Journal of Universal Computer Science, vol. 19, no. 13 (2013), 1871-1891
submitted: 5/11/12, accepted: 28/6/13, appeared: 1/7/13 © J.UCS

Web Resource Sense Disambiguation in Web of Data

Farzam Matinfar
(Computer Engineering Department, University of Isfahan, Iran
f.matinfar@eng.ui.ac.ir)

Mohammadali Nematbakhsh
(Computer Engineering Department, University of Isfahan, Iran
nematbakhsh@eng.ui.ac.ir)

Georg Lausen
(Albert-Ludwigs-Universität, Freiburg, Germany
lausen@informatik.uni-freiburg.de)

Abstract: This paper introduces the use of WordNet as a resource for RDF web resources sense disambiguation in Web of Data and shows the role of designed system in interlinking datasets in Web of Data and word sense disambiguation scope. We specify the core labelling properties in semantic web to identify the name of entities which are described in web resources and use them to identify the candidate senses for a web resource. Moreover, we define the web resource's context to identify the most appropriate sense for each of the input web resources. Evaluation of the system shows the high coverage of the core labelling properties and the high performance of the sense disambiguation algorithm.

Keywords: Semantic web, Linked Data, Interlinking, Word Sense Disambiguation
Categories: H.3.1, H.3.3, M.7

1 Introduction

In recent years, the web has evolved to one where both documents and data are linked [Bizer, 2009]. The goal of the LOD[1] community project is publishing and interlinking datasets following the rules [Lee, 2006] for linking data. Technically, Lined data refers to the published data on the web that is machine-readable, has explicit meaning, is linked to other external datasets, and can be linked to from external datasets [Bizer, 2009]. The RDF links generate triples where the subject resource is a URI reference in the namespace of one dataset and the object is a URI reference in the target dataset [Bizer, 2007]. Applying the Lined Data principles, each web resource describes and represents an entity.

This paper connects two research areas Word Sense Disambiguation (WSD) and interlinking in Web of Data. This paper specifies the core labelling properties used in Web of Data by analysing web resources' descriptions and used ontologies. These labels specify the titles of the web resources. Since two web resources with the same title do not necessarily represent the same entity, it is needed to identify the sense and meaning of each title. To the best of our knowledge, our system, which we refer to as

[1] Linking Open Data

RESDS[1], is the first RDF web resource sense disambiguation system. Given an input web resource, RESDS system has the ability to specify the name of the entity which is described in a web resource and identify the corresponding sense in WordNet database. The designed system aims to interlink datasets by comparing and matching RDF web resources' senses. Given two web resources, the relation between the corresponding senses shows the relation between the web resources too.

There are many applications that could use and benefit from this system. In WSD scope, identifying the most appropriate sense for a word in a given context is desired. WSD is considered as AI-problem that refers to the task of identifying the meaning of a word in context in a computational manner [Navigli, 2009]. There are many supervised algorithms to solve the WSD problem. However, these approaches require large training data where words are annotated by human experts. It is estimated that it takes one person one minute to annotate a word in a corpus [Edmonds, 2000]. Therefore, making training data including large number of words with various senses each supported by hundred examples of a sense require several years. Another category of existing algorithms for WSD are knowledge-based approaches that exploit the knowledge of large, wide-coverage databases. In these approaches, usually the word in a corpus is mapped to a sense in a sense inventory. One of the most used sense inventories in WSD systems is WordNet database [Fellbaum, 1998] which is used in SENSEVAL[2] competitions too. Therefore, mapping RDF web resources to WordNet enable such systems to map the text words to web resources too. Considering the Web of Data as a huge knowledge base, annotating texts with web resources can highly increase the texts' semantic and understanding. Moreover, more information can be obtained about the annotated terms in a text through exploring the corresponding web resources.

Interlinking tools can benefit from RESD system too. The 4th principle of the Linked Data emphasizes linking web resources to each other to discover more things and information. Therefore, there is a need to design methods to interlink datasets. As Lined data Datasets usually consist of a large number of web resources (things), there is a need to design automatic or semi-automatic tools to generate RDF links and interlink datasets. In some domains, there are accepted naming schemas which are common in entity description. For example, ISBN and ISSN in publishing domain, ISIN in financial domain, EAN and EPC in identifying products, and some accepted identifiers in gene and Molecules domain. However, in most of the datasets, there are not such accepted naming schemas. Therefore, it works for a limited number of domains. One of the common approaches to do the interlinking process is comparing the web resources' descriptions. The more similarity between web resources' description, the more probable they are to describe the same entities. One of the main obstacles with this approach is schema heterogeneity. Dataset owners usually use various and different ontologies and properties to describe their dataset. Therefore, web resources investigation and comparison become more difficult than before. Interlinking tools can benefit from RESD system to discover the same entities in target datasets. Web resources that are linked to each other with *owl:sameAs* predicate should have the same sense. Therefore, discovering the senses of web resources enables us to identify the web resources that describe the same entity. Moreover,

[1] RDF Entity Sense Disambiguation System
[2] http://www.senseval.org/

knowing the sense of an RDF web resource makes it possible to extract related knowledge from WordNet to find other related web resources.

This paper is organized as follows: in section 2, a summarized related work is presented. Section 3, identifies labelling properties used in Web of Data and explains the proposed method for web resource sense disambiguation. Experimental results and the conclusion are presented in section 4 and 5 respectively.

2 Related work

In recent years, various approaches and tools [Ngomo, 2011] have been designed to do the interlinking process. Some of them use string matching and similarity measures [Scharffe, 2009; Volz, 2009]. In [Nikolov, 2008] string similarity and adaptive learning are used to make the fusion ontology. Some of the approaches are designed for linking specific domains such as linking movie datasets [Hassanzadeh, 2009), linking music datasets [Raimond, 2008), and interlinking multimedia [Hausenblas, 2009). Some approaches do not support multi ontology environments [Nikolov, 2008] or require the alignment of ontologies as input to do the interlinking process [Volz, 2009). Ontology matching algorithms [(Cruz, 2001), (Jain, 2010), and (Gil, 2012)] can be applied before interlinking too. User contribution is another way to interlink datasets [Hausenblas, 2008] which is not effective for large datasets.

