Measuring Semantic Distance for Linked Open Data-enabled Recommender Systems

Guangyuan Piao
Insight Centre for Data Analytics
National University of Ireland Galway
IDA Business Park, Lower Dangan, Galway, Ireland
guangyuan.piao@insight-centre.org

John G. Breslin
Insight Centre for Data Analytics
National University of Ireland Galway
IDA Business Park, Lower Dangan, Galway, Ireland
john.breslin@nuigalway.ie

ABSTRACT

The Linked Open Data (LOD) initiative has been quite successful in terms of publishing and interlinking data on the Web. On top of the huge amount of interconnected data, measuring relatedness between resources and identifying their relatedness could be used for various applications such as LOD-enabled recommender systems. In this paper, we propose various distance measures, on top of the basic concept of *Linked Data Semantic Distance* (*LDSD*), for calculating Linked Data semantic distance between resources that can be used in a LOD-enabled recommender system. We evaluated the distance measures in the context of a recommender system that provides the top-N recommendations with baseline methods such as *LDSD*. Results show that the performance is significantly improved by our proposed distance measures incorporating normalizations that use both of the resources and global appearances of paths in a graph.

CCS Concepts

•Information systems → Similarity measures; Recommender systems;

Keywords

Linked Data; Semantic similarity; Recommender system

1. INTRODUCTION

In recent years, the Linked Open Data (LOD) cloud[1] has been increasing in popularity. As a result of the success of the LOD, many semantic datasets are freely available on the Web in machine-understandable format (primarily RDF [1]) related to different domains. The LOD has been adopted in recommender systems in order to improve the performance of such systems as well as reduce the *cold-start* problem inherent to recommender systems [2, 8, 12–14]. Also, the need for a semantic representation of data and user profiles has been identified as one of the next challenges in the field of recommender systems [11]. The use of semantic data in recommendation tasks not only can provide better knowledge (e.g., a richer representation of data), but also can facilitate the easy adoption of the same approach to other domains [3]. Moreover, problems related to a keyword-based approach such as synonymy and polysemy are resolved since resources in LOD datasets are identified by unique URIs and semantically interlinked to each other [11]. All of the information within the LOD can be exploited and used in an LOD-enabled, content-based recommender system where the domain knowledge plays a fundamental role. In this regard, measuring the distance between resources and identifying their relatedness plays a significant role as it can be adopted to recommender systems for providing recommendations. To this end, various approaches to measure the distance/similarity between two resources in LOD datasets such as DBpedia[2] have been proposed [6, 9, 15, 19]. *Linked Data Semantic Distance* (*LDSD*) [15] is one of the most popular approaches to measure the semantic distance between two resources, which has been adopted to a recommender system in the music domain [14]. Di Noia et. al [3, 4] proposed applying one of the most popular models in information retrieval: the Vector Space Model (VSM) [17] in a LOD-based setting and representing the whole RDF graph as a matrix. Items and user profiles can be represented in terms of the VSM and the similarity between items and users can be identified by using the *cosine similarity* measure.

In this paper, based on the basic concept of *LDSD*, we present various semantic distance measures by incorporating the number of connected resources via a link, different normalization strategies and a statistical approach for calculating the semantic distance between two resources. We then evaluate these distance measures in the context of a recommender system that provides the top-N recommendations. It is noteworthy that although we performed our experiments using the DBpedia dataset, the distance mea-

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

[2]http://dbpedia.org/

SAC 2016, April 04-08, 2016, Pisa, Italy

DOI: http://dx.doi.org/10.1145/2851613.2851839

sures we proposed are not tied to this particular dataset and can be adopted to any similar LOD datasets as well.

The rest of the paper is organized as follows: Section 2 gives some related work. In Section 3 we propose various semantic distance measures for calculating the distance between two resources. Section 4 is dedicated to the evaluation and Section 5 presents the results. Finally, Section 6 concludes the paper and gives some ideas for future work.

