Cross-language semantic matching for discovering links to e-gov services in the LOD cloud

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

[1] Department of Information Science, Systems Theory, 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. The large diffusion of e-gov initiatives is increasing the attention of public administrations towards the Open Data initiative. The adoption of open data in the e-gov domain produces different advantages in terms of more transparent government, development of better public services, economic growth and social value. However, the process of data opening should adopt standards and open formats. Only in this way it is possible to share experiences with other service providers, to exploit best practices from other cities or countries, and to be easily connected to the Linked Open Data (LOD) cloud.

In this paper we present CroSeR (Cross-language Service Retriever), a tool able to match and retrieve cross-language e-gov services stored in the LOD cloud. The main goal of this work is to help public administrations to connect their e-gov services to services, provided by other administrations, already connected to the LOD cloud. We adopted a Wikipedia-based semantic representation in order to overcome the problems related to match really short textual descriptions associated to the services. A preliminary evaluation on an open catalog of e-gov services showed that the adopted techniques are promising and are more effective than techniques based only on keyword representation.

1 Introduction and Motivations

The main motivation behind the success of the Linked Open Data (LOD) initiative is related to well-known advantages coming from the interconnection of information sources, such as improved discoverability, reusability, and utility of information [11]. In the last years, many governments decided to make public their data about spending, service provision, economic indicators, and so on. These datasets are also known as Open Government Data (OGD). As of February 2013, more than 1,000,000 OGD datasets have been put online by national and local governments from more than 40 countries in 24 different languages[3].

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

As the interest of governments in LOD has grown over the last years, a roadmap consisting of three data-processing stages, namely the open stage, the link stage, and the reuse stage, has been proposed to drive the transition from OGD to Linked Open Government Data (LOGD) [2].

The SmartCities project[4] is worth of mentioning in this context. The general aim of that project is to create an innovation network between governments and academic partners leading to excellence in the domain of the development and uptake of e-services, setting a new baseline for e-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].

LGSL is a great opportunity for local and national governments all over Europe. Linking national or local service catalogs to LGSL allows to make local or national services searchable 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) [1]. 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 [13]. Finally, by linking e-service catalogs to LGSL additional information can be exploited, e.g., 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 e-service catalogs, often consisting of several hundreds - or thousands - of services, to LGLS requires a lot of effort, which often prevents administrations from taking advantage of becoming part of the LOD cloud.

Automatic cross-language ontology matching methods can support local and national administrations in linking their service catalogs to LGSL, and therefore to the LOD cloud, by reducing the cost of this activity. Although some cross-language ontology matching methods have been proposed [16], the application of these methods to the problem of linking local and national service catalogs has to deal with the poor quality of the descriptions available in the catalogs. 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 German service (literally translated as) *Acquisition of children daycare contributions* and the Dutch service (literally translated as) *Grant Babysitting/Child Services* have been manually linked to

[4] http://www.smartcities.info/aim
[5] http://www.esd.org.uk/esdtoolkit/
[6] http://lod-cloud.net/

the English service *Nursery education grant* by domain experts. Therefore, the automatic matching of the service text labels is not a trivial task.

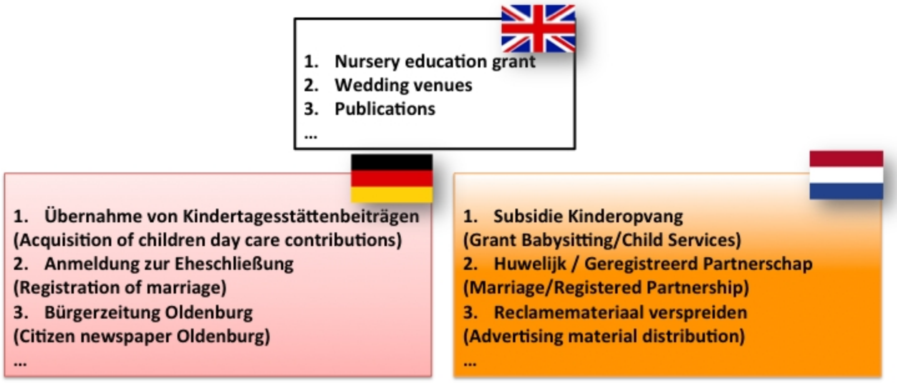
1. Nursery education grant
2. Wedding venues
3. Publications
...
1. Übernahme von Kindertagesstättenbeiträgen
(Acquisition of children day care contributions)
2. Anmeldung zur Eheschließung
(Registration of marriage)
3. Bürgerzeitung Oldenburg
(Citizen newspaper Oldenburg)
...
1. Subsidie Kinderopvang
(Grant Babysitting/Child Services)
2. Huwelijk / Geregistreerd Partnerschap
(Marriage/Registered Partnership)
3. Reclamemateriaal verspreiden
(Advertising material distribution)
...

