Crowdsourced Affinity: A Matter of Fact or Experience

Chun Lu[1,2(✉)], Milan Stankovic[1,2], Filip Radulovic[1], and Philippe Laublet[2]

[1] Sépage, 27 rue du chemin vert, 75011 Paris, France
{chun,milstan,filip}@sepage.fr

[2] STIH, Université Paris-Sorbonne, 28 rue Serpente, 75006 Paris, France
philippe.laublet@paris-sorbonne.fr

Abstract. User-entity affinity is an essential component of many user-centric information systems such as online advertising, exploratory search, recommender system etc. The affinity is often assessed by analysing the interactions between users and entities within a data space. Among different affinity assessment techniques, content-based ones hypothesize that users have higher affinity with entities similar to the ones with which they had positive interactions in the past. Knowledge graph and folksonomy are respectively the milestones of Semantic Web and Social Web. Despite their shared crowdsourcing trait (not necessarily all knowledge graphs but some major large-scale ones), the encoded data are different in nature and structure. Knowledge graph encodes factual data with a formal ontology. Folksonomy encodes experience data with a loose structure. Many efforts have been made to make sense of folksonomy and to structure the community knowledge inside. Both data spaces allow to compute similarity between entities which can thereafter be used to calculate user-entity affinity. In this paper, we are interested in observing their comparative performance in the affinity assessment task. To this end, we carried out a first experiment within a travel destination recommendation scenario on a gold standard dataset. Our main findings are that knowledge graph helps to assess more accurately the affinity but folksonomy helps to increase the diversity and the novelty. This interesting complementarity motivated us to develop a semantic affinity framework to harvest the benefits of both data spaces. A second experiment with real users showed the utility of the proposed framework and confirmed our findings.

Keywords: Crowdsourcing · Affinity · Similarity · Semantic · Knowledge graph · Folksonomy · Travel · e-tourism · Semantic affinity framework · Diversity · Novelty

1 Introduction

User-entity affinity is the likelihood of a user to be attracted by an entity or to perform an action (click, purchase, like, share) related to an entity. The entity can be book, film, artist etc. User-entity affinity has a big impact from both economic and user experience point of view. It is an essential component of many user-centric information systems such as online advertising systems, exploratory search systems and recommender systems. It is crucial for predicting the click-through rate which is central to the multi-billion-dollar online advertising industry [1]. In exploratory search systems, it is

E. Blomqvist et al. (Eds.): ESWC 2017, Part I, LNCS 10249, pp. 554–570, 2017.
DOI: 10.1007/978-3-319-58068-5_34

leveraged to retrieve interesting entities that might satisfy a user's fuzzy intention [2]. It is proper to recommender systems which are designed to mitigate the information overload by suggesting entities in affinity with the user.

Among different common affinity assessment techniques [3, 4], content-based ones [5] hypothesize that users would have higher affinity with entities that are similar to the ones with which they had positive interactions in the past. The emergence of knowledge graph and folksonomy has boosted this family of techniques by providing a large amount of data about entities [6].

Knowledge graph and folksonomy are respectively the milestones of Semantic Web and Social Web. On the Semantic Web, people contribute to the creation of large public knowledge graphs like DBpedia[1] and Wikidata[2] [7]. On the Social Web, people annotate and categorize entities with freely chosen texts called tags which form the folksonomy. Despite their shared crowdsourcing trait (not necessarily all knowledge graphs but some major large-scale ones above-mentioned), the encoded data are different in nature and structure. Knowledge graph encodes factual data with a formal ontology. Folksonomy encodes experience data with a loose structure. We give a concrete example to illustrate their difference. On DBpedia, the film dbr:Jumanji is linked to facts like dbr:Joe_Johnston by the property dbo:director and to dbr:Robin_Williams by the property dbo:starring. In the folksonomy of users of MovieLens[3], the same film is abundantly tagged with "nostalgic", "not funny", "natural disaster" etc. Even though these folksonomy tags are less formally structured, they reflect the experience that different users had with the film and thus a sort of intersubjectivity which is lacking in factual knowledge graph.

After in-depth study of the literature (Sect. 2), we struggled to find helpful insights about which data space is more effective in affinity-based systems. While both data spaces continue to proliferate on the web (Twitter hashtags, Instagram, Flickr, Mendeley) [8], it is more necessary than ever to shed some light on their comparative performance and contribution to the user-entity affinity assessment.

We conducted a first experiment within a travel destination recommendation scenario (Sect. 3). The findings motivated us to develop a semantic affinity framework which harvests the benefits of both Social Web and Semantic Web (Sect. 4). We used the proposed framework to compute a travel affinity graph which was evaluated in a second experiment with real users (Sect. 5). Section 6 concludes the paper with some advice for future development of affinity-based systems.

2 Related Work

For the past ten years, researchers have been closely studying the relatedness between knowledge graph and folksonomy. They are respectively the milestones of Semantic Web and Social Web. The general idea behind many research efforts is to enhance semantics in the Social Web with the help of Semantic Web technologies [9]. In the case

[1] http://wiki.dbpedia.org/about.
[2] https://www.wikidata.org/.
[3] https://movielens.org/.

of folksonomy, semantics are leveraged to (1) guide and control the tagging process (2) make sense of folksonomy. In [10], the authors proposed the MOAT ontology and a collaborative framework to guide tagging system users to specify the meaning of a tag with an existing resource on the Semantic Web. In [11], the author stated that although the semantics is much more implicit in folksonomy, the collective actions of a large number of individuals can still lead to the emergence of semantics. He suggested building lightweight ontologies from folksonomies. In [12], the authors used Semantic Web and natural language processing techniques to automatically classify folksonomy tags to four categories based on the intention: content-based, context-based, subjective, organisational. Some authors tried to ground folksonomy tags semantically by mapping pairs of tags in Del.icio.us to pairs of synsets in WordNet. Then WordNet-based measures for semantic distance are used to derive semantic relations between the mapped tags [13].

