Cross-language Semantic Retrieval and Linking of E-gov Services

Fedelucio Narducci[1], Matteo Palmonari[1], and Giovanni Semeraro[2]

[1] Department of Computer Science, Systems and Communication
University of Milano-Bicocca, Italy
surname@disco.unimib.it
[2] Department of Computer Science
University of Bari Aldo Moro, Italy
giovanni.semeraro@uniba.it

Abstract. Public administrations are aware of the advantages of sharing Open Government Data in terms of transparency, development of improved services, collaboration between stakeholders, and spurring new economic activities. Initiatives for the publication and interlinking of government service catalogs as Linked Open Data (LOD) support the interoperability among European administrations and improve the capability of foreign citizens to access services across Europe. However, linking service catalogs to reference LOD catalogs requires a significant effort from local administrations, preventing the uptake of interoperable solutions at a large scale. The web application presented in this paper is named CroSeR (Cross-language Service Retriever) and supports public bodies in the process of linking their own service catalogs to the LOD cloud. CroSeR supports different European languages and adopts a semantic representation of e-gov services based on Wikipedia. CroSeR tries to overcome problems related to the short textual descriptions associated to a service by embodying a semantic annotation algorithm that enriches service labels with emerging Wikipedia concepts related to the service. An experimental evaluation carried-out on e-gov service catalogs in five different languages shows the effectiveness of our model.

1 Introduction and Motivations

As of May 2013, more than 1,000,000 Open Government Data sets (OGD) have been put online by national and local governments from more than 40 countries in 24 different languages[3]. The interconnection of OGD coming from different sources supports the retrieval, integration and analysis of information at a larger scale [12]. These advantages motivated the uptake of Linked Open Data (LOD), where information is interconnected by means of semantic links [1], as a paradigm for the publication of OGD on the web. Data linking is therefore a crucial step in the transition from OGD to Linked Open Government Data (LOGD) [3].

[3] http://logd.tw.rpi.edu/iogds_data_analytics

Initiatives such as the European Local Government Service List (LGSL) have published catalogs of services provided by public administrations of different European countries as LOGD. Linking service catalogs described in languages other than the ones available in the LGSL is a big opportunity for a large number of administrations to make their services more accessible and comparable at a cross-national level. However, discovering links between services described in different languages requires a significant human effort because of the number of service descriptions that have to be compared and because of the linguistic and cultural barriers.

Automatic cross-language semantic matching methods can support local and national administrations in linking their service catalogs to the LOD cloud, by reducing the cost of this activity. However, this domain poses several challenges to cross-language ontology matching methods proposed so far [17] because of the poor quality of the descriptions, which often consist of the name of the service and very few other data, and the semantic heterogeneity of names refereeing to linkable service, which is due to cultural differences across countries.

In this paper we propose *Cross-language Service Retriever* (CroSeR), a tool to support the linkage of a source service catalog represented in any language to a target catalog represented in English (i.e., LGSL), where both the source and target catalogs are characterized by minimal descriptions. Our tool is based on a cross-language semantic matching method that i) *translates* service labels in English using a machine translation tool, ii) automatically *extracts* a Wikipedia-based semantic representation from the translated service labels using the Explicit Semantic Analysis (ESA) technique [8], and iii) *evaluates* the similarity between two services using their Wikipedia-based representations. The user can select a service in a source catalog and use the ranked list of matches suggested by CroSeR to select the equivalent service in the LGSL. Our method is independent from the language adopted in the source catalog and it does not assume the availability of information about the services other than very short text descriptions used as service labels.

We conducted experiments with all the catalogs in five different languages available in the LGSL dataset; the experimental results show that CroSeR is effective in providing meaningful suggestions to the user and that the cross-language matching method presented in this paper outperforms several alternatives.

To the best of our knowledge, this is the first attempt to address the problem of linking e-gov service catalogs described in different languages; moreover, previous work using ESA to support cross-language link discovery [11], or ESA variants to support retrieval [18], extract semantic representation from reasonably long documents; the application of ESA to support cross-language link discovery between resources, for which only minimal descriptions are available, is a novel contribution of this paper. The development of effective cross-language matching techniques is also acknowledged as one of the challenges for realizing a multilingual Web of Data [9].

The rest of this paper is organized as follows. Section 2 describes the problem and presents the architecture and the functionalities of CroSeR. Section 3 provides an in-depth explanation of the cross-language matching method introduced in this paper. Experimental results are presented in Section 4. Finally, related work is discussed in Section 5 and conclusions are drawn in Section 6.

2 CroSeR: Cross-language Service Retriever

In this section, we first describe the problem context and characterize the critical issues that have to be considered by the cross-language matching techniques presented in this paper; afterwards, we provide an overview of CroSeR in terms of global architecture and functionalities.

2.1 Local Government Services in the lod Cloud

The SmartCities project[4] has the goal of creating an innovation network between governments and academic partners leading to excellence in the domain of the development and uptake of e-gov services, setting a new baseline for e-gov service delivery in the whole North Sea region. The project involves seven countries of the North Sea region: England, Netherlands, Belgium, Germany, Scotland, Sweden, and Norway. One of the main interesting results of this project is the European Local Government Service List (LGSL) as part of the Electronic Service Delivery (ESD)-toolkit website[5]. The goal of the LGSL is to build standard lists (i.e., ESD-standards) which define the semantics of public sector services. Each country involved into the project is responsible to build and maintain its list of public services delivered to the citizens, and all of those services are interlinked to the services delivered by other countries. The ESD-standards are already linked to the LOD cloud[6].