2. RELATED WORK

Maedche et. al [10] defined a set of similarity measures for comparing ontology-based metadata by considering different aspects of an ontology separately. They propose differentiating across three dimensions for comparing two resources: taxonomic, relational and attribute similarities. However, the similarity measures depend on some strong assumptions about the model such as "Ontologies are strictly hierarchical such that each concept is subsumed by only one concept", which is not the case in terms of DBpedia. Passant [15] proposed a measure named *LDSD* to calculate semantic distance on Linked Data. The distance measure considers direct links from resource A to resource B and vice versa. In addition, it also considers the same incoming and outgoing nodes via the same properties of resources A and B in a graph. However, several aspects of relatedness that mentioned in this paper have been ignored. Leal et al. [9] presents an approach for computing the semantic relatedness of resources in DBpedia. In the paper, they proposed a similarity measure based on a notion of proximity, which measures how connected two resources are, rather than how distant they are. This means that the similarity measure considers both distance and paths between two nodes. However, they do not consider incoming nodes (resources) and properties of the resources as *LDSD* did. Based on the *Shakti* measure, Strobin et al. [19] propose a method to find the weights automatically by using Genetic Algorithms (GA) [18] based on a training dataset from Last.fm[3]. This method is quite efficient at learning the weights automatically. However, it needs a gold standard dataset (e.g., Last.fm dataset for the music domain) to learn the weights of properties which is not always available in other domains. There are also some supervised learning approaches based on VSM model for LOD-enabled recommender systems. Tommaso et. al [3] adapt the VSM model to a LOD-based setting and represent the whole RDF graph as a matrix. On top of the VSM representation, they use the Support Vector Machine as a classifer to predict if a user would like an item or not. Using the same representation, they also propose to assign a weight to each property that represents its worth with respect to the user profile [4]. In this regard, they use GA to learn the weights of properties that minimize the misclassification errors. Our study is different from these model-based approaches since we focused on the top-N recommendation task while they focused on the prediction task. In addition, our proposed distance measures can be easily adopted in different domains since they do not require any learning process with a gold standard dataset. However, adaptive algorithms such as GA can also be applied to the semantic distance measures to learn the different weights of various paths (e.g., direct or indirect paths with different lengths). This, although interesting, is beyond the scope of this paper and we aim to explore it in future work.

[3]http://last.fm

3. LINKED DATA SEMANTIC DISTANCE MEASURES

LDSD was one of the first approaches for measuring the semantic distance between two resources on LOD datasets such as DBpedia and used for recommender systems [15]. The distance measure (equation (1)) considers direct links from resource r_a to resource r_b and vice versa. In addition, it also considers the same incoming and outgoing nodes via the same properties of resource r_a and r_b. The distance measure has a scale from 0 to 1, where a larger value denotes less similarity between two resources. We use the definition of a dataset following the Linked Data principles outlined in [14] and the definition of a path as below:

Definition 1. A dataset following the Linked Data principles is a graph G such as $G = (R, L)$ in which $R = \{r_1, r_2, ...r_n\}$ is a set of resources identified by their URI, $L = \{l_1, l_2, ...l_n\}$ is a set of typed links identified by their URI. A path is a sequence of resources and links between two resources, such as $p_i = \left[\ldots, l_{(i)j}, r_n, l_{(o)j}, \ldots\right]$. The direction of a link l_j in terms of the first resource can be represented by $l_{(i)j}$ (incoming) or $l_{(o)j}$ (outgoing).

For example, in the example graph (Fig. 1), we have paths such as $\left[l_{(o)associatedMusicArtist}\right]$, $\left[l_{(i)musicalguests}, r_{List\_of\_The\_Tonight\_Show\_with\_Jay\_Leno\_episodes\_(2013-14)}, l_{(o)musicalguests}\right]$ from the resource r_{Ariana_Grande} to r_{Selena_Gomez}.