Some authors tried to mine user interests from folksonomies and other platforms of the Social Web. [14] observed that users have multiple profiles in various folksonomies and they proposed a method for the automatic consolidation of user profiles across different social platforms. Wikipedia categories was used to represent user interests. A similar method was proposed in [15] for the automatic creation and aggregation of interoperable and multi-domain user interest profile. Instead of using Wikipedia categories, user interests are represented with semantic concepts. The paper [16] also studied the multi-folksonomies problem. The authors found that the overlap between different tag-based profiles from different tagging systems is small and aggregating tag-based profiles lead to significantly more information about individual users, which impacts positively the personalized recommendations. Some recent work [17, 18] focused on one particular social platform: Twitter. They extract semantic concepts from tweets. These concepts are then enriched within the Wikipedia categorization system [17] or within a whole knowledge graph with multi-strategy enrichment [18].

Some authors studied the advantage of using folksonomies in some recommendation scenarios. The authors of [19] argued that although folksonomies provide structures that are formally weak or unmotivated, they are strongly connected with the actual use of the terms in them and the resources they describe. They may provide data about the perceptions of users, which is what counts in the recommendation context. The authors showed the advantage of folksonomies over using keywords in a movie recommendation scenario. Another study [20] in the cultural heritage domain showed that using both static official descriptions of items and user-generated tags about items allows to increase the precision of recommendations than using solely one of them.

User data mined from folksonomies are certainly useful for understanding the user and finding entities in affinity with him/her. But these data are not always easy to acquire for a system which is not built within social platforms. Il requires the system to ask users to log in with their social accounts or to purchase user data from some data management platforms. In this paper, we are not interested in social tagging actions of one particular user but the community knowledge generated by all users' tagging actions.

In [21], a class of applications called collective knowledge systems was proposed and was aimed at unlocking the collective intelligence of the Social Web with knowledge representation and reasoning techniques of the Semantic Web. On the location-based social network Foursquare, the venue similarity partially relies on the "taste" similarity[4]. Their taste map[5] was launched on 2014. It contains more than 10,000 short descriptive tags for restaurants sourced from 55 million tips[6]. Tastes can be as simple as a favourite dish like "soup dumplings" or a vibe like "good for dates". This collective knowledge is useful to characterize different venues and calculate their similarity. In [22], the authors proposed a data structure named "tag genome" which extends a traditional tagging model. It records how strongly each tag applies to each item on a continuous scale. It encodes each item based on its relationship to a common set of tags. As a case study, a tag genome is computed for MovieLens[7]. Since the tag genome is a vector space model, the entity similarity can be calculated with measures like cosine similarity.

In the literature about the semantic similarity in knowledge graph, we can find papers in two general directions: (1) similarity between classes [23] (2) similarity between entities. In this paper, we are interested in the latter direction.

In [24–28], different algorithms were proposed to compute the semantic similarity between entities by exploring the semantic properties. In [24], the author proposed an algorithm named Linked Data Semantic Distance (LDSD) which calculates the dissimilar degree of two resources in a semantic dataset such as DBpedia. In [25], the authors proposed an algorithm which is built on the top of LDSD. The modification consists of incorporating normalizations that use both resources and global appearances of paths. In [26], the authors proposed to use a vector space model to compute the similarity between RDF resources. It calculates a similarity score between two entities at the same property then sum all property-level similarity scores. These papers consider only three types of properties between two entities (1) direct property (2) outbound property pointing to the same entity (3) inbound property pointing from the same entity. The paper [27] proposed the *SPrank* algorithm to compute top-N item recommendations exploiting the information available on Linked Open Data. It consists of exploring paths in a semantic graph to find items that are related to the ones the user is interested in. A supervised learning to rank approach is leveraged to find what paths are most relevant for the recommendation task. A recent approach named RDF2Vec learns latent numerical representations of entities in RDF graphs [28].

Some variants of the spreading activation algorithm were presented in [29–31]. In [29], the algorithm aims to make cross-domain recommendation in DBpedia. [30] is in the same direction and the authors gave a concrete example of recommending places of interest from music artists. In [31], the algorithm is used to support an exploratory search system. The system retrieves similar/related entities to the ones initially entered by the user.

After in-depth study of the literature, we did not find any study comparing how the entity similarity calculated with knowledge graph and folksonomy performs in the user-entity affinity assessment task. We also did not find any approach with clear instructions

[4] http://engineering.foursquare.com/2015/12/08/finding-similar-venues-in-foursquare/.
[5] http://engineering.foursquare.com/2014/10/21/exploring-the-foursquare-taste-map/.
[6] Statistics reported in this article: http://www.theverge.com/2014/8/6/5973627/foursquare-8-review-the-ultimate-food-finder.
[7] http://www.movielens.org.

on how to tackle the affinity challenge in modern-day e-commerce systems such as e-tourism systems. This paper represents an initiative towards shedding light on these issues. We hope that the findings of our study can guide future design and development of affinity-based systems.

3 First Experiment: Gold Standard Study

We conducted an experiment within a travel destination recommendation scenario. It is a real and important problem. Web is today one of the most important sources for travel inspiration and purchase. More than 80% of people do travel planning online[8]. However, travelers feel bogged down by the myriad of options[9]. They feel overwhelmed and are obliged to spend a lot of time browsing multiple websites[10] before booking a trip. 68% of travelers begin searching online without having a clear travel destination in mind. Recommender systems can help travelers find more efficiently destinations in affinity with them. There is no consensual definition of travel destination, it can be a village, a city, a region or a country. In this paper, we consider cities as travel destinations. We use both terms "city" and "travel destination" interchangeably. In this section, we present an experiment within the travel destination recommendation scenario. We compare two representative approaches of knowledge graph and folksonomy within a gold standard dataset.