While interlinking methods try to find the desired web resources, in WSD scope the concentration is on recognizing the best sense of a word. There have been various research studies to solve WSD problem. Supervised algorithms [(Hearst, 1998), [(Lee, 2004), and (Ng, 1997)] have shown high performances. But the drawback of the supervised algorithms is that they require sense tagged data and their performances depend on the amount of the training data. In contrast, knowledge-based approaches [(Lesk, 1986), (Rada, 2005), (Navigli, 2005), and (Walker, 1986)] usually use dictionaries to identify the target sense and WordNet is one of the widely used disciotnaries in this scope. Moreover, this dictionary has been used for measuring the relatedness of concepts [(Pedersen, 2004), (Budanitsky, 2006)]. Hence, some researches merge and enrich WordNet concepts to make course-grained word senses to increase the performance of related applications [Gharib, 2012].

3 Proposed Algorithm Overview

The proposed algorithm and the designed system aim to link RDF web resources to the corresponding senses in WordNet database and interlink them based on the matching result. The architecture of the designed system is shown in Fig. 1 and consists of the following steps:

- Identifying core labelling properties
- Specifying candidate senses
- Making disambiguation contexts of candidate senses
- Making disambiguation context of a web resource
- Mapping contexts
- Enriching the contexts of WordNet senses

- Enriching the context of a web resource

In this section, the proposed architecture and its phases are discussed in detail.

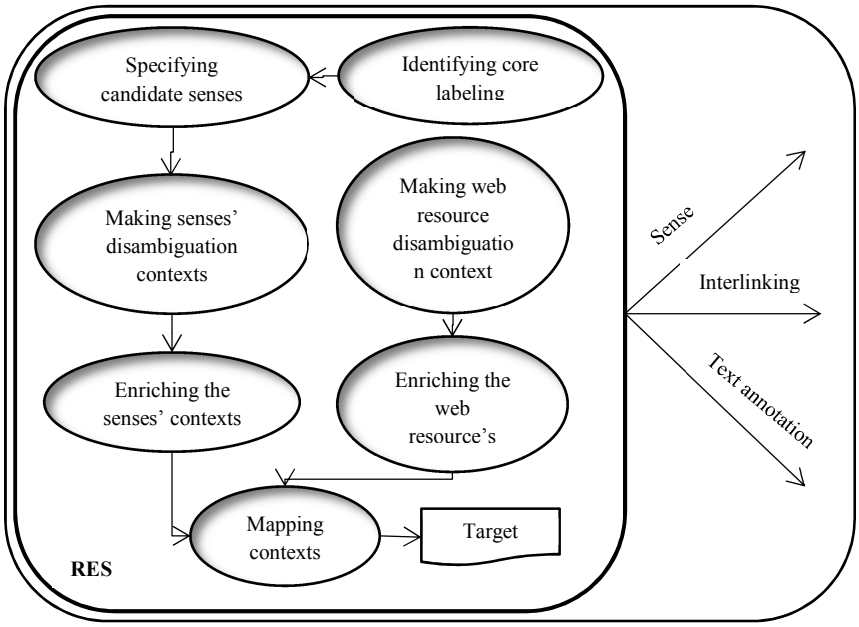

Figure 1: workflow of RESD

3.1 Identifying Core Labelling Properties

In Web of Data each web resource represents and describes an entity. Each entity usually has a title property which can be selected from any ontology. Different applications such as browsers, search engines, and interlinking tools benefit from these properties to do their tasks. We use these properties to identify the candidate senses in WordNet for an entity.

Different ontologies define different terms for labelling purposes such as rdfs:label, dc:title, and foaf:name. The Web of Data is an open space and hence individuals and organizations are free to publish their data considering their favourites and preferences. Therefore, organizations use arbitrary ontologies to describe their web resources. This makes the web space heterogeneous.

In (Ell, Vrandecic, Simperl 2011), the most used properties for labelling purposes in BTC-2010 have been discovered by finding string values in page description and a number of label-related metrics have been introduced. Additionally, we have conducted an empirical study to identify which ontologies and the corresponding labelling terms have been widely used in Web of Data. We have used the BTC-2011 dataset for our empirical study. We have randomly selected 90 million triples in the BTC-2011 and then extracted their predicates. We have categorized the predicates based on the domain of the URI addresses. These domain addresses represent the URI

addresses of ontologies that have been used. Finally, we have selected and ranked ontologies based on three criteria in our study:

1) Ontology-density

The first ranking is based on the extent of each ontology usage. Predicates with the same domain URI address are gathered in a group and the size of each group divided by the total size represents each ontology density. We ranked 196 ontology-density values by descending order. Our results show that some ontologies play a more important role than others and as it is shown in Fig. 2, 20% of highly used ontologies cover more than 99% of ontology usages. Hence, we selected 38 top ontologies in this stage. The top 3 ontologies are shown in table 1.

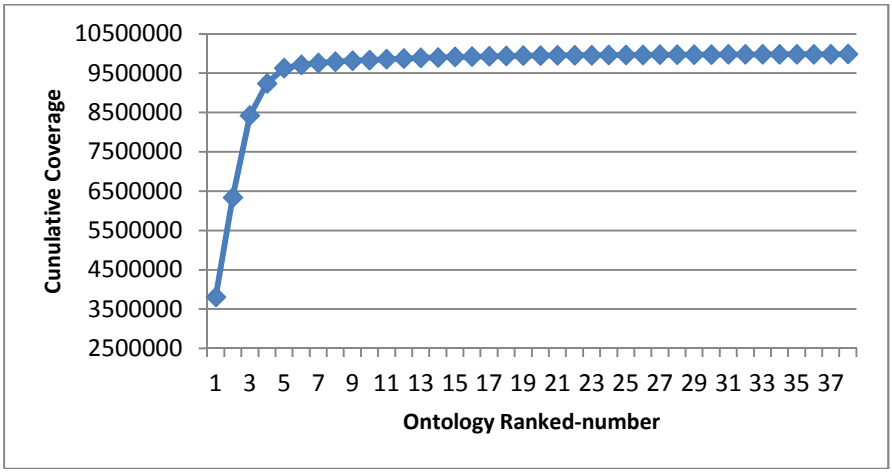

Figure 2: 38 ontologies cover more than 9.98 million predicates in each 10 million triples

2) Averaged-ontology-predicate-density

Dataset owners can use different ontologies to publish their datasets. The preferences of datasets owners, popularity of ontologies, size of ontologies, domain of ontologies, etc. are some of the criteria which influence ontology selection.