LDSD consists of two C_d functions with $C_{ii}(l_i, r_a, r_b)$ and $C_{io}(l_i, r_a, r_b)$. C_d is a function that computes the number of direct and distinct links between resources in a graph G. $C_d(l_i, r_a, r_b)$ equals 1 if there is a link l_i from resource r_a to resource r_b. Otherwise, if there is no link from resource r_a to resource r_b, $C_d(l_i, r_a, r_b)$ equals to 0. By extension C_d can be the total number of nodes via l_i from r_a ($C_d(l_i, r_a)$). For example, in the example graph (Fig. 1), we have:

$C_d(l_{influences}, r_{Ariana\_Grande}, r_{Selena\_Gomez}) = 1$
$C_d(l_{influences}, r_{Ariana\_Grande}) = 1$
$C_d(l_{musicalguests}, r_{List\_of\_The\_Tonight\_Show\_with\_Jay\_Leno\_episodes\_(2013-14)}) = 2$

C_{ii} and C_{io} are functions that compute the number of indirect and distinct links, both incoming and outgoing, between resources in a graph G. $C_{ii}(l_i, r_a, r_b)$ equals 1 if there is a resource r_n linked to both r_a and r_b via an incoming property l_i, and 0 if not. Similarly, $C_{io}(l_i, r_a, r_b)$ equals 1 if there is a resource r_n linked to both r_a and r_b via an outgoing property l_i, and 0 if not. By extension C_{ii} and C_{io} can be used to compute the total number of resources linked indirectly to r_a via l_i ($C_{ii}(l_i, r_a)$ and $C_{io}(l_i, r_a)$). In the example (Fig. 1), we have $C_{ii}(l_{musicalguests}, r_{Ariana\_Grande}, r_{Selena\_Gomez}) = 1$ (via incoming property from $r_{List\_of\_The\_Tonight\_Show\_with\_Jay\_Leno\_episodes\_(2013-14)}$) and

$$LDSD(r_a, r_b) = \frac{1}{1 + \sum_i \frac{C_d(l_i, r_a, r_b)}{1+\log(C_d(l_i, r_a))} + \sum_i \frac{C_d(l_i, r_b, r_a)}{1+\log(C_d(l_i, r_b))} + \sum_i \frac{C_{ii}(l_i, r_a, r_b)}{1+\log(C_{ii}(l_i, r_a))} + \sum_i \frac{C_{io}(l_i, r_a, r_b)}{1+\log(C_{io}(l_i, r_a))}} \quad (1)$$

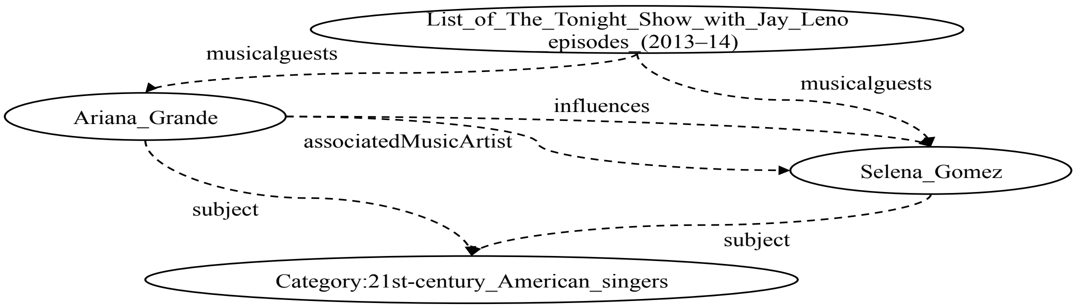


Figure 1: Example of relationships of two resources in DBpedia

$C_{io}(l_{subject}, r_{Ariana\_Grande}, r_{Selena\_Gomez}) = 1$ (via outgoing property to $r_{Category:21st-century\_American\_singers}$).

On top of the basic concept of *LDSD*, we propose four different distance measures incorporating various aspects of relatedness between two resources (Section 3.2-3.5).

