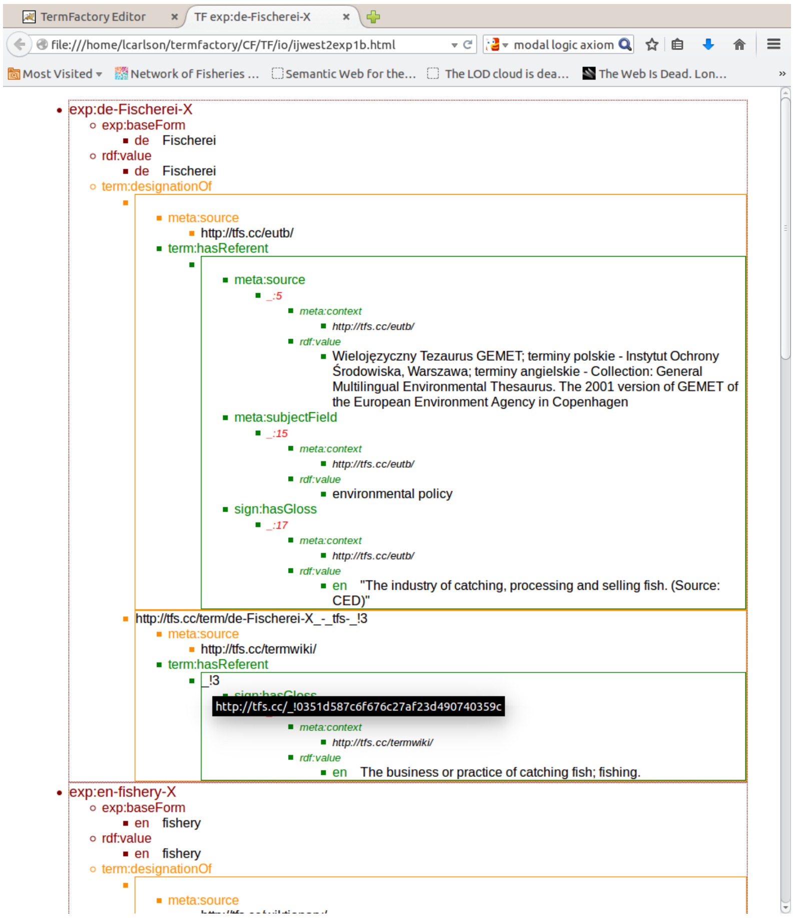
Figure 2. Lemma oriented entries

A TF toolkit utility Factor offers operations that help reduce clutter generated by query engine or reasoner. Operation lean identifies duplicate blanks. TF leanification is an instance of the subgraph isomorphism algorithm described in [16] with some of the optimizations described in [14]. Operation lean adds `owl:sameAs` triples relating blanks that can be merged. Operation merge removes `owl:sameAs` triples by merging identical nodes. An excerpt of the resulting lemma oriented graph is shown in Figure 2.

The lemma oriented arrangement makes it easy to inspect and edit polysemous terms coming from different datasets without losing provenance. TermFactory defines a two-way conversion between RDF and HTML which allows using standard HTML editors to edit TF terms.

The TF built in HTML/RDF editor [17] can be used to edit datasets in the form of contexted models in HTML and commit the results of the edits back into named models in a datastore. The TF editor can thus be used to move triples between named graphs in a datastore.

The editor has access to the TF Factor utilities. To merge entries in the editor, it is enough to add `owl:sameAs` triples and run the Factor merge facility.



```
• exp:de-Fischerei-X
    ○ exp:baseForm
        ▪ de   Fischerei
    ○ rdf:value
        ▪ de   Fischerei
    ○ term:designationOf
        ▪
            ▪ meta:source
                ▪ http://tfs.cc/eutb/
            ▪ term:hasReferent
                ▪
                    ▪ meta:source
                        ▪ _:5
                            ▪ meta:context
                                ▪ http://tfs.cc/eutb/
                            ▪ rdf:value
                                ▪ Wielojęzyczny Tezaurus GEMET; terminy polskie - Instytut Ochrony
                                  Środowiska, Warszawa; terminy angielskie - Collection: General
                                  Multilingual Environmental Thesaurus. The 2001 version of GEMET of
                                  the European Environment Agency in Copenhagen
                    ▪ meta:subjectField
                        ▪ _:15
                            ▪ meta:context
                                ▪ http://tfs.cc/eutb/
                            ▪ rdf:value
                                ▪ environmental policy
                    ▪ sign:hasGloss
                        ▪ _:17
                            ▪ meta:context
                                ▪ http://tfs.cc/eutb/
                            ▪ rdf:value
                                ▪ en   "The industry of catching, processing and selling fish. (Source:
                                  CED)"
        ▪ http://tfs.cc/term/de-Fischerei-X_-_tfs-_!3
            ▪ meta:source
                ▪ http://tfs.cc/termwiki/
            ▪ term:hasReferent
                ▪ _!3
                    ▪ sign:hasGloss
      http://tfs.cc/_!0351d587c6f676c27af23d490740359c
                            ▪ meta:context
                                ▪ http://tfs.cc/termwiki/
                            ▪ rdf:value
                                ▪ en   The business or practice of catching fish; fishing.
• exp:en-fishery-X
    ○ exp:baseForm
        ▪ en   fishery
    ○ rdf:value
        ▪ en   fishery
    ○ term:designationOf
        ▪
            ▪ meta:source
```