3.1 Experiment Dataset

To the best of our knowledge, there is no publicly available and widely used dataset for the evaluation of travel destination recommender systems. Thus, we use a dataset of our recent previous work [32]. It is constructed from Yahoo! Flickr Creative Commons 100 M (YFCC100 M) dataset[11] [33]. The original dataset contains 100 million of geotagged photos and videos published on Flickr. We processed the original dataset to make it suitable for our use case. Firstly, we took the file "yfcc100m_dataset". We filtered all the lines where latitude and longitude data were missing and where the accuracy level was below 16 (the highest accuracy level in Flickr). In other words, we retained only geotagged photos and videos with the highest geo-location accuracy. Secondly, we mapped each photo/video to a travel attraction entity in a travel knowledge graph constructed during that work. In the travel knowledge graph, travel attraction entities are linked to their city entities. So, once the mapping is done, we know the cities users have been to. We eliminated users who have been to only one city because in our evaluation, we need at least one city as user profile and another city as ground truth. Finally, for each user, we sorted the visited cities in a chronological order by considering the

[8] The 2013 Traveler: http://www.thinkwithgoogle.com//research-studies/2013-traveler.html.
[9] Reaching the connected customer: http://info.boxever.com/reaching-the-connected-customer.
[10] Custom Research: Exploring the Traveler's Path to Purchase: https://info.advertising.expedia.com/travelerspathtopurchase.
[11] Yahoo Webscope: http://webscope.sandbox.yahoo.com.

dates photos/videos were taken. After the processing, we know the travel sequence of each user. A travel sequence is a list of cities that a user visited in a chronological order. For example, the travel sequence "dbr:Munich, dbr:Stockholm, dbr:New_York_City" means that the user has visited respectively Munich, Stockholm and New York City. Table 1 shows the statistics about the experiment dataset.

Table 1. Statistics about the experiment dataset

# users	3878
# cities	705
avg# cities per user	5.27

The dataset is published for future benchmarking and reproducibility[12]. In the dataset, some users have the same sequence. We intentionally retained these seemly duplicated records because in a real-world scenario, some sequences are more frequent than others. A system which can produce high quality recommendations based on these sequences should be somehow "rewarded" or on the contrary "penalized".

3.2 Folksonomy Engineering

For the folksonomy part, we crawled data from the website of a collaborative travel platform where users are invited to tag cities after their trips there. They are restricted to use existing tags such as "Kayaking", "Great for wine", "People watching". The crawled dataset contains 234 tags about 26,237 cities in 154 countries. To give readers a clearer idea about the dataset, in Fig. 1, we show the distribution of tags in terms of the number of times they are applied (Applications) and the number of cities on which they are applied (Cities). However, due to the space limit and for a better readability,

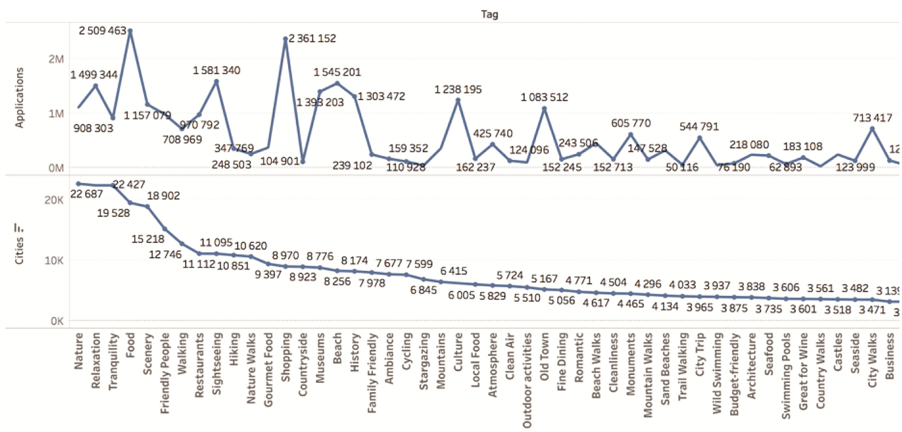

Fig. 1. Distribution of folksonomy tags in the travel domain

12 https://bitbucket.org/sepage/semantic-affinity-framework.

we can only show a part of the distribution. We modeled this folksonomy dataset in a tag genome fashion [22]. The tag relevance score is calculated with the Term Frequency-Inverse Document Frequency scheme. In our case, terms are tags and documents are cities. As in [22], we compute the similarity between cities by calculating the cosine of the angle between their vectors.

3.3 Knowledge Graph Engineering

For the knowledge graph part, we manually selected some inbound and outbound properties shown in Table 2. We put *skos:broader* in brackets because it is not directly linked to cities but indirectly linked via *dct:subject*.

For each of the 705 cities in the evaluation dataset, we ran SPARQL queries with all selected properties. We gave a special treatment to the property *dct:subject*. For each retrieved direct linked category, we also retrieved its parent categories by using *skos:broader*. We put all direct and parent categories together, deduplicated the list and put it under the property *dct:subject*. Then we eliminated nodes which are linked to only one city because they do not contribute to the similarity calculation between two cities. 501365 nodes were initially retrieved. After the cleaning, only 29743 nodes were retained.

