A Recommendation Engine based on Social Metrics

Ofelia Cervantes[1], Francisco Gutiérrez[1] Ernesto Gutiérrez[1], J. Alfredo Sánchez[1], Muhammad Rizwan[2], and Wan Wanggen[2]

[1] Universidad de las Américas Puebla, Puebla, México
[2] Institute of Smart City, Shanghai University, Shanghai, China
ofelia.cervantes@udlap.mx, francisco.gutierrez@udlap.mx, ernesto.gutierrezca@udlap.mx, alfredo.sanchez@udlap.mx, rizwan@shu.edu.cn, wanwg@staff.shu.edu.cn

Abstract. Recommender systems have changed the way people find products, points of interest, services or even new friends. The technology behind recommender systems has evolved to provide user preferences and social influences. In this paper, we present a first approach to develop a recommendation engine based on social metrics applied to graphs that represent object's characteristics, user profiles and influences obtained from social connections. It exploits graph centrality measures to elaborate personalized recommendations from the semantic knowledge represented in the graph. The graph model and selected graph algorithms for calculating graph centralities that are the core of the recommender system are presented. Semantic concepts such as semantic predominance and similarity measures are adapted to the graph model. Implementation challenges faced to solve performance issues are also discussed.

Keywords: recommender systems, social metrics, graph models, massive graph processing

1 Introduction

The widespread availability of products and services through web-based and mobile applications makes it difficult for end users to select the right item, according to their preferences. A recommender system must make use of different sources of information for providing users with suggestions of items that better correspond to their expectations. Those sources can include user preferences, item descriptions or social information.

Several approaches have been proposed to tackle the problem of selecting automatically the list of items that really contributes to satisfy the needs of end users. Approaches based on demographics or modeling user profiles are oriented to exploit user features and preferences for filtering available choices. The main challenge of this approach is to create the user profile from scratch. Some systems invite users to select their preferences from a predefined list of categories or allow the system to extract their profiles from other applications. Other systems register every user action to dynamically build models based upon the user's behavior. Some research works focus on collecting ratings made by users that have evaluated the product or service providing a first-hand perception of the quality of the evaluated item. Other approaches have focused on describing the main features of every item and trying to match them with user requirements.

Modeling social networks using graphs has opened opportunities for exploring new alternatives for implementing recommender systems. Social metrics, such as flow centralities calculated on graph-based models provide interesting measures to represent the semantic predominance of concepts featuring users' preferences as well as item characteristics. A first proof of concept was accomplished and reported in [6]. A more detailed description of the model is presented here as well as implementation problems that were solved to provide a complete recommendation engine that can be integrated in recommendation applications.

The remainder of the paper is structured as follows: Section 2 presents related work. Then, Section 3 introduces the proposed graph-based recommendation model. Section 4 discusses selected graph algorithms used to calculate social metrics, particularly flow centralities. Next, Section 5 describes the framework developed in the prototype and the performance challenges we had to overcome. Finally, in Section 6 we report the results we have obtained thus far and discuss future work.

2 Related Work

Traditionally, recommender systems are classified into the following categories: demographic, content-based, collaborative filtering, social-based, context-aware and hybrid approaches. However, the ever increasing amount of information available in social media enables the development of knowledge-based recommender systems [4]. The rich knowledge that has been accumulated in social media can be exploited to improve recommendation outcomes, enhance user experience and to develop new algorithms. In addition to the classification of recommender systems, Zanker [24] illustrates the fundamental building blocks of a recommender system, out of which we highlight three: user model, community (social network), and a recommendation algorithm. Thereby, we present a brief overview of related work regarding knowledge-base recommender systems from building blocks perspectives.