Figure 3. TF terms in HTML

5. PUBLISHING TERMS

Once term candidates have been matched and merged, the question arises how to publish them as quality checked terms.

The graph of Figure 2 uses a noncommittal lightweight vocabulary prepared for trawling term candidates from the Web. Quality checked terms are couched in TermFactory terminology schema. The vocabulary conversion can be done with a SPARQL script. The script also makes the epistemic logic inference from provenance to source: from'context C says so and so' to'so and so by source C'.

```
CONSTRUCT {
        ?d exp:baseForm ?b ;
          term:designationOf      [      term:hasReferent      ?r      ;
meta:source ?c ] .
        ?s ?p ?o .
}
WHERE    {
        {
         ?d   rdf:value   ?b   ;   sign:labels   [   meta:context   ?c   ;
rdf:value ?r ] .
         }
         UNION
        {
         ?s ?p ?o .
         FILTER ( ?p != sign:labels ) .
         } .
```

The result of the translation is shown in Figure 3 in TermFactory HTML format. The remaining context blanks can be removed from with a SPARQL script that reduces the contexted model (dataset) to its union graph. The converted source indications may be inherited between concepts, terms and designations using OWL2 property chain axioms.

Traditionally, words and ideas have no proper names. Words are identified by themselves and distinguished by grammatical properties when needed. Concepts are identified by words and disambiguated by more words (definitions, explanations, examples) as necessary. The web has changed all this: now anything at all, including words, can have a globally unique name. What is worse, they can have many names. The problem of homonymy gets replaced by an equally hairy problem of synonymy, or ontology matching, sometimes called ontology hell.

Given that terminological entities have natural key properties, it would seem natural to identify such entities by their properties rather than by name, thereby avoiding ontology hell. But this goes against Linked Data recommendations [18]:

The scope of blank nodes is limited to the document in which they appear, meaning it is not possible to create RDF links to them from external documents, reducing the potential for interlinking between different Linked Data sources. In addition, it becomes much more difficult to merge data from different sources when blank nodes are used, as there is no URI to serve as a common key. Therefore, all resources in a data set should be named using URI references.

Wisely enough, the Linked Data initiative wants to avoid logical complexities inherent in blank node matching at this stage of the push.

TF offers the following compromise. TF terminological entities are identified by key properties, *and* they can be systematically named by IRIs constructed by rule on the key properties. The TF Factor utility supports introduction of such descriptive IRIs for anonymous term data (operation names) and parsing key properties back from descriptive IRIs (operation keys). The mapping between descriptive names and keys can also be expressed in SPARQL 1.1. queries.
The naming convention is this: the local name of a linguistic expression (word or phrase) is formed from the expression's language code, lexical form, and category (usually, part of speech code is sufficient, but it can be another sufficiently distinguishing tag if not).

The three parts of the local name are separated by hyphens. For example, the English language greeting *hello* in TFS (if it belonged to the TFS vocabulary) could get the descriptive name

```
syn:en-hello-X
```

Here `x` is the part of speech tag or homonym distinguisher, and `syn` is a conventional prefix for the TFS syntax namespace '`http://tfs.cc/syn/`'. Usual terminological conventions are to be followed: base (dictionary lemma) form if appropriate, number singular if available, no capitalization, no booleans (and/or/not), no metatext, and no punctuation. The expression part should qualify as a search string for the expression without further parsing. It should not contain any notation belonging to metalanguage, such as parentheses, unless of course parentheses are really part of the designation. On the other hand, we must allow for variation in the name caused by standard encoding of reserved characters in IRIs (or URIs/URLs, as the case may be).