Average number in 10 million triples	**Ontology URI**
3808474	http://www.w3.org/2000/01
2529060	http://purl.org/goodrelations
2083347	http://www.w3.org/1999/02

Table 1: 3 ontologies with the highest ontology-density

Ontology-predicate density: Dividing the size of each ontology usage by the number of used predicates of that ontology represents the Averaged-ontology-predicate density measure. This is a normalized measure regarding the number of predicates. For example, while two ontologies *Goodrelations* and *Rdfs* have a great portion in ontology usages in our experiment, the number of used predicates is 25 and 10

respectively. Therefore, each predicate of the *rdfs* ontology has a higher usage in average compared with the former ontology. 21 Ontologies with an Average-ontology-predicate density higher than 220 with at least 5 used predicates in our experiment are shown in table 2.

Ontology-predicate density	**Ontology URI**
380847.4	http://www.w3.org/2000/01
101162.4	http://purl.org/goodrelations
740.08	http://www.w3.org/1999/02
4346.40	http://data-gov.tw.rpi.edu/vocab/p/90
5732.22	http://xmlns.com/foaf/0.1
1395.96	http://purl.org/dc
587.73	http://purl.uniprot.org/core
3469.5	http://purl.org/net
1378.62	http://purl.org/rss/1.0
2976.71	http://www.aktors.org/ontology
922.78	http://www.w3.org/2002/07
536.5	http://purl.org/ontology
239.79	http://www.rdfabout.com/rdf/schema
387.07	http://swrc.ontoware.org
1224	http://rdf.opiumfield.com/lastfm
541.14	http://www.geonames.org
1405	http://www.semanlink.net/2001/00
527	http://www.daml.org/2002/02/telephone/1
508	http://transport.data.gov.uk/0/ontology
296.54	http://semanticscience.org/ontology
244.1	http://semantic-mediawiki.org/swivt

Table 2: 21 ontologies with the highest Average-ontology-predicate density (each include at least 5 used predicates)

3) Ontology popularity and topic coverage

In [Nikolov, 2010], wide coverage and popular ontologies are introduced by analyzing existing instance-level links. We also take some of these ontologies into consideration.

Finally, we unified the mentioned three ontology sets which were created with different criteria. We probed the resulting set manually to extract the terms they use for labeling purposes. Our result contains most of the labeling properties demonstrated in [Ell, 2011]. It additionally includes 13 more labeling terms. Hence, we considered 49 ontology terms for labeling purpose in our study which are presented in table 3.

Ontology Terms
http://www.w3.org/2000/01/rdf-schema#label http://xmlns.com/foaf/0.1/nick http://purl.org/dc/elements/1.1/title http://purl.org/rss/1.0/title http://xmlns.com/foaf/0.1/name http://purl.org/dc/terms/title http://www.geonames.org/ontology#name http://xmlns.com/foaf/0.1/nickname http://swrc.ontoware.org/ontology#name http://sw.cyc.com/CycAnnotations_v1#label
http://rdf.opiumfield.com/lastfm/spec#title http://www.proteinontology.info/po.owl#ResidueName http://www.proteinontology.info/po.owl#Atom http://www.proteinontology.info/po.owl#Element http://www.proteinontology.info/po.owl#AtomName http://www.proteinontology.info/po.owl#ChainName http://purl.uniprot.org/core/fullName http://purl.uniprot.org/core/title http://www.aktors.org/ontology/portal#has-title http://www.w3.org/2004/02/skos/core#prefLabel
http://www.aktors.org/ontology/portal#name http://xmlns.com/foaf/0.1/givenName http://www.w3.org/2000/10/swap/pim/contact#fullName http://xmlns.com/foaf/0.1/surName http://swrc.ontoware.org/ontology#title http://swrc.ontoware.org/ontology#booktitle http://www.aktors.org/ontology/portal#has-pretty-name http://purl.uniprot.org/core/orfName http://purl.uniprot.org/core/name http://www.daml.org/2003/02/fips55/fips-55-ont#name
http://www.geonames.org/ontology#alternateName http://purl.uniprot.org/core/locusName http://www.w3.org/2004/02/skos/core#altLabel http://creativecommons.org/ns#attributionName http://www.aktors.org/ontology/portal#family-name http://www.aktors.org/ontology/portal#full-name http://xmlns.com/foaf/0.1/primaryTopic http://xmlns.com/foaf/0.1/topic http://www.w3.org/2004/02/skos/core#prefLabel http://rdf.data-vocabulary.org/#name
http://dbpedia.org/ontology/name http://rdf.freebase.com/ns/type.object.name http://purl.org/goodrelations/v1#legalName http://purl.org/dc/terms/subject http://www.w3.org/2004/12/q/contentlabel#hasLabel http://www.w3.org/2006/vcard/ns#title http://dbpedia.org/property/name http://ogp.me/ns#title http://rdfs.org/sioc/ns#topic

Table 3: all Labeling Terms acquired from various resources and criteria

Given an RDF web resource *w*, we select triples in *w* which their subject resource are equal to URI address of *w* and their predicates are equal to one of the properties shown in table 3. The object (value) parts of these triples show the name of the described entity. For example, by querying the web address *http://dbpedia.org/resource/Plant* several values are retrieved which are shown in value column of table 4.