Services in the LGSL describe abstract functionalities of services that are concretely offered by a number of providers at a local level; a LGSL service such as *Homeless support*, represents more a category of services, rather than an individual service. However, following an approach also used by other e-gov service representation models, these categories are represented in a knowledge base as instances and can be referred to as abstract services [15][7]. For this reason, two services that are considered equivalent by domain experts and that belong to different catalogs in different languages are linked through a *sameAs* link. The aim of CroSeR is therefore to support the discovery of *sameAs* links according to the semantics adopted in the ESD-toolkit.

By linking national or local service catalogs to LGSL, a large number of local and national governments all over Europe can make their services searchable

[4] http://www.smartcities.info/aim

[5] http://www.esd.org.uk/esdtoolkit/

[6] http://lod-cloud.net/

[7] The ESD-toolkit allows local administrations to specify for each service links to web documents describing concrete services offered by individual providers; however, only a limited number of abstract services are linked to these concrete services.

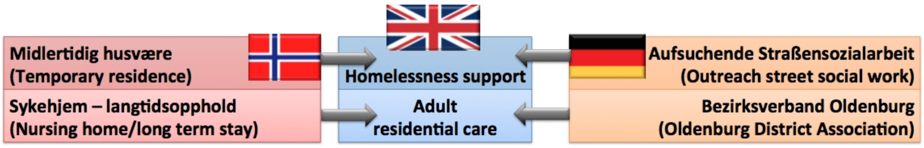

Midlertidig husvære
(Temporary residence)
Sykehjem – langtidsopphold
(Nursing home/long term stay)
Homelessness support
Adult
residential care
Aufsuchende Straßensozialarbeit
(Outreach street social work)
Bezirksverband Oldenburg
(Oldenburg District Association)

Fig. 1: Examples of linked services in the LGSL. Services linked by an arrow have an *owl:sameAs* relation in the LGSL. The automatic English translation powered by Bing is reported in brackets.

in several languages, improving also the capability of EU citizens to access services in a foreign language country, an explicit objective of the Digital Agenda for Europe (DAE) [2]. Moreover, local and national governments can learn best practices of service offerings across Europe and compare their service to make their service offering more valuable [15]. Finally, by linking e-gov service catalogs to LGSL additional information can be exploited, e.g., English services in the LGSL are linked to a taxonomy of life events, which is useful to enrich the service catalogs and support navigation. However, manually linking service catalogs, often consisting of several hundreds - or thousands - of services, to LGSL requires a lot of effort, which often prevents administrations from taking advantage of becoming part of the LOD cloud.

Automatic cross-language matching methods, which can reduce the effort needed to manually link these catalogs, have to deal with the poor quality of the service descriptions. Services are represented by minimal descriptions that often consist of the name of the service and very few other data. Furthermore, as showed in Figure 1, the labels associated with services linked in the LGSL are not a mere translation from a language to another. As an example, the Norwegian service (literally translated as) *Temporary residence* and the German service (literally translated as) *Outreach street social work* have been manually linked to the English service *Homeless support* by domain experts. Therefore, the automatic matching of the service text labels is not a trivial task.

2.2 CroSeR: Architecture and Functionalities

CroSeR is based on the hypothesis that extracting semantic annotations from service descriptions can support effective matching methods even if the available descriptions are poor and equivalent services can be described very differently in different countries; this is in fact the case for most of the service catalogs considered in the LGSL and for most of service catalogs provided by local administrations. We therefore assume that each service is described only by short textual description (i.e., service *label*) and represents a high-level description of a concrete service offered by one or more providers[8].

Figure 2 depicts the general architecture of CroSeR. We can observe three main components: the *Web GUI*, the *Content Analyzer* and the *Retriever*. The

[8] http://www.smartcities.info/files/Smart_Cities_Brief_What_is_a_service_list.pdf