The descriptive label for a sign (term or sense) is formed by concatenating the descriptive label of its designation with some namespace prefix for its referent and the referent's local name, separated by hyphen. Since namespace prefixes are not globally registered, this naming convention is only suggestive of the referent of the term, and main purpose of the prefix is to serve as a suggestive distinguisher. The corresponding namespace is found on the referent of the term. The designation and referent parts of the term label are separated by the string `_-_` . For example, say we have a concept (meaning) for greetings with name `sem:Greeting`. Then the sense of English `hello` as a greeting could be labeled descriptively as

```
sign:en-hello-S_-_sem-Greeting
```

Hyphen and underscore are used as separators as the least reserved punctuation-like characters in the many Semantic Web character set conventions. Separator strings should not occur inside the parts they separate in a way that could cause ambiguity. Here sign is a conventional prefix for TF sign namespace and `sem` is the prefix for TF meaning namespace.

The key properties are supposed to be keys in the sense of OWL2 hasKey construct, so that they uniquely identify the expression or term. Two different items should not end up with the same descriptive name, and optimally one item should get only one such name. The main brunt of identification is borne by the site URL prefix, so if two sites are to share a resource, the site prefix had better be the same. But even if not, and two sites happen to define the same term, the local part of the descriptive name should help identify the key features of the named resources for harmonization.

Given descriptive names, a TF term ontology containing nothing but designations or terms carrying descriptive resource names can be a useful resource as such. It is already a well-formed TF term ontology. For searching and browsing, it may be enough to look for descriptive identifiers. If the descriptive name is properly constructed, the key properties of the resource can be directly read off it.

For publishing nameless resources (such as meanings of terms), the TF Factor utility provides operation skolem and its inverse blanks, for converting TF blank nodes into arbitrarily named constants in a given namespace. The local name of a TF Skolem constant starts with underscore and exclamation mark as distinguished from blank variables that start with underscore and colon.. If the referent of a descriptive name of a term is a Skolem constant, the term name is also one, and gets replaced by a blank in the blanks operation. An example of a TF skolem constant is shown in the black and white tooltip in Fig 3.

6. QUERYING TERMS

One reason for shunning blank nodes in RDF in general and in Linked Data publishing in particular is complexity. Leanification of RDF is intractable in the worst case. (However, empirical investigation [14] indicates that the worst case does not arise often in real life.)

Complexity aside, blank nodes raise the granularity of RDF facts. Given blanks, the size of logically independent RDF fact can grow to a graph of arbitrary size. (In general, blanks are not supposed to be cross-identified between graphs, though the scope of blanks across the same dataset is not fully settled as yet.)

The granularity problem comes up in querying of terms in TF as well. It arises in the connection of SPARQL DESCRIBE queries. Which triples about a term x in a graph properly describe it? Terms typically refer to other terms; dictionaries are known to be multiply connected and circular. Where to draw the limits of a term entry?

The notion of blank node closure [19] is a logical candidate, as it logically constitutes the smallest closed first order sentence which contains all mentions of x. With the stronger OWL semantics, the grain can be cut smaller, as some blanks can be uniquely identified by id (inverse functional) properties and OWL 2 keys [20].

TF uses these insights in its definition of DESCRIBE queries. A TF designation (lexeme) is uniquely keyed by its string, language and category (part of speech). A TF term is keyed by designation and referent (concept). A DESCRIBE query on one term includes the properties of that term plus as much of the neighboring ones that is needed to uniquely identify them, either by IRI or by key properties.

7. EVALUATION

In this section, we evaluate the proposed method of harvesting terms from half open sources by comparing findings from four web sources to terminological information already in linked data (specifically, DBpedia). We organized the evaluation in three steps: choose a set of sample terms by frequency information in COCA and Google books; calculate the number of candidate equivalents and languages for the sample terms in the different platforms and in DBpedia; and evaluate the data quality in the compared sources.

7.1. Sample

The Corpus of Contemporary American English (COCA) [21], a corpus of 450 million words collected from 1990 to 2012, provides frequency data of English. We selected five sample sets of terms of decreasing frequency from a list of 6000 items sampling every tenth word of the 60,000 most frequent words in the corpus [22].

We checked the frequencies of our sample words with Google Ngram Viewer [23] until items within each group had frequencies of the same order of magnitude in both sources. Table 2 shows the result of the sample selection process.