URI Address	Property	Value
http://dbpedia.org/ resource/Plant	http://dbpedia.org/property/name	Plant
	http://www.w3.org/2000/01/rdf-schema#label	Plantae
		Plants
	http://xmlns.com/foaf/0.1/name	Plants
	http://xmlns.com/foaf/0.1/primary Topic	http://en.wikipedia.org/ wiki/Plant

Table 4: identifying the name of the described entities in http://dbpedia.org/resource/Plant

3.2 Specifying the Candidate Senses

Given an RDF web resource *w*, we define *candidate_senses(w)* as a collection of candidate senses for *w*:

$$candidate_seanses(w) = \{s_1, s_2, \dots, s_n\}$$

We query WordNet using values discovered in the previous section to identify the candidate senses for *w*. In our example, the term *plant* is one of the retrieved values. Therefore, WordNet is queried by *plant* and four senses are provided in output as follows:

1) plant, works, industrial plant -- (buildings for carrying on industrial labor; "they built a large plant to manufacture automobiles")

2) plant, flora, plant life -- (a living organism lacking the power of locomotion)

3) plant -- (something planted secretly for discovery by another; "the police used a plant to trick the thieves"; "he claimed that the evidence against him was a plant")

4) plant -- (an actor situated in the audience whose acting is rehearsed but seems spontaneous to the audience)

3.3 Disambiguation Context of a WordNet Sense

Given a WordNet sense *s* and its corresponding synset *ss*, we use the following relationships and information to define the disambiguation context *C(s)*:

- **Synonymy**: all the synonyms of *s* in synset *ss*. For example, in the first sense of *plant*, the sysnset is "plant, works, industrial plant" and terms *plant*, *works*, and *industrial plant* are added to the context of the first sense.
- **Hypernym**: this relation represents the "s *is kind of u*" relationship which generalizes a sense. All synonyms of *u* are added to the context. For example, in the first sense of the term *plant* we have the following relationship:

plant, works, industrial plant ----> is kind of -----> building complex, complex

Therefore, terms *building complex* and *complex* are added to the context.

- **Hyponym**: this relation shows the "*u is a kind of* s" relationship which specializes a sense to a narrower sense. All synonyms of *u* are added to the context. For example, for the first sense of the plant, two of the relationships are as follows (there are other synsets too):

distillery, still ----> is kind of -----> plant, works, industrial plant
factory, mill, manufacturing plant, manufactory ---> is kind of ----> plant, works, industrial plant
Therefore, terms *distillery, still, factory, mill, manufacturing plant,*and *manufactory* are added to the context.

- **Part_Meronym**: this relation shows the "s *is part of u*" relationship.
- **Part_Holonym**: the inverse of Part_Meronym.
- **Member_Meronym**: this relation shows the "s *is member of* something" relationship.
- **Member_Holonym**: the inverse of Member_Meronym.

All terms provided by the mentioned relationships are added to *C(s)*.

Depth parameter (D): given a WordNet sysnset *p*, and a collection of relationships *r*, depth parameter shows the maximum allowed number of edges connecting *p* to any sysnset *q* using *r*.

To make the context of a synset, it is possible to continue exploring WordNet repeatedly using the mentioned relationships. The depth parameter determines the maximum allowed distance and therefore bounds the exploration. For a given sense *s*, the senses that are reachable through the following paths are discovered to make the context *C(s)*:

$$path(s) = \{r_1, r_2, \dots, r_n\} \qquad n \leq D \ , r_i \in r$$

For example, the contexts of the *plant*'s candidate senses are shown in table 5.

Sense	Context
plant, works, industrial plant	plant, works, industrial plant, building complex, complex, distillery, still, factory, mill, manufacturing plant, manufactory, packinghouse, packing, plant, recycling plant, refinery, saltworks, ...
plant, flora, plant life	Flora, plant life, life form, organism, being, living thing, Plantae, kingdom Plantae, plant kingdom, plant part, phytoplankton, ornamental, acrogen, ...
plant	Plant, contrivance, stratagem, dodge, ...
plant	Plant, actor, histrion, player, thespian, role player, ...

Table 5: The contexts of the candidate senses of the term plant. D=1

3.4 Disambiguation Context of a Web Resource

Given a web resource *w*, we are interested in to the literals that are found in *w* and connected RDF pages to make the context *C(w)*.

Algorithm:

The triples in the description of *w* are extracted. In each triple, if the object is a literal, then it is added to *C(w)* and if it is an RDF URI reference, then it is added to the queue *q*. The URIs in the *q* are analysed too and the retrieved literals are added to *C(w)*. This process continues till the queue becomes empty or end criterion happens. In our algorithm, each literal has its own weight which is determined based on several parameters. The pseudo code of the algorithm is presented in Algorithm 1.

Local Distance: Given an input web resource *x*, consider a triple as p -- predicate -- q in *x*'s description; we define the local distance between *x* and *q* as follows:

$$\boldsymbol{dis}(\boldsymbol{x},\boldsymbol{q}) = \begin{cases} \mathbf{1}, & \boldsymbol{x} = \boldsymbol{p} \\ \mathbf{2}, & \boldsymbol{x} \neq \boldsymbol{p} \end{cases}$$

Where $q \neq x$

Distance: given a web resource *w*, several RDF resources are probed to make the *C(w)* . The distance of each web resource *y* is acquired by summing the distances of web resources located virtually between *x*, *y*.

$$\boldsymbol{dis}(\boldsymbol{w},\boldsymbol{y}) = \boldsymbol{dis}(\boldsymbol{w},\boldsymbol{\alpha_1}) + \boldsymbol{dis}(\boldsymbol{\alpha_1},\boldsymbol{\alpha_2}) + \boldsymbol{dis}(\boldsymbol{\alpha_2},\boldsymbol{\alpha_3}) + \cdots + \boldsymbol{dis}(\boldsymbol{\alpha_n},\boldsymbol{y})$$
$$\boldsymbol{\alpha_i} \in \boldsymbol{intermediated\ web\ resources}$$

The distance of each web resources has a reverse relation with the weight of that resource.

Weight: given the initial web resource *w* (that we are interested in making disambiguation context) the weight of a literal *l* that is extracted from RDF page *p,* is defined as follows:

$$weight(l) = \begin{cases} \dfrac{\alpha}{\mathrm{dis(w,p)} + 1} & if\ subject\ resource\ of\ l = p \\ \dfrac{\beta}{\mathrm{dis(w,p)} + 1} & if\ subject\ resources\ of\ l \neq p \end{cases}$$

Max_Size (θ) and **MAX_URIs (λ)**: If the number of words in *C(w)* and the number of the probed URI addresses reach θ and λ respectively, the process is terminated.