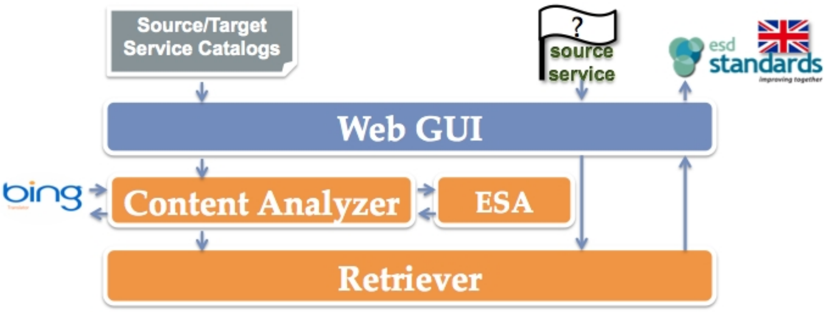


Fig. 2: CroSeR general architecture

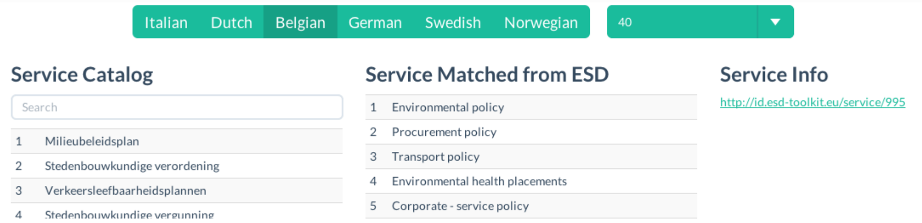
user interacts with CroSeR using the *Web GUI*. The *Content Analyzer* processes the service labels and builds a semantic annotation for each service; the content analyzer is used to process both the source and the target catalogs; in our case the source catalog is represented by a list of services labeled in any natural language, while the target catalog is represented by the LGSL labeled in English. The *Retriever* takes a semantically annotated service (in any language in which a source service catalog is available) as input and returns a ranked list of matching services; these services are the candidate targets of a *owl:sameAs* link from the input service. The *Content Analyzer* uses external automatic translation tools and Explicit Semantic Analysis (ESA) techniques to annotate services with Wikipedia concepts; however, other annotation method can be easily plugged into the CroSeR architecture (in fact, several semantic annotation methods are compared to the one proposed in this paper in Section 4). The *Retriever* component evaluates the similarity between a query represented by an input service, and the services in the target catalog. Before explaining the techniques adopted by the *Content Analyzer* and the *Retriever* components (see Section 3) we provide more insight on the functionality provided by the application and on the *Web GUI*.

The first step that the CroSeR user should perform is to upload his own service catalog into the system. After that, the catalog will be semantically analyzed and indexed. This step is generally not time expensive (according to the bandwidth available to the user). The user is now able to explore the catalog just uploaded by scrolling the whole list of services or by performing a keyword-based search (see Figure 3). Next, the user selects a source service from his own catalog and CroSeR retrieves a list of *candidate target services* from the LGSL that are potentially linkable by a *owl:sameAs* statement. The number of retrieved services is configurable by the user.

Municipalities often argue that they are different and that local government is different in different countries. Of course that is true and elected representatives have different priorities for local government. But much of the basic public services delivered locally are common to many countries[9]. However, sometimes

[9] http://esd-toolkit.eu/guidance/Standards.aspx

the connection between the source service and the target service could be not straightforward by simply comparing service labels. Hence, user can then select a candidate service and looks at further details (Service Info box) directly gathered from the ESD-toolkit.



Service Catalog

1	Milieubeleidsplan
2	Stedenbouwkundige verordening
3	Verkeersleefbaarheidsplannen
4	Stedenbouwkundige vergunning

Service Matched from ESD

1	Environmental policy
2	Procurement policy
3	Transport policy
4	Environmental health placements
5	Corporate - service policy

Service Info

http://id.esd-toolkit.eu/service/995

Fig. 3: CroSeR screenshot

Finally, user can switch on the feedback mode of CroSeR and thus the system stores the relation between the source service and the LGSL service after the selection of a candidate service from the retrieved list.

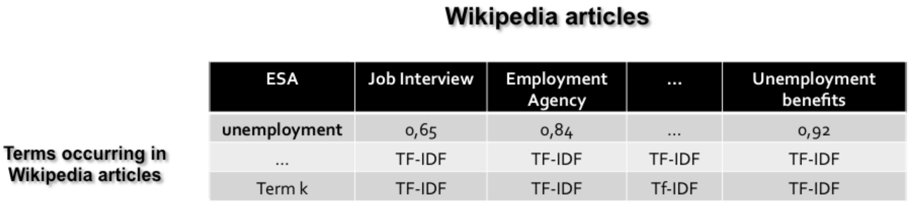
Please consider that a set of catalogs already linked to the LGSL (i.e., Dutch, Belgian, German, Swedish, Norwegian) are already uploaded in the demo of CroSeR available online[10]. In that case the web application shows the gold standard (by highlighting in green the service connected by *owl:sameAs* statement). Only for the Italian catalog the human annotation is not yet available. From a preliminary test, it seems that our model is effective on the Italian language, as well. As an example, given the Italian service *Arbitrati e conciliazioni* (translated as *Arbitrations and conciliations*), CroSeR is able to retrieve the correct service *Legal - litigation support* in LGSL that has not any common keyword with the input service.

3 Cross-language Service Annotation and Matching with Explicit Semantic Analysis

A simple and convenient way to represent textual description is called *bag of words* (BOW) and is the most used representation both in Information Filtering and Retrieval applications. In a BOW each item (e.g., service) is represented by the set of words in the text, together with their number of occurrences. In CroSeR we adopted an enhaced version of the BOW that, according to Sorg et al. [18], is called *bag of concepts* (BOC). In a BOC each service is represented by a set of Wikipedia concepts most related to the service label. To this purpose we exploited (ESA) [8] that allows to represents terms and documents using Wikipedia pages (concepts). Accordingly, for each service label a set of Wikipedia concepts is generated.

[10] http://siti-rack.siti.disco.unimib.it:8080/croser/

ESA views an encyclopedia (i.e., Wikipedia) as a collection of concepts (articles), each one provided with a large textual description (the article content). The power of ESA is the capability of representing Wikipedia's knowledge base in a way that is directly used by a computer software, without the need for manually encoded common-sense knowledge. Therefore, ESA uses Wikipedia as a space of concepts explicitly defined and described by humans. Formally, given the space of Wikipedia concepts $C = \{c_1, c_2, ..., c_n\}$, a term t_i can be represented by its *semantic interpretation vector* $v_i =< w_{i1}, w_{i2}, ..., w_{in} >$, where w_{ij} represents the strength of the association between t_i and c_j. Weights are obtained from a matrix T, called ESA-*matrix*, in which each of the n columns corresponds to a concept, and each row corresponds to a term of the Wikipedia vocabulary (i.e., the set of distinct terms in the corpus of all Wikipedia articles). Cell $T[i, j]$ contains w_{ij}, the TF-IDF value of term t_i in the article (concept) c_j. Therefore, the semantic interpretation vector for a given term is the corresponding row vector in the ESA-*matrix*. As an example, the meaning of the generic term *unemployment* can be described by a list of concepts (the semantic interpretation vector) it refers to (e.g., the Wikipedia articles for: *job interview, employment agency, unemployment benefits,...*) (see Figure 4). The semantic interpretation vector for a text fragment f (i.e. a sentence, a document, a service label) is obtained by computing the centroid (average vector) of the semantic interpretation vectors associated with terms occurring in f.