3.1 Linked Data Semantic Distance incorporating the number of linked resources via a link

In this section, we introduce $LDSD_\alpha$ (equation (2)) that incorporates the number of resources linked to r_a and r_b via l_i. C'_{ii} (C'_{io}) of $LDSD_\alpha$, is equal to the number of resources linked to r_a and r_b via an incoming (outgoing) property l_i while C_{ii} (C_{io}) of *LDSD* equals 1 if there is a resource r_n linked to r_a and r_b via an incoming (outgoing) property l_i. The intuition behind this is that two resources are more similar if there are a greater number of linked resources via a property l_i. For instance, if two music artists have 10 `dbpedia:MusicalArtist`(s) in common via the property `dbpedia-owl:associatedMusicalArtist`, then they are more similar than two other music artists that have 1 `dbpedia:MusicalArtist` in common via the same property. The prefix `dbpedia` is used for the namespace `http://dbpedia.org/resource/` and the prefix `dbpedia-owl` is used for the namespace `http://dbpedia.org/ontology/`.

3.2 Linked Data Semantic Distance with normalizations considering both resources

In our previous study [16], we showed that *LDSD* with normalizations considering both resources can produce symmetric results for r_a and r_b and the reversed order of them, which is an important property as a distance measure [7]. On top of $LDSD_\alpha$, we further modify the normalizations of C'_{ii} and C'_{io} by considering both resources r_a and r_b (see equation (3)). That is, the normalization of C'_{ii} is carried out by the average of $C_{ii}(l_i, r_a)$ and $C_{ii}(l_i, r_b)$ while for C_{ii} in *LDSD* it is carried out by considering the first resource r_a only. Similarly, the normalization of C'_{io} is carried out by the average of $C_{io}(l_i, r_a)$ and $C_{io}(l_i, r_b)$. We use $LDSD_\beta$ to refer to the distance measure in the rest of the paper.

3.3 Linked Data Semantic Distance with global normalizations

These aforementioned distance measures use local normalizations, i.e., normalizations that are carried out in the local context of r_a and r_b. Instead of using local normalizations, we use global normalizations of a path to investigate the impact on calculating the distance between two resources. The distance measure can be defined as equation (4) and we use $LDSD_\gamma$ to refer to the distance measure in the rest of the paper.

This measure penalizes the importance of a path between two resources according to the global appearances of the path in the whole graph. In *LDSD*, the normalizations of $C_d(l_i, r_a, r_b)$ and $C_d(l_i, r_b, r_a)$ are carried out using $C_d(l_i, r_a)$ that computes the number of resources r_n from r_a via l_i

$$LDSD_\alpha(r_a, r_b) = \frac{1}{1 + \sum_i \frac{C_d(l_i, r_a, r_b)}{1+\log(C_d(l_i, r_a))} + \sum_i \frac{C_d(l_i, r_b, r_a)}{1+\log(C_d(l_i, r_b))} + \sum_i \frac{C'_{ii}(l_i, r_a, r_b)}{1+\log(C_{ii}(l_i, r_a))} + \sum_i \frac{C'_{io}(l_i, r_a, r_b)}{1+\log(C_{io}(l_i, r_a))}} \quad (2)$$

$$LDSD_\beta(r_a, r_b) = \frac{1}{1 + \sum_i \frac{C_d(l_i, r_a, r_b)}{1+\log(C_d(l_i, r_a))} + \sum_i \frac{C_d(l_i, r_b, r_a)}{1+\log(C_d(l_i, r_b))} + \sum_i \frac{C'_{ii}(l_i, r_a, r_b)}{1+\log(\frac{C_{ii}(l_i, r_a)+C_{ii}(l_i, r_b)}{2})} + \sum_i \frac{C'_{io}(l_i, r_a, r_b)}{1+\log(\frac{C_{io}(l_i, r_a)+C_{io}(l_i, r_b)}{2})}} \quad (3)$$

$$LDSD_\gamma(r_a, r_b) = \frac{1}{1 + \sum_i \frac{C_d(l_i, r_a, r_b)}{1+\log(C_{dp}(l_i))} + \sum_i \frac{C_d(l_i, r_b, r_a)}{1+\log(C_{dp}(l_i))} + \sum_i \sum_j \frac{C_{ii}(l_i, r_j, r_a, r_b)}{1+\log(C_{iip}(l_i, r_j))} + \sum_i \sum_j \frac{C_{io}(l_i, r_j, r_a, r_b)}{1+\log(C_{iop}(l_i, r_j))}} \tag{4}$$

$$LDSDLLR(r_a, r_b) = \frac{1}{1 + \sum_i L_d(l_i, r_a, r_b) + \sum_i L_d(l_i, r_b, r_a) + \sum_i \sum_j L_{ii}(l_i, r_j, r_a, r_b) + \sum_i \sum_j L_{io}(l_i, r_j, r_a, r_b)} \tag{5}$$