URI address	Context
http://dbpedia.org/resource/Plant	archaeplastida, eukaryote, plants, plantae, planta, rostliny, pflanzen, ...
http://lod.geospecies.org/kingdoms/Ab.rdf	GeoSpecies Knowledge Base: Kingdom RDF DescriptionPlanta, 13140804, Plantae, Plants, Peter J. DeVries, About: Kingdom Plantae, GeoSpecies Database: Kingdom RDF DescriptionPlantae, ...

Table 6: part of the contexts of two different URIs

Algorithm 1:
w,q, obj, uri, triple_collection, context_collection, weight, text, α and β are variables
Input: w, α β, θ and λ
Output: context_collection
Output: text*

```
1:  q=an empty queue;
2:  Text=empty*
3:  number_of_words=0;
4:  number_of_uris=0;
5:  add (w, 0) to q;
6:  while (q is not empty) && !(end criteria)
7:      obj=get the first object in q and delete it from q
8:      uri=obj.firstfield
9:      distance=obj.secondfield
10:     number_of_uris++
11:     triple_collection=empty
12:     extract triples from uri and put them in triple_collection
13:     for every triple in triple_collection do
14:           if the object is a literal
15:              add literal to the string Text*
16:              if (subject==uri)
17:                   weight = α/(distance + 1)
18:              else
19:                   weight = β/(distance + 1)
20:              end if
21:           add (object, weight) to context_collection
22:           number_of_words++;
23:           end if
24:           if the object is an rdf uri reference
25:              if (subject=w)
26:                   add(object, depth+1) to q
27:              else
28:                   add(object,depth+2) to q
29:              end if
30:           end if
31:     end do
32: end while
33: return context_collection
```

end criteria= (number_of_words < θ) && (number_of_uris < λ)

---

* These codes are just used for algorithm 2 later

For example, the context of the two URIs are shown in table 6.

3.5 Context Mapping Algorithm

The goal of context mapping is to link a RDF web resource to a corresponding WordNet sense:

$$map(w) = \begin{cases} s & if\ a\ correspandence\ exists \\ \emptyset & no\ correspandence\ exists \end{cases}$$

Given an RDF web resource *w* and a set of candidate senses $\{s_1, s_2, \ldots s_n\}$ in WordNet, we use conditional probability *p(s|w)* to find the sense of *w*. The sense that maximizes this probability is selected as the most appropriate sense and is calculated as follows:

$$\boldsymbol{map(w) = argmax_{s \in candidate\ sense(w)} p(s|w)}$$
$$\boldsymbol{= argmax_{s \in candidate\ sense(w)} \frac{p(s,w)}{p(w)}}$$

P(w) is constant for all cases. Therefore:

$$\boldsymbol{map(w) = argmax_{s \in candidate\ sense(w)} p(s,w)}$$

Therefore, the joint probability of sense s and RDF web resource *w* is determinant. *P(s,w)* is estimated as follows:

$$\boldsymbol{p(s,w) = \frac{u(s,w)}{\sum_{s_i \in candidate_senses(w)} u(s_i,w)}}$$

Where $u(s,w)=|C(s) \cap C(w)|$

This formula computes the size of intersection of contexts of a sense and an RDF web resource. Therefore, the sense *s* whose disambiguation context has a higher overlap with the disambiguation context of a web resource *w* is selected as the target sense in WordNet. In the example, *http://dbpedia.org/resource/Plant* is mapped to the second sense i.e. *plant, flora, plant life.*

3.6 Enriching the Disambiguation Context of a Web Resource

The drawback of our method is the situations where the literal is a text value which may contain several sentences. Some of the properties such as *rdfs:comment*, *dbpedia-owl:abstract* naturally present text values. These text values in most of the cases cannot influence the result of context mapping. That is because of the very low occurrence probability of the same text in the target contexts. For example, the address URI "http://dbpedia.org/resource/Plant" contains the following triple:

Subject: http://dbpedia.org/resource/Plant
Predicate: rdfs:comment
Object: Plants are living organisms belonging to the kingdom Plantae. Precise definitions of the kingdom vary, but as the term is used here, plants include familiar organisms such as flowering plants, conifers, ferns, mosses, and green algae, but do not include seaweeds like kelp, nor fungi and bacteria. The group is also called green plants or Viridiplantae in Latin.

The text in object part of the triple does not exist in the contexts of candidate senses and hence this text becomes useless. However, there may be important terms in the text that are beneficial and can improve the result of disambiguation process. To overcome this drawback, we use the frequency metric to find the key terms in a text value. To do that, we collect all the strings in a single bag and finally, the terms that their frequencies are more than μ * AVG are added to disambiguation context of an entity. Parameter μ is a constant higher than 1 and it shows the intensity of desire to add terms from the texts exist in an RDF page. Values closer to 1 result in adding a

greater number of terms to the context. The pseudo code of the steps is presented in Algorithm 2. In our example (dbpedia-plant), several important terms such as *organism*, *kingdom*, *plantae* are added to context.

3.7 Enriching the Disambiguation Context of a WordNet sense

To make the disambiguation context of a WordNet sense, we used the structural information. In our investigation, it was revealed that the sizes of the contexts of senses are much lower than the size of the context of web resources. While we benefit from structural and text information in a page description, we only have some limited structural related words in the context of a sense. Hence, to enrich the context of a sense, we added the words in the glossary part of a sense definition to the context (StopWords are removed).

Algorithm 2:

Input: Text (that is produced in line 15 of Algorithm 1), μ
Output: Enriched_context_collection
1: Enriched_context_collection=context_collection
2: Text is tokenized
3: Put the tokens in token_collection
4: For each Term in token_collection
5: If Term is a Stopword
6: Delete it
7: All = number of tokens in token_collection
8: For each Term in token_collection
9: Frequency(Term)= the number of Term occurrences in token_collection
10: Calculate the percentage of each Term: Percentage(Term) = Frequency(Term)/All
11: Calculate the average of percentages as AVG
12: For each Term in token_collection
13: If Percentage(Term) > μ * AVG
14: Add Term to Enriched_context_collection
15: Return Enriched_context_collection

---

4 Experiments

First we evaluated the effectiveness of the specified labelling properties in section 3.1 and then we performed several experiments to evaluate the RESD system. We implemented our experiments in Java environment and we used several APIs including Jena[1], JWordNet, Simpack. Simmetrics, SecondString[2], Lucene[3], etc.