Wikipedia articles
Terms occurring in Wikipedia articles

ESA	Job Interview	Employment Agency	...	Unemployment benefits
unemployment	0,65	0,84	...	0,92
...	TF-IDF	TF-IDF	TF-IDF	TF-IDF
Term k	TF-IDF	TF-IDF	Tf-IDF	TF-IDF

Fig. 4: The ESA-matrix

The motivation behind the use of ESA in CroSeR is twofold: 1) ESA is able to perform a sort of word sense disambiguation (WSD) based on the semantics *explicitly* used by humans [8] ; 2) ESA is able to generate new knowledge in terms of Wikipedia concepts most related to a given unstructured text.

Please consider the service label *bank account.* The term *bank* is a polysemous word (meanings related to finance, geography, computing, etc.). If we extract the semantic interpretation vector for *bank*=⟨The bank (1915 film) (0.50), Memory bank (0.49), ..., Bank account (0.47), ...⟩ and *account*=⟨Bank account (0.75), Savings account (0.70), ..., Cynthia Cooper (accountant)(0.08), ...⟩ , by computing their centroid vector we obtain *bank account* = ⟨Bank Account (0.61), Savings Account (0.35), ...⟩ by boosting in the first position the most related concept in that specific context. As regards the second motivation behind the adoption of ESA, we can consider the service label *Home Schooling.* ESA generates (as centroid vector) the Wikipedia articles ⟨Home (0.67), School (0.55), Education (0.48), Family (0.35), ...⟩ by adding new knowledge that is not di-

rectly extractable from the input text and thus enriching the short service label.
Semantic Annotation. CroSeR supports any language for which an automatic translation is available. Indeed, before the generation of a Wikipedia based representation, an automatic translation process powered by Bing[11] is performed and every *service label* is translated in English. Subsequently, the translated labels are used by another component called ESA that is able to generate an ESA-based representation of the services[12]. Therefore, for each service s, a set of Wikipedia concepts W_s (i.e., the semantic interpretation vector) semantically related to the service label is generated. The Wikipedia-based representations are then indexed by Lucene. This step is performed by the *Content Anayzer* (see Figure 2).
Service Matching. Indexed services are represented by using the Vector Space Model (VSM). A multidimensional space in which each dimension is a Wikipedia concept is thus built. Accordingly, a service is a point in that space. Formally, each service is represented as a vector $\boldsymbol{s} =< w_1, \ldots, w_n >$ where w_k is the TF-IDF value of the the Wikipedia concept. Finally, the similarity between two services (vectors) is computed in terms of cosine similarity.

Therefore, given a source service in one of the supported languages (the query), CroSeR is able to return a ranked list of the most similar English services from the LGSL. This last step is performed by the *Retriever* (see Figure 2).

4 Experimental Evaluation

We carried out an *in-vitro* evaluation of CroSeR on five catalogs already linked to the LGSL. Links between services belonging to different catalogs are in terms of *owl:sameAs* statement and are made by human experts. The goals of our experiment were to evaluate: (1) the effectiveness in retrieving the correct service in a list of n service to be presented to the user, (2) the capability in boosting the correct service in the first positions of the ranked list.

We compared the representation based on ESA with other state-of-the-art Wikipedia-based representations.
Other Annotating techniques. In order to validate our experimental results, we adopted also other techniques for semantically annotate service labels with a set of Wikipedia concepts. In particular, we adopted three well-known on-line services that perform semantic annotation, namely Wikipedia Miner, Tagme, DBpedia Spotlight. The on-line services take as input a text description (the service label), and return a set of *Wikipedia concepts* that emerge from the input text. All those services allow to configure some parameters in order to favor recall or precision. Given the conciseness of the input text in our domain, we set those parameters for improving the recall instead of precision.

[11] http://www.microsoft.com/en-us/translator/

[12] Since other implementations of ESA available online did not satisfy our requirements (e.g., do not comply with all heuristics defined by Gabrilovich and colleagues [8]) we developed a new version of ESA.

- **Wikipedia Miner.** Wikipedia Miner is a tool for automatically cross-referencing documents with Wikipedia [13]. The software is trained on Wikipedia articles, and thus learns to disambiguate and detect links in the same way as Wikipedia editors [5].
- **Tagme.** Tagme is a system that performs an accurate and on-the-fly semantic annotation of short texts via Wikipedia as knowledge base [6]. The annotation process is composed of two main phases: the *disambiguation* and the *pruning.*
- **DBpedia Spotlight.** DBpedia Spotlight [12] was designed with the explicit goal of connecting unstructured text to the LOD cloud by using DBpedia as hub. Also in this case the output is a set of Wikipedia articles related to a text retrieved by following the URI of the DBpedia instances.