Table 2. Selected properties for calculating city similarity in knowledge graph

Inbound		Outbound	
dbo:birthPlace	dbo:broadcastArea	dbo:isPartOf	dbo:part
dbo:location	dbo:nearestCity	dbo:country	dbo:twinTown
dbo:deathPlace	dbo:ground	dbo:timeZone	dbo:saint
dbo:city	dbo:foundationPlace	dbo:Mayor	dbo:district
dbo:capital	dbo:assembly	dbo:region	dct:subject
dbo:hometown	dbo:restingPlace	dbo:province	(skos:broader)
dbo:recordedIn	dbo:place	dbo:leaderName	
dbo:residence	dbo:locationCity		
dbo:headquarter			

We adopted a simple-to-implement and low-computational-cost similarity measure: Jaccard index. It has been thoroughly evaluated and compared with three other more sophisticated similarity measures in [34]. It has been proved to produce highly accurate recommendations. The features of an entity are modeled as a set of nodes in its surroundings. For two entities e_1 and e_2, we do graph walking to collect their surrounding nodes at a specific distance d: $N_d(e_1)$ and $N_d(e_2)$.

$$J(e_1, e_2) = \frac{|N_d(e_1) \cap N_d(e_2)|}{|N_d(e_1) \cup N_d(e_2)|} \quad (1)$$

Our knowledge graph is modelled in such a way that we only need to set d to 1 to get all interesting nodes. With this measure, we can easily consider two interesting graph patterns in Fig. 2. For example, GP1 allows to capture that a person was born (dbo:birthPlace) in a city and resides (dbo:residence) in another city. GP2 allows to capture entities linked to different direct categories which have a common broader category, for example:

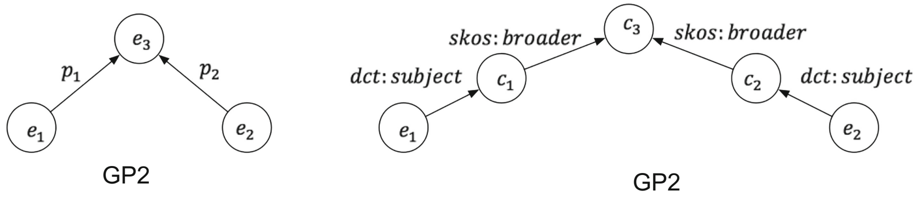

Fig. 2. Examples of two interesting graph patterns exploited by our similarity measure

dbr:Pushkar → *dbc:Hindu_holy_cities* → *dbc:Holy_cities* ← *dbc:Bethlehem* ← *dbr:Bethlehem*

dbr:New_Delhi → *dbc:Capitals_in_Asia* → *dbc:Capitals* ← *dbc:Capitals_in_Europe* ← *dbr:Athens*

One-hop category has been proven to be useful in many personalization tasks [18, 26].

3.4 Candidate Approaches

Since our experiment aims to compare the performance of knowledge graph and folksonomy on user-city affinity assessment, we implemented two candidate approaches: FOLK and KG which use respectively the data and techniques presented in Sects. 3.2 and 3.3.

3.5 Common Affinity Prediction Algorithm

To ensure a fair environment for comparing them, we used a common affinity prediction algorithm. This assessment methodology was adopted in a comparative study on knowledge graph and similarity measure choices [34]. Given a user profile $profile(u)$ containing a list of cities that the user u has visited in the past, the affinity score of a candidate city c_i is calculated with Eq. 2, which is the sum of pairwise similarity with each city in the user profile divided by the total number of cities in the user profile. The pairwise similarity $Sim(c_i, c_j)$ is calculated by the candidate approaches and feeds the common affinity prediction algorithm. The affinity score is only influenced by the similarity score calculated by the candidate approaches.

$$affinity(u, c_i) = \frac{\sum_{c_j \in profile(u)} Sim(c_i, c_j)}{|profile(u)|} \tag{2}$$

3.6 Protocol

We use the "all but n" protocol. It is aligned with the common practice of offline experiments in the recommender system community [5, 35, 36]. For each user in the evaluation datasets, we split his/her cities into two parts: profile and ground truth. In the travel domain, the user history is much poorer than some other domains like music and movie. In our dataset, the number of visited cities range from 2 to 112 (the vice champion user has 64 cities), in average, each user has visited 5.27 cities. We adopted the "all but 1" strategy. For a travel sequence containing *n* cities, the first *n−1* cities constitute the profile and the *n-th* city constitutes the ground truth. There is no standard about the number of recommendations that should be computed. We made a choice based on our past research work [37, 38], experience with the clients of Sépage company and practices on several popular travel websites (expedia, tripadvisor, kayak etc.). The number of recommendations depends on different contexts. In a recommendation or an advertising banner, the number is relatively limited, in an inspirational browsing environment, more cities are displayed. For these reasons, we decided to compute top-10, top-20 and top-30 recommendations.

3.7 Quality Dimensions and Metrics

In the recommendation scenario, the user-entity affinity assessment capacity can be most reflected by the accuracy of the recommendations. To measure the accuracy, we used two metrics: Success and Mean Reciprocal Rank (MRR).

$$Success = \frac{\sum_{u \in U} rel_{g,u}}{|U|} \text{ where } rel_{g,u} = \begin{cases} 1, & if\ ground\ truth\ g\ is\ in\ top - N \\ & 0, otherwise \end{cases} \tag{3}$$

The Success metric (Eq. 3) calculates the number of users for whom the candidate approaches recommend cities that are in affinity with the users divided by the total number of users. It is an alternative metric to classic precision and recall which are not perfectly adapted in our case. Because each user's ground truth contains only one city. The precision and recall of each user are binary values. 1/N or 0 for the precision, 1 or 0 for the recall. For this reason, it would be more intuitive to compare the number of users for whom the system can actually recommend the ground truth.

The Mean Reciprocal Rank is calculated as in Eq. 4.

$$MRR = \frac{1}{|U|} \sum_{u \in U} \frac{1}{rank_u} \tag{4}$$

where $rank_u$ is the rank position of the ground truth of the user *u*. It shows how early the ground truth appears in the recommendation list. A higher MRR reveals a better capacity of a candidate approach to detect affinity cities. It is crucial if we can only recommend a very limited number of cities.

Currently, the recommender system community has a growing interest in generating diverse and novel recommendations, even at the expense of the accuracy. Apart from our main focus, we are also interested in knowing how our approaches perform on these two quality dimensions.