[1] Jena API is used to retrieve the RDF page contents and make queries
[2] Three APIs Simpack, Simmetrics and SecondString are used to make string comparisons
[3] Lucene API is used to make functions such as tokenizing, stemming, removing StopWords, etc.

4.1 Labelling Properties Coverage

We conducted an experiment to find out the role of the labelling properties in the RDF web resources. We selected 100 random URI addresses from the BTC-2011 dataset that were used as the seed URIs. We extracted 5500 dereferenceable URI addresses from the Web of Data online using seed URIs. Triples in each web resource were retrieved to investigate whether they had used labelling properties. The results are shown in table 7. It shows that more than of 95% of the investigated resources contain at least one of the properties in table 3 and makes us highly able to identify the title of entities which are described in RDF resources.

RDF pages	RDF pages with labeling properties	RDF pages without labeling properties
5500	5263 (95.7%)	237 (4.3%)
The most used labeling property: http://www.w3.org/2000/01/rdf-schema#label		

Table 7: Performance of labeling properties

Number of senses	Algorithm	Annotators
0	38	12
1	214	193
2	194	185
3	177	169
4	89	86
5	37	54
More than 5	61	101
Average number of senses per page	2.53	2.96

Table 8: statistics of number of candidate senses. Column 2 and 3 show the number of candidate senses based on the result of our algorithm and the annotators point of view respectively.

	Precision	Recall	F-measure
MSC (most similar contexts)	96.45	93.27	94.83
MFS	52.37	53.17	52.77
Random	46.37	47.08	46.72

*Table 9: performance of the RDF Entity Sense Disambiguation algorithm with the values $\mu=2$, $\alpha=1$ and $\beta=0.3$ $\theta=150$, $\lambda=30$ and $D=2$. F-measure=2*P*R/(P+R)*

4.2 Evaluation of RDF Entity Sense Disambiguation

In the first experiment, we created a gold standard to evaluate the quality of RDF entity sense disambiguation. We selected 800 RDF pages from BTC-2011 dataset randomly and three annotators identified their corresponding senses in WordNet. In cases that the sense tags provided by two annotators were different, we voted to identify the correct sense (there was no case in which all the three annotators provided

different senses for an entity). The annotators were asked to query WordNet with the possible titles of a web resource to find the right sense. Table 8 provides the statistics of the number of candidate senses for 800 web resources and table 9 shows the best precision and recall of the entity sense disambiguation algorithm compared with the most frequent WordNet sense (MFS) and random sense selection approaches. The results show that our approach is able to find more than 96% of the annotated senses and it highly improves the quality of sense disambiguation compared with the other two methods.

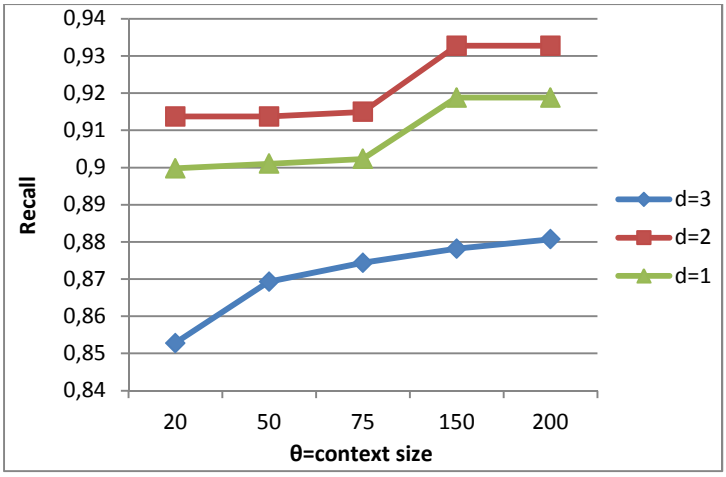

Figure 3: Recall of the RDF sense disambiguation algorithm. λ=30

To discover the role of some of the input parameters, we conducted several experiments. We assigned input parameters μ=2, α=1 and β=0.3 and executed our disambiguation algorithm with different values for the other input parameters θ, λ and D. In the experiments, we used the enriched contexts for both web resources and WordNet senses. Figures 3, 4, and 5 show the recall, precision, and F-measure of the algorithm respectively.

For most of the experiments, the best value for parameter D is 2 and probing WordNet deeper than this value not only does not improve the performance, but also results in worse precision and recall. It means that terms with depth 3 semantically are not strong enough to distinguish senses. The results also show that 150-200 terms can make the context of an RDF web resource effectively and using more terms does not enrich the context considerably. However, lower size values can highly decrease the effectiveness of sense disambiguation quality. Since making 150 terms is more efficient than making 200 terms, we use θ=150 for the best performance. Moreover, our experiments show that exploring 20-30 web resources is needed to provide 150 terms for the context. Table 10 Shows the F-measure values regarding to θ and λ.