We can observe that while the intuition behind Wikipedia Miner, Tagme, and DBpedia Spotlight is quite similar, ESA implements a different approach. Indeed, the first three tools identify Wikipedia concepts already present in the text, conversely ESA generates new articles related to a given text by using Wikipedia as knowledge base. As an example, let us suppose that we want to annotate the service label *Home Schooling.* Wikipedia Miner, Tagme and DBpedia Spotlight link it to the Wikipedia article *Homeschooling*, while ESA generates (as centroid vector) the Wikipedia articles *Home, School, Education, Family, ...* Hence, we can state that the three first tools perform a sort of topic *identification* of a given text, while ESA performs a feature *generation* process by adding new knowledge to the input text. Another example enforces the motivation behind the need of producing a semantic annotation of the service labels. Let's consider the English service label *Licences - entertainment* and the corresponding Dutch service *Vergunning voor Festiviteiten* (translated as: *Permit for Festivities*). A keyword-based approach never matches these two services. Conversely, the Tagme annotation generates for the English Service the Wikipedia concepts *License, Entertainment*, and for the translated Dutch label the concepts *License, Festival.* In addition to those Wikipedia-based representations, we evaluated our system also by setting hybrid representations obtained by merging the keywords extracted from the label associated to the service with the corresponding Wikipedia concepts.

Experimental Design and Dataset. We adopted two different metrics: *Accuracy@n (a@n)* and *Mean Reciprocal Rank* (MRR) [20]. The *a@n* is calculated considering only the first n retrieved services. If the correct service occurs in the *top-n* items, the service is marked as correctly retrieved. We considered different values of $n = 1, 3, 5, 10, 20, 30$. The second metric (MRR) considers the rank of the correct retrieved service and is defined as follows:

$$MRR = \frac{\sum_{i=1}^{N} \frac{1}{rank_i}}{N}, \tag{1}$$

where $rank_i$ is the rank of the correctly retrieved $service_i$ in the ranked list, and N is total number of services into the catalog. The higher is the position of the services correctly retrieved in the list, the higher is the MRR value for a given representation.

The dataset is extracted from the ESD-toolkit catalogue freely available online[13]. We indexed English, Dutch, German, Belgian, Swedish, and Norwegian catalogs. It is worth noting that even if Dutch and Belgian services are represented in the same language (i.e., Dutch), services links to the same LGSL item have generally different labels. For example the English service *Primary school places* has the label *Leerplicht* in the Dutch catalog, whereas has the label *Basisonderwijs* in the Belgian one. The labels have an average length of about three words. The catalogs have different size and each catalog links a different number of services to the LGSL (Dutch = 225, German = 190, Belgian = 341, Norwegian = 165, Swedish = 66, LGSL = 1,425, **TOTAL = 2,422 services**).

Results and Discussion. The baseline of our experiment is the keyword-based representation. For that representation, only stemming and stopword elimination are performed on the text.

Generally speaking, results in terms of *a@n* follow the same trend for all languages (see Figures 5,6,7,8,9). ESA is the representation with the best accuracy for the most of n values and representations. It is also the representation with the largest gap with respect to the baseline (i.e., keyword). Furthermore, ESA is the only representation that does not show any improvement by combining Wikipedia concepts with keywords (i.e., *esa+keyword*). This is due to the fact that ESA generally outperforms the keyword-based representation and thus the merging does not produce any benefit. The worst representation is generally Wikipedia Miner, followed by Tagme. As opposed to ESA, those representations improve their accuracy by merging Wikipedia concepts with keywords, but they generally do not outperform the representation only based on keywords (except for *dbpedia+keyword* that shows a slight improvement). There are also differences in terms of highest accuracy values among the different catalogs. The system seems to be more accurate on the Norwegian, Dutch and Belgian catalogs. The motivation behind these differences could be related also to the efficiency of the translation process from the different languages.