In ESWC 2014 Challenge on Linked Open Data-enabled Book Recommendation [35], the diversity was considered with respect to two properties in DBpedia: *dbo:author* and *dct:subject*. In our case, we considered the diversity with respect to *dbo:country* and *dct:subject*. The Eqs. 5 and 6 measure the intra-list similarity (ILS)

$$ILS_u@N = \sum_{i \in L_u^N} \sum_{j \in L_u^N} \frac{sim(i,j)}{|pairs|} \tag{5}$$

$$ILS@N = \frac{1}{|U|} \sum_{u \in U} ILS_u@N \tag{6}$$

where $sim(i,j)$ is the aggregated similarity score with respect to the two properties. We give equal importance (0.5) to them in this calculation. The higher ILS is, the less diverse the recommendation list is.

We calculate the novelty with respect to the capacity of recommending long-tail cities. Following the power law distribution[13], we consider the 80% less popular cities as long-tail cities. We use the DBpedia pagerank[14] value as the popularity index.

$$Novelty@N = \frac{number\ of\ recommended\ long - tail\ cities}{N * |U|} \tag{7}$$

3.8 Results and Discussions

In Table 3, we show the scores of the two approaches on four metrics when top-10, top-20 and top-30 recommendations are computed. Paired t-tests show that the differences between the two approaches among all metrics in all settings are statistically significant with $p < .01$.

Table 3. Scores of two candidate approaches on four metrics when top-10, top-20 and top-30 recommendations are computed

	Top-10		Top-20		Top-30	
	KG	FOLK	KG	FOLK	KG	FOLK
Success	**0.232**	0.06	**0.33**	0.116	**0.386**	0.166
MRR	**0.047**	0.003	**0.047**	0.003	**0.047**	0.003
ILS	0.257	**0.089**	0.208	**0.072**	0.176	**0.065**
Novelty	0.717	**0.824**	0.722	**0.772**	0.723	**0.755**

We can observe a net advantage of KG over FOLK in terms of success and MRR. Higher scores on success and MRR reflect the capacity of a system to detect cities in high affinity with the user and to give them better rankings. Recommendations produced

[13] https://en.wikipedia.org/wiki/Power_law.
[14] http://people.aifb.kit.edu/ath/.

by FOLK are generally more diverse and novel than those produced by KG. Actually, in the folksonomy, we ignore some aspects that are considered by DBpedia such as the geography (*dbo:country*, *dbo:region*), the people (*dbo:birthPlace, dbo:residence)*, the related categories (*dct:subject, skos:broader)*. The folksonomy contains travel-related traits like “Luxury Brand Shopping”, “Clean Air”, “Traditional food”. These traits can be shared by different cities in the world and by less popular cities.

The very different performances of the two approaches on different quality dimensions led to us to believe in their complementarity in obtaining a balanced trade-off and in yielding recommendations that are equitably accurate, diverse and novel.

4 Semantic Affinity Framework

Motivated by the findings of our first experiment, we propose a Semantic Affinity Framework which harvests the benefits of both Semantic Web and Social Web. It is designed for user-centric information systems which aim to provide personalised user experience by leveraging the affinity. The framework integrates, aggregates, enriches and cleans entity (here we refer to main objects of the system, e.g., book, film, city) data from knowledge graphs and folksonomies. The Fig. 3 shows the pipeline of the framework.

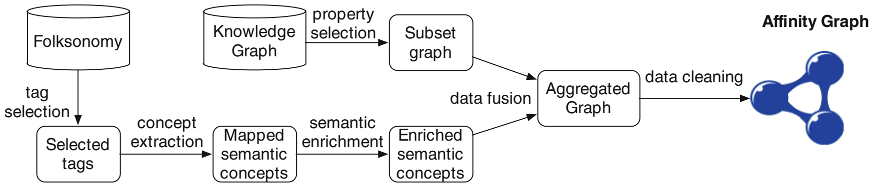

Fig. 3. Pipeline of semantic affinity framework, from folksonomy and knowledge graph to affinity graph

For the folksonomy data, a tag selection rule should be specified. For example, for data which are structured in the tag genome fashion, we can define a threshold above which tags can be considered as relevant for the entity. Then selected tags are mapped to semantic concepts by using concept extractors. For example, the tag “Skyline” can be mapped to “dbr:Skyline” with tools like Babelfy[15] and Dandelion[16]. After this, we can conduct semantic enrichment operations on mapped concepts, such as the 1-hop category enrichment (GP2 in Fig. 2). For example, “dbr:Skyline” can be enriched with “dbc:Skycrappers” and “dbc:Towers”. The enriched semantic concepts are fusioned to an aggregated graph together with the subset knowledge graph resulting from a property selection process (like in Sect. 3.3). More precisely, the enriched semantic concepts are linked to the entities that they describe. In the aggregated graph, data from the knowledge graph use their original properties, data from the folksonomy use a common

15 http://babelfy.org/.
16 http://dandelion.eu/.

"has_characteristic" property. In the current version of the framework, the data cleaning process uses three heuristics: 1. Eliminating nodes linked to only one entity in the graph; 2. Deduplicating nodes which appear multiple times due to the fusion of processed knowledge graph and folksonomy data; 3. Privileging properties from the knowledge graph than "has_characteristic".

Finally, an affinity graph is generated. In addition to the recommendation usage, it provides the possibility to explain the recommendations [24, 39] in a feature style [36]. For example, one can recommend "Ljubljana" because of the feature "dbc:Capitals_in_Europe". Given a user profile containing a list of entities, the affinity graph searches the most common features shared by the entities. Since features are linked to entities explicitly with different properties, we have control on the diversity of the features to display to the user. We developed a diversity function which maximizes the number of properties of the displayed features. The function iterates on the list of features ordered by their occurrences and selects a feature if no other feature sharing the same property has been selected previously. The iteration ends when desired number of features are selected. In case it is not reached after an iteration on the whole list, more iterations are then conducted on the unselected features.