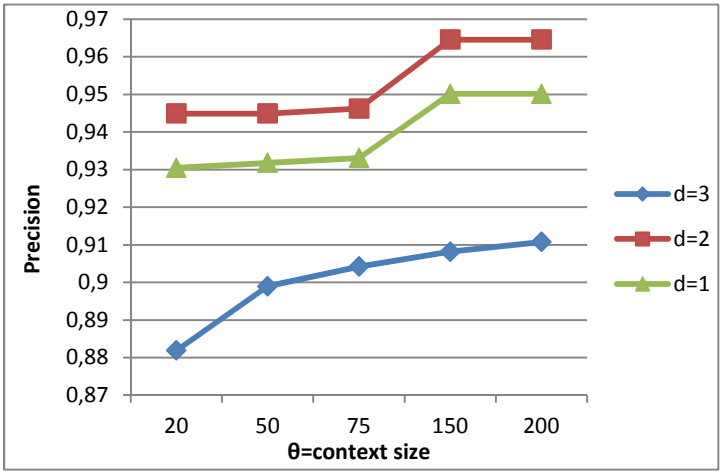

Figure 4: Recall of the RDF sense disambiguation algorithm. λ=30

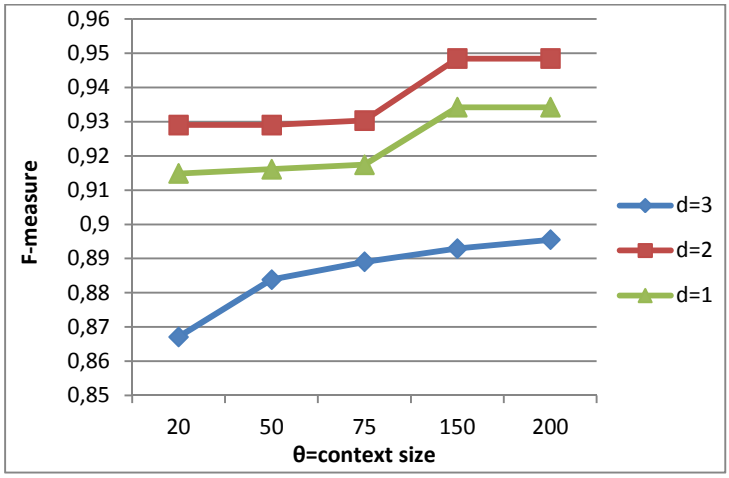

Figure 5: F-measure of the RDF sense disambiguation algorithm. λ=30

θ \ λ	10	15	20	30
20	0.891613	0.903226	0.922581	0.929032
50	0.891613	0.903226	0.922581	0.929032
75	0.891613	0.913548	0.929032	0.930323
150	0.891613	0.914839	0.947097	0.948387
200	0.891613	0.917419	0.947097	0.948387

Table 10: F-measure values regarding to θ and λ

We further investigated the role of WordNet's glossaries and frequent words in entity description. We conducted several experiments to identify the effect of richer contexts. The values of standard precision, recall and F-measure metrics are shown in table 11. Regarding the results, combination of enriched contexts (using glossaries and frequent words) dominated the other options. In addition, the effect of frequent words in the entity descriptions is more than the effect of glossaries in WordNet. Therefore, the abstracts, comments or any other texts in the content of web resources have an important role in context construction.

4.3 Role of RESD in interlinking

In the second experiment, we investigated the performance of RESD in interlinking process in Web of Data. Given two RDF web resources *w1*, and *w2*, if they map to the same entry in WordNet, they can be linked with *owl:sameAs* predicate. We extracted URIs pairs from the Web of Data where each pair describes the same entity and hence the same sense and then we evaluated the performance of our approach based on its capability to find the *owl:sameAs* relation between them.

RDF page	WordNet	Precision	Recall	F-measure
Main context	Structure	93.96	90.86	92.38
Main context + frequent words	Structure	95.66	92.51	94.06
Main context	Structure + glossary	94.35	91.24	92.77
Main context + frequent words	Structure + glossary	96.45	93.27	94.83

Table 11: the effect of glossary and frequent words in performances of disambiguation algorithm

To create the gold standard dataset, we extracted 20 million random triples from the BTC-2011 dataset. Among these triples, 4894 triples were separated where their predicates were *http://www.w3.org/2002/07/owl#sameAs* (triples with their object parts containing *sparql* queries or a blank node were removed). From 4894 triples, in 3780 triples at least one of the subject or object resources were not retrievable in RDF

format[1]. We probed the content of subject and object resources of the remaining 1114 triples automatically. Using labeling properties, we identified 1047 title pairs. We queried WordNet with the title pairs to identify their availability in WordNet. At least one of the titles in 719 title pairs did not exist in WordNet and 328 title pairs were included in it.

Number of senses	**Number**
1	324
2	116
3	101
4	21
5	45
More than 5	49
Average number of senses per page= 2.28	

Table 12: statistics of number of senses

	Recall (328 pairs)	Recall (all 1114 pairs)
MSC	96.64	30.27
MFS	64.32	20.15
Random	57.62	18.05

Table 13: the performance of sense disambiguation in interlinking process to find owl:sameAs relationship

We applied the sense disambiguation algorithm to 656 (328 pairs) to find the possible *owl:saemAs* links between them[2]. Table 12 provides the statistics of the titles' senses. Results in table 13 show that our algorithm is able to find 317 (more than 96%) pairs of equivalent entities and it is highly preferable to the results of MFS and Random sense methods. However, if we compare the results (i.e., discovering 317 links) with the whole links (i.e., 1047), we were able to discover 30.27% of the *owlsameAs* links. It is due to the fact that 68.72% (i.e., 719 out of 1047) of titles did not exist in WordNet database and therefore no sense is detected for further analysis. Hence, if the entities in datasets, which we desire to interlink, are defined in WordNet, the proposed algorithm has a high performance; otherwise it can be used as a complementary component for other interlinking approaches.

5 Conclusion and Future Work

In this paper, we identified the core labelling properties and presented a method for RDF Entity Sense Disambiguation. Our results show the specified labelling properties

[1] Some of the pages are in other formats such as html, etc., and some of the pages encounter errors such as page not found, IO exception, etc.

[2] Before applying our algorithm, we removed any existing links between URIs in their contents; therefore, there was no relationship between them beforehand

are able to discover the titles of web resources with a large coverage and they identify the candidate senses for RDF web resources effectively. The performance of the proposed sense disambiguation method shows that it is highly qualified and outperforms the MFS and Random sense selection methods. The results show that frequent words in entity description, glossaries in WordNet, and assigning appropriate parameter values have a considerable effect on the performance of the algorithm. Moreover, by disambiguating the sense of an entity, we are able interlink the datasets in Web of Data. However, its performance highly depends on the existence of entities in WordNet knowledge base. Our experiment showed that if this condition is met, a great number of *owl:sameAs* links are discoverable.

In future, we will conduct experiments to use several interlinking methods simultaneously with our sense detection algorithm to interlink datasets. Moreover, we will use other knowledge bases for disambiguation and interlinking purposes.

Acknowledgements

This research is supported by the Research Institute for ICT, University of Isfahan, and Databases and Information Systems in Institute for Informatik (Freiburg). We appreciate the anonymous people who helped us by reviewing the paper and performing the empirical study.

References

[Bizer, 2007] Bizer, C., Cyganiak, R., Heath, T.: "How to publish linked data on the web", (2008), http://wifo5-03.informatik.uni-mannheim.de/bizer/pub/LinkedDataTutorial/

[Bizer, 2009] Bizer, C., Heath, T., Lee, T. B.: "The Story So Far"; International Journal on semantic Web and Information Systems(IJSWIS),5 (2009), 1-22.