Table 2. Sample word sets

rank 1	problem, place, system, government, point
rank 500	knot, prosecute, fake, capitalism, disagreement

rank 700	officer, goal, reduce, article, career
rank 1000	entice, swelling, radiant, checkpoint, authorization
rank 6000	gynaecological, chondrosarcoma, omani, land-sea, muleta, broad-tail, slide-slip

7.1. Language statistics

We evaluate our method by comparing the number of candidate equivalents found in four online term sites against Wiktionary data in DBPedia.

EuroTermBank (ETB) is a centralized online terminology bank for languages of new EU member countries interlinked to other terminology banks and resources.' [24] There are currently 4 linked databases, 133 local resources, and 4 external resources (TermNet.lv, IATE, OSTEN, MoBIDic) in 33 languages in ETB. [25] In our test, we trawled terms using url http://www.eurotermbank.com/Getentrysmart.aspx?lang_from=en&lang_to=en&where=etb%20extres&term=<key>.

IATE (InterActive Terminology for Europe) provides a multilingual web-based platform for inter-institutional communication among EU countries to facilitate the terminology standardization and data availability for all EU institutions and agencies. IATE is currently available in 24 languages. [26] The trawl URL we used for IATE termbase was 'http://iate.europa.eu/SearchByQuery.do?method=search&valid=Search+&sourceLanguage=s&targetLanguages=s&domain=&typeOfSearch=t&query=<key>'.

TermWiki.com is a commercial collaborative knowledge sharing platform connecting people of similar interests on the basis of 'specific subject-oriented terms and short encyclopedia-like entries'. At the time of writing, Termwiki reported altogether 7163367 terms of 1768 categories in 102 languages [27]. For our test, we trawled terms from TermWiki at URL 'http://en.termwiki.com/Special:Search/all/<key>?namespace=EN'.

Wiktionary is a free content dictionary launched as the sister project of Wikipedia by Wikimedia Foundation. At the time of writing, Wiktionary contained lexical information in 159 languages. Separate Wiktionary sites exist for different languages, such as the English Wiktionary [28] and German Wiktionary [29]. Each of these independent sites may contain duplicate entries for the same words in many languages. The exact structure forthe entries differs between languages of Wiktionary, and slightly also between language entries within each site, so that Danish and Dutch entries in the English Wiktionary mayuse different kinds of wiki templates. [30] In our testing process, we trawled terms from the English Wiktionary using the following XML API 'http://en.wiktionary.org/w/api.php?action=parse&format=xml&page=<key>'.

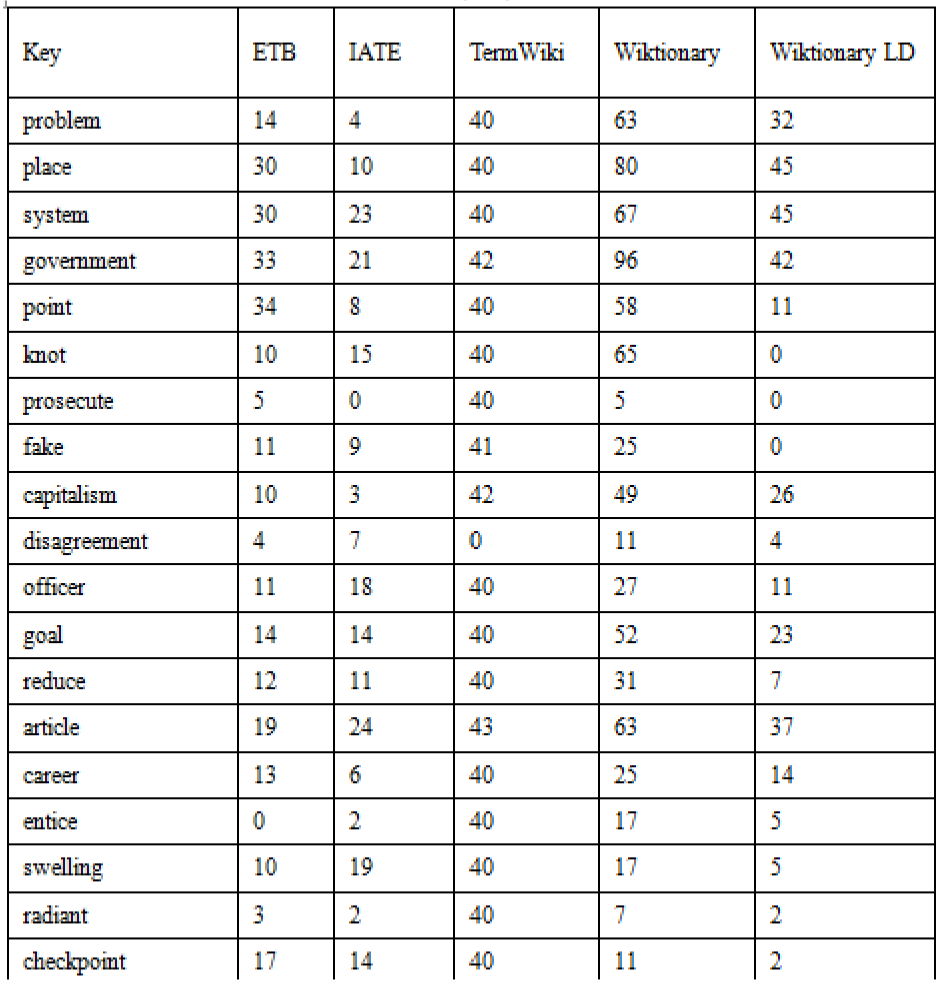
Terminology sites typically use some fuzzy matching condition that retrieves a hit list around the search key. We prune the hit list to contain only exact matches in our further processing of the trawl.