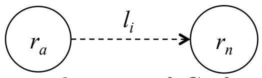


Figure 2: Local normalization of C_d function in equations (1), (2) and (3): the number of resources from r_a to r_n via l_i

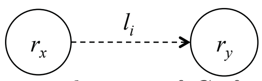


Figure 3: Global normalization of C_d function in equation (4): the number of appearances from r_x to r_y via l_i in a graph

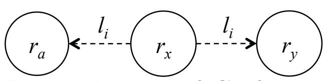


Figure 4: Local normalization of C_{ii} function in equations (1), (2) and (3): the number of resources linked to a resource via incoming property l_i as r_a

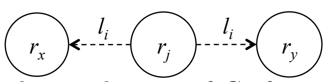


Figure 5: Global normalization of C_{ii} function in equation (4): the number of appearances of the path from r_x to r_y via the path $\left[l_{(i)i},\, r_j,\, l_{(o)i}\right]$

(see Fig. 2). In contrast, in $LDSD_\gamma$, the normalizations of C_d functions are carried out using $C_{dp}(l_i)$ that computes the global appearances of the path $\left[l_i\right]$ between any two resources in DBpedia (see Fig. 3). Furthermore, for indirect paths between two resources, $LDSD_\gamma$ normalizes each indirect path by the number of global appearances of it. Taking incoming indirect paths for example, $C_{ii}(l_i, r_j, r_a, r_b)$ equals 1 if there is a path $\left[\, l_{(i)i},\, r_j,\, l_{(o)i}\right]$ from r_a to r_b, and 0 if not. The normalization of $C_{ii}(l_i, r_j, r_a, r_b)$ is then carried out using $C_{iip}(l_i, r_j)$ that computes the global appearances of the path $\left[l_{(i)i},\, r_j,\, l_{(o)i}\right]$ between any two resources in DBpedia (see Fig. 5).

3.4 Linked Data Semantic Distance by log-likelihood

In this section, we adopt a statistical approach, *log-likelihood ratio* [5] for measuring the semantic distance between two resources. The approach is adopted and implemented as a similarity metric named *log-likelihood similarity* in the *Apache Mahout*[4], which is a widely known open source project to build an environment for creating machine learning applications with scalable performance as well as collaborative filtering based recommender systems. We adopt the implementation of the similarity in Mahout for our experiment.

[4]http://mahout.apache.org/

The *log-likelihood similarity* is good at handling rare events. In our approach, we treat a resource and direct/indirect paths of it appearing in a graph as an event. The distance measure can be defined as equation (5) and we use $LDSDLLR$ to refer to this measure in the rest of the paper. $L_d(l_i, r_a, r_b)$ denotes the *log-likelihood ratio* of two events [5]: an event for the resource r_a with the path $[l_{(o)i}]$ and the other event for the resource r_b with the path $[l_{(i)i}]$. Similarly, $L_{ii}(l_i, r_j, r_a, r_b)$ denotes the *log-likelihood ratio* of two events: the resource r_a with the incoming link l_i from a resource r_j, and the resource r_b with the incoming link l_i from the resource r_j.

4. EVALUATION SETUP

We evaluate the proposed semantic distance measures in terms of the performance of recommender systems when using them as recommendation algorithms. To this end, we use a subset of the dataset from the second Linked Open Data-enabled recommender systems challenge[5]. The dataset was collected from Facebook[6] profiles about personal preferences ("likes") in the music domain, which consists of 52,072 users and 21 liked items on average. The items available in the dataset have been mapped to their corresponding DBpedia URIs. We randomly select 500 users with 10,590 preference records for the experiment. The main details of the dataset from the challenge and its subset for our experiment are presented in Table 1.