2.1 User Modeling

In order to personalize recommendations, it is necessary to know information about each user. User models are representations of users' needs, goals, preferences, interests, and behaviors along with users' demographic characteristics [19]. Several user modeling approaches have been proposed, from typical weighted vectors to domain ontologies. In [1] the authors defined a user model based on fuzzy logic and proposed an approach to infer the degree of genre presence in a movie by exploiting the tags assigned by the users. In [24] the authors presented a simple attribute-value pair dictionary to model the user through the explicit elicitation of user requirements. A richer user model is presented in [10], where the authors used a machine learning process to capture the user profile and context into a domain ontology.

Our work tries to balance between simple [24] and complex models [10] with the goal of having an efficient but still rich user model. Other works, like Cantador et al. and Moahedian et al. [5, 14], are similar to our proposed user model, since we use tags and keywords to build a lax ontology.

2.2 Recommendation Algorithms and Techniques

There is a wide range of recommendation algorithms and techniques. They vary according to data availability, recommender filtering type as well as user and object representation. Several methods have demonstrated to have an acceptable performance such as: association rule learning [24], Bayesian networks [8], nearest neighbors [2], genetic algorithms [13], neural networks [3], clustering [20], latent semantic features, among others that can be found with more detail in [4].

In this work, we use graph metrics commonly used in social network analysis [21]. We propose semantic social network analysis that integrates semantic methods of knowledge engineering and natural language processing with classic social network analysis. Advantages of semantic social network analysis are: its knowledge foundation and its non probabilistic nature. In contrast, one disadvantage is its computational cost. Therefore, enhancement techniques are needed to process graph metrics more efficiently.

2.3 Information Extracted from Social Networks

Recommender systems are creating unique opportunities to assist people to find relevant information when browsing the web, and making meaningful choices with the success of emerging Web 2.0, and various social network Websites. In [7], the author has proposed a novel approach for recommendation systems that uses data collected from social networks.

Wang et al. [22] studied the problem of recommending new venues to users who participate in location-based social networks (LBSNs) and propose algorithms that create recommendations based on: past user behavior (visited places), the location of each venue, the social relationships among the users, and the similarity between users.

Ye et al. [23] exploited the social and geographical characteristics of users and locations/places to research issues in realizing location recommendation services for large-scale location-based social networks. They observed the strong social and geospatial ties among users and their favorite locations/places in the system via the analysis of datasets collected from Foursquare.

Similar to our work, Savage [18] investigated the design of a more complete, ubiquitous location-based recommendation algorithm that is based on a text classification problem. The system learns user preferences by mining a person's social network profile. The author also defined a decision-making model, which considers the learned preferences, physical constraints, and how the individual is currently feeling.

We can state that novel approaches rely mainly on the fusion of information inferred from a user's social network profile and other data sources (e.g. mobile phone's sensors). In this sense, it is necessary to develop new strategies that produce recommendations from rich but still incomplete information.

3 Graph Model

Our recommendation engine is based on a graph representation of users and objects of interest linked through concepts (denoted as terms). Figure 1 shows the graph model where every node falls in one of three categories: *User*, *Term* or *Object of Interest*, and every edge represents the semantic relation between nodes: *Predominance*, *Similarity* or *Friendship*. Every *Term* node of the graph in Figure 1 acts as a semantic descriptor of both: *Users* and *Objects of Interest*. In other words, every user and every object are correspondingly described by the terms linked to them. In general, users are described by their tastes, preferences, and interest (user model) whereas objects are described by tags and keywords (object model). In this manner, when a term is shared between a user and an object, it shows the possibility that the user could be interested in that particular object, even though the object had never been seen or rated by user.

A graph-based representation allows us to apply graph algorithms (e.g. social metrics) to discover topological features, key relationships, and important (prestigious) nodes. Then, with these features we can make relevant recommendations to users, such as suggesting friends or places. Therefore, the foundation of our recommender system relies on a knowledge base constructed from both: a **user model** (see section 3.3) and an **object model** (see section 3.4).