Fig. 1: Examples of linked services in the LGSL. Services with the same ID number are linked by the *owl:sameAs* relation in the LGSL. The automatic English translation powered by Bing is reported in brackets.

In this paper we propose *Cross-language Service Retriever (CroSeR)*, a tool to support the linkage of a source e-service catalog represented in any language to a target catalog represented in English, where both the source and target catalogs are characterized by minimal descriptions. Ultimately, the aim of CroSeR is to support human annotators in order to simplify the selection of possibly matching services. Our tool exploits a cross-language ontology matching technique that uses an off-the-shelve machine translation tool and annotates the translated descriptions with Wikipedia concepts in order to extrapolate semantic representations of the services; candidate links are retrieved by evaluating the similarity between the extracted semantic representations. Our method is independent from the language adopted in the source catalog and does not assume the availability of further information about the services other than very short text descriptions used as names for the services.

We conduct an experiment using the English, German and Dutch catalogs from the LGSL dataset. In the experiment we compare several configurations of our system that leverage different semantic annotation tools and the Explicit Semantic Analysis (ESA) [7] method. Our preliminary results show that the method based on ESA outperforms both the methods based on other annotation tools and a baseline where no semantic representation is used.

The rest of this paper is organized as follows. Section 2 analyzes the state of the art. Section 3 describes the general architecture of our system, and Section 4 shows the tools exploited for obtaining a Wikipedia-based representation of e-gov services. Finally, experimental results are presented in Section 5 and in Section 6 the conclusion and the future work are summarized.

2 Related Work

Ontology matching, link discovery, and entity linking 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 [16]. 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) [17]; *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 [17]. These definitions can be easily extended to semantic matching tasks over other types of information sources (e.g., cross-language and multi-lingual 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 leveraging automatic machine translation tools [17, 6, 19]. 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 [17]. This method uses features defined by combining string-based and structural similarity metrics. A translation process powered by Microsoft Bing[7] 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 [6] and instances [19], 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 [14]. 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 [8]. 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,

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

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 [9]. 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.

Finally, the impact of the translation quality on the quality matching in cross-lingual scenarios is investigated in [5]. From that research it emerges that good translation quality is a prerequisite for achieving good quality cross-lingual matches. However, this is likely true only compared to accuracy of monolingual matching [17].

3 CroSeR: Cross-language Service Retriever

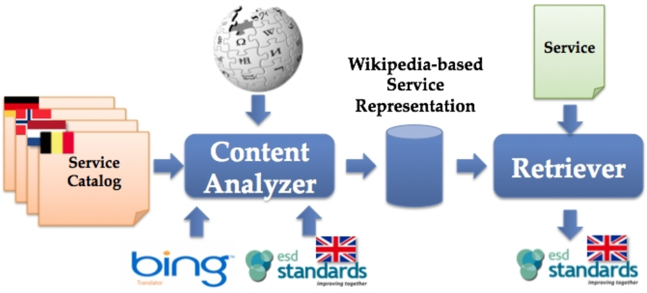

Fig. 2: CroSeR general architecture

We assume that each service is *labeled* by a short textual description (called *service label* in the following), e.g., see examples in Figure 1, and represents an abstract service, i.e., a high-level description of concrete services offered by one or more providers[8] [13]. The intuitive semantics of a link between a source

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

and a target service description is that the two descriptions refer to the same abstract service. Although service descriptions conceptually represent categories of concrete services, which are offered by actual providers, these descriptions are represented as ontology instances and, consistently with the ESD approach, we represent the links with *owl:sameAs* relations. In order to discover links between a *source list* of services described in an arbitrary language $\mathcal{L}$, and a *target list* of services described in English (LGSL), we design a matching algorithm for retrieving the top-k services that are best candidate to be linked to a given source service. We implemented the matching algorithm in a system called Cross-Language Service Retriever (CroSeR).

Given a source service described in an arbitrary language $\mathcal{L}$, and a *target list* of services described in English (LGSL), the CroSeR system returns a ranked list of k services in LGSL that are candidate to be *owl:sameAs*-related to the source service. The user will use the service list returned by CroSeR to validate the link between the source and the target service.