For each user, 5 liked items were blinded out to construct a candidate list for recommendations while the rest of the items were used to construct preference profiles of users. In the end, the candidate list consists of 1,132 items of type `dbpedia:MusicalArtist` or `dbpedia:MusicalBand`.

A user profile can be represented as a set of resources that they liked before (equation (6)). The similarity between a user u_i and an item can be measured using equation (7) where $dist(r_a, r_b)$ denotes the distance measure deployed in the recommender system.

$$Profile(u_i) = \{r_1, r_2, ..., r_m\} \tag{6}$$

[5]http://2015.eswc-conferences.org/important-dates/call-RecSys

[6]https://www.facebook.com/

Table 1: Descriptive statistics of the dataset

Dataset	# of users	# of liked items			
		min.	max.	avg.	std.
Total	52,072	15	37	21	6.20
Subset	500	15	37	21	6.18

$$sim(u_i, r_a) = \frac{\sum_{r_b \in Profile(u_i)} (1 - dist(r_a, r_b))}{|Profile(u_i)|} \quad (7)$$

The recommender system then provides the top-N recommendations based on the similarity of a user profile and an item from the candidate list. We evaluate the performance of the recommender system using our proposed distance measures and taking *LDSD* as a baseline. Additionally, a model-based approach [3] for RDF data with VSM [17] is adopted as another baseline. The model considers all properties of items (resources) that a user liked, where each resource is represented as a unique vector of weights and each weight indicates the degree of association between the item and the resource with respect to a property. In this regard, both items and users can be represented in terms of the VSM. We then use the *cosine similarity* measure for measuring the similarities between a user and an item. Finally, we add an item-based collaborative filtering (*ITEMCF*) approach with *log-likelihood similarity* in *Mahout* as another baseline which does not exploit LOD.

The performance of the recommender system was evaluated by standard evaluation methods for top-N ($N = 1, 5, 10, 20$) recommendation tasks: recall at N (*R@N*), precision at N (*P@N*) and Mean Reciprocal Rank (*MRR*). *R@N* is the fraction of items that are relevant to the users that are successfully retrieved in the top-N recommendations, and *P@N* is the fraction of the top-N recommended items that are relevant to the user. *MRR* indicates at which rank the first item relevant to the user occurs on average.

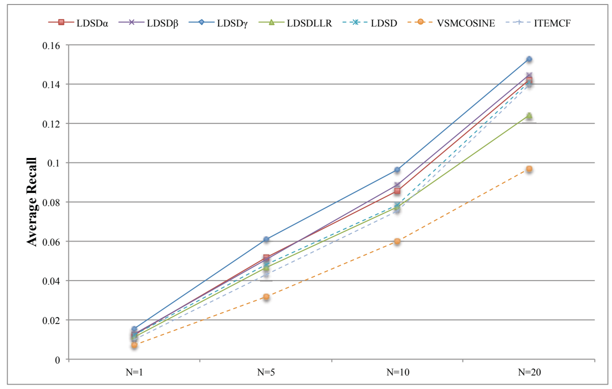
DBpedia provides a large set of properties for each item. Hence, selecting a subset of domain-dependent properties is necessary [4, 12]. Referring to the music domain, in which we performed the evaluation, we selected the 15 properties (see Table 2) for calculating the semantic distance between resources. `dct:subject` relates a resource to its categories. In addition, we decided to leverage the properties belonging to the DBpedia Ontology since they represent high-quality, clean and well-structured information [13]. The prefix `dct` is used for the namespace `http://purl.org/dc/terms/subject`.