In order to construct the user and object models, we applied a linguistic analysis over user and object text descriptions. Basically, we conducted preprocessing (removal of stopwords and selection of most descriptive words) and statistical linguistic analysis (using weighting schemes: tf-idf and okapi BM25) to define a bond between text descriptions and semantic relations represented in the graph (see section 3.2). It is possible to obtain user and object descriptions from social networks (Facebook, Foursquare, Twitter), web pages (Wikipedia, web search results, etc.), human experts contributions or other textual resources.

3.1 Weighted Graph Definition

Formally, we define a weighted graph $G = (V, E, f_E)$ where $V = \{v_1, \ldots, v_n\}$ is a set of vertices vertex (nodes), $E = \{e_1, \ldots, e_n\} \subset \{\{x, y\} \mid x, y \in V\}$ is a set of edges, and $f_E : E \to \mathbb{R}$ the function on weights for every edge. In our recommender system $V = U \cup T \cup O$ where U is the set of users, T is the set of terms, and O is the set of objects of interest. $E = P \cup S \cup F$ where P is

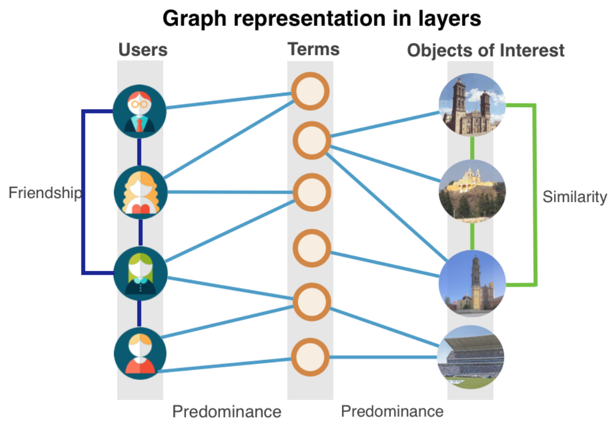

Fig. 1. Graph model. Node layers: User, Term, and Objects of Interest linked trough Friendship, Predominance and Similarity edges.

the set of predominance edges, S the set of similarity edges, and F is the set of friendship edges. Function f_E is adapted according to each type of edge. For instance, we can obtain the sub-graph of users as $G_U = (U, F, f_F)$ (see User layer in Figure 1), the sub-graph of objects as $G_O = (O, S, f_S)$ (see Objects of Interest layer in Figure 1), and the sub-graph of user and object profiles as $G_{U \cup T \cup O} = (U \cup T \cup O, P, f_P)$.

3.2 Semantic Relations

In order to build the semantic relations of the graph, it is necessary to obtain text descriptions of users and objects. As a result, we have two collections: the users text collection (UTC) and the objects text collection (OTC), where each text description is considered a document D in a vector space model.

We define three types of semantic relations (edges of the graph): predominance, similarity, and friendship. Each semantic relation links different types of nodes and has a different weighting function. Predominance is the edge between a user or an object and a term, similarity is the edge between two objects and friendship is the edge between two users.

Predominance is the semantic relation between a term and a user or an object. A term acts as a descriptor of users and objects. We define a weighting function over the edge of predominance based on linguistic analysis. We apply Okapi BM25 ranking function [17] to each independent document collection (UTC and OTC) using Equation 1.

In Equation 1, $pred$ is the predominance of the term T in document D, I_{doc} is the number of indexed documents (size of collection), T_{doc} is the number of documents containing term T, TF is the term frequency relative to document D, DL is the document length, $avgDL$ is the average document length among the entire collection, K and B are free parameters (usually $K = 1.2$ and $B = 0.75$).

$$pred_{(D,T)} = \log_{10}\left(\frac{I_{doc}+0.5}{I_{doc}+0.5}\right) * \left(\frac{TF*(K+1)}{TF+K*((1-B)+B*\frac{DL}{avgDL})}\right) \quad (1)$$

Similarity is the semantic relation between two objects. This measure indicates the degree of affinity between objects. We apply the cosine similarity measure (Equation 2) to obtain this value. The Similarity is calculated after the predominance, since it relies on shared terms. Then, every object is a vector of predominances as shown below Equation 2.