[Budanitsky, 2006] Budanitsky, A., Hirst, G.: "Evaluating WordNet-based Measures of Lexical Semantic Relatedness"; Journal of Computational Linguistics, 32, 1 (2006), 13-47.

[Cruz , 2001] Cruz, I., Palmonari, M., Caimi, F., Stroe, C.: "Towards on the go matching of linked open data ontologies"; In: IJCAI Workshop on Discovering Meaning on the Go in Large Heterogeneous Data, Spain (2011), 37-42.

[Edmonds 2000] Edmonds, P.: "Designing a task for SENSEVAL-2"; Technical report, University of Brighton, U.K. (2000).

[Ell, 2011] Ell, B., Vrandecic, D., Simperl, E.: "Labels in the Web of Data"; Proc of the 10th international conference on The semantic web, (2011), 162-176

[Fellbaum 1998] Fellbaum, C.: "WordNet: An Electronic Database"; MIT Press, Cambridge, MA (1998).

[Gharib, 2012] Gharib, T. F., Badr, N., Haridy, S., Abraham, A.: " Enriching Ontology Concepts Based on Texts from WWW and Corpus"; Journal of Universal Computer Science (J.UCS), 18, 16 (2012), 2234-2252.

[Gil, 2012] Gil, J. M., Delgado, I. N., Montes, J. F.: "MaF: An Ontology Matching Framework"; Journal of Universal Computer Science (J.UCS), 18, 2 (2012), 194-217.

[Hassanzadeh, 2009] Hassanzadeh, O., Mariano, C.: "Linked Movie Database"; Proc. World Wide Web workshop on Linked Data on the Web (2009).

[Hausenblas, 2008] Hausenblas, M., Halb, W., Raimond, Y.: "Scripting User Contributed Interlinking"*;* 4th Workshop on Scripting for the Semantic Web (SFSW), collocated with the European Semantic Web Conference (ESWC) (2008).

[Hausenblas, 2009] Hausenblas, M., Troncy, R., Raimond, Y., Bürger, T.:"Interlinking Multimedia: How to Apply Linked Data Principles to Multimedia Fragments"; Proc. Linked Data on the Web Workshop (LDOW), in conjunction with 18th Int. World Wide Web Conference, Madrid, Spain (2009).

[Hearst, 1998] Hearst, M.A., Dumais, S., Osman, E., Platt, J., Scholkopf, B.: "Support vector machines"; Intelligent Systems and their Applications, IEEE, 13, 4 (1998), 18–28.

[Jain, 2010] Jain, P., Hitzler, P., Sheth, A., Verma, K., Yeh, P.: "Ontology alignment for linked open data"; ISWC'10 Proceedings of the 9th international semantic web conference on the semantic web, (2010), 402-417.

[Lee, 2006] Lee, T. B.: "Linked Data", (2006), http://www.w3.org/DesignIssues/LinkedData.html

[Lee, 2004] Lee,Y. K., Ng, H. T., Chia, T. K.: "Supervised word sense disambiguation with support vector machines and multiple knowledge sources"; Proc. of Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text (2004), 137–140.

[Lesk, 1986] Lesk, M.: "Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone"; Proc. of the 5th annual international conference on Systems documentation.

[Navigli, 2009] Navigli, R.: "Word Sense Disambiguation: A survey"; ACM Computing Surveys, 41, 2 (2009), 1–69.

[Navigli, 2005] Navigli, R., Velardi, P.: "Structural semantic interconnections a knowledge-based approach to word sense disambiguation"; IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 27 (2005).

[Ng, 1997] Ng, H.T.: "Exemplar-based word sense disambiguation: Some recent improvements"; Proc. of the Second Conference on Empirical methods in natural Language Processing (1997), 208–213.

[Ngomo, 2011] Ngomo, A. C. N., Auer, S.: "LIMES - A Time-Efficient Approach for Large-Scale Link Discovery on the Web of Data"; Proc. International Joint Conferences on Artificial Intelligence (IJCAI) (2011).

[Nikolov, 2010] Nikolov, A., Motta, E.: "Capturing emerging relations between schema ontologies on the Web of Data; Proc. Consuming Linked Data at the 9th International Semantic Web Conference (ISWC), 7-11 November (2010).

[Nikolov, 2008] Nikolov, a., Uren, V., Motta, M., Roeck, A.: "Integration of semantically annotated data by the KnoFuss architecture"; Proc. 16th International Conference on Knowledge Engineering and Knowledge Management (EKAW), Acitrezza, Italy (2008).

[Pedersen, 2004] Pedersen, T., Siddharth, P., Jason, M.: " WordNet::Similarity: measuring the relatedness of concepts"; Proc. HLT-NAACL—Demonstrations, USA (2004), 38-41.

[Rada 2005] Rada, M.: 2005.: "Large vocabulary unsupervised word sense disambiguation with graph-based algorithms for sequence data labeling"; Proc. of the Joint Human Language

Technology and Empirical Methods in Natural Language Processing Conference (HLT/EMNLP) (2005), 411–418.

[Raimond, 2008] Raimond, Y., Sutton, C., Sandler, M.: "Automatic Interlinking of Music Datasets on the Semantic Web"; Proc. Linked Data on the Web Workshop at 17th Int. World Wide Web Conf., Beijing, China (2008).

[Scharffe, 2009] Scharffe, F., Liu, Y., Zhou, C.: *"RDF-AI: an Architecture for RDF Datasets Matching, Fusion and Interlink";* Proc. IJCAI workshop on Identity, reference, and knowledge representation (IR-KR), Pasadena, US (2009).

[Volz, 2009] Volz, J., Bizer, C., Gaedke, M., Kobilarov, G.: "Silk – A Link Discovery Framework for the Web of Data"; Proc. 2nd Linked Data on the Web, Madrid, Spain (2009).

[Walker, 1986] Walker, D., Amsler, R.: "The use of machine readable dictionaries in sublanguage analysis"; In Analyzing Language in Restricted Domains, Grish-man and Kittredge (eds), LEA Press, (1986), 69–83.