5 Second Experiment: User Study

To assess the usefulness and the efficiency of the Semantic Affinity Framework, we conducted a qualitative user study which is complementary to the quantitative evaluation on a gold-standard dataset. We computed an affinity graph with data from both knowledge graph and folksonomy. Apart from the aspects that were already mentioned in the first experiment, in this user study, we are especially interested in the explanation capacity of different approaches. This aspect is of qualitative nature and can only be assessed by questioning real users. Three approaches are compared: KG, FOLK (which are compared in the first experiment) and AG (which uses the affinity graph). The explanation generation mechanism is described for AG in Sect. 4. KG and FOLK compute explanations within their respective data spaces and follow the same logic with the exception that the diversity function is not applied on FOLK because there are no differentiated semantic properties.

5.1 Protocol and Metrics

We used a well-oiled protocol [40] to simulate a travel planning process. Firstly, participants put themselves into the scenario of looking for the destination for the next trip. They were free to choose to plan for a weekend trip or a long holiday. Secondly, they went to the evaluation interface where they could visualize the 705 cities. To reduce bias towards cities shown at the top of the page, the presentation order of the cities was randomized. Thirdly, they chose several cities that appealed to them at first glance. Fourthly, they submitted their choices and got three sets of top-5 highest scored cities generated by the three candidate approaches accompanied by 5 semantic concepts to explain the recommendations. In Fig. 4, we show an example about how the recommendations and explanations were presented. Finally, participants rated respectively the recommendations and the explanations as a whole in a five-level Likert scale on different quality dimensions. For the recommendations, the same dimensions as in the first experiment were reused: relevance, diversity and novelty. For the explanations, instead of novelty, we opted for interestingness. This dimension measures the capacity of arousing the attention or interest. Participants were guided by the exact meaning of the scale. For example, on the relevance dimension, the scale was: 1 – not relevant, 2 – weakly relevant, 3 – moderately relevant, 4 – relevant, 5 – strongly relevant. We consider 4 and 5 as positive ratings. We use the percentage of positive ratings as our metric.

You submitted:	You might like:	We recommend you:
dbr:Rome	dbc:Clothing	dbr:The_Hague
dbr:Florence	dbr:Food	dbr:Haarlem
dbr:Amsterdam	dbr:David_de_Haen	dbr:Naples
	dbr:Italy	dbr:Milan
	dbr:History	dbr:Turin

Fig. 4. Example of recommendations and explanations generated by the AG approach for a user having submitted dbr:Rome, dbr:Florence and dbr:Amsterdam

5.2 Results and Discussions

37 people participated in our study. They work in different companies located at the “Pépinière 27” in Paris, France at the moment of this study. They have between 25 to 38 years old, 19 males, 18 females.

The results on the recommendations are in line with the results obtained in our first experiment. KG yields clearly better accuracy than FOLK. FOLK has more advantage on diversity and novelty. AG obtains indeed balanced and good scores on all three dimensions (Fig. 5).

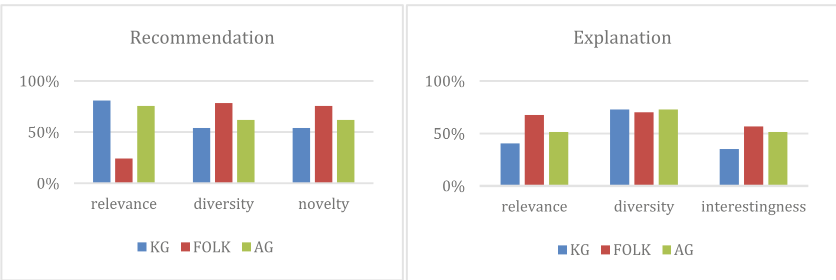

Fig. 5. Percentage of users having given positive ratings to recommendations and explanations

We discuss more about the explanations part which is not covered in the first experiment. The results showed that explanations provided by FOLK were the most appreciated. Our folksonomy dataset was crowdsourced by travelers, it is by nature highly relevant and it covers different travel aspects (food, activity, transport). The explanation capacity of AG was boosted by the inclusion of features from FOLK which allowed it to outperform KG where only knowledge graph features were used. Participants were relatively skeptical about some knowledge graph features. Some users found the features very general, for example, “dbc:Leisure” which comes from the 1-hop category enrichment. A possible solution to this problem is to use the DBpedia category tree [17]. Since we know the level of all categories, we can define a threshold above which categories and its related concepts are too general for the explanation task. Some users found some features difficult to understand such as “dbr: China_Record_Corporation”. The user who got this explanation submitted “dbr:Shanghai”, “dbr:Shenzhen”, “dbr:Beijing”. “dbr: China_Record_Corporation” is linked to all these three cities by “dbo:location”. Actually we picked “dbo:location” because it allows to capture interesting links, for example it can link two cities via a television series. However, this property is also used by entities having the type “dbo:Company”. This problem can be solved with additional engineering efforts, such as blacklisting certain types.

To sum up, knowledge graph allows to better yield entities in high affinity with the users, folksonomy performs better on diversity and novelty, and it also brings high quality explanations. Harvesting both data spaces, the affinity graph results in equitable and competitive performance on multiple quality dimensions in both recommendation and explanation tasks. To make explanations more user-friendly, additional engineering efforts are needed and it can be very helpful to leverage knowledge graphs especially ontology types and DBpedia category tree.

