Executing, Comparing, and Reusing Linked Data-Based Recommendation Algorithms With the Allied Framework

Cristhian Figueroa, Iacopo Vagliano, Oscar Rodríguez Rocha, Marco Torchiano, Catherine Faron Zucker, Juan Carlos Corrales, Maurizio Morisio

Executing, Comparing and Reusing Linked Data-Based Recommendation Algorithms with the AlLied Framework

Cristhian Figueroa
Universidad Antonio Nariño, Colombia

Iacopo Vagliano
ZBW – Leibniz Information Centre for Economics, Germany

Oscar Rodríguez Rocha
University Côte d'Azur, CNRS, Inria, I3S, France

Marco Torchiano
Politecnico di Torino, Italy

Catherine Faron-Zucker
University Côte d'Azur, CNRS, Inria, I3S, France

Juan Carlos Corrales
Universidad del Cauca, Colombia

Maurizio Morisio
Politecnico di Torino, Italy

ABSTRACT

Data published on the Web following the Linked Data principles has resulted in a global data space called the Web of Data. These principles led to semantically interlink and connect different resources at data level regardless their structure, authoring, location, etc. The tremendous and continuous growth of the Web of Data also implies that now it is more likely to find resources that describe real-life concepts. However, discovering and recommending relevant related resources is still an open research area. This chapter studies recommender systems that use Linked Data as a source containing a significant amount of available resources and their relationships useful to produce recommendations. Furthermore, it also presents a framework to deploy and execute state-of-the-art algorithms for Linked Data that have been re-implemented to measure and benchmark them in different application domains and without being bound to a unique dataset.

Keywords: Semantic Recommendation, Web of data, Evaluation Framework, DBpedia, Interlinked Data, Linked Data datasets, Recommender System Structure

INTRODUCTION

The Web of Data has emerged as a way to make the Web machine-readable, relying on structured data that follow the Linked Data principles (Moyano, Sicilia, & Barriocanal, 2018). Thanks to the rise of the Web of Data, users are more likely to find resources that describe or represent real-life concepts.

However, due to the increase in the amount of structured data published on the Web, discovering and recommending related resources is still an open research area (Ricci, Rokach & Shapira, 2011). This problem can be addressed by analyzing the categories of resources, their explicit references to other resources and by combining both approaches (Figueroa, Vagliano, Rodríguez Rocha & Morisio, 2015). Accordingly, many works are addressing this problem, typically focusing on specific application domains and datasets. In contrast, we seek a solution which can fit more than one domain and dataset and we intend to generalize existing approaches. In this context, the research described in this chapter aims to answer the following research questions:

- How can we choose state-of-the-art algorithms for discovering and recommending resources from the web based on the characteristics of a given application domain and a given dataset?
- How can we benchmark the existing algorithms to select the one that best suits specific discovering and recommendation needs?
- How can we develop an algorithm that is dynamically adaptable to the characteristics of the dataset and independent of the application domain?

This chapter presents a framework named *AlLied* for deploying and executing recommendation algorithms based on Linked Data. This framework allows developers and researchers to test different configurations of these algorithms in a range of application domains and datasets. Additionally, *AlLied* provides a set of APIs to be exploited as the primary component for Recommender Systems (RS)' architectures: developers do not need to deal with the execution platform of the algorithms but only focus their efforts either on selecting the algorithm that best fits their needs or on writing a customized one.

After conducting an in-depth analysis of the state-of-the-art recommendation algorithms executed in *AlLied*, the authors proposed a generic resource discovery and recommendation algorithm named ReDyAl which dynamically adapts itself to the characteristics of the dataset and the application domain.

This chapter additionally provides an overview of the research problem in discovering and recommending resources as well as the various existing types of RS. It gives a detailed review of the Linked Data based RS and summarizes the results of an evaluation of ReDyAl deployed in the *AlLied* framework. Finally, this chapter shows how it is possible to choose the more appropriate state-of-the-art resource recommendation algorithms for a given application domain and dataset by measuring its performance and accuracy.

BACKGROUND

Recommendation problem

The recommendation problem is finding an item which maximizes the utility of the item for a given user. A formal definition is described in Equation 1 (Adomavicius & Tuzhilin, 2005).

$$\forall u \in U \; i^{max,u} = arg\, max_{i \in I}\, f(i,u) \qquad (1)$$

U is the set of users considered by the recommender system and I the set of items; they can be both extremely large. The utility function f: $U \times I \rightarrow R$ represents the usefulness of an item $i \in I$ for a given user $u \in U$, where R is a totally ordered set (e.g. nonnegative numbers within a given range).

The utility of an item is often represented by a rating, which indicates how a particular user liked a given item (Di Noia & Ostuni, 2015). For instance, a user gave the movie *The Green Mile* the rating 4 out of 5. The utility is not defined on the whole $U \times I$ space, but only a subset is available. In fact, only a portion of ratings is known for each user. Thus, a recommender system has to assess the utility function from the available data and use it to predict unknown values. Typically, the recommendations are provided by selecting for each user the best N items, i.e. the items with the highest utility (top-N recommendations).

Recommendation techniques

RS are software tools and techniques to suggest items or objects (films, music, news, people, messages, etc.) (Ricci, Rokach, & Shapira, 2011). The most popular classes of RS are content-based, collaborative filtering, knowledge-based, and hybrid. Content-based algorithms provide recommendations to a user based on their previous preferences and the content of items (e.g., keywords, size, pixels, genre, etc.) (Lops, Gemmis, & Semeraro, 2011). Collaborative-filtering (CF) approaches consider the ratings that users with similar preferences gave to the same items (Felfernig, Jeran, Ninaus, Reinfrank, & Reiterer, 2013). Knowledge-based infer similarities between user requirements and items' features described in a knowledge base (Dell'Aglio, Celino, & Cerizza, 2010). Hybrid techniques combine one or more approaches to compensate weaknesses of single methods. For example, CF methods suffer from the issue known as "new user," which is the difficulty of generating recommendations because of the lack or insufficiency of ratings that a new user may have issued. This problem, however, is not a limitation for content-based methods since the prediction of the new items is not focused on user ratings but the description of the features of these items (Ricci et al., 2011).

Knowledge-based RS have some advantages over other types (Dell'Aglio et al., 2010). First, these algorithms require few information about user profiles. Second, they are less subject to the "cold start" problem (new users or items do not contain enough ratings) (Tomeo, Fernández-Tobías, Cantador, & Di Noia, 2017). Third, they can explain recommendations. The main problems of knowledge-based RS are the significant computational complexity due to the processing of large amounts of data, and the high effort required to construct and maintain the knowledge base. Furthermore, the knowledge base depends on the application domain and may require frequent updates.

A new kind of knowledge-based RS known as "linked data-based RS" (Figueroa, Vagliano, Rodríguez Rocha, & Morisio, 2015), or semantics-aware RS (de Gemmis, Lops , Musto, Narducci, Semeraro, 2015; Di Noia & Ostuni, 2015), has emerged. These RS suggest items relying on datasets published on the Web of Data (Damljanovic, Stankovic, & Laublet, 2012). Unlike traditional knowledge-based RS, linked data-based RS use datasets built, modeled, and maintained by different organizations and communities around the world. These datasets may contain knowledge from different domains and sources in the Web of Data. Linked data-based RS still struggle to generate recommendations with an acceptable accuracy for end users. The causes include the need for information from both user profiles and descriptions of the items; the necessity of knowledge bases to be frequently updated and maintained; high computational complexity; and the required manual extraction of a subset of the knowledge bases representing a portion specialized on a domain of interest.

More research on how to apply the different techniques of RS and the web of data in real-world situations is required (Musto, Lops, de Gemmis, & Semeraro, 2017; Park, Kim, Choi, & Kim, 2012). The main objective of the work addressed in this chapter is to develop a linked data-based RS to recommend resources dynamically analyzing their relationships and considering the knowledge of the Web of Data.

RELATED WORK

This work classifies the related studies into four main types based on their recommendation algorithms: graph-based, machine learning, memory-based, and probabilistic. Though, there are also other methods that are less used.

Graph-based methods

The Web of Data can be analyzed from the perspective of its graph structure (Moyano et al., 2018). Graph-based algorithms exploit this structure for computing relevance scores for items represented as nodes in a graph. Algorithms in this category are classified into semantic exploration and path-based.

- *Semantic Exploration* techniques explore the graph structure of datasets using structural relationships to compute distances and generate recommendations. For example, *HyProximity*

(Damljanovic et al., 2012) exploits hierarchical links among Wikipedia categories in DBpedia, while dbRec (Kitaya, Huang, & Kawagoe, 2012; Passant, 2010a; Yang et al., 2013) is a music recommender system, which mainly relies on the number of direct and indirect links between two resources. ReDyAl (Vagliano et al., 2016) combines this two approaches and generalize them to be applied to any dataset and domain. Other methods are based on page rank (Blanco, Cambazoglu, Mika, & Torzec, 2013; Narducci, Musto, Semeraro, Lops, & de Gemmis, 2013; Nguyen et al., 2015), semantic clustering (Ko, Kim, Ko, & Chang, 2014), feature selection (Musto, Basile, Lops, de Gemmis, & Semeraro, 2017), and the Vector Space Model (VSM) (Khrouf & Troncy, 2013; Narducci et al., 2013; V. C. C. Ostuni et al., 2013).

- *Path-based* algorithms use information about semantic paths within a graph structure to compute similarities useful to produce recommendations. For example, spreading activation (Cheekula, Kapanipathi, Doran, & Jain, 2015; Chicaiza, Piedra, López-Vargas, & Tovar-Edmundo, 2014; Hajra, Latif, & Tochtermann, 2014; Marie, Gandon, Legrand, & Ribière, 2013), random walk (Cantador, Konstas, & Jose, 2011); path-weights for vertex discovery (Strobin & Niewiadomski, 2014). Modern methods combine machine learning to learn the best path to consider relying on learning to rank (Di Noia, Ostuni, Tomeo, & Di Sciascio, 2016) or deep learning (Palumbo, Rizzo, & Troncy, 2017; Rosati, Ristoski, Di Noia, De Leone, & Paulheim, 2016) techniques.