$$Similarity_{(A,B)} = cos\,\Theta = \frac{A \cdot B}{\|A\|\,\|B\|} \quad (2)$$

$$ObjectA = \left[pred_{(A,T_1)}, pred_{(A,T_2)}, \cdots, pred_{(A,T_n)}\right]$$
$$ObjectB = \left[pred_{(B,T_1)}, pred_{(B,T_2)} \cdots, pred_{(B,T_n)}\right]$$

In Equation 2, the similarity between object A and object B is determined by the weights of the terms they have in common. In this manner, a high similarity value indicates a higher semantic correspondence between objects.

Friendship is the semantic relation between two users. This measure indicates the degree of affinity between two users. Our current model does not distinguish between close friends, friends or acquaintances. Therefore, the users' sub-graph is only a friend-of-a-friend (FOAF) node-link type.

3.3 User Model

As part of the graph-based representation, users are defined as sub-graphs. A user model is composed of two sub-graphs: user profile $G_u = (u, P, f_P)$ and user FOAF network $G_U = (U, F, f_F)$. Figure 2A shows the user profile network and 2B the user FOAF network. In the user profile sub-graph, each user is linked to a set of terms that indicate tastes, preferences and interests. Tastes are general inclinations of user towards some entities and they are generally expressed with actions such as likes (e.g. Foursquare, check-ins and Facebook likes *football, beer, steak, coffee, etc*). Preferences are user inclinations towards taste features. Preferences are more fine grained than tastes and are usually expressed in users' reviews and ratings (e.g. starred reviews: *I like the double espresso, I don't like diet soda*). Interests are defined as contextual user inclinations or intentions (e.g. *I want to try Chinese food, I'm going to watch minions movie*).

$$f_P = \begin{cases} pred_{(U,T)} & Case\ A \\ 1 & Case\ B \\ \#stars - 2 * \frac{1}{3} & Case\ C \end{cases} \quad (3)$$

Our scheme to weight edges within a user profile is indicated in Equation 3. Case A occurs when only text descriptions are used; this means that terms

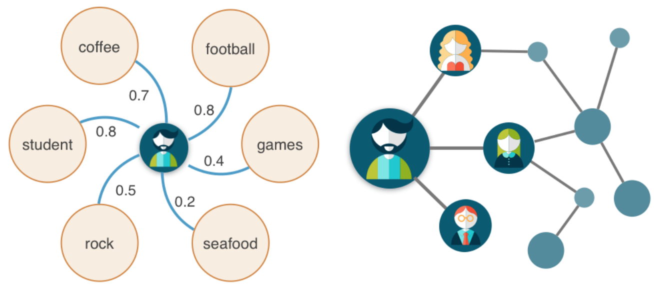

Fig. 2. A)User profile and B)User FOAF Network

are weighted according to the Okapi measure (*pred*, as shown in Equation 1). Case B occurs when explicit likes are found in Foursquare or Facebook. Case C occurs when terms extracted from starred reviews are used to describe a user. In addition to Equation 3, we use a threshold value to limit the number of terms connected with a given user. In fact, we use the first quartile as threshold value. An example of user profile is shown in Figure 2A, where, it is possible to notice that a user likes football, rock and coffee, and is likely to be student. In user friendship networks, as mentioned earlier, there are no differences among friendship types. Then, in FOAF network all weights are equal to 1 ($f_F = 1$). A user FOAF network is shown in Figure 2B.

3.4 Object Model

Objects of interest are sets of items that can be of potential interest to a user. Depending upon the application, objects of interest can be of different grain size. For instance, they can be coarse-grained as a point of interest (POI) or fine-grained as items inside places. An object is described by the category to which it belongs (e.g. dog is an animal). Also, it is described by tags designed by users or keywords found in object's description. The object model is consists of two sub-graphs: object profile $G_o = (o, P, f_P)$ and object similarity sub-graph $G_O = (O, S, f_S)$. Object profile was built with data gathered from Foursquare, Wikipedia and results from web searches. The weights of edges that link objects and terms were calculated using the predominance formula shown in Equation 1. This means that the weight function on edges is $f_P = pred_{(O,T)}$.