5. RESULTS

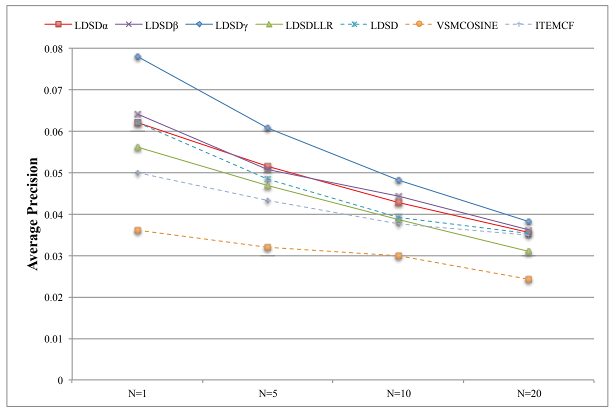
Figure 6(a) and 6(b) show the *R@N* and *P@N* results of the recommender system with the proposed distance measures

Table 2: Properties Selected for the Music Domain

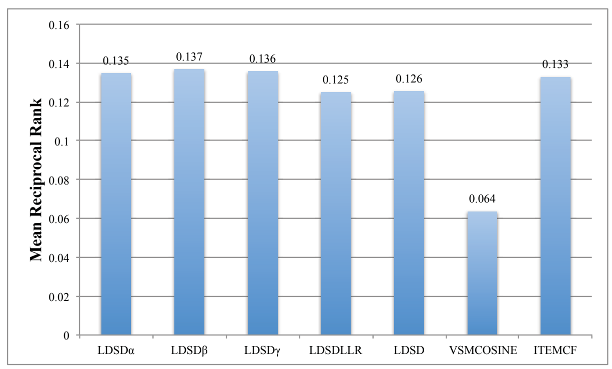
`dct:subject, dbpedia-owl:genre, dbpedia-owl:associatedBand, dbpedia-owl:associatedMusicalArtist, dbpedia-owl:instrument, dbpedia-owl:formerBandMember, dbpedia-owl:currentMember, dbpedia-owl:influencedBy, dbpedia-owl:pastMember, dbpedia-owl:associatedAct, dbpedia-owl:influenced, dbpedia-owl:recordLabel, dbpedia-owl:occupation, dbpedia-owl:hometown, dbpedia-owl:bandMember`



(a) Recall



(b) Precision



(c) MRR

Figure 6: Results of the recommender system using different distance measures

(solid lines) and baseline methods (dashed lines). Figure 6(c) shows the MRR results of the recommender system. The *paired t-test* is used for testing the significance where the significance level was set to 0.05 unless otherwise noted.

Overall, $LDSD_\gamma$ performs best, which considers global appearances of paths within the whole graph, followed by $LDSD_\beta$ and $LDSD_\alpha$. More specifically, the *R@5, 10* have improved significantly ($p < 0.01$), as well as *R@20* using $LDSD_\gamma$ compared to using *LDSD*. Also, the *R@10* has improved significantly by using $LDSD_\beta$ compared to using

LDSD. Similar observations can be found in the precision results. *LDSDLLR* performs worst among these proposed distance measures, and worse than *LDSD*. All of the proposed distance measures perform significantly better than the VSM model with *cosine similarity* measure. In addition, we can observe the improvements by exploiting LOD based on the recall and precision results compared to using the *ITEMCF* approach.

In terms of *MRR* (see Figure 6(c)), $LDSD_\alpha$, $LDSD_\beta$ and $LDSD_\gamma$ have similar performance and the performance is significantly better than the VSM model with *cosine similarity* measure and slightly better than *LDSD* and *ITEMCF* (but not significant).

The results show that incorporating the number of linked resources and adopting different normalization strategies such as local normalizations by considering both resources ($LDSD_\beta$), and the global normalizations of paths ($LDSD_\gamma$) can improve the performance of the recommender system. In addition, the best performance achieved by $LDSD_\gamma$ indicates that global normalizations of paths represent the importance of paths better than local ones of them.

6. CONCLUSION AND FUTURE WORK

In this paper, we proposed and investigated various semantic distance measures for calculating the distance between resources. These measures were adopted into a LOD-based recommender system to recommend the top-N items to users based on the preference of a user. We found that the performance of the recommender system is significantly improved by incorporating the number of resources via indirect paths, local normalizations considering both resources ($LDSD_\beta$) and global appearances of paths ($LDSD_\gamma$) compared to baseline methods.