6 Conclusion

In this paper, we are interested in the problem of user-entity affinity assessment which is essential in many user-centric information systems. Among different assessment techniques, content-based ones predict higher affinity scores for entities similar to the ones with which a user had positive interactions in the past. Knowledge graph and folksonomy have boosted the similarity calculation by providing a large amount of data on entities. Despite the shared crowdsourcing trait between knowledge graph (some major large-scale ones e.g. DBpedia and Wikidata) and folksonomy, the encoded data are different in nature and structure. Knowledge graph encodes factual data with a formal ontology. Folksonomy encodes experience data with a loose structure. Existing work has proven their efficiency in separate settings. To the best of our knowledge, this paper is the first work trying to shed some light on their comparative performance in the affinity assessment task. We made a comprehensive state of the art. We have selected the most representative approach of each category for comparison. We conducted two experiments. The first one within a travel destination recommendation scenario on a gold standard dataset has shown a net advantage of knowledge graph in affinity assessment accuracy. However, folksonomy contributes more to two other important quality dimensions which are diversity and novelty. This interesting complementarity motivated us to develop the Semantic Affinity Framework to harvest the benefits of both knowledge graph and folksonomy. The framework integrates, aggregates, enriches and cleans entity data from both spaces, and finally produces an affinity graph. A second experiment with real users confirmed the findings of the first experiment and showed the utility and the efficiency of the proposed semantic affinity framework. In addition to the recommendation task, we evaluated the capacity of explaining the recommendations. The inclusion of folksonomy data in the affinity graph has clearly increased the relevance and interestingness of the explanations. The travel domain within which our two experiments were conducted is the predominant domain of e-commerce. We hope that our findings can guide the design and the development of affinity-based systems in this important domain. We also hope that the ideas and the methodology of this paper can serve as an instigator of further similar comparative studies in other domains.

References

1. McMahan, H.B., Holt, G., Sculley, D., Young, M., Ebner, D., Grady, J., Chikkerur, S.: Ad click prediction: a view from the trenches. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 1222–1230. ACM (2013)
2. Waitelonis, J., Sack, H.: Towards exploratory video search using linked data. Multimed. Tools Appl. **59**(2), 645–672 (2012)
3. Krulwich, B.: Lifestyle finder: intelligent user profiling using large-scale demographic data. AI Mag. **18**(2), 37 (1997)
4. Candillier, L., Meyer, F., Boullé, M.: Comparing state-of-the-art collaborative filtering systems. In: Perner, P. (ed.) MLDM 2007. LNCS (LNAI), vol. 4571, pp. 548–562. Springer, Heidelberg (2007). doi:10.1007/978-3-540-73499-4_41
5. Lops, P., De Gemmis, M., Semeraro, G.: Content-based recommender systems: state of the art and trends. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 73–105. Springer, New York (2011)
6. Di Noia, T., Ostuni, V.C.: Recommender systems and linked open data. In: Faber, W., Paschke, A. (eds.) Reasoning Web 2015. LNCS, vol. 9203, pp. 88–113. Springer, Cham (2015). doi:10.1007/978-3-319-21768-0_4
7. Pellissier Tanon, T., Vrandečić, D., Schaffert, S., Steiner, T., Pintscher, L.: From freebase to wikidata: the great migration. In: Proceedings of the 25th International Conference on World Wide Web, pp. 1419–1428 (2016)
8. Schmachtenberg, M., Bizer, C., Paulheim, H.: Adoption of the linked data best practices in different topical domains. In: Mika, P., et al. (eds.) ISWC 2014. LNCS, vol. 8796, pp. 245–260. Springer, Cham (2014). doi:10.1007/978-3-319-11964-9_16
9. Bontcheva, K., Rout, D.: Making sense of social media streams through semantics: a survey. Semant. Web **5**(5), 373–403 (2014)
10. Passant, A., Laublet, P.: Meaning of a tag: a collaborative approach to bridge the gap between tagging and linked data. In: Proceedings of Linked Data on the Web Workshop (2008)
11. Mika, P.: Ontologies are us: a unified model of social networks and semantics. Web Semant. Sci. Serv. Agents World Wide Web **5**(1), 5–15 (2007)
12. Cantador, I., Konstas, I., Jose, J.M.: Categorising social tags to improve folksonomy-based recommendations. Web Seman. Sci. Serv. Agents World Wide Web **9**(1), 1–15 (2011)
13. Cattuto, C., Benz, D., Hotho, A., Stumme, G.: Semantic grounding of tag relatedness in social bookmarking systems. In: Sheth, A., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 615–631. Springer, Heidelberg (2008). doi:10.1007/978-3-540-88564-1_39