For Linked Data, we use the DBpedia SPARQL endpoint for Wiktionary [31]. Not all Wiktionaries are available from the endpoint. The current RDF dump contains content from English, German, French, and Russian Wiktionaries. The dump currently contains 100 million triples, including the duplication between the different Wiktionary sites. In the test, we run the following query in the endpoint 'http://wiktionary.dbpedia.org/sparql and the query'.

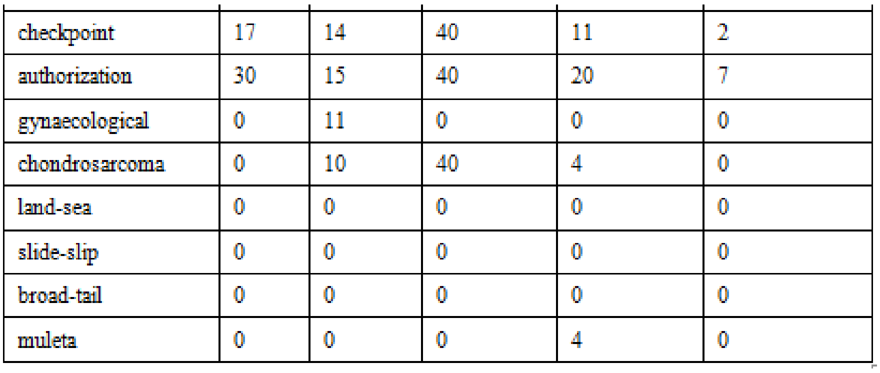
```
PREFIX t: <http://wiktionary.dbpedia.org/terms/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
SELECT ?label (COUNT(distinct ?lang) as ?langs)
WHERE {
      ?r    rdfs:label    ?label    ;    t:hasTranslation    ?t    .    ?t
dc:language ?lang .
      FILTER(?label
IN("problem"@en,"place"@en,"system"@en,"government"@en,"point"@en,"knot
"@en,"prosecute"@en,"fake"@en,"capitalism"@en,"disagreement"@en,"office
r"@en,"goal"@en,"reduce"@en,"article"@en,"career"@en,"entice"@en,"swell
ing"@en,"radiant"@en,"checkpoint"@en,"authorization"@en,"gynaecological
"@en,"chondrosarcoma"@en,"Omani"@en,"land-sea"@en,"muleta"@en,"broad-
tail"@en,"slide-slip"@en)) }}
```

The trawl results and the query report from DBpedia endpoint are displayed together in both a table and a line chart. (See Table.2 and Figure 4 below) As expected, direct web access to Wiktionary serves equivalents in more languages than the DBpedia endpoint at present. Termwiki seems to impose a ceiling at 40 languages, which it reaches in all but a few cases. The European term banks are naturally limited to European languages, but their coverage for special field terms at the lower end of the frequency scale is good.