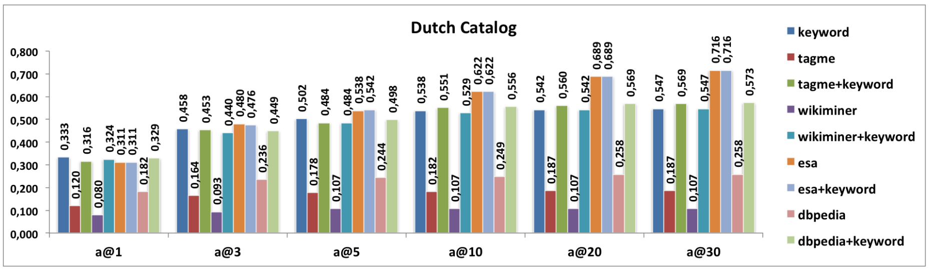


Fig. 5: Accuracy for the Dutch catalog

[13] http://standards.esd-toolkit.eu/EuOverview.aspx

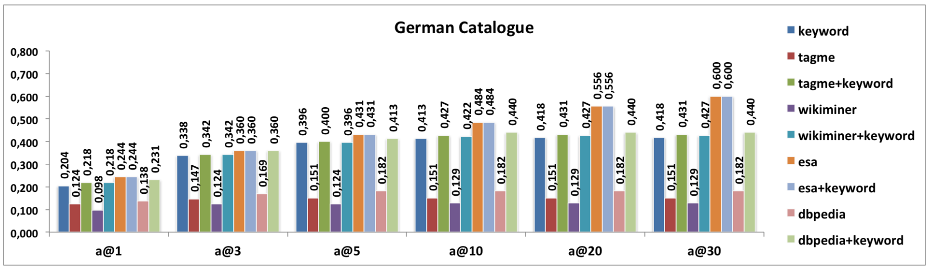


Fig. 6: Accuracy for the German catalog

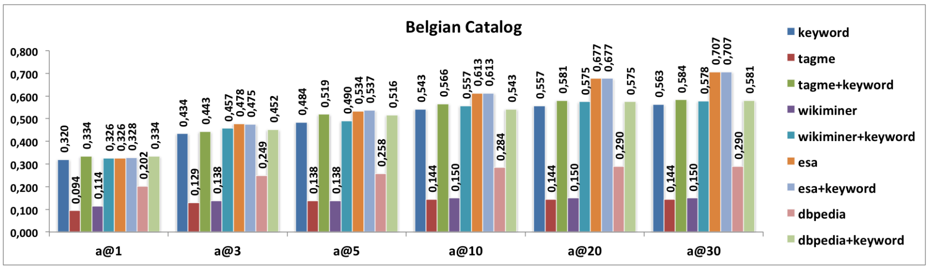


Fig. 7: Accuracy for the Belgian catalog

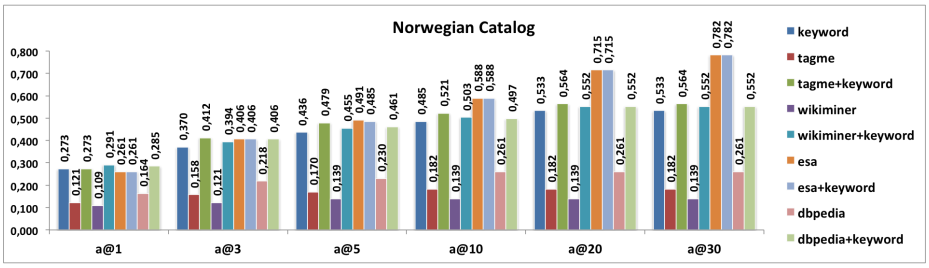


Fig. 8: Accuracy for the Norwegian catalog

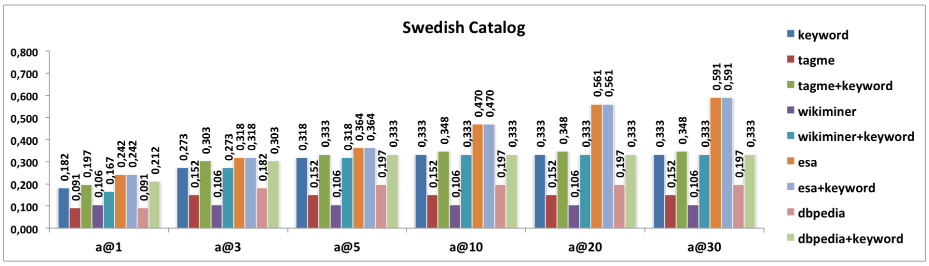


Fig. 9: Accuracy for the Swedish catalog

We can certainly state that ESA is the most effective representation in terms of accuracy. In order to statistically validate our experiment we compared results obtained by the keyword-based representation with results obtained by the ESA-based one. Table 1 reports levels of significance (*p-value*) obtained by performing the Wilcoxon Matched Pairs Test. More specifically, the number reported in each cell shows the statistic significance (*p-value*) of the differences between keywords and ESA for each n value of *a@n*. Empty cells show no statistically significant difference. We can observe that for the Belgian catalog (that is also the richest one) the improvement of ESA is statistically significant for each n value. Conversely, other catalogs show statistically significant differences from $n = 10$ onwards. These results can be considered actually satisfying, since starting from 10 retrieved items CroSeR becomes significantly better than a keyword-based model. The second analysis focuses the attention on the capability of CroSeR to boost relevant services in the first positions of the retrieved list. Results in terms of MRR for each representation are reported in Table 2. For Belgian, German, and Swedish catalogs the representation based on ESA shows the highest values, but differences with other representations (*keyword* and *wikifi+keyword*) are really slight. Accordingly, there is not a representation that decisively outperform the baseline for this metric. However, since CroSeR is a retriever system, and not, for example, a question-answering engine (for which to have the correct answer in the first position plays a crucial role), we can consider these results good. Indeed, the average rank of the correct service for the ESA representation is between the

Table 1: Wilcoxon test for Keyword-based vs. ESA-based representation

Catalog	**a@1**	**a@3**	**a@5**	**a@10**	**a@20**	**a@30**
Dutch				0.01	0.01	0.01
Belgian	0.05	0.01	0.01	0.01	0.01	0.01
German				0.01	0.01	0.01
Norwegian				0.01	0.01	0.01
Swedish				0.01	0.01	0.01

Table 2: MRR values for each representation

Representation	**Dutch**	**Belgian**	**German**	**Norwegian**	**Swedish**
keyword	**0.333**	0.320	0.242	0.273	0.182
tagme	0.120	0.094	0.147	0.121	0.091
tagme+keyword	0.316	0.334	0.258	0.273	0.197
wikifi	0.080	0.114	0.116	0.109	0.106
wikifi+keyword	0.324	0.326	0.258	**0.291**	0.167
esa	0.311	0.326	**0.289**	0.261	**0.242**
esa+keyword	0.311	**0.328**	**0.289**	0.261	**0.242**
dbpedia	0.182	0.202	0.163	0.164	0.091
dbpedia+keyword	0.329	0.334	0.274	0.285	0.212

third and *fifth* position of the retrieved list[14]. Therefore, even tough the reranking is surely an aspect that needs to be further investigated, we can assume that these results can be considered satisfying.