In the future, we plan to extend the algorithms by incorporating longer paths, in order to recommend items that are more than one node away from the seed one. Moreover, paths with different lengths would contribute to the semantic distance in a different way and the weights of them can be optimized by using GA as we mentioned in Section 2.

7. ACKNOWLEDGMENTS

This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under Grant Number SFI/12/RC/2289 (Insight Centre for Data Analytics).

8. REFERENCES

[1] D. Brickley and R. V. Guha. {RDF vocabulary description language 1.0: RDF schema}. 2004.

[2] T. Di Noia, I. Cantador, and V. C. Ostuni. Linked open data-enabled recommender systems: ESWC 2014 challenge on book recommendation. In *Semantic Web Evaluation Challenge*, pages 129–143. Springer, 2014.

[3] T. Di Noia, R. Mirizzi, V. C. Ostuni, and D. Romito. Exploiting the web of data in model-based recommender systems. In *Proceedings of the sixth ACM conference on Recommender systems*, pages 253–256. ACM, 2012.

[4] T. Di Noia, R. Mirizzi, V. C. Ostuni, D. Romito, and M. Zanker. Linked Open Data to Support Content-based Recommender Systems. In *Proceedings of the 8th International Conference on Semantic Systems*, I-SEMANTICS '12, pages 1–8, New York, NY, USA, 2012. ACM.

[5] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational linguistics*, 19(1):61–74, 1993.

[6] V. Groues, Y. Naudet, and O. Kao. Adaptation and evaluation of a semantic similarity measure for dbpedia: A first experiment. In *Semantic and Social Media Adaptation and Personalization (SMAP)*, pages 87–91. IEEE, 2012.

[7] S. Harispe, S. Ranwez, S. Janaqi, and J. Montmain. Semantic Measures for the Comparison of Units of Language, Concepts or Instances from Text and Knowledge Base Analysis. *arXiv preprint arXiv:1310.1285*, 2013.

[8] B. Heitmann and C. Hayes. Using Linked Data to Build Open, Collaborative Recommender Systems. In *AAAI spring symposium: linked data meets artificial intelligence*, pages 76–81, 2010.

[9] J. P. Leal, V. Rodrigues, and R. Queirós. Computing semantic relatedness using dbpedia. 2012.

[10] A. Maedche and V. Zacharias. Clustering ontology-based metadata in the semantic web. In *Principles of Data Mining and Knowledge Discovery*, pages 348–360. Springer, 2002.

[11] S. E. Middleton, D. De Roure, and N. R. Shadbolt. Ontology-based recommender systems. In *Handbook on ontologies*, pages 779–796. Springer, 2009.

[12] C. Musto, P. Basile, P. Lops, M. de Gemmis, and G. Semeraro. Linked Open Data-enabled Strategies for Top-N Recommendations. In *CBRecSys*, page 49, 2014.

[13] V. C. Ostuni, T. Di Noia, E. Di Sciascio, and R. Mirizzi. Top-n recommendations from implicit feedback leveraging linked open data. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 85–92. ACM, 2013.

[14] A. Passant. dbrec: Music Recommendations Using DBpedia. In *ISWC 2010 SE - 14*, pages 209–224, 2010.

[15] A. Passant. Measuring Semantic Distance on Linking Data and Using it for Resources Recommendations. In *AAAI Spring Symposium: Linked Data Meets Artificial Intelligence*, volume 77, page 123, 2010.

[16] G. Piao, S. showkat Ara, and J. G. Breslin. Computing the Semantic Similarity of Resources in DBpedia for Recommendation Purposes. In *Semantic Technology*. Springer International Publishing, 2015.

[17] G. Salton, A. Wong, and C.-S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[18] M. Srinivas and L. M. Patnaik. Genetic algorithms: A survey. *Computer*, 27(6):17–26, 1994.

[19] L. Strobin and A. Niewiadomski. Evaluating semantic similarity with a new method of path analysis in RDF using genetic algorithms. *COMPUTER SCIENCE*, 21(2):137–152, 2013.