The architecture of CroSeR is depicted in Figure 2. The CroSeR system consists of two components: the *Content Analyzer* analyzes the service descriptions in the source and target lists and builds a semantic annotation for each service; the *Retriever* discovers the links between the source and target services by computing the similarity between the semantic annotations.

Content Analyzer. The input to the *Content Analyzer* are the service catalogs in different languages to be linked to the LGSL. The first step performed by this component is to translate the service labels in English by leveraging the Bing API[9]. Next, the translated labels are exploited for obtaining the *Wikipedia-based annotation* of the services. For each service s, a set of Wikipedia concepts W_s semantically related to the service label is generated; we call *Wikipedia-based annotation* of s the set W_s. The set W_s is built by processing the short text in the label of the service with semantic annotation techniques. The Wikipedia concepts are generated for all the services (English ones, as well), since we need to adopt an unified representation.

The Wikipedia-based annotation aim to capture the main topics (represented by the corresponding Wikipedia concepts) related to a service. In this context, a Wikipedia concept is defined as the title of a Wikipedia article. This solution is also able to perform a sort of word sense disambiguation of a natural language text without the application of elaborate algorithms based on lexical ontologies such as Wordnet[10]. Furthermore, the annotation of a service with a set of Wikipedia concepts represents an additional link between the service and the LOD cloud (by using DBpedia as input).

Retriever. The *Retriever* adopts the Vector Space Model (VSM) for representing services in a multidimensional space where each dimension is a Wikipedia concept. Therefore, each service is represented as a point in that space. In this first implementation we weighing each concept in W_s by adopting the simplest schema represented by the number of occurrences of the concept in W_s. For-

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

[10] http://wordnet.princeton.edu/

mally, each service is represented as a vector $\boldsymbol{s} =< w_1, \ldots, w_n >$ where w_k is the occurrence of the the Wikipedia concept k in W_s. We guess that more sophisticated weighing measures such as TF-IDF and BM25 [15] do not improve the performance of our system at this step.

Finally, the similarity between two services (vectors) is computed in terms of cosine similarity. Therefore, given a service in one of the supported languages (the query), the retriever is able to return a ranked list of the most similar English services from the LGSL.

Please, note that we adopt a variety of techniques (described in details in Section 4) for annotating the services, each of which represents a different CroSeR configuration. In addition to those Wikipedia-based configurations, we evaluated our system also by setting hybrid configurations obtained by merging the keywords extracted from the label associated to s with the corresponding Wikipedia concepts in W_s. In that case, all the above-mentioned definitions are still valid, with the slight difference that the vector space is built both on keywords and Wikipedia concepts (instead of Wikipedia concepts alone).

4 Semantic annotation of e-gov services

We exploited different 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, and we implemented a semantic feature generation tool based on the Explicit Semantic Analysis (ESA) technique. 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. Also ESA generates a set of Wikipedia concepts as output, but the insight behind it is quite different. 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.

Wikipedia Miner. Wikipedia Miner is a tool for automatically cross-referencing documents with Wikipedia [12]. The software is trained on Wikipedia articles, and thus learns to disambiguate and detect links in the same way as Wikipedia editors [3]. The first step is the disambiguation of terms within the text by means of a classifier. Several features are exploited for learning the classifier. The two main features are *commonness* and *relatedness* [10]. The *commonness* of a Wikipedia article is defined by the number of times it is used as destination from some anchor text. For example, the anchor text *tree* has a higher *commonness* value for the article *Tree (plant)* than the article *Tree (data structure)*. However, this feature is not sufficient to disambiguate a term. Therefore, the algorithm compares each possible sense (Wikipedia article) for a given term with its surrounding context by computing the *relatedness* value. This measure computes the similarity of two articles by comparing their incoming and outgoing links.

Tagme. Tagme is a system that performs an accurate and on-the-fly semantic annotation of short texts via Wikipedia as knowledge base [4]. The annotation process is composed of two main phases: the *anchor disambiguation* and the *anchor pruning*. The disambiguation is based on a process called "collective agreement". Given an anchor text a associated with a set of candidate Wikipedia pages P_a, each other anchor in the text casts a vote for each candidate annotation in P_a. As in Wikipedia Miner, the vote is based on the *commonness* and the *relatedness* between the candidate page $p_i \in P_a$ and the candidate pages associated to the other anchors in the text. Subsequently, an anchor pruning is performed by deleting the candidate pages in P_a considered to be not meaningful. This process takes into account the probability of the anchor text to be used as link in Wikipedia and the coherence between the candidate page and the candidate pages of other anchors in the text.