It is worth noting that for some source services CroSeR fails to find the correct match or cannot find any match. For example, consider the two services in Figure 1 *Adult residential care* (English) and the corresponding German service *Oldenburg District Association.* It is really hard for an automatic tool to find a correspondence between those two labels without any additional content that explains that the adult residential care is in charge to the Oldenburg District Association in Germany. Furthermore, in several cases when the match defined as correct in the Gold Standard is not returned in the top results by CroSeR, the suggested candidate services are still semantically related to the best match selected by domain experts. For example, for the Norwegian service *Nursing home/long term stay*, CroSeR suggests the services *Care at home, Care - home assessment, Residential care home registration* that are surely semantically related to the target service chosen by the human expert (*Adult residential care*). However, in the experimental evaluation that is considered as a mistake.

5 Related Work

Most of e-gov services are not described with rich Semantic Web Service (SWS) representation models [4]. Therefore, sophisticated matchmaking methods proposed for SWSs cannot be applied in our domain, even in the case of mono-lingual service descriptions. Relevant work to CroSeR can be found in ontology matching, link discovery, and entity linking, which are tightly related research areas.

In all of these areas, automatic or semi-automatic matching techniques are applied to discover correspondences among *semantically related entities* that appear in a source and a target information source [17]. Different types of correspondences have been addressed (e.g., *equivalence*, *subclass*, *same as*, and so on), depending on the types of considered entities (e.g., ontology concepts, ontology instances, generic RDF resources) and information sources (web ontologies, linked datasets, semi-structured knowledge bases). *Cross-language* ontology matching is the problem of matching a source ontology that uses terms from a natural language $\mathcal{L}$ with a target ontology that uses terms from another natural language $\mathcal{L}'$ (e.g., $\mathcal{L}$ is German and $\mathcal{L}'$ is English) [19]; *multi-lingual ontology matching* is the problem of matching two ontologies that use more than one language each, where the languages used in each ontology can also overlap [19]. These definitions can be easily extended to semantic matching tasks over other types of information sources (e.g., cross-language matching of two document corpuses). In the following we discuss the most relevant approaches to cross-language matching proposed over different information sources.

The most adopted approach for cross-language ontology matching is based on transforming a cross-lingual matching problem into a monolingual one by lever-

[14] Please remember that RR is 1 if a relevant document was retrieved at rank 1, if not it is 0.5 if a relevant document was retrieved at rank 2 and so on.

aging automatic machine translation tools [19, 7, 21]. However, the accuracy of automatic machine translation tools is limited and several strategies have been proposed to improve the quality of the final matchings. One of the most recent approaches uses a Support Vector Machine (SVM) to learn a matching function for ontologies represented in different languages [19]. This method uses features defined by combining string-based and structural similarity metrics. A translation process powered by Microsoft Bing[15] is used to build the feature vectors in a unique reference language (English). A first difference with respect to our work is that the proposed approach is deeply based on structural information derived from the ontology; this information is very poor in our scenario and is not used in our method. Also other translation-based approaches use structural information, i.e., neighboring concepts [7] and instances [21], which is not available in our scenario.

Two ontology matching methods have been recently proposed, which use the concepts' names, labels, and comments to build *search keywords* and query web data. A first approach queries a web search engine and uses the results to compute the similarity between the ontology concepts [16]. The system supports also cross-language alignment leveraging the Bing API to translate the keywords. A second approach submit queries to the Wikipedia search engine [10]. The similarity between a source and target concept is based on the similarity of the Wikipedia articles retrieved for the concepts. Cross-language matching is supported by using the links between the articles written in different languages, which are available in Wikipedia, and by comparing the articles in a common language. The authors observe that their approach has problems when it tries to match equivalent ontology elements that use a different vocabulary and lead to very different translations (e.g., *Autor von(de)* and *has written(en)*). Despite we do also leverage Wikipedia, our matching process uses semantic annotation tools and ESA. We can therefore incorporate light-weight disambiguation techniques (provided by the semantic annotation tools) and match entities that, when translated, are represented with significantly different terms (in particular when the system use the ESA model).

Another interesting work presented in literature applies the Explicit Semantic Analysis (ESA) for cross-language link discovery [11]. The goal of that paper is to investigate how to automatically generate cross-language links between resources in large document collections. The authors show that the semantic similarity based on ESA is able to produce results comparable to those achieved by graph-based methods. However, in this specific domain, algorithms can leverage a significant amount of text that is not available in our case. A cross-language version of ESA (CL-ESA that does not require any translation process of the input text) is proposed in [18] for cross-lingual and multilingual retrieval. CL-ESA was evaluated on multilingual documents provided with a quite large textual descriptions. We preliminarily evaluated CL-ESA in CroSeR, however, likely due to the the concise descriptions available in our domain, results were not satisfying.

[15] http://www.bing.com/translator

Finally, we mention that the preliminary experiments on cross-language semantic matching of e-Gov services have been presented in a previous paper [14]. However, the CroSeR web application based on ESA represents a novel contribution of this paper and previous experiments have been significantly extended in terms of languages considered and in-depth analysis of the results.