Table 2. Language statistics



Key	ETB	IATE	TermWiki	Wiktionary	Wiktionary LD
problem	14	4	40	63	32
place	30	10	40	80	45
system	30	23	40	67	45
government	33	21	42	96	42
point	34	8	40	58	11
knot	10	15	40	65	0
prosecute	5	0	40	5	0
fake	11	9	41	25	0
capitalism	10	3	42	49	26
disagreement	4	7	0	11	4
officer	11	18	40	27	11
goal	14	14	40	52	23
reduce	12	11	40	31	7
article	19	24	43	63	37
career	13	6	40	25	14
entice	0	2	40	17	5
swelling	10	19	40	17	5
radiant	3	2	40	7	2
checkpoint	17	14	40	11	2



checkpoint	17	14	40	11	2
authorization	30	15	40	20	7
gynaecological	0	11	0	0	0
chondrosarcoma	0	10	40	4	0
land-sea	0	0	0	0	0
slide-slip	0	0	0	0	0
broad-tail	0	0	0	0	0
muleta	0	0	0	4	0

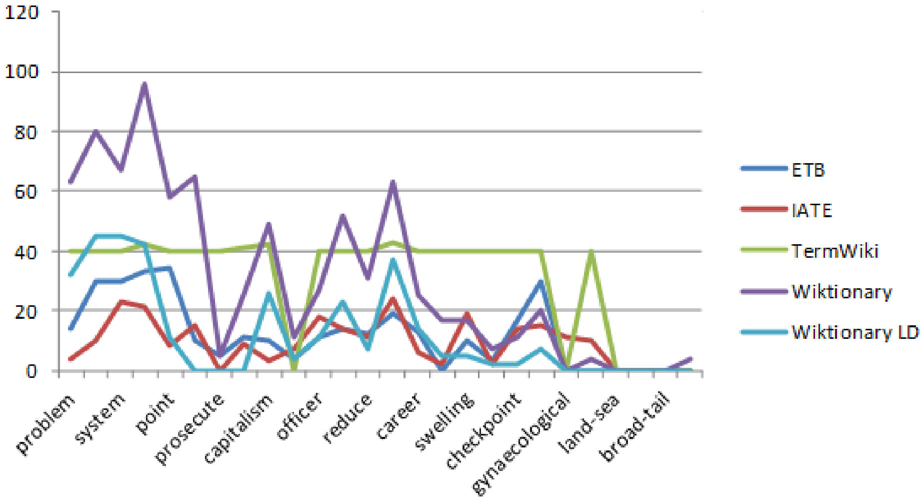


Figure 4. Language statistics

A terminologist would like to quantify the quality and preferentiality of the candidates as well. One advantage of using multiple sources marked by provenance is that it is possible to apply a voting criterion, assuming that more reliable and preferred terms are proposed by more sites. As an example, our sample has two candidates for *government* in Chinese, of which the preferred term gets nine votes and a less common equivalent one.

?label	?count
"政府"@zh	9
"治理"@zh	1

Similar voting criteria can be applied to merge and distinguish senses. Synonymy and translation relations generate a graph that is denser around well defined concepts where synsets of full synonyms or equivalents appear as strongly connected components of the graph [32]. This idea will be explored further in forthcoming work.

8. RELATED WORK

Some RDF datastores extend triples to longer tuples internally [33,34]. The ARQ query engine [35] works with datasets but not on quads directly; it evaluates triple patterns on each graph in turn. This restricts the statement and handling of cross-graph queries. OWL reasoning remains to be extended to quads as well. This might involve adding a context index to description logic tableau construction along the lines of modal logic tableaux.

Some proposals for marking provenance that want to avoid anonymous resources attack the vocabulary instead. Sahoo et al. [36, 37] create new URIs for the subject, property and/or object from a triple's context and associate provenance to the new URIs. Nguyen et al. [38] propose extending property vocabulary with singleton properties. A singleton property like `:p#1` is a subproperty of the original `:p` whose domain and range are singletons, so it identifies its triple. These proposals tend to bloat the vocabulary and run the risk of accidental synonymy.

Many writers that extend triples to longer n-tuples [39, 40, 41, 11, 42] have proposed to structure the context slot further. One motivation is that triple provenance may be divided between many graphs due to reasoning and updates.

An initiative to publish linguistic items as named resources in Linked Open data has been formulated by the Open Linguistics Working Group [43, 44].

9. DISCUSSION

The extension of TermFactory tools to the trawling and processing of term candidates from non RDF sources is quite recent. We cannot present quantitative proof as yet, but the new tools and workflows should expedite the preparatory work of a professional terminologist in looking for and sifting through term candidates in the web of documents and data.

The use of RDF and the associated tools involves a learning curve at the present state of the toolkit, but as experience of best practices accrue, the most productive workflows can be packaged further into prefabricated tool chains and user friendly interfaces.

The SW approach to professional terminology work presents a workable compromise between expressive power and tractability for the needs of terminology work. The graph formalism backed by description logic helps bridge the plethora of vocabularies and presentation formats that characterises the field.