Machine learning

These algorithms use techniques to analyze, predict and classify data extracted from datasets. This allows them to learn from data to produce recommendations. Algorithms in this type are classified in Supervised learning and unsupervised learning (Portugal, Alencar, & Cowan, 2018):

- *Supervised learning.* A model is prepared through a training process and correct answers to produces predictions (Welling, 2011). These algorithms predict class labels from attributes. For example, kNN (Ahn & Amatriain, 2010; Ristoski, Loza Mencía, Paulheim, & Menc, 2014), decision trees (Khrouf & Troncy, 2013; V. C. Ostuni, Di Noia, Di Sciascio, & Mirizzi, 2013; Ristoski et al., 2014), logistic regression (Moreno et al., 2014; Narducci et al., 2013; V. C. Ostuni et al., 2013; Zhang, Wu, Sorathia, & Prasanna, 2014), Support Vector Machines (SVM) (Di Noia, Mirizzi, Ostuni, & Romito, 2012; Kushwaha & Vyas, 2014; V. V. Ostuni, Di Noia, Mirizzi, Di Sciascio, & Noia, 2014), random forest (Narducci et al., 2013; V. C. Ostuni et al., 2013), naive Bayes (Schmachtenberg, Strufe, & Paulheim, 2014) and bayesian classifiers (Lopes, Leme, Nunes, & Casanova, 2014).
- *Unsupervised learning.* Unlike in the supervised learning, input data is not labelled and does not have a known result, so the aim of these algorithms is to try to discover the structure or distribution of the data (Welling, 2011). For example, K-Means (Manoj Kumar, Anusha, & Santhi Sree, 2015; Moreno et al., 2014), fuzzy-C means, self-organizing map (SOM); principal component analysis (PCA) (V. V. Ostuni et al., 2014).

Memory-based

Memory-based algorithms recommends items based on the entire collection of previously rated path queries. For example, rating prediction (Kushwaha & Vyas, 2014; Moreno et al., 2014; Narducci et al., 2013); singular value decomposition (SVD) (Ko & Son, 2015; Moreno et al., 2014; Rowe, 2014); implicit feedback; and matrix factorization (Lommatzsch, Kille, Albayrak, & Berlin, 2013).

Probabilistic

These algorithms exploit probabilistic techniques applied to linked data such as Latent Dirichlet Allocation (LDA) (Hopfgartner & Jose, 2010; Khrouf & Troncy, 2013; Zhang et al., 2014), Random Indexing (RI) (Damljanovic et al., 2012), Bayesian ranking (Lopes, André, et al., 2014), and beta probability distribution (Maccatrozzo, Ceolin, Aroyo, & Groth, 2014).

Others

Other types of algorithms exist. For example, evolutionary computation, automated planning, semantic reasoning, social network analysis (SNA), and text mining on user review.

- *Evolutionary computation.* This class include stochastic methods inspired from natural evolution such as genetic algorithm (Khrouf & Troncy, 2013; Lommatzsch, Kille, Albayrak, et al., 2013), biological classification (Chawuthai, Takeda, & Hosoya, 2015) and particle swarm optimization (Juang, Tung, & Chiu, 2011).
- *Automated planning.* These methods use artificial intelligence to create strategies that are executed by intelligent agents such as (Gordea, Lindley, & Graf, 2011).
- *Semantic reasoning.* These techniques are based on rules to infer logical consequences from a set of asserted facts or axioms (Calì, Capuzzi, Dimartino, & Frosini, 2013; Cantador, Castells, & Bellogín, 2011; Ozdikis, Orhan, & Danismaz, 2011). Some approaches combine reasoning with context-awareness (Karpus, Vagliano, & Goczyła, 2017; Karpus, Vagliano, Goczyła, & Morisio, 2016).
- *Social network analysis.* These algorithms exploit relationships found in social networks related to items and users (Lopes, André, et al., 2014; Lopes, Leme, et al., 2014).
- *Semantic annotation of user review.* New approaches are emerging, which combine semantic annotation of user reviews with additional information from the Linked Data cloud (Vagliano, Monti, Morisio, & Scherp).

Linked data-based RS still have some issues. Graph-based algorithms for RS suffer from high computational complexity for exploiting semantic features due to the large data and inconsistency of datasets (Damljanovic et al., 2012; Passant, 2010a). Machine learning algorithms are time-consuming for the training phase, and some of them only use linked data just for representation, so they do not take into account the intrinsic semantic structure of datasets (Arturo et al., 2014; V. C. Ostuni et al., 2013; Ristoski et al., 2014). Other algorithms require user's profile information to produce recommendations, they suffer from the cold-start problem (Kushwaha & Vyas, 2014; Lommatzsch, Kille, & Albayrak, 2013; Moreno et al., 2014). Finally, existing hybrid recommendation techniques are not organized in a conceptual architecture based on their functionalities. This conceptual architecture would be useful to execute and test various configurations of algorithms for creating novel RS (Lommatzsch, Kille, & Albayrak, 2013; Lommatzsch, Kille, Kim, & Albayrak, 2013; Moreno et al., 2014).

ALLIED: A FRAMEWORK FOR EXECUTING RESOURCE RECOMMENDATION ALGORITHMS BASED ON LINKED DATA

AlLied is a framework to select, evaluate, and create algorithms to recommend resources from Linked Data belonging to different application domains. This framework integrates the algorithms to execute specific tasks in the process of recommendation (i.e., resource generation, ranking, categorization, and presentation). Accordingly, the framework is suitable to compare the results of different configurations of these algorithms and to enable the development of innovative applications on top of it.

The work described in this chapter focuses on algorithms that rely only on linked data, but it is not limited to them. In fact, developers can extend it to consider other approaches in combination with Linked Data. The recommendation process of *AlLied* is shown in Figure 1.

Figure 1. Steps of the recommendation process. Source: (Figueroa et al., 2017)

Resource generation. The first step is intended to generate a set of candidate resources (CR) that maintain semantic relationships with an initial resource (*ir*). The semantic relations may be direct or indirect links between two resources in a dataset.

Results ranking. This step sorts the candidate resources generated in the previous one by considering the semantic similarity with the initial resource. Different semantic similarity measures can be used to calculate the semantic similarity between pairs of resources.

Grouping. The list of ranked candidate resources generated in the previous step may be too general, that is, a recommendation may include resources from unrelated domains of knowledge. For this reason, this optional third step groups these resources already ranked into meaningful clusters that represent common knowledge domains.

Results presentation. Finally, the results of the last step are graphically presented through different facets to allow the end-users to visualize the recommendations.

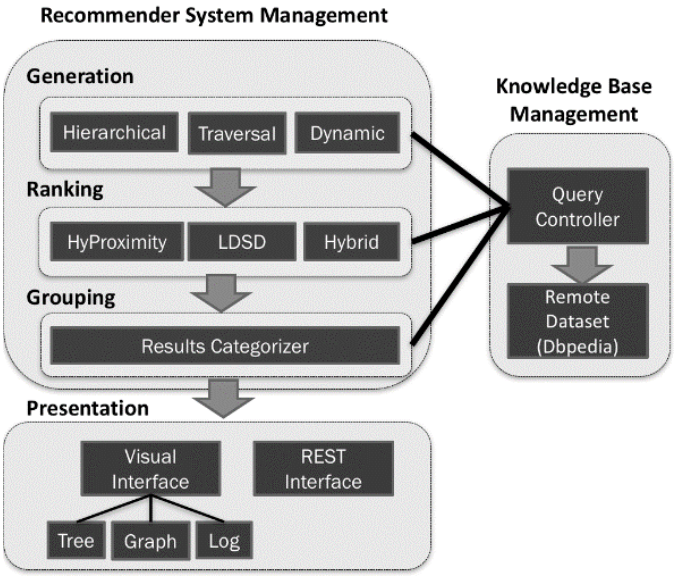
Implementation

The framework *AlLied* is composed of three subsystems (
Figure *2*):

- *KB Management* is related to the Knowledge Base Layer as it provides the interfaces needed to access to local or remote Linked Data datasets. The components of this subsystem include 1) a query controller to execute queries on the local/remote datasets; 2) a category tree which is a hierarchical structure that allows the algorithms to perform hierarchical-dependent operations; and 3) local and remote datasets which are the source of the structured data.
- *RS Management* not only provides mechanisms for retrieving, searching, discovering and ranking resources, but also performs management tasks such as creating new connections to other remote/local datasets. It contains the main components of the RS as it is the central subsystem. It also controls the execution process of the RS. This subsystem includes the RS Executor to control the execution of recommendation algorithms, the Dataset manager to access the functionalities provided by the KB Management, the RS Controller to manage the recommendation algorithms as well as the datasets of the KB Management subsystem.
- *RS Presentation:* this subsystem provides the user interface.

Each subsystem contains functional components. These functional components are located in a set of layers for each subsystem conforming to their functionality. For example, the layers generation, ranking, and grouping are responsible for the recommender system management activities, and the knowledge base core layer oversees accessing the datasets and the presentation layer for showing the results to the users.