6 Conclusions and Future Work

In this paper we presented a web application called CroSeR that supports the linking of multilingual catalogs of e-gov services to the LOD cloud. CroSeR adopts a Wikipedia-based representation of services based on Explicit Semantic Analysis. We carried out an *in-vitro* experiment on five different languages and results showed the effectiveness of our approach. Furthermore, one of the strongest point of our model is the that the extension to other languages is straightforward. Therefore, CroSeR could be a valuable solution for supporting public bodies in linking their own service catalogs to the LOD cloud, profiting by all advantages that this connection entails.

We are also investigating the effectiveness of another service based on CroSeR, actually implemented in very preliminary stage, that is able to get as input a query formulated in natural language. In the future work we will improve this prototype. We will also try to improve the accuracy of CroSeR by gathering additional content related to services (for example Google snippets or other documents retrieved with web searches). Finally, we will validate the *in-vitro* results by carrying out an experiment with real users on an Italian catalog of e-gov services (already available in the online demo of CroSeR) and investigate the effectiveness of CroSeR when a different language is used in the target catalog.

Acknowledgements

The work presented in this paper has been partially supported by the Italian PON project PON01_00861 SMART (Services and Meta-services for smART eGovernment) and the EU project FP7-INCO 295006 SIERA (integrating Sina Institute into the European Research Area). The authors wish to thank Dr Claudio Baldassarre and Dr Marco Cremaschi for implementing the Web GUI of CroSeR.

References

1. C. Bizer, T. Heath, and T. Berners-Lee. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, 5(3):1–22, 2009.
2. European Commission. A digital agenda for europe. *COM(2010) 245 final/2*, 2010.
3. L. Ding, V. Peristeras, and M. Hausenblas. Linked Open Government Data. *IEEE Intelligent Systems*, 27(3):11–15, 2012.

4. D. Fensel, F. Michele Facca, E. Paslaru Bontas Simperl, and I. Toma. *Semantic Web Services*. Springer, 2011.
5. S. Fernando, M. Hall, E. Agirre, A. Soroa, P. Clough, and M. Stevenson. Comparing taxonomies for organising collections of documents. In *Proceedings of COLING '12*, pages 879—894. Indian Institute of Technology Bombay, 2012.
6. P. Ferragina and U. Scaiella. Tagme: on-the-fly annotation of short text fragments (by Wikipedia entities). In *Proceedings of CIKM '10*, pages 1625–1628. ACM, 2010.
7. B. Fu, R. Brennan, and D. O'Sullivan. Using pseudo feedback to improve cross-lingual ontology mapping. In *Proceedings of ESWC '11*, pages 336–351. Springer-Verlag, 2011.
8. E. Gabrilovich and S. Markovitch. Wikipedia-based semantic interpretation for natural language processing. *Journal of Artificial Intelligence Research*, 34:443–498, 2009.
9. J. Gracia, E. Montiel-Ponsoda, P. Cimiano, A. Gómez-Pérez, P. Buitelaar, and J. McCrae. Challenges for the multilingual web of data. *Web Semantics*, 11:63–71, March 2012.
10. S. Hertling and H. Paulheim. WikiMatch - Using Wikipedia for Ontology Matching. In *Proceedings of the 7th International Workshop on Ontology Matching (OM 2012)*. CEUR, 2012.
11. P. Knoth, L. Zilka, and Z. Zdrahal. Using explicit semantic analysis for cross-lingual link discovery. In *Proceedings of 5th International Workshop on Cross Lingual Information Access: Computational Linguistics and the Information Need of Multilingual Societies*, 2011.
12. P. N. Mendes, M. Jakob, A. García-Silva, and C. Bizer. DBpedia Spotlight: Shedding light on the web of documents. In *Proceedings of I-SEMANTICS '10*, pages 1–8. ACM, 2011.
13. D. Milne and I. H. Witten. Learning to link with Wikipedia. In *Proceedings of CIKM '08*, pages 509–518. ACM, 2008.
14. F. Narducci, M. Palmonari, and G. Semeraro. Cross-language semantic matching for discovering links to e-gov services in the LOD cloud. In *Proceedings of the 2nd International Workshop on Knowledge Discovery and Data Mining Meets Linked Open Data, co-located with ESWC '13'*. CEUR Workshop, 2013.
15. M. Palmonari, G. Viscusi, and C. Batini. A semantic repository approach to improve the government to business relationship. *Data Knowl. Eng.*, 65(3):485–511, 2008.
16. H. Paulheim. WeSeE-Match results for OEAI 2012. In *Proceedings of the 7th International Workshop on Ontology Matching (OM 2012)*, 2012.
17. P. Shvaiko and J. Euzenat. Ontology matching: State of the art and future challenges. *IEEE Trans. Knowl. Data Eng.*, 25(1):158–176, 2013.
18. P. Sorg and P. Cimiano. Exploiting Wikipedia for cross-lingual and multilingual information retrieval. *Data & Knowledge Engineering*, 74(0):26 – 45, 2012. Applications of Natural Language to Information Systems.
19. D. Spohr, L. Hollink, and P. Cimiano. A machine learning approach to multilingual and cross-lingual ontology matching. In *Proceedings of ISWC 2011*, pages 665–680. Springer-Verlag, 2011.
20. E. M. Voorhees. TREC-8 question answering track report. In *Proceedings of TREC-8*, pages 77–82. NIST Special Publication 500-246, 1999.
21. S. Wang, A. Isaac, B. Schopman, S. Schlobach, and L. Van Der Meij. Matching multi-lingual subject vocabularies. In *Proceedings of ECDL '09*, pages 125–137, Berlin, Heidelberg, 2009. Springer-Verlag.