10. CONCLUSION

We have described a SW tool workflow to prepare terminological data trawled from multiple Web terminology sources for matching and merging. We have showed how problems that arise with the use of blank nodes in RDF graphs can be tamed in TermFactory without losing their advantages. For coding provenance, we proposed a minor but apparently novel variant of RDF reification. To handle redundancy, TF implements state of the art methods for reasoning with RDF graphs containing anonymous (blank) nodes. We also discussed ways for RDF format terminology to conform with Linked Data recommendations without losing the advantages of blank nodes.

ACKNOWLEDGEMENTS

TermFactory was developed under grants from TEKES project ContentFactory and EU project MOLTO.

REFERENCES

[1] Anon. http://www.iso.org/iso/iso_technical_committee.html%3Fcommid%3D48104. Last accessed: 2014.07.27.

[2] Lauri Carlson. Introduction to TermFactory. http://www.helsinki.fi/~lcarlson/CF/TF/doc/TFIntro_en.html. Last accessed: 2014.09.07.

[3] World Health Organisation. (1992). ICD-10 Classifications of Mental and Behavioural Disorder: Clinical Descriptions and Diagnostic Guidelines. Geneva. World Health Organisation.

[4] Olga Caprotti, et al. (2012) High quality translation: MOLTO tools and applications. Swedish Language Technology Conference 2012, Lund, October 24-26, 2012.

[5] Shanshan, Wang, Kun, Ji., and Lauri, Carlson. Fishing for terms in half open data with TermFactory.

[6] Chen, Lei, Haifei Zhang, Ying Chen, and Wenping Guo. "Blank Nodes in RDF."Journal of Software 7, no. 9 (2012): 1993-1999.

[7] Barwise , Jon, and John Perry. Situations and Attitudes. Bradford books. MIT Press, 1983.

[8] Carroll, Jeremy J., et al. "Named graphs, provenance and trust." Proceedings of the 14th international conference on World Wide Web. ACM, 2005.

[9] Hintikka, Jaakko. Models for Modalities: Selected Essays. Synthese Library. Kluwer, 1969.

[10] Anon. ProvenanceRDFNamedGraph. http://www.w3.org/2011/prov/wiki/ProvenanceRDFNamedGraph. Last accessed: 2014.09.09.

[11] Schueler, Bernhard, Sergej Sizov, Steffen Staab, and Duc Thanh Tran. Querying for meta knowledge. In Proceedings of the 17th international conference on World Wide Web, pages 625–634. ACM, 2008.

[12] Cui Tao, Jyotishman Pathak, Harold R Solbrig, Wei-Qi Wei, and Christopher G Chute. Lexrdf model: An rdf-based unified model for heterogeneous biomedical ontologies. In CEUR workshop proceedings, volume 521, page 3. NIH Public Access, 2009.

[13] Sandro Hawke. Rdf spaces and datasets. https://dvcs.w3.org/hg/rdf/raw-file/default/rdf-spaces/index.html, 2012. Last accessed: 2014.09.07.

[14] Hogan, Aidan. et al. (2014) Everything you always wanted to know about blank nodes. Web Semantics: Science, Services and Agents on the World Wide Web (2014). http://dx.doi.org/10.1016/j.websem.2014.06.004

[15] Arenas, Marcelo, Mariano Consens, and Alejandro Mallea. "Revisiting blank nodes in rdf to avoid the semantic mismatch with sparql." W3C Workshop: RDF Next Steps, Palo Alto, CA. 2010.

[16] Lee, Jinsoo, Wook-Shin Han, Romans Kasperovics, and Jeong-Hoon Lee. "An in-depth comparison of subgraph isomorphism algorithms in graph databases." In Proceedings of the VLDB Endowment, vol. 6, no. 2, pp. 133-144. VLDB Endowment, 2012.

[17] Kun Ji, Lauri Carlson. RDF/XHTML: Ontology Editing in HTML. KEOD 2012: 365-368

[18] Heath, Tom and Bizer, Christian (2011). Linked Data: Evolving the Web into a Global Data Space. Morgan & Claypool Publishers. http://dx.doi.org/10.2200/S00334ED1V01Y201102WBE001

[19] Stickler, Patrick. "Cbd-concise bounded description." W3C Member Submission 3 (2005).

[20] Ding, Li, Tim Finin, Yun Peng, Paulo Pinheiro Da Silva, and Deborah L. McGuinness. "Tracking rdf graph provenance using rdf molecules." (revision 2)

[21] Davies, Mark. "The Corpus of Contemporary American English as the first reliable monitor corpus of English." Literary and linguistic computing 25.4 (2010): 447-464.