Figure 2: Diagram of the implementation of the AlLied framework. Source (Figueroa, 2017)



Knowledge base management

This subsystem represents the data layer of the *AlLied* framework. It is the primary data source containing knowledge about resources and their structural relationships. The knowledge base may be a tuple (R, T, L) composed of resources (R), categories (T), and links (L):

- **Resources** are representation of real-world ideas and objects such as persons, cities, movies, or topics.
- **Categories** are classes which hierarchically group resources. For instance, DBpedia provides information about the hierarchical relationships in three different classification schemata: Wikipedia Categories, YAGO Categories[1], and WordNet Synset Links[2]. This implementation uses the Wikipedia categories to describe the hierarchical information of resources and their relationships. Wikipedia categories are concepts of the Simple Knowledge Organization System (*SKOS*[3]) vocabulary.
- **Relationships** are the links (also known as properties) connecting resources or categories along the whole dataset. Any knowledge base may contain three types of relationships.
    - *Resource-Resource:* these are the traversal relationships between resources, which are those links between resources that do not refer to hierarchical classifications. Most of the links of DBpedia belong to this type.
    - *Resource-Category:* these are relationships between a resource and a category. They can be represented by using the SKOS properties `skos:subject` (hasCategory) and `skos:isSubjectOf` (IsCategoryOf). However, `skos:subject` and `skos:isSubjectOf` are deprecated and consequently not used in DBpedia. Therefore, DBpedia relates resources to their Wikipedia categories using `dcterms:subject` instead. Accordingly, `dcterms:subject` is used in *AlLied* for both relationships.
    - *Category-Category*: these are hierarchical relationships between categories within a hyponymy structure (a category tree). They can be represented by using the SKOS properties `skos:broader` (isSubCategoryOf) and `skos:narrower` (isSuperCategoryOf).

By default, AlLied uses the DBpedia dataset as the knowledge base, but developers can easily extend it to other datasets. DBpedia is one of the most significant datasets, frequently fed with data from Wikipedia, and one of the most interlinked datasets in the Web of Data (Schmachtenberg, Bizer, & Paulheim, 2014). The Wikipedia categories (SKOS concepts) were selected because they are the most linked in DBpedia.

Generation layer

This layer aims at discovering resources related to a given one through semantic relationships. Given an initial resource (or a set of initial ones) it generates a set of candidate resources located at a predefined distance. For this layer, three generators were implemented based on the semantic relationships found in the Linked Data through SPARQL queries (Ayala, Przyjaciel-Zablocki, Hornung, Schätzle, & Lausen, 2014).

Traversal generator

The Traversal generator looks for resources that are directly related to a given initial resource and those found through a third resource (indirect relationships). Its implementation is inspired in the *dbrec* recommender (Passant, 2010b).

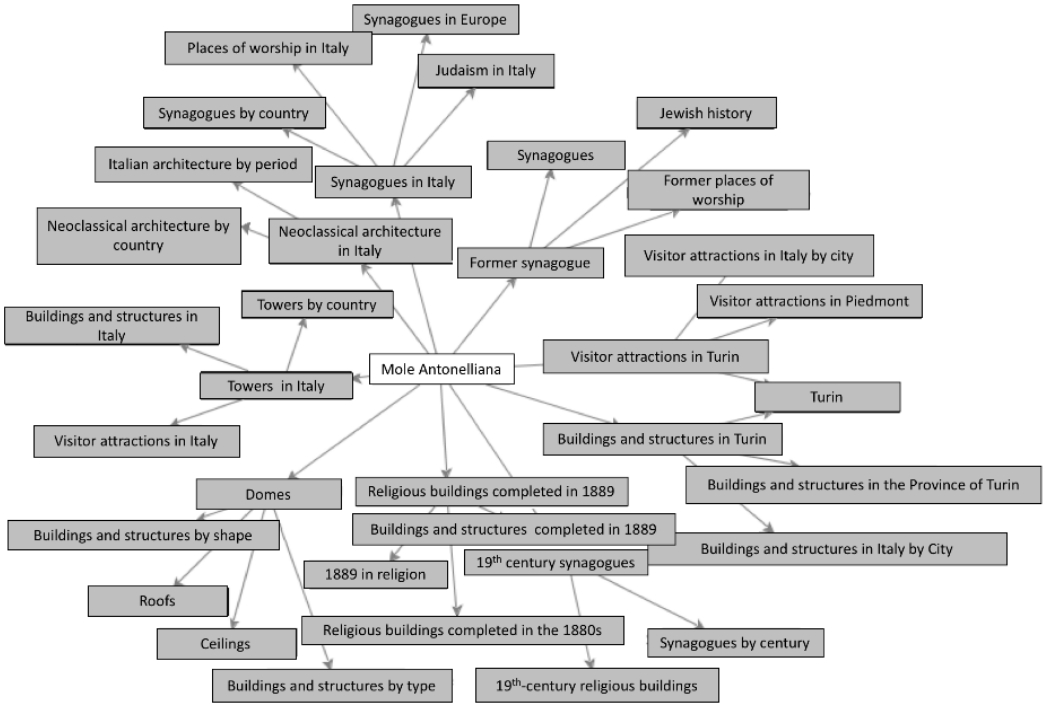
Hierarchical generator

The hierarchical generator generates a set of candidate resources located at a specified distance in a hierarchy of categories taken from a category tree described in a dataset. The implementation of this module is inspired by (Damljanovic et al., 2012), which obtains candidate resources navigating a tree of Wikipedia categories.

The hierarchical generator firstly extracts categories of an initial resource (`<inURI>`) and then looks for broader categories until a maximum distance (which may be user-defined) is reached. This maximum distance is the hierarchical distance of a broader category from its base categories. It is inversely proportional to the specificity of a category (the higher the distance the lower the specificity of a category).

After extracting categories, this module retrieves subcategories for all the broader categories at maximum distance (it descends one level into the category tree) to increase the possibility for finding more candidate resources. Then, the algorithm obtains candidate resources for each category (including sub-categories). Therefore, the module creates a "category graph", including the initial resource, its category tree, and the candidate resources retrieved for each category. For example, *Figure 3* shows an example of the category graph for the resource <`http://dbpedia.org/resource/Mole_Antonelliana`>.

Figure 3: Example of a category graph for the resource "Mole Antonelliana" (candidate resources are not included for space reasons). Source (Figueroa et al., 2017)



Dynamic generator

It is a "hybrid" generator, which takes advantage of both the traversal and the hierarchical approaches, giving priority to the existing interlinking between resources, that is, one of the four principles of Linked Data (Bizer, Heath, & Berners-Lee, 2009). The innovative algorithm of this generator is explained in section *A Dynamic Algorithm for Recommendation.*

Ranking layer

This layer uses semantic similarity functions to rank the candidate resources obtained in the previous layer. This layer sorts the candidate resources according to the values of a semantic similarity function, which measures the similarity between the initial resource and each candidate resource. The framework in its current implementation includes (but is not limited to) three ranking algorithms.

Traversal LDSD ranking

The traversal LDSD ranking algorithm calculates the Linked Data Semantic Distance *(LDSD)* This measure that was initially proposed by (Passant, 2010a), is based on the number of indirect and direct links between two resources. The similarity of two resources (r_1, r_2) is measured in Equation (1) as the basic form of the *LDSD* distance.

$$LDSD(r_1, r_2) = \frac{1}{1+Cd_{out}+Cd_{in}+Ci_{out}+Ci_{in}} \quad (1)$$

In the LDSD equation, Cd_{out} is the number of direct input links (from r_1 to r_2), Cd_{in} is the number of direct output links, Ci_{in} is the number of indirect input links, and Ci_{out} is the number of indirect output links.

Unlike the implementation developed by Passant, which is limited to links from a specific domain, the LDSD function implemented in *AlLied* considers all resources from the dataset. However, it can be customized to defined types of links belonging or not to a specific domain by adding a set of *forbidden links*. Two SPARQL queries count direct and indirect input and output links between an initial resource and a resource of the set of candidate resources.

HyProximity ranking

The *HyProximity* ranking algorithm is based on the similarity measure defined by (Stankovic, Breitfuss, & Laublet, 2011). This measure can be used to calculate both traversal and hierarchical similarities. The *HyProximity* in its general form is shown in Equation (2) as the inverted distance between two resources, balanced with a pondering function.

$$hyP(r_1, r_2) = \frac{p(r_1,r_2)}{d(r_1,r_2)} \quad (2)$$

In this equation, d is the distance function between two resources and p is a weighting function used to give a level of importance to different distances. Based on the structural relationships (hierarchical and traversal), different distances and weighting functions may be used to calculate the *HyProximity* similarity.

- **Hierarchical HyProximity**: The definition of this similarity function relies on the work of (Stankovic et al., 2011). It was calculated using the maximum distance of categories of the hierarchical generator algorithm such that: $d(ir, cr_i) = maxLevel$. Here ir is the initial resource and cr_i is each one of the candidate resources generated in the hierarchical algorithm. The weighting function is defined in Equation (3), which is an adaptation of the informational content function (Seco, Veale, & Hayes, 2004). In this equation, $hypo(C)$ is the number of descendants of category C and $|C|$ is the total number of categories in the category Graph of C.

$$p(C) = 1 - \frac{\log(hypo(C)+1)}{log(|C|)} \quad (3)$$

This function was selected because it minimizes the complexity of calculation of the informational content with regard to other functions that employ an external corpus (Hadj Taieb, Ben Aouicha, Tmar, & Hamadou, 2011).

- **Traversal HyProximity:** in this similarity function $d(ir, cr_i) = maxLevel$ if the generator of resources is hierarchical, otherwise $d(ir, cr_i) = 1$ for resources connected to the initial resource through direct traversal links or $d(ir, cr_i) = 2$ for indirect traversal links. The weighting function is defined in Equation (4): $p_{trav}(r_1, r_2)$ depends on the number of resources n connected over a specific property and the total number of resources of the dataset, M:

$$p_{trav}(r_1, r_2) = -\log\frac{n}{M} \quad (4)$$

Nonetheless, in *AlLied*, this algorithm is not limited to a specific property, and optionally can be configured to support a set of forbidden links or allowed links in a similar way as shown in the *Generation Layer*. The number of direct and indirect links was calculated with SPARQL queries. The value of M was fixed to the number of resources contained in DBpedia.