DBpedia Spotlight. DBpedia Spotlight [11] 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. The annotation process works in four-stages. First, the text is analyzed in order to select the phrases that may indicate a mention of a DBpedia resource. In this step any spots that are only composed of verbs, adjectives, adverbs and prepositions are disregarded. Subsequently, a set of candidate DBpedia resources is built by mapping the spotted phrase to resources that are candidate disambiguations for that phrase. As in the abovementioned tools, the disambiguation process uses the context around the spotted phrase to decide for the best choice amongst the candidates. Finally, there is a configuration step whose goal is to set the best parameter values for the text to be disambiguated.

Explicit Semantic Analysis. Explicit Semantic Analysis (ESA) is a technique proposed by Gabrilovich and Markovitch [7], that uses Wikipedia as a space of concepts explicitly defined and described by humans. The idea is that the meaning of a generic term (e.g. *ball*) can be described by a list of concepts it refers to (e.g. the Wikipedia articles: *volleyball, soccer, football,...*). 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. The semantic interpretation vector for a text fragment f (i.e. a sentence, a document, a service label) is obtained by computing the centroid of the semantic interpretation vectors associated with terms occurring in f.

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 sematic annotation of the service labels. Let's consider the English service label *Licences - entertainment* and the corresponding Dutch service in LGSL *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.*

5 Experimental Evaluation

In this experiment we evaluated the different configurations in terms of:
(1) effectiveness in retrieving the correct service in a list of k service to be presented to the user, (2) capability in boosting the correct service in the first positions of the ranked list.

Design and Dataset. We adopted two different metrics: *Accuracy@n (a@n)* and *Mean Reciprocal Rank (MRR)* [18]. 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 the number of the services correctly retrieved. The higher is the position of the services correctly retrieved in the list, the higher is the MRR value for a given configuration. We decided to adopt this normalization instead of considering N as the total number of the services in the catalogue in order to evaluate the ranking of each configuration independently from its coverage. Hence, a configuration with a good ranking, but a small coverage will obtain a higher MRR value than a configuration with a better coverage, but a worse ranking.

The dataset is extracted from the esd-toolkit catalogue freely available online[11]. We indexed English, Dutch, and German services. The number of Dutch services is 225, and the number German services is 190. For each service we extracted and represented its textual label in terms of Wikipedia concepts by exploiting the methods described in Section 4. The labels have an average length of about three words.

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

Results. The baseline of our experiment is the keyword-based configuration. For that configuration, only stemming and stopword elimination are performed on the text. We compared the baseline with the above-mentioned four different Wikipedia-based configurations (i.e., Wikipedia Miner, Tagme, DBpedia Spotlight, ESA) as well as with a combination of keywords and Wikipedia configurations. The latter configurations were obtained by adding to the keywords the corresponding Wikipedia concepts generated by the different methods.

Results for the Dutch language are reported in Table 1. We can observe that the best configuration in terms of $a@n$ is ESA. The configuration becomes more effective by returning several services ($n > 5$), since the matching is particularly difficult in this domain. The worst configuration is that based on Wikipedia Miner. This is likely due to a low effectiveness of Wikipedia Miner in identifying topics from very short text. Most configurations seem to improve their accuracy by merging the Wikipedia concepts with keywords; however they do not generally overcome the baseline. The only tool for which the keywords do not generally improve the performance is ESA. However, by analyzing the MRR values we can observe that ESA produces the worst ranking of the retrieved list of services. Conversely, the method with the best ranking is Wikipedia Miner, but it has a very small coverage (only 24 services). Hence, we can state that ESA is able to identify the correct correspondence for the largest number of services ($\sim$ 82% of services) but under the condition to extend the list of retrieved service. Very similar results are shown for the German services (see Table 2). Also in this experiment ESA is the configuration with the best accuracy, while Wikipedia Miner achieves the best ranking. These results are very promising since the service labels in LGSL are written and matched by human experts and they are not always mere translations of the English labels.