[22] COCA Word frequency data: genres-sample.xls http://www.wordfrequency.info/sample.asp Accessed: 2014.10.27

[23] Goole Books Ngrams Viewer. https://books.google.com/ngrams . Last accessed: 2014.10.29.

[24] Anon. About. http://www.eurotermbank.com/about.aspx . Last accessed: 2014.10.29.

[25] Anon. http://www.eurotermbank.com/default.aspx . Last accessed: 2014.10.29.

[26] Anon. Interactive Terminology for Europe. https://books.google.com/ngrams. Last accessed: 2014.10.29..

[27] Anon. Discover, Learn and Meet New Friends. http://en.termwiki.com/TermWiki:About#. Last accessed: 2014.10.29.

[28] Anon. Wiktionary, the free dictionary. http://en.wiktionary.org/wiki/Wiktionary:Main_Page. Last accessed: 2014.10.29.

[29] Anon. Wiktionary, das freie Wörterbuch. http://de.wiktionary.org/wiki/Wiktionary:Hauptseite. Last accessed: 2014.10.29.

[30] Anon. Wiktionary RDF extraction. http://dbpedia.org/Wiktionary. Last accessed: 2014.10.29.

[31] Anon. Virtuoso SPARQL Query Editor. http://wiktionary.dbpedia.org/sparql. Last accessed: 2014.10.29

[32] Levary, David, et al. "Loops and self-reference in the construction of dictionaries." Physical Review X 2.3 (2012): 031018.

[33] Kiryakov, Atanas, and Vassil Momtchev. "Triplesets: tagging and grouping in RDF datasets." W3C Workshop ‘‘RDF Next Steps’’, Stanford (June 2010). 2010.

[34] Krieger, Hans-Ulrich. "A Temporal Extension of the Hayes and ter Horst Entailment Rules for RDFS and OWL." AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning. 2011.

[35] Anon. Apache ARQ. https://jena.apache.org/documentation/query/arq-query-eval.html, 2014. Accessed: 2014.07.07.

[36] Satya S Sahoo, Olivier Bodenreider, Pascal Hitzler, Amit Sheth, and Krishnaprasad Thirunarayan. Provenance context entity (pace): Scalable provenance tracking for scientific rdf data. In Scientific and Statistical Database Management, pages 461–470. Springer, 2010.

[37] Satya S Sahoo, Vinh Nguyen, Olivier Bodenreider, Priti Parikh, Todd Minning, and Amit P Sheth. A unified framework for managing provenance information in translational research. BMC bioinformatics, 12(1):461, 2011.

[38] Vinh Nguyen, Olivier Bodenreider, and Amit Sheth. Don’t like RDF reification?: Making statements about statements using singleton property. In Proceedings of the 23rd International Conference on World Wide Web,WWW’14, pages 759–770, Republic and Canton of Geneva, Switzerland, 2014. InternationalWorld WideWeb Conferences Steering Committee.

[39] E. Dumbill. Tracking provenance of rdf data. Technical report, ISO/IEC, 2003.

[40] Giorgos Flouris, Irini Fundulaki, Panagiotis Pediaditis, Yannis Theoharis, and Vassilis Christophides. Coloring rdf triples to capture provenance. In Proceedings of the 8th International Semantic Web Conference, ISWC ’09, pages 196–212, Berlin, Heidelberg, 2009. Springer-Verlag.

[41] Panagiotis Pediaditis, Giorgos Flouris, Irini Fundulaki, and Vassilis Christophides. On explicit provenance management in rdf/s graphs. In Workshop on the Theory and Practice of Provenance, 2009.

[42] Umberto Straccia, Nuno Lopes, Gergely Lukacsy, and Axel Polleres. A general framework for representing and reasoning with annotated semantic web data. In AAAI, 2010.

[43] Chiarcos, Christian, John McCrae, Philipp Cimiano, and Christiane Fellbaum
Towards Open Data for Linguistics: LinguisticLinked Data.

[44] Chiarcos, Christian, et al. "The Open Linguistics Working Group." LREC. 2012.

Authors

Kun Ji, PhD student in translation studies at the University of Helsinki. Her main topic is collaborative multilingual terminology management and has been working on the area for four years.

Shanshan Wang, PhD student in translation studies at the University of Helsinki. She has been working on ontology-based multilingual terminology for four years.

Lauri Carlson, Professor of Linguistics and translation at the University of Helsinki. Lately, he has been working on applying semantic web technologies to multilingual language technology and terminology.