Grouping layer

Since this implementation of the framework relies on DBpedia, which is a general-purpose dataset, the results obtained may contain an inherent ambiguity due to the generality of the data used to produce recommendations. Moreover, a single ranked list of recommendations may not always be an excellent way to show this kind of general results because users may require results arranged according to their personal needs or knowledge domain. The grouping layer addressed this requirement, because it provides mechanisms to group the results obtained from the ranking layer into meaningful clusters that represent domains of knowledge.

Currently, the grouping layer relies on Algorithm 1. This algorithm provides a mechanism to efficiently clusters the recommended items. Although in the current implementation of *AlLied* the resulting clusters correspond to Wikipedia categories, custom clusters can be easily defined by aggregating many categories or relying on other category schemas, such as YAGO categories.

Algorithm 1. Hierarchical classification algorithm: Source (Figueroa et al., 2017)

```
Require: CR, inURI, maxLevel, optionally Gc_in
Ensure: A Category graph G_C
1:    if Gc_in = null then
2:        Gc = createCategoryGraph(CR, maxLevel)
3:    else
4:        Gc = Gc_in
5:    end if
6:                        C_maxLevel = getMaxLevelCategories(Gc)
7:    for each pair of categories (c_i, c_j) ∈ C_maxLevel do
8:        c_lcb = getLesCommonBroaderCategory(c_i, c_j)
9:        Add c_lcb to Gc
10:       Add edge (c_i, c_lcb) and edge (c_j, c_lcb) to Gc
11:   end for
```

```
12:   intersectCategories(Gc)
13:                              deleteEmptyCategories(Gc)
13:   return Gc
```

Algorithm 1 receives as input a set of ranked candidate resources (CR), an initial resource $inURI$, and optionally an initial category graph (Gc_{in}) if the latter is already available. If Gc_{in} is not given, then the algorithm creates a new category graph Gc containing categories for the initial resource and the set of candidate resources until a maximum distance ($maxLevel$) (Lines 1 - 5). In this implementation $maxLevel$ is set to 2 because with this value it is possible to obtain a reasonable relationship between the number of categories and the time consumed.
Afterward, the algorithm extracts categories at the highest distance ($C_{maxLevel}$) and creates pairs of categories combining the elements of $C_{maxLevel}$ (Lines 6 - 7). Next, the function *getLessCommonBroaderCategory*, which is based on the less common ancestor, is executed to find a set of broader categories subsuming the categories of $C_{maxLevel}$. These new broader categories are then added to Gc including their edges, i.e. (c_i, c_{lcb}) and (c_j, c_{lcb}) (Lines 8 - 11).

Presentation layer

Developers can easily integrate *AlLied* to any application that requires recommendations based on linked data. The current implementation includes three main interfaces that provide mechanisms to present results to the final user: a Web interface, a standalone interface, and a RESTful interface. These are described elsewhere (Vagliano et al., 2016; Figueroa, 2017).

A DYNAMIC ALGORITHM FOR RECOMMENDATION

ReDyAl (Vagliano et al., 2016) is an algorithm that was developed considering the different types of relationships between data published according to the Linked Data principles. It aims to discover related resources from datasets that may contain either "well-linked" resources as well as "poor-linked" resources. A resource is said to be "well-linked" if it has many links higher than the average number of links in the dataset; otherwise, it is "poor-linked." The algorithm can dynamically adapt its behavior to find a set of candidate resources to be recommended, giving priority to the implicit knowledge contained in the Linked Data relationships.

The execution of *ReDyAl* contains three stages: 1) discovering related resources by analyzing the interlinking between them; 2) Examining the categorization of the given initial resource and discovering similar resources located in common categories; and 3) intersecting the results of the previous stages, given priority to those found in the first stage.

Figure 4: Flowchart of ReDyAl. Source (Figueroa et al., 2017)

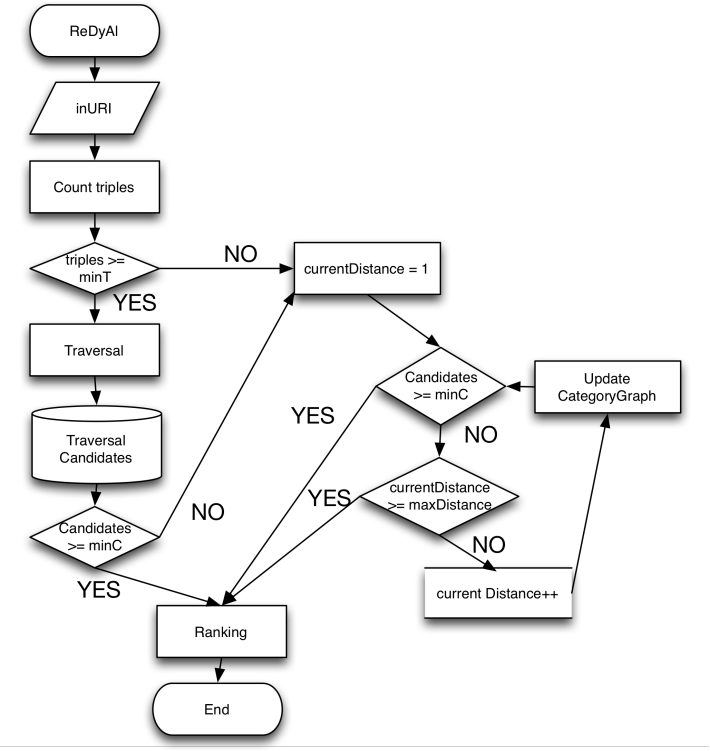


Figure 4 shows a flowchart of the *ReDyAl* algorithm, which receives as input an initial resource by specifying its corresponding URI ($inURI$), and three values ($minT$, $minC$, $maxDistance$) for configuring its execution. $minT$ is the minimum number of links (or triples involving the initial resource) to consider a resource as "well-linked" If the initial resource is "well linked", traversal interlinking has a higher priority in the generation of candidate resources, otherwise the algorithm gives priority to the hierarchical relationships.$minC$ is the minimum number of candidate resources that the algorithm is expected to generate, while $maxDistance$ limits the distance (number of hierarchical levels) that the algorithm considers in the category tree. The value of $maxDistance$ may be defined manually and it is useful when there are not enough candidate resources from the categories found at a certain distance (i.e., the number of candidate resources retrieved is lower than *minC*). In this case, the algorithm increases the distances to find more resources, and if the *maxDistance* value is reached with less than *minC* candidate resources, the algorithm ranks only the candidate resources found until that moment. Additionally, the algorithm may receive a list of "forbidden links" (*FL*) to avoid searching for candidate resources over a predefined list of undesired links.

Algorithm 2. ReDyAl algorithm: Source (Figueroa et al., 2017)

```
Require: inURI, minT, minC, FL, maxDistance
Ensure: A set of candidate resources CR
1:                              L_in = readAllowedLinks(inURI, FL)
2:     if |L_in| ≥ minT then
3:         for all l_k ∈ L_in do
4:             DR_{l_k} = getDirectResources(l_k)
5:             IR_{l_k} = getIndirectResources(l_k)
6:             Add DR_{p_k} to CR_tr
```

```
7:          Add CR_{p_k} to CR_{tr}
8:        end for
9:        if |CR_{tr}| ≥ minC then
10:          return CR_{tr}
11:       else
12:          currentDistance = 1
13:          Gc = createCategoryGraph(inURI, currentDistance)
14:          while currentDistance ≤ maxDistance do
15:             CR_{hi} = getCandidateResources(Gc)
16:             if |CR_{hi}| ≥ minC then
17                Add CR_{tr} and CR_{hi} to CR
18:               return CR
19:            end if
20:            increase currentDistance
21:            updateCategoryGraph(currentDistance)
22:         end while
23:         Add CR_{tr} and CR_{hi} to CR
24:       end if
25:   end if
26:   return CR
```

ReDyAl (*Algorithm 2*) starts by retrieving a list of allowed links from the initial resource. Allowed links are those that are not specified as forbidden (FL) and that are explicitly defined in the initial resource. If there is a considerable number of allowed links, i.e., the initial resource is well-linked, the algorithm obtains a set of candidate resources located through direct (DR_{l_k}) or indirect links (IR_{l_k}) (Lines 1-8). Next, the algorithm counts the number of candidate resources generated until this point (CR_{tr}), and if it is greater than or equal to $minC$, the execution terminates returning the results (Lines 9-10). Otherwise, the algorithm generates a category graph (Gc) with categories of the first distance. Subsequently, it applies iterative updates on the categories at a distance n from the initial resource to obtain broader categories until at least one of two conditions is fulfilled: either, the number of candidate resources is enough ($CR > minC$), or the maximum distance is reached ($currentDistance > maxDistance$). At each iteration, candidate resources (CR_{hi}) are extracted from the broader categories of maximum distance (Lines 14- 23). In any case the algorithm combines these results with the results obtained in Lines 3 – 8 (Adding CR_{tr} and CR_{hi} to CR). Finally, the set CR of candidate results is returned (Line 26).

EVALUATION

AlLied enables the comparative evaluation of any new algorithm to state-of-the-art algorithms. Using *AlLied*, the *ReDyAl* algorithm was evaluated concerning accuracy and novelty. This evaluation aimed to answer the following questions:

RQ1: Which of the considered algorithms is more accurate?

RQ2. Which of the considered algorithms provides the highest number of novel recommendations?

This section compares *ReDyAl* with algorithms that rely exclusively on Linked Data to produce recommendations.