3.5 User Global and Local Network

In order to apply social metrics (centrality measures) and relate them to pertinent recommendations, we defined two networks from user perspective: a user global network (UGN_u) and a user local network (ULN_u). User global network is the whole graph (all nodes: users U, terms T and objects O and all edges: similarities S, predominances P and friendships F) centered in current user.

Therefore, $UGN_u = (U \cup T \cup O, S \cup P \cup F)$ (see Figure 1). Whereas user local network is the sub-graph defined by current user node u, term nodes adjacent to user T_u and object nodes adjacent to terms node O_T linked trough predominance edges from user P_u and from objects P_O. Hence, $ULN_u = (u \cup T_u \cup O_T, P_u \cup P_O)$ (see Figure 3). It is important to highlight the difference between user global and local networks, since it will lead to different semantics interpretations when calculating centrality measures over them.

4 Centrality Measures and Recommender Engine

Centrality measures have been used extensively in the past to exploit networks and discover the most relevant nodes in a graph. In social network analysis (SNA) graph centralities are used to identify the most relevant persons, communities and even detect strange behaviors in the network. However, given the takeoff of social networks, people have increased their interaction not only to meet people and friends but to search about things they like, give their impressions, and reach points of interest and objects of interest. These spatio-temporal interactions can also be represented in a graph, thus, an accurate user profiling representation can give us a great deal of insight about the user behavior. Because of these approaches in exploiting graph measures we have explored the use of centralities to exploit the topological structure of our user global network and flow centrality to get the most of our weighted global and local networks in terms of a recommender engine.

4.1 Centrality Algorithms

Centrality in graphs is widely used to measure the importance of a node in a graph, especially in SNA [9]. Our recommender engine implements these central-

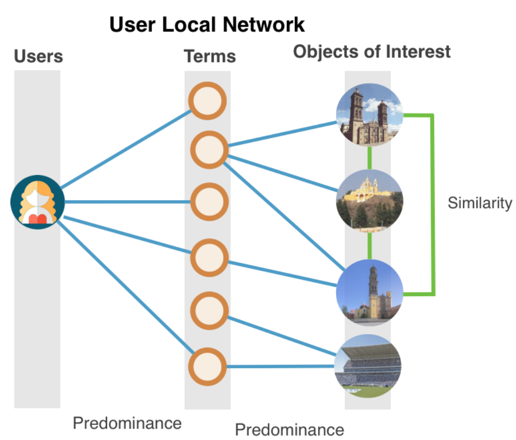

Fig. 3. User Local Network

ities to measure the relevance of people in the social network. Some centrality measures like closeness and betweenness are based on the calculation of the shortest distance to reach all other nodes in the graph. Our algorithms to calculate centralities are applied to the network of persons so we can infer the most popular nodes (degree), the capacity of a node to reach any other in the network (closeness), and to identify the leaders interconnected within a neighborhood in the graph (betweenness) [15]. Degree centrality is a measure that counts the direct relationships a node has, and thus, the nodes that are in direct contact. Closeness is defined as the inverse sum of the shortest paths to each other node and betweenness is defined as the number of shortest paths from all vertices to all others that pass through that node. Centrality measures are calculated over the network at a topological level given a scale-free graph of persons. Thus, these measures are not exploiting our weighted graph, they are applied only at a social-network level. Terms and objects of interest can be seen as sub-graphs of the global network that can be exploited by using flow-based centrality measures.

4.2 Flow-Centrality Algorithms

We are using flow centralities [15] to measure the betweenness, closeness, and eccentricity in between the objects of interest, terms, and people profiles. Flow centralities allow us to exploit the semantic relationships between the user and the profiles of the objects of interest. Flow centralities reveal the most relevant nodes in the graph given their weights, for instance, given a set of terms associated to a user profile we can better understand a user preferences and give a better recommendation.