14. Szomszor, M., Alani, H., Cantador, I., O'Hara, K., Shadbolt, N.: Semantic modelling of user interests based on cross-folksonomy analysis. In: Sheth, A., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) ISWC 2008. LNCS, vol. 5318, pp. 632–648. Springer, Heidelberg (2008). doi:10.1007/978-3-540-88564-1_40
15. Orlandi, F., Breslin, J., Passant, A.: Aggregated, interoperable and multi-domain user profiles for the social web. In: Proceedings of the 8th International Conference on Semantic Systems, pp. 41–48. ACM (2012)
16. Abel, F., Herder, E., Houben, G.J., Henze, N., Krause, D.: Cross-system user modeling and personalization on the social web. User Model. User-Adap. Inter. **23**(2–3), 169–209 (2013)
17. Kapanipathi, P., Jain, P., Venkataramani, C., Sheth, A.: User interests identification on twitter using a hierarchical knowledge base. In: Presutti, V., d'Amato, C., Gandon, F., d'Aquin, M., Staab, S., Tordai, A. (eds.) ESWC 2014. LNCS, vol. 8465, pp. 99–113. Springer, Cham (2014). doi:10.1007/978-3-319-07443-6_8
18. Piao, G., Breslin, J.: Exploring dynamics and semantics of user interests for user modeling on Twitter for link recommendations. In: Proceedings of the 12th International Conference on Semantic Systems (SEMANTiCS 2016), Leipzig, Germany (2016)
19. Szomszor, M., Cattuto, C., Alani, H., O'Hara, K., Baldassarri, A., Loreto, V., Servedio, V.D.P.: Folksonomies, the semantic web, and movie recommendation. In: Proceedings of the Workshop on Bridging the Gap between Semantic Web and Web 2.0 at the 4th ESWC (2007)
20. Semeraro, G., Lops, P., De Gemmis, M., Musto, C., Narducci, F.: A folksonomy-based recommender system for personalized access to digital artworks. J. Comput. Cult. Heritage (JOCCH) **5**(3), 11 (2012)
21. Gruber, T.: Collective knowledge systems: where the social web meets the semantic web. Web Web Semant. Sci. Serv. Agents World Wide Web **6**(1), 4–13 (2008)
22. Vig, J., Sen, S., Riedl, J.: The tag genome: encoding community knowledge to support novel interaction. ACM Trans. Interact. Intell. Syst. **2**(3), Article 13 (2102)
23. Zhu, G., Iglesias, C.A.: Computing semantic similarity of concepts in knowledge graphs. IEEE Trans. Knowl. Data Eng. **29**(1), 72–85 (2016)
24. Passant, A.: dbrec—music recommendations using DBpedia. In: Patel-Schneider, P.F., et al. (eds.) Proceedings of the 9th International Semantic Web conference. LNCS, vol. 6497, pp. 209–224. Springer, Heidelberg (2010)
25. Piao, G., Breslin, J.: Measuring semantic distance for linked open data-enabled recommender systems. In: Proceedings of the 31st Annual ACM Symposium on Applied Computing. ACM (2016)
26. Di Noia, T., Mirizzi, R., Ostuni, V.C., Romito, D., Zanker, M.: Linked open data to support content-based recommender systems. In: Proceedings of the 8th International Conference on Semantic Systems, pp. 1–8. ACM (2012)
27. Di Noia, T., Ostuni, V.C., Tomeo, P., Di Sciascio, E.: Sprank: semantic path-based ranking for top-n recommendations using linked open data. ACM Trans. Intell. Syst. Technol. (TIST) **8**, 9 (2016)
28. Ristoski, P., Paulheim, H.: RDF2Vec: RDF graph embeddings for data mining. In: Groth, P., Simperl, E., Gray, A., Sabou, M., Krötzsch, M., Lecue, F., Flöck, F., Gil, Y. (eds.) ISWC 2016. LNCS, vol. 9981, pp. 498–514. Springer, Cham (2016). doi:10.1007/978-3-319-46523-4_30
29. Heitmann, B.: An open framework for multi-source, cross-domain personalisation with semantic interest graphs. Doctoral dissertation. National University of Ireland, Galway (2014)
30. Kaminskas, M., Fernández-Tobías, I., Ricci, F., Cantador, I.: Knowledge-based identification of music suited for places of interest. Inf. Technol. Tourism **14**(1), 73–95 (2014)
31. Marie, N.: Linked data based exploratory search. Doctoral dissertation. Université de Nice Sophia-Antipolis (2014)

32. Lu, C., Laublet, P., Stankovic, M.: Travel attractions recommendation with knowledge graphs. In: Blomqvist, E., Ciancarini, P., Poggi, F., Vitali, F. (eds.) EKAW 2016. LNCS (LNAI), vol. 10024, pp. 416–431. Springer, Cham (2016). doi:10.1007/978-3-319-49004-5_27
33. Thomee, B., Shamma, D.A., Friedland, G., Elizalde, B., Ni, K., Poland, D., Li, L.J.: YFCC100M: the new data in multimedia research. Commun. ACM **59**(2), 64–73 (2016)
34. Nguyen, P.T., Tomeo, P., Noia, T., Sciascio, E.: Content-based recommendations via DBpedia and freebase: a case study in the music domain. In: Arenas, M., Corcho, O., Simperl, E., Strohmaier, M., d'Aquin, M., Srinivas, K., Groth, P., Dumontier, M., Heflin, J., Thirunarayan, K. (eds.) ISWC 2015. LNCS, vol. 9366, pp. 605–621. Springer, Cham (2015). doi: 10.1007/978-3-319-25007-6_35
35. Noia, T., Cantador, I., Ostuni, V.C.: Linked open data-enabled recommender systems: ESWC 2014 challenge on book recommendation. In: Presutti, V., Stankovic, M., Cambria, E., Cantador, I., Iorio, A., Noia, T., Lange, C., Reforgiato Recupero, D., Tordai, A. (eds.) SemWebEval 2014. CCIS, vol. 475, pp. 129–143. Springer, Cham (2014). doi: 10.1007/978-3-319-12024-9_17
36. Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Recommender systems survey. Knowl.-Based Syst. **46**, 109–132 (2013)
37. Lu, C., Laublet, P., Stankovic, M.: Ricochet: context and complementarity-aware, ontology-based POIs recommender system. In: Proceedings of SALAD at the 14th Extended Semantic Web Conference, pp. 10–17 (2014)
38. Lu, C., Stankovic, M., Laublet, P.: Leveraging semantic web technologies for more relevant E-tourism behavioral retargeting. In: Proceedings of the 24th International Conference on World Wide Web, pp. 1287–1292. ACM (2015)
39. Vig, J., Sen, S., Riedl, J.: Tagsplanations: explaining recommendations using tags. In: Proceedings of the 14th International Conference on Intelligent User Interfaces, pp. 47–56. ACM (2009)
40. Lu, C., Stankovic, M., Laublet, P.: Desperately searching for travel offers? formulate better queries with some help from linked data. In: Gandon, F., Sabou, M., Sack, H., d'Amato, C., Cudré-Mauroux, P., Zimmermann, A. (eds.) ESWC 2015. LNCS, vol. 9088, pp. 621–636. Springer, Cham (2015). doi:10.1007/978-3-319-18818-8_38