Experiment

A user study was conducted involving 109 participants. The participants were mainly students of Politecnico di Torino (Italy) and University of Cauca (Colombia) enrolled in IT courses. The average age of the participants was 24 years old and they were 91 males, 14 females, and 4 did not provide any information about their sex. Although the proposed algorithm is not bounded to any particular domain, this evaluation was focused on movies because in this domain a quite large amount of data is available on DBpedia. Additionally, finding participants was not too difficult, since no specific skills are required to be able to express an opinion about movies. The evaluation was conducted as follows. A list of 20 recommendations generated from a given initial movie was presented to the participants. For each recommendation two questions were asked:

Q1: Did you already know this recommendation? Possible answers were yes, yes but I haven't seen it (if it is a movie) and no.

Q2: Is it related to the movie you have chosen? Possible answers were I strongly agree, I agree, I don't know, I disagree, I strongly disagree. Each answer was assigned respectively a score from 5 to 1.

Each list of 20 recommendations was pre-computed. Recommendations were generated for each of the 45 initial movies with the four different algorithms implemented within *AlLied.* Then, the recommendations generated by each algorithm were merged in a list of 20 recommendations to be shown to the participants.

The authors developed a website[4] to collect the answers from the participants. The participants were able to choose an initial movie from a list of 45 movies selected from the IMDB[5] top 250 list. The first 50 movies were considered, and five movies were excluded because they were not available in DBpedia. When a participant selected an initial movie, the tool provided the corresponding list of recommendations with the questions mentioned above. Participants were able to evaluate recommendations from as many initial movies as they wanted. As a result, the recommendations of the lists for 40 out of 45 initial movies were evaluated by at least one participant and each movie was evaluated by an average of 6.18 participants. The dataset with the initial movies and the lists of recommendations is available online[6].
About the questions stated at the beginning of this section, to answer RQ1, the Root Mean Squared Error (RMSE) (Shani & Gunawardana, 2011) was computed and to answer RQ2 the ratio between the number of evaluations. For the RMSE measure, scores given by the participants were normalized in the interval [0, 1] and considered as a reference. Then these scores were compared with the similarities computed by each algorithm, since each algorithm ranks its recommendations by using its own semantic similarity function.

Results

The results of the evaluation are summarized in Figure 5, which compares the algorithms to their RMSE and novelty.

Figure 5: Accuracy and novelty of the algorithms: Source (Figueroa et al., 2017)

The "sweet spot" area represents the conditions in which an algorithm has a good trade-off between novelty and prediction accuracy. In effect, presenting a high number of recommendations not known to the user is not necessarily good because it may prevent him to assess the quality of the recommendations: for example, having in the provided recommendation a movie which he has seen and which he liked may increase the trust of the user in the RS.

Concerning the RQ1, *HyProximity* accounted the lowest RMSE measures (with 25% and about 36% for the hierarchical and traversal versions respectively). Though, these results are less significant due to the low number of answers to Q2 for these algorithms (this means that the RMSE was computed over a low number of recommendations). For both *ReDyAl* and dbrec the RMSE is roughly 45%. Concerning RQ2, the two versions of *HyProximity* account for the highest values (hierarchical approximately 99%, while traversal about 97%). The high values of novelty mean that the algorithm can recommend more novel objects that have not been noticed by the user before, however, these low values in performance scored by *HyProximity* hierarchical and traversal imply that most of these novel results are not relevant. In this regard, *ReDyAl* and dbrec scored good values for novelty accounting respectively for about 60% and 45%. while presenting also good values for performance.

HyProximity generated recommendations based in both traversal and hierarchical algorithms, which only obtained few answers to Q2. Table 1 shows that most of the recommendations generated were unknown to the users. As a consequence, the results for both algorithms are less definitive than for the other algorithms. This is especially meaningful for RQ1, since only the evaluations for which the answer to Q1 was either yes or yes but I haven't seen it (if it is a movie) were considered for computing the accuracy measures. Furthermore, the Fleiss' kappa measure was evaluated for assessing the agreement of the participants answering Q2. The recommendations that were not evaluated by at least one participant were excluded. The scored value for the Fleiss' kappa was 0.79; which corresponds to a substantial agreement.

Figure 5 shows also that *ReDyAl* and dbrec provide a good trade-off among accuracy and novelty (sweet spot area), although *ReDyAl* performs better in novelty. *HyProximity* hierarchical and *HyProximity* traversal seems to be excellent performers since their RMSE is low and their novelty is high. However, it should be

noticed that RMSE was computed on few evaluations. A further analysis of these two algorithms is needed to verify if the user can benefit from such a high novelty and if novel recommendations are relevant. In addition, more research is needed on poorly-linked resources, since the choice of the initial films focused on selecting well-known films could ease the evaluation from participants. On poorly-linked resources *ReDyAl* and *Hyproximity* hierarchical are expected to score good values of the accuracy of the recommendations, since they can rely on categories, while dbrec and *HyProximity* traversal are likely to provide much less recommendations since they only rely on direct links between resources.

About the execution time, *ReDyAl* scored the best relationship candidate resources-execution time with a value of 3.7 resources per millisecond and the worse was scored by dbrec (0.28). These results showed, that generating candidate resources dynamically, not only allows improving the accuracy and novelty but also the execution time.

Table 1: Performance for generation layer algorithms

Generator algorithm	Time (ms)	Candidate Films
ReDyAl	1911.3	7069.0
dbrec	5379.2	11513.1
HyProximity hierarchical	15637.5	4404.5

Table 1 shows the results of the performance for the ranking layer algorithms. The algorithms evaluated were the *ReDyAl*, dbrec, and *HyProximity* hierarchical. The ranker algorithms were tested with a fixed number of candidate resources; otherwise it is not possible to compare the algorithms. Though, this is not a realistic situation because the generation layer algorithms may generate a different number of candidate resources. We selected 23000 resources because it was the mean value among the generation algorithms. This experiment demonstrated that the fastest ranking algorithm was *ReDyAl* and the lowest was *HyProximity* hierarchical. This fact was expected as the *HyProximity* hierarchical computes the similarity values based on both hierarchical and traversal links.

These results demonstrated that there is still a need for improving graph-based algorithms because, although the accuracy measures are good enough, the execution times are not suitable to quickly serve recommendations to users. For more details about this study please refer to (Figueroa, 2017).

DISCUSSION

Linked Data empowers RS to expand the results of recommendations to resources coming from different data sources and types. For example, the evaluation of the algorithms implemented in *AlLied* recommended resources about movies, however many of the results were not only resources of this type but also about producers, actors, cities, etc. This feature is an advantage in the case of RS that require the use of heterogeneous information, for example, in the domain of tourism, the recommended items may not be only points of interest such as museums, but also important personalities living in the city, typical foods, among others. Furthermore, Linked data-based RS are useful to present explanations of the recommendations, because of the graph structure of the datasets in which the items are interlinked. In this case, following the links of the graph to which the recommended items belong is enough.

One of the features of the *AlLied* is that RS developed on top of it do not require user profile information to generate recommendations, they only need to consider the relationships of the concepts extracted from a resource with the concepts on the same or other existing structured datasets. Nonetheless, the knowledge of Linked Data should not be limited only to exploit relationships among items, but also to enrich data about items and users and to generate implicit knowledge about them and their relationships. In this way, RS would be able to produce personalized recommendations relying on the derived implicit knowledge.

AlLied is the first framework for Linked Data based RS that splits the recommendation process into meaningful phases embodied by layers. Each layer represents one task of the recommendation process that may be executed by various algorithms. In this way, *AlLied* allows developers to create and evaluate different configurations and combinations of algorithms at each layer to develop novel RS based on Linked Data. Hence, these RS are more suitable to users' requirements, applications, domains, and contexts.

Concerning the performance, obtaining resources with a reasonable degree of accuracy while keeping low-execution times is still subject to research. This work encourages other scientists to study this interesting type of RS using *AlLied*. Its layered architecture allows developers to create compositions of recommendation algorithms and to determine the most suitable combinations for the desired context.

It is also useful for the reproducibility of the results for the research community of RS because algorithms for recommendation are classified in the corresponding *AlLied*'s layer. Hence, these may be tuned to improve the accuracy and performance of the overall RS for a specific application. This is especially important taking into account that the difficulty to reproduce results is widely agreed in the community.

This study demonstrated that some of the algorithms for RS are more suitable to generate candidate resources under certain conditions of the initial resource. For example, if the initial resource contains a minimum number of links (properties) to other resources it should be desirable to execute the traversal algorithm that is specialized on finding related resources through the traversal links. In consequence, a new algorithm named ReDyAl was designed to automatically choose the best algorithm to recommend resources based on the relations of the initial resource. This algorithm demonstrated its ability to generate novel recommendations, which is useful when users do not want to receive recommendations about items they already know or have previously consumed.

FUTURE RESEARCH DIRECTIONS

The graph-based algorithms studied in this chapter were explicitly developed to deal with Linked Data datasets. They take advantage of the relationships among the items represented through resources of the Web of Data. Nevertheless, the execution time of the graph-based algorithms is still an open issue because they far exceed the performance of machine learning algorithms. Therefore, a logical evolution of this study is the combination of graph-based algorithms with machine learning algorithms. This evolution can make possible to obtain accurate results in an execution time adapted to the needs of the end users. Research and experimentation are still needed in RS based on Linked Data, to explore more techniques from the vast amount of machine learning algorithms to determine which of them are more suitable to deal with Linked Data datasets.

Furthermore, a recent method known as "Propagated Linked Data Semantic Distance (PLDSD)" for the traversal ranking, improves the results of the algorithms based on the semantic distance LDSD proposed initially by Passant. The PLDSD uses the Floyd-Warshall algorithm to expand semantic distance calculations beyond resources that are just one or two links (Alfarhood, Labille, & Gauch, 2017). The implementation of this measure into the *AlLied* framework means an improvement of the traversal ranking algorithm because unlike the LDSD, it extends the number of items considered for the calculation of the semantic distance.