Flow Betweenness In SNA betweenness is one of the most common referenced centralities. Flow betweenness (see Equation4) is defined in [11] as the max flow that passes through node x_i, by the total flow between all pairs of points (p_i) where x_i is neither a source nor a sink. A node with a high flow betweenness centrality has a large influence in the network because of the flow that passes through it. Due to the relevance of a node with high betweenness, in our recommender model, a node with high betweenness should be recommended as the things the user cannot miss (see Figure 4).

$$F_b(i) = \frac{\sum_{j<k}^{n} m_{jk}(x_i)}{\sum_{j<k}^{n} m_{jk}} \tag{4}$$

Flow Closeness Closeness is just a measure of distance and is defined as the inverse of the average distance to other vertices. A node with high flow closeness centrality has a fast communication within all the nodes in the graph. In equation 5, flow closeness is defined as the inverse sum of the max flow to every other resource. In our recommender model, elements with high flow closeness should be recommended to the user as things that could be interesting, because those weighted elements are close to the user profile (see Figure 5).

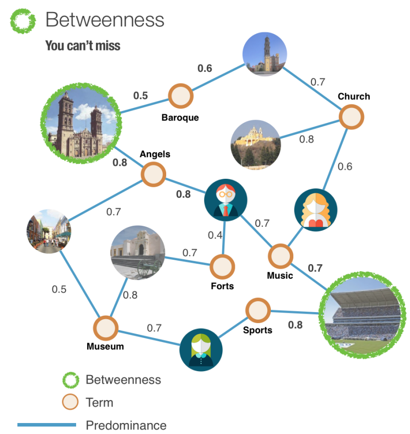

Fig. 4. "You can't miss" as a result of computing flow betweenness.

$$F_c(i) = \frac{1}{\sum_{j<k}^{n} m_{jk}(x_i)} \tag{5}$$

Eccentricity On the other hand, eccentricity is the maximum distance taking in consideration the weighted paths of the network. Eccentricity lets us find the nodes that are far away from the most central node in the network. In equation 6 eccentricity is defined as the maximum distance between pairs of nodes given their maximum flow in the network. In our recommender model, an item with

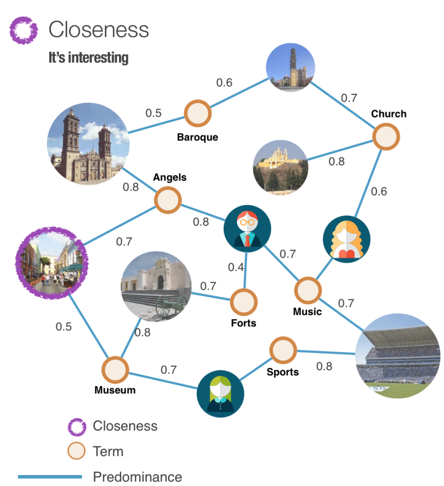

Fig. 5. "Could be interesting" as a result of computing closeness.

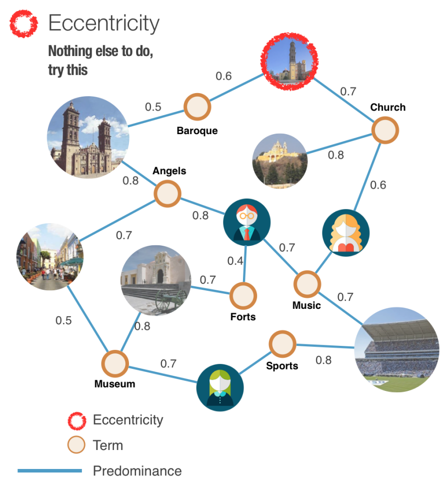

Fig. 6. "Nothing else to do? Try this" as a result of computing betweenness.

Table 1. Centrality-based recommendations.