Table 1: Accuracy@n and MRR for the Dutch language.
The highest values are reported in bold **(total services = 225).**

Configuration	**a@1**	**a@3**	**a@5**	**a@10**	**a@20**	**a@30**	**MRR**	**N**
keyword	**0.333**	0.458	0.502	0.538	0.542	0.547	0.610	123
tagme	0.120	0.164	0.178	0.182	0.187	0.187	0.643	42
tagme+keyword	0.316	0.453	0.484	0.551	0.560	0.569	0.555	128
wikiminer	0.080	0.093	0.107	0.107	0.107	0.107	**0.750**	24
wikiminer+keyword	0.324	0.440	0.484	0.529	0.542	0.547	0.593	123
esa	0.311	**0.480**	0.538	**0.622**	**0.689**	**0.716**	0.378	**185**
esa+keyword	0.311	0.476	**0.542**	**0.622**	**0.689**	**0.716**	0.378	**185**
dbpedia	0.182	0.236	0.244	0.249	0.258	0.258	0.707	58
dbpedia+keyword	0.329	0.449	0.498	0.556	0.569	0.573	0.574	129

Table 2: Accuracy@n and MRR for the German language.
The highest values are reported in bold **(total services = 190).**

Configuration	**a@1**	**a@3**	**a@5**	**a@10**	**a@20**	**a@30**	**MRR**	**N**
keyword	0.204	0.338	0.396	0.413	0.418	0.418	0.489	94
tagme	0.124	0.147	0.151	0.151	0.151	0.151	0.824	34
tagme+keyword	0.218	0.342	0.400	0.427	0.431	0.431	0.505	97
wikiminer	0.098	0.124	0.124	0.129	0.129	0.129	**0.759**	29
wikiminer+keyword	0.218	0.342	0.396	0.422	0.427	0.427	0.510	96
esa	**0.244**	**0.360**	**0.431**	**0.484**	**0.556**	**0.600**	0.350	**157**
keyword+esa	**0.244**	**0.360**	**0.431**	**0.484**	**0.556**	**0.600**	0.350	**157**
dbpedia	0.138	0.169	0.182	0.182	0.182	0.182	0.756	41
dbpedia+keyword	0.231	0.360	0.413	0.440	0.440	0.440	0.525	99

6 Conclusions and Future Work

In this paper we proposed a tool called CroSeR that is able to perform a cross-language matching of e-gov services. Four different semantic Wikipedia-based representations were investigated. The most accurate representation turned out to be ESA that, given a Dutch or German service, is able to retrieve the corresponding English service for most of services. Hence, adding new external knowledge for representing a very short textual description is an effective solution in this specific domain. However, the correct service is generally not boosted in the first positions of the ranked list. Therefore, as a future work we want to combine different representations to generate the ranked list. For example, we can start by adopting the representation with the highest MRR value, and then shifting to the representations with a worse ranking but a better accuracy. We want also to extend the experiment to the other languages in the esd-toolkit catalogue (Belgian, Norwegian, Swedish). Another idea is to exploit also the other relations stored into the catalogue (life event, interactions) for improving the service matching. Finally, we will carry out a user study where users can directly formulate their information need instead of using the service label as query.

7 Acknowledgments

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).

References

1. European Commission. A digital agenda for europe. *COM(2010) 245 final/2*, 2010.
2. L. Ding, V. Peristeras, and M. Hausenblas. Linked Open Government Data. *IEEE Intelligent Systems*, 27(3):11–15, 2012.

3. Samuel Fernando, Mark Hall, Eneko Agirre, Aitor Soroa, Paul Clough, and Mark Stevenson. Comparing taxonomies for organising collections of documents. In *Proceedings of COLING '12*, pages 879–894. The COLING 2012 Organizing Committee, 2012.
4. 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.
5. B. Fu, R. Brennan, and D. O'Sullivan. Cross-lingual ontology mapping — an investigation of the impact of machine translation. In *Proceedings of ASWC '09*, pages 1–15. Springer-Verlag, 2009.
6. 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.
7. E. Gabrilovich and S. Markovitch. Wikipedia-based semantic interpretation for natural language processing. *Journal of Artificial Intelligence Research (JAIR)*, 34:443–498, 2009.
8. 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.
9. 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.
10. O. Medelyan, I.H. Witten, and D. Milne. Topic indexing with Wikipedia. In *Proceedings of AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy*, pages 19–24. AAAI Press, 2008.
11. 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.
12. D. Milne and I. H. Witten. Learning to link with Wikipedia. In *Proceedings of CIKM '08*, pages 509–518. ACM, 2008.
13. 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.
14. H. Paulheim. WeSeE-Match results for OEAI 2012. In *Proceedings of the 7th International Workshop on Ontology Matching (OM 2012)*, 2012.
15. Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, April 2009.
16. P. Shvaiko and J. Euzenat. Ontology matching: State of the art and future challenges. *IEEE Trans. Knowl. Data Eng.*, 25(1):158–176, 2013.
17. 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.
18. E. M. Voorhees. TREC-8 question answering track report. In *Proceedings of TREC-8*, pages 77–82. NIST Special Publication 500-246, 1999.
19. 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.