In the same way, it is necessary a broad study of more types of algorithms such as evolutionary computation, automated planning, among others to analyze their relevance under different domains, which is required to increase the available set of techniques for each layer of the *AlLied* framework. This study may lead to an improvement of the performance and accuracy of the *ReDyAl* algorithm while maintaining its novelty.

Another issue which is gaining interest is mining microblogging data and text reviews. Opinion mining and sentiment analysis techniques can support recommendation methods that take into account the evaluation of aspects of items expressed in text reviews. Extracting information from the raw text in the form of Linked Data can ease its exploitation and the integration.

A closely related research area is exploratory search. It refers to cognitive consuming search tasks such as learning or topic investigation. Exploratory search systems also recommend relevant topics or concepts. An open question not addressed in this work is how to leverage the data semantics richness for successful exploratory search.

Finally, a current problem with the knowledge published in linked data is the poor quality of data. It is demonstrated that data extracted from semi-structured or even structured sources, often contains inconsistencies as well as misrepresented and incomplete information (Zaveri, Maurino, & Equille, 2014). A next step for improving the results of the algorithms implemented in *AlLied* is to develop an additional layer for addressing data quality in both local and remote datasets.

CONCLUSION

This chapter reviewed state-of-art Linked Data based recommendation algorithms and presented *AlLied*, a framework for deploying and executing these algorithms. *AlLied* allows developers to create and evaluate different configurations and combinations of algorithms at each layer to develop novel RS based on Linked Data. Hence, these RS are more suitable to users' requirements, applications, domains, and contexts. It is also useful for the reproducibility of the results for the research community of RS because algorithms for recommendation are classified in the corresponding *AlLied*'s layer. Hence, these may be tuned to improve the accuracy and performance of the overall RS for a specific application.

Additionally, a hybrid algorithm named ReDyAl is presented. This algorithm dynamically integrates both traversal and hierarchical approaches for discovering resources. The ReDyAl algorithm was designed based on the analysis of state-of-the-art algorithms that were implemented within the *AlLied* framework. The current version of this framework contains a set of three state-of-the-art traversal and hierarchical algorithms and the ReDyAl algorithm.

In this work, a user study was conducted to analyze the ReDyAl algorithm and a set of state-of-the-art algorithms re-implemented within *AlLied*. This framework facilitated the study a common framework contained all the algorithms involved, and the results of the generated recommendations were aligned.

The study demonstrated that, although ReDyAl improved the novelty of the results discovered, the accuracy of the algorithm was not the highest due to its inherent complexity.

Future work includes studying the relevance under different domains and improving the accuracy of ReDyAl while maintaining its novelty. This can lead us to discover resources faster and make the algorithm usable for real-time applications.

ACKNOWLEDGMENT

The first author was supported by the fellowship No 511 from the Administrative Department of Science, Technology and Innovation (COLCIENCIAS) of Colombia.

REFERENCES

Adomavicius, G., & Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*. https://doi.org/10.1109/TKDE.2005.99

Ahn, J., & Amatriain, X. (2010). Towards Fully Distributed and Privacy-Preserving Recommendations via Expert Collaborative Filtering and RESTful Linked Data. In *2010 IEEE/WIC/ACM International*

Conference on Web Intelligence and Intelligent Agent Technology (Vol. 1, pp. 66–73). IEEE. https://doi.org/10.1109/WI-IAT.2010.53

Alfarhood, S., Labille, K., & Gauch, S. (2017). PLDSD: Propagated linked data semantic distance. In *Proceedings - 2017 IEEE 26th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE 2017* (pp. 278–283). https://doi.org/10.1109/WETICE.2017.16

Arturo, A., Caraballo, M., Moura, N., Jr, A., Nunes, B. P., Lopes, G. R., … Casanova, M. A. (2014). TRTML - A Tripleset Recommendation Tool Based on Supervised Learning Algorithms. In V. Presutti, E. Blomqvist, R. Troncy, H. Sack, I. Papadakis, & A. Tordai (Eds.), *The Semantic Web: ESWC 2014 Satellite Events SE - 58* (Vol. 8798, pp. 413–417). Springer International Publishing. https://doi.org/10.1007/978-3-319-11955-7_58

Ayala, V. A. A., Przyjaciel-Zablocki, M., Hornung, T., Schätzle, A., & Lausen, G. (2014). Extending SPARQL for Recommendations. In *Proceedings of Semantic Web Information Management on Semantic Web Information Management* (pp. 1–8). https://doi.org/10.1145/2630602.2630604

Bizer, C., Heath, T., & Berners-Lee, T. (2009). Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems*, *5*, 1–22. https://doi.org/10.4018/jswis.2009081901

Blanco, R., Cambazoglu, B. B., Mika, P., & Torzec, N. (2013). Entity recommendations in web search. In *The Semantic Web–ISWC 2013* (pp. 33–48). https://doi.org/10.1007/978-3-642-41338-4_3

Calì, A., Capuzzi, S., Dimartino, M. M., & Frosini, R. (2013). Recommendation of text tags in social applications using linked data. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 8295 LNCS, pp. 187–191). https://doi.org/10.1007/978-3-319-04244-2_17

Cantador, I., Castells, P., & Bellogín, A. (2011). An Enhanced Semantic Layer for Hybrid Recommender Systems: Application to News Recommendation. *International Journal on Semantic Web and Information Systems*, *7*(1), 44–78. https://doi.org/10.4018/jswis.2011010103

Cantador, I., Konstas, I., & Jose, J. M. (2011). Categorising social tags to improve folksonomy-based recommendations. *Web Semantics: Science, Services and Agents on the World Wide Web*, *9*(1), 1–15. https://doi.org/10.1016/j.websem.2010.10.001

Chawuthai, R., Takeda, H., & Hosoya, T. (2015). Link Prediction in Linked Data of Interspecies Interactions Using Hybrid Recommendation Approach. In T. Supnithi, T. Yamaguchi, J. Z. Pan, V. Wuwongse, & M. Buranarach (Eds.), *Semantic Technology SE - 9* (Vol. 8943, pp. 113–128). Springer International Publishing. https://doi.org/10.1007/978-3-319-15615-6_9

Cheekula, S. K., Kapanipathi, P., Doran, D., & Jain, P. (2015). Entity Recommendations Using Hierarchical Knowledge Bases. In *Proceedings of the 4th Workshop on Knowledge Discovery and Data Mining Meets Linked Open Data co-located with 12th Extended Semantic Web Conference (ESWC 2015)*. Retrieved from http://ceur-ws.org/Vol-1365/

Chicaiza, J., Piedra, N., López-Vargas, J., & Tovar-Edmundo. (2014). Domain Categorization of Open Educational Resources Based on Linked Data. *Knowledge Engineering and the Semantic Web*, (JANUARY). https://doi.org/http://dx.doi.org/10.1007/978-3-319-11716-4_2

Damljanovic, D., Stankovic, M., & Laublet, P. (2012). Linked Data-Based Concept Recommendation: Comparison of Different Methods in Open Innovation Scenario. In E. Simperl, P. Cimiano, A. Polleres, O. Corcho, & V. Presutti (Eds.), *The Semantic Web: Research and Applications* (pp. 24–38). Heraklion, Crete, Greece: Springer Berlin Heidelberg. Retrieved from http://link.springer.com/chapter/10.1007/978-3-642-30284-8_9

Dell'Aglio, D., Celino, I., & Cerizza, D. (2010). Anatomy of a Semantic Web-enabled Knowledge-based Recommender System. In *4th international workshop Semantic Matchmaking and Resource Retrieval in the Semantic Web, at ISWC* (Vol. 667, pp. 115–130). Bonn, Germany. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?rep=rep1&type=pdf&doi=10.1.1.204.3679

Di Noia, T., Mirizzi, R., Ostuni, V. C., & Romito, D. (2012). Exploiting the web of data in model-based recommender systems. In *6th ACM conference on Recommender systems - RecSys '12* (p. 253). New York, New York, USA: ACM Press. https://doi.org/10.1145/2365952.2366007

Felfernig, A., Jeran, M., Ninaus, G., Reinfrank, F., & Reiterer, S. (2013). Toward the Next Generation of Recommender Systems : Applications and Research Challenges. In *Multimedia Services in Intelligent*

Environments (pp. 81–98). Springer. Retrieved from http://link.springer.com/chapter/10.1007/978-3-319-00372-6_5#

Figueroa, C. (2017). *Recommender Systems based on Linked Data*. Politecnico di Torino - Universidad del Cauca. https://doi.org/10.6092/polito/porto/2669963

Figueroa, C., Vagliano, I., Rocha, O. R., Torchiano, M., Zucker, C. F., Corrales, J.-C. C., & Morisio, M. (2017). Allied: A Framework for Executing Linked Data-Based Recommendation Algorithms. *International Journal on Semantic Web and Information Systems*, *13*(4), 134–154. https://doi.org/10.4018/IJSWIS.2017100107

Figueroa, C., Vagliano, I., Rodríguez Rocha, O., & Morisio, M. (2015). A systematic literature review of Linked Data-based recommender systems. *Concurrency and Computation: Practice and Experience*, *27*(17), 4659–4684. https://doi.org/10.1002/cpe.3449

Gordea, S., Lindley, A., & Graf, R. (2011). Computing Recommendations for Long Term Data Accessibility basing on Open Knowledge and Linked Data. In *RecSys 2011 Workshop on Human Decision Making in Recommender Systems affiliated with the 5th ACM Conference on Recommender Systems* (pp. 1–8). Chicago, USA. Retrieved from http://ceur-ws.org/Vol-811/paper8.pdf

Hadj Taieb, M. A., Ben Aouicha, M., Tmar, M., & Hamadou, A. B. (2011). New information content metric and nominalization relation for a new WordNet-based method to measure the semantic relatedness. *Cybernetic Intelligent Systems (CIS), 2011 IEEE 10th International Conference on*. https://doi.org/10.1109/CIS.2011.6169134