Measure	Recommendation
Similar nodes	What else?
Flow betweenness	You can't miss
Flow closeness	Also discover
Eccentricity	Be different

high eccentricity should be recommended if the user has nothing left to do (see Figure 6).

$$F_e(i) = max_{j \epsilon_v} \sum m_{jk}(x_i) \tag{6}$$

4.3 Graph recommendations

As we have shown, our recommender model relies on the continuous computation of predominance and similarity between items in the graph. As the graph evolves from user interactions between the user and objects of interest, the recommendations get more accurate over time. However, in order to give recommendations, computation of centralities is required. As shown in Table 1, we can recommend similar items if the user is asking "What else?", then we can show him similar items to the recommended item. Flow betweenness is used to recommend things the user "can't miss" because of their relevance in the network. Flow closeness is used to recommend central items that could be things that the user "would like to discover", and eccentricity is used to show items to the user that are far away from the more central nodes in the network and could cause a "being different" impression.

5 Implementation

We decided to implement the discussed model and graph measures in a weighted graph. We explored different graph databases and graph processing frameworks, to select the tools to build a graph processing framework that could give us the flexibility to calculate those metrics with ease.

5.1 Census framework

We defined an architecture (see Figure 7) based on the graph model discussed before. We named our graph processing framework "Census" which is built with Play framework[1] and is intended to have multiple instances of Signal/Collect while processing our graph in Google Compute Engine[2]. Census uses Neo4j[3] Graph database to store the graph, Neo4j give us the flexibility to do queries over the computed network through custom plugins that serve queries through a REST API. Census process requests from Census Control which uses an orchestrator to administrate compute requests and instances of Census in the graph.

5.2 Proof of concept

With Census we explored a first approach to implement the presented recommender model. The graph database was populated with nodes of persons and

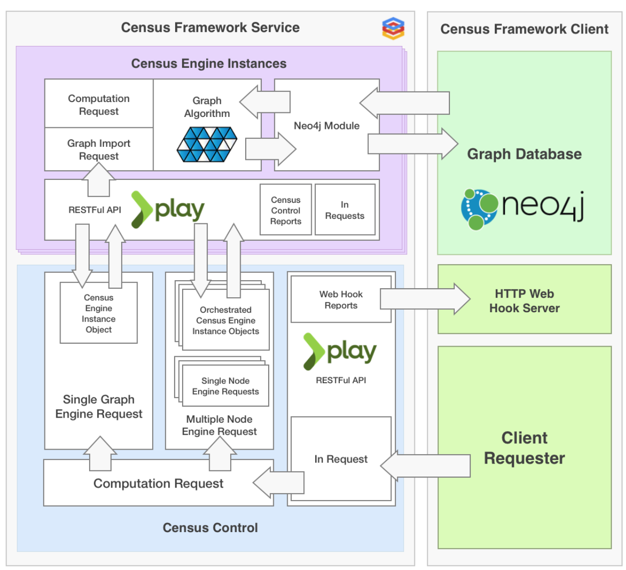

Fig. 7. Architecture of Census, graph processing framework.

[1] https://www.playframework.com/
[2] https://cloud.google.com/compute/
[3] http://neo4j.com/

Table 2. Centrality computation results

Point of Interest	**Flow Betweenness**	**Flow Closeness**	**Flow Eccentricity**	**Predominance**
Centro Expositor	0.00	0.00	0.39	0.39
Africam	1.05	0.04	0.28	0.47
Museo Revolución	**6.00**	**2.08**	0.46	1.22
Convento Sta. Monica	2.21	0.73	0.46	0.67
Museo Amparo	0.00	0.00	0.65	0.65
Museo Regional Puebla	2.77	**1.04**	0.65	0.66
Fuerte Loreto	0.00	0.00	0.78	0.78
Casa Alfeñique	0.00	0.00	0.78	0.40
Catedral Puebla	**7.13**	1.01	0.31	0.64
Exconvento Calpan	0.00	0.00	0.40	0.40
Convento San Gabriel	0.00	0.00	**0.84**	0.84
Capilla Rosario	0.00	0.00	**0.84**	0.40
Fuerte Guadalupe	0.00	0.00	0.23	0.23
Plazuela Sapos	2.04	0.16	0.35	0.64
Barrio Analco	0.00	0.00	0.41	0.41
Paseo San Francisco	0.00	0.00	0.25	0.25
Barrio Artista	0.00	0.00	0.25	0.21
Estadio Cuauhtémoc	1.05	0.12	**0.85**	1.38