Hajra, A., Latif, A., & Tochtermann, K. (2014). Retrieving and Ranking Scientific Publications from Linked Open Data Repositories. In *Proceedings of the 14th International Conference on Knowledge Technologies and Data-driven Business* (p. 29:1--29:4). New York, NY, USA: ACM. https://doi.org/10.1145/2637748.2638436

Hopfgartner, F., & Jose, J. M. (2010). Semantic user profiling techniques for personalised multimedia recommendation. *Multimedia Systems*, *16*(4–5), 255–274. https://doi.org/10.1007/s00530-010-0189-6

Juang, Y.-T., Tung, S.-L., & Chiu, H.-C. (2011). Adaptive fuzzy particle swarm optimization for global optimization of multimodal functions. *Information Sciences*, *181*(20), 4539–4549. https://doi.org/10.1016/j.ins.2010.11.025

Khrouf, H., & Troncy, R. (2013). Hybrid event recommendation using linked data and user diversity. In *Proceedings of the 7th ACM conference on Recommender systems - RecSys '13* (pp. 185–192). New York, New York, USA: ACM Press. https://doi.org/10.1145/2507157.2507171

Kitaya, K., Huang, H.-H., & Kawagoe, K. (2012). Music Curator Recommendations Using Linked Data. In *Second International Conference on the Innovative Computing Technology (INTECH 2012)* (pp. 337–339). Casablanca: IEEE. https://doi.org/10.1109/INTECH.2012.6457799

Ko, H., Kim, E., Ko, I.-Y., & Chang, D. (2014). Semantically-based recommendation by using semantic clusters of users' viewing history. In *Big Data and Smart Computing (BIGCOMP), 2014 International Conference on* (pp. 83–87). IEEE.

Ko, H., & Son, J. (2015). Multi-Aspect Collaborative Filtering based on Linked Data for Personalized Recommendation. In *Proceedings of the 1st Workshop on New Trends in Content-based Recommender Systems co-located with the 8th ACM Conference on Recommender Systems (RecSys 2014)* (Vol. New Trends, pp. 49–50).

Kushwaha, N., & Vyas, O. P. (2014). SemMovieRec: Extraction of Semantic Features of DBpedia for Recommender System. In *Proceedings of the 7th ACM India Computing Conference* (p. 13:1--13:9). New York, NY, USA: ACM. https://doi.org/10.1145/2675744.2675759

Lommatzsch, A., Kille, B., & Albayrak, S. (2013). Learning hybrid recommender models for heterogeneous semantic data. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing - SAC '13* (p. 275). New York, New York, USA: ACM Press. https://doi.org/10.1145/2480362.2480420

Lommatzsch, A., Kille, B., Albayrak, S., & Berlin, T. U. (2013). A Framework for Learning and Analyzing Hybrid Recommenders based on Heterogeneous Semantic Data Categories and Subject Descriptors. In *10th Conference on Open Research Areas in Information Retrieval* (pp. 137–140). Lisbon, Portugal: ACM.

Lommatzsch, A., Kille, B., Kim, J. W., & Albayrak, S. (2013). An Adaptive Hybrid Movie Recommender

Based on Semantic Data. In *Proceedings of the 10th Conference on Open Research Areas in Information Retrieval* (pp. 217–218). Paris, France, France: LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE. Retrieved from http://dl.acm.org/citation.cfm?id=2491748.2491795

Lopes, G. R., André, L., Rabello Lopes, G., Paes Leme, L., Pereira Nunes, B., Casanova, M., & Dietze, S. (2014). Two Approaches to the Dataset Interlinking Recommendation Problem. *Web Information Systems Engineering – WISE 2014 SE - 25*, *8786*, 324–339. https://doi.org/10.1007/978-3-319-11749-2_25

Lopes, G. R., Leme, L. A. P. P., Nunes, B. P., & Casanova, M. A. (2014). RecLAK: Analysis and recommendation of interlinking datasets. In *CEUR Workshop Proceedings* (Vol. 1137).

Lops, P., Gemmis, M. De, & Semeraro, G. (2011). Content-based Recommender Systems: State of the Art and Trends. In *Recommender Systems Handbook* (pp. 73–105). https://doi.org/10.1007/978-0-387-85820-3

Maccatrozzo, V., Ceolin, D., Aroyo, L., & Groth, P. (2014). A Semantic Pattern-Based Recommender. In V. Presutti, M. Stankovic, E. Cambria, I. Cantador, A. Di Iorio, T. Di Noia, … A. Tordai (Eds.), *Semantic Web Evaluation Challenge: SemWebEval 2014 at ESWC 2014, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers* (pp. 182–187). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-319-12024-9_24

Manoj Kumar, S., Anusha, K., & Santhi Sree, K. (2015). Semantic Web-based Recommendation: Experimental Results and Test Cases. *International Journal of Emerging Research in Management & Technology*, *4*(6), 215–222. Retrieved from http://www.ermt.net/docs/papers/Volume_4/6_June2015/V4N6-252.pdf

Marie, N., Gandon, F., Legrand, D., & Ribière, M. (2013). Discovery Hub: a discovery engine on the top of DBpedia. In *3rd International Conference on Web Intelligence, Mining and Semantics - WIMS '13* (p. 1). New York, New York, USA: ACM Press. https://doi.org/10.1145/2479787.2479820

Moreno, A., Ariza-Porras, C., Lago, P., Jiménez-Guarín, C. L., Castro, H., & Riveill, M. (2014). Hybrid Model Rating Prediction with Linked Open Data for Recommender Systems. In *SemWebEval 2014 at ESWC 2014, Anissaras, Crete, Greece, May 25-29, 2014, Revised Selected Papers* (pp. 193–198). https://doi.org/10.1007/978-3-319-12024-9_26

Moyano, A. N., Sicilia, M. A., & Barriocanal, E. G. (2018). On the Graph Structure of the Web of Data. *International Journal on Semantic Web and Information Systems (IJSWIS)*, *14*(2), 70–85. https://doi.org/10.4018/IJSWIS.2018040104

Musto, C., Basile, P., Lops, P., de Gemmis, M., & Semeraro, G. (2017). Introducing linked open data in graph-based recommender systems. *Information Processing and Management*, *53*(2), 405–435. https://doi.org/10.1016/j.ipm.2016.12.003

Musto, C., Lops, P., de Gemmis, M., & Semeraro, G. (2017). Semantics-aware Recommender Systems exploiting Linked Open Data and graph-based features. *Knowledge-Based Systems*, *136*, 1–14. https://doi.org/10.1016/j.knosys.2017.08.015

Narducci, F., Musto, C., Semeraro, G., Lops, P., & de Gemmis, M. (2013). Exploiting Big Data for Enhanced Representations in Content-Based Recommender Systems. In C. Huemer & P. Lops (Eds.), *14th International Conference, EC-Web 2013, Prague, Czech Republic, August 27-28, 2013. Proceedings* (Vol. 152, pp. 182–193). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-39878-0_17

Nguyen, P. T., Tomeo, P., Di Noia, T., Di Sciascio, E., Noia, T. Di, & Sciascio, E. Di. (2015). An Evaluation of SimRank and Personalized PageRank to Build a Recommender System for the Web of Data. In *Proceedings of the 24th International Conference on World Wide Web* (pp. 1477–1482). Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee. https://doi.org/10.1145/2740908.2742141

Ostuni, V. C. C., Gentile, G., Noia, T. Di, Di Noia, T., Mirizzi, R., Romito, D., & Di Sciascio, E. (2013). Mobile movie recommendations with linked data. In *IFIP WG 8.4, 8.9, TC 5 International Cross-Domain Conference, CD-ARES 2013, Regensburg, Germany, September 2-6, 2013. Proceedings* (Vol. 8127 LNCS, pp. 400–415). https://doi.org/10.1007/978-3-642-40511-2_29

Ostuni, V. C., Di Noia, T., Di Sciascio, E., & Mirizzi, R. (2013). Top-N recommendations from implicit

feedback leveraging linked open data. In *Proceedings of the 7th ACM conference on Recommender systems - RecSys '13* (pp. 85–92). New York, New York, USA: ACM Press. https://doi.org/10.1145/2507157.2507172

Ostuni, V. V., Di Noia, T., Mirizzi, R., Di Sciascio, E., & Noia, T. Di. (2014). A Linked Data Recommender System Using a Neighborhood-Based Graph Kernel. In M. Hepp & Y. Hoffner (Eds.), *E-Commerce and Web Technologies SE - 10* (Vol. 188, pp. 89–100). Springer International Publishing. https://doi.org/10.1007/978-3-319-10491-1_10

Ozdikis, O., Orhan, F., & Danismaz, F. (2011). Ontology-based recommendation for points of interest retrieved from multiple data sources. *International Workshop on Semantic Web Information Management - SWIM '11*, 1–6. https://doi.org/10.1145/1999299.1999300

Park, D. H., Kim, H. K., Choi, I. Y., & Kim, J. K. (2012). A literature review and classification of recommender systems research. *Expert Systems with Applications*, *39*(11), 10059–10072. https://doi.org/10.1016/j.eswa.2012.02.038

Passant, A. (2010a). dbrec - Music recommendations using DBpedia. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 6497 LNCS, pp. 209–224). https://doi.org/10.1007/978-3-642-17749-1_14

Passant, A. (2010b). dbrec — Music Recommendations Using DBpedia. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (Vol. 6497 LNCS, pp. 209–224). https://doi.org/10.1007/978-3-642-17749-1_14

Portugal, I., Alencar, P., & Cowan, D. (2018). The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*. https://doi.org/10.1016/j.eswa.2017.12.020

Ricci, F., Rokach, L., & Shapira, B. (2011). Introduction to Recommender Systems Handbook. In *Recommender Systems Handbook* (pp. 1–35). Springer. https://doi.org/10.1007/978-0-387-85820-3_1

Ristoski, P., Loza Mencía, E., Paulheim, H., & Menc, E. L. (2014). A Hybrid Multi-strategy Recommender System Using Linked Open Data. In V. Presutti, M. Stankovic, E. Cambria, I. Cantador, A. Di Iorio, T. Di Noia, … A. Tordai (Eds.), *Semantic Web Evaluation Challenge SE - 19* (Vol. 475, pp. 150–156). Springer International Publishing. https://doi.org/10.1007/978-3-319-12024-9_19

Rowe, M. (2014). Transferring Semantic Categories with Vertex Kernels: Recommendations with SemanticSVD++. In P. Mika, T. Tudorache, A. Bernstein, C. Welty, C. Knoblock, D. Vrandečić, … C. Goble (Eds.), *13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I* (Vol. 8796, pp. 341–356). Springer International Publishing. https://doi.org/10.1007/978-3-319-11964-9_22

Schmachtenberg, M., Bizer, C., & Paulheim, H. (2014). Adoption of the Linked Data Best Practices in Different Topical Domains. In P. Mika, T. Tudorache, A. Bernstein, C. Welty, C. Knoblock, D. Vrandečić, … C. Goble (Eds.), *The Semantic Web – ISWC 2014 SE - 16* (Vol. 8796, pp. 245–260). Springer International Publishing. https://doi.org/10.1007/978-3-319-11964-9_16

Schmachtenberg, M., Strufe, T., & Paulheim, H. (2014). Enhancing a Location-based Recommendation System by Enrichment with Structured Data from the Web. In *Proceedings of the 4th International Conference on Web Intelligence, Mining and Semantics (WIMS14) - WIMS '14* (pp. 1–12). https://doi.org/10.1145/2611040.2611080

Seco, N., Veale, T., & Hayes, J. (2004). An Intrinsic Information Content Metric for Semantic Similarity in WordNet. *European Conference on Artificial Intelligence*.

Stankovic, M., Breitfuss, W., & Laublet, P. (2011). Discovering Relevant Topics Using DBPedia: Providing Non-obvious Recommendations. In *2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology* (pp. 219–222). IEEE. https://doi.org/10.1109/WI-IAT.2011.32

Strobin, L., & Niewiadomski, A. (2014). Recommendations and Object Discovery in Graph Databases Using Path Semantic Analysis. In L. Rutkowski, M. Korytkowski, R. Scherer, R. Tadeusiewicz, L. Zadeh, & J. Zurada (Eds.), *13th International Conference, ICAISC 2014, Zakopane, Poland, June 1-5, 2014, Proceedings, Part I* (Vol. 8467, pp. 793–804). Springer International Publishing. https://doi.org/10.1007/978-3-319-07173-2_68

Tomeo, P., Fernández-Tobías, I., Cantador, I., & Di Noia, T. (2017). Addressing the Cold Start with

Positive-Only Feedback Through Semantic-Based Recommendations. *International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems*, *25*. https://doi.org/10.1142/S0218488517400116

Vagliano, I., Figueroa, C., Rodriguez, O., Torchiano, M., Faron-Zucker, C., & Morisio, M. (2016). ReDyAl: A Dynamic Recommendation Algorithm based on Linked Data. In *3rd Workshop on New Trends in Content-based Recommender Systems - CBRecSys 2016* (pp. 31–39). Boston, MA, USA: CEUR Workshop Proceedings. Retrieved from http://ceur-ws.org/Vol-1673/paper6.pdf

Welling, M. (2011). *A First Encounter with Machine Learning*. Donald Bren School of Information and Computer Science - University of California Irvine. Retrieved from https://www.ics.uci.edu/~welling/teaching/ICS273Afall11/IntroMLBook.pdf

Yang, R., Hu, W., Qu, Y., Li, G., Wang, P., Yu, B., … Qu, Y. (2013). Using Semantic Technology to Improve Recommender Systems Based on Slope One. *Semantic Web and Web Science SE - 2*, *2*(1), 11–23. https://doi.org/10.1007/978-1-4614-6880-6_2

Zaveri, A., Maurino, A., & Equille, L.-B. (2014). Web Data Quality: Current State and New Challenges Amrapali. *International Journal on Semantic Web and Information Systems*, *10*(2), 1–6. https://doi.org/10.4018/ijswis.2014040101

Zhang, Y., Wu, H., Sorathia, V., & Prasanna, V. K. (2014). Event recommendation in social networks with linked data enablement. In *ICEIS 2013 - Proceedings of the 15th International Conference on Enterprise Information Systems* (Vol. 2, pp. 371–379). Retrieved from http://www.scopus.com/inward/record.url?eid=2-s2.0-84887649972&partnerID=tZOtx3y1

ADDITIONAL READING

Portugal, I., Alencar, P., & Cowan, D. (2018). The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*. https://doi.org/10.1016/j.eswa.2017.12.020

Moyano, A. N., Sicilia, M. A., & Barriocanal, E. G. (2018). On the Graph Structure of the Web of Data. *International Journal on Semantic Web and Information Systems (IJSWIS)*, *14*(2), 70–85. https://doi.org/10.4018/IJSWIS.2018040104

Zhang, X., Lin, E., & Lv, Y. (2018). Multi-Target Search on Semantic Associations in Linked Data. *International Journal on Semantic Web and Information Systems (IJSWIS)*, *14*(1), 71–97. https://doi.org/10.4018/IJSWIS.2018010103

Sabou, M., Aroyo, L., Bontcheva, K., Bozzon, A., & Qarout, R. K. (2018). Semantic web and human computation: The status of an emerging field. *Semantic Web*, (Preprint), 1–12.

Alfarhood, S., Labille, K., & Gauch, S. (2017). PLDSD: Propagated linked data semantic distance. In *Proceedings - 2017 IEEE 26th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE 2017* (pp. 278–283). https://doi.org/10.1109/WETICE.2017.16

Musto, C., Lops, P., de Gemmis, M., & Semeraro, G. (2017). Semantics-aware Recommender Systems exploiting Linked Open Data and graph-based features. *Knowledge-Based Systems*, *136*, 1–14. https://doi.org/10.1016/j.knosys.2017.08.015

Tomeo, P., Fernández-Tobías, I., Cantador, I., & Di Noia, T. (2017). Addressing the Cold Start with Positive-Only Feedback Through Semantic-Based Recommendations. *International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems*, *25*. https://doi.org/10.1142/S0218488517400116

Figueroa, C., Ordoñez, H., Corrales, J.-C., Cobos, C., Wives, L. K., & Herrera-Viedma, E. (2016). Improving business process retrieval using categorization and multimodal search. *Knowledge-Based Systems*, *110*, 49–59. https://doi.org/10.1016/j.knosys.2016.07.014

Rocha, O. R., Vagliano, I., Figueroa, C., Cairo, F., Futia, G., Licciardi, C. A., Morando, F. (2015). Semantic annotation and classification in practice. *IT Professional*, *17*(2).

KEY TERMS AND DEFINITIONS

Linked Data: They are a set of good practices or principles for publishing and linking structured data on the Web. Linked Data provide the means to make the Semantic Web a reality.

Linked Data Datasets: They are sets of structured data using the principles of the linked data. Much of them are interlinked so it is possible to not only exploit the knowledge of a single dataset but also the power of the semantic interconnection among them.

Linked Data Endpoints: they are mechanism used in Linked Data to provide access to the available datasets. Endpoints may be interfaces to execute queries to the datasets in a similar way as in a database.

Resource Description Framework (RDF): This framework is a recommendation of the W3C that provides a generic graph-based data model for describing resources, including their relationships with other resources. The graph data model of the RDF framework is composed of triples or statements. Each triple contains a subject, a predicate, and an object.

Linked Data based RS: A kind of knowledge-based RS which uses linked data as source of data to infer relationships or to represent items and users, with the aim to improve the recommendation process.

Recommender System: A kind of algorithms or frameworks designed to recommend items of interest for an end user. These items can belong to different categories or types, e.g. songs, places, news, books, films, events, etc.

The Web of Data: Unlike the World Wide Web that consists of human-readable documents linked via hyperlinks, the “Web of Data” refers to a global space of structured and machine-readable data.

APPENDIX 1: BACKGROUND OF THE RESEARCH UNIT

The research described in this chapter was conducted in collaboration of two Ph.D. programs. The Ph.D. Course in Telematics Engineering (Universidad del Cauca), and the Ph.D. Course in Computers and Systems Engineering (Politecnico di Torino). The former focus on telematics systems and services oriented to the development of the Information and Communication Technologies (ICT). The latter addresses computer and systems engineering, with main interest in automation, informatics and operational research. The Ph.D. advisors were: Prof. Juan Carlos Corrales (Universidad del Cauca), whose research areas are services composition and data analysis; and Prof. Maurizio Morisio (Politecnico di Torino) whose interests lie in finding, applying and evaluating the most suitable techniques and process to produce better software faster.

Additionally, the research was supported by the Wimmics joint research team between Inria Sophia Antipolis - Méditerranée and I3S (CNRS and Université Nice Sophia Antipolis). Wimmics stands for Web-Instrumented Man-Machine Interactions, Communities, and Semantics, and their challenge is to bridge formal semantics and social semantics on the web.

ENDNOTES

[1] http://www.mpi-inf.mpg.de/departments/databases-and-information-systems/research/yago-naga/yago/
[2] https://wordnet.princeton.edu/
[3] http://www.w3.org/2004/02/skos/
[4] http://natasha.polito.it/RSEvaluation/
[5] http://www.imdb.com/chart/top
[6] http://natasha.polito.it/RSEvaluation/faces/resultsdownload.xhtml