points of interest from the city of Puebla, Mexico, then we selected documents from the Web to create a profile of the points of interest using our semantic approach described before. We created different users with their respective profiles setting them with random characteristics as weights in the relationships of the graph. We calculated similarities between those points of interest and the terms describing them. The result was a big graph database with approximately 3,000 nodes and 10,000 weighted relationships. Over the global network we calculated the graph measures to discover the relevant nodes in the graph.

5.3 Results

After computing all the algorithms, we focused our attention on the local network of a user in particular. Results of centrality computation are presented in Table 2. We can notice that higher centrality measure values allow us to suggest the most relevant points of interest for this user. For example we can see that “Catedral Puebla” is an element with high betweenness which means that it is a relevant place in the city and that element should be recommended as “You can’t miss”. Another relevant element is the “Museo Revolución” because it shows the highest flow closeness. In the case of flow eccentricity, we can see the elements that are far away from user preferences giving the opportunity to explore new things and be different.

6 Conclusions

We have presented a first approach to a graph-based recommendation model that takes advantage of social metrics and recommends points of interest to citizens and/or visitors in a smart city. The proposed model expresses the semantics of relationships that exist between users and points of interest through terms that define a profile for the items. This new approach, using particularly flow centralities, considers semantic predominance of terms for defining and exploiting the relationships among user profile preferences as well as the descriptive characteristics of points of interest. Recommendations can then be extracted based on the knowledge represented in the graph.

In order to validate the recommendation model, the recommendation engine was implemented and has shown that interesting recommendations could be suggested to users, considering not only their preferences, but also taking into account suggestions coming out from preferences of other members of the social network related to them by the friendship relationship. The graph-based recommendation model also proposes to explore points of interest that are very different to user preferences, inviting him to explore new points of interest in the city.

The implementation of the recommendation engine is a challenging task because of the data volume and the complexity of required calculations to evaluate flow centralities and semantic predominance. This challenge not only raised new questions but also opened interesting opportunities for dealing with performance issues. Preliminary results were presented, showing that the use of social metrics in any real recommendation system must include a specialized components for solving distributed and concurrent processing tasks. Even though nowadays there are advanced and efficient solutions for managing big data, adequate use of graph-based solutions for modeling social networks still remains as the core problem of a recommendation engine.

The framework presented in this paper was tested through a prototype that demonstrated the validity of our proposal. The recommendation engine is available through a REST APIs. These web services can be easily integrated into web or mobile apps. Application domains include intelligent tourism, (as in those described in this paper), as well as other areas of interest for citizens, such as administrative services in a Smart City.

7 Future Work

The prototype will be extended and adapted to include specific information on the cities of Puebla in Mexico and Shanghai in China. Also, different aspects could still be improved in the recommendation engine to contribute to enrich the user experience in a smart city:

- Incorporation of new semantic filters to propose lists of objects of interest; proposing for instance only the points of interest in the proximity of the user's

location and considering the time when the user queries the recommendation system.
- In absence of explicit evaluation of user preferences, we will explore the integration of the Sentiment Analysis component developed by our group [12], to offer the possibility to add open comments and to evaluate automatically their polarity.
- Routes recommendation: from the list of recommended points of interest, different alternative routes can be built. A prototype of a mobile application has been already developed for evaluating the interaction with users [16]. The prototype exploits data from Foursquare and recommends points of interest based on ratings made by users. The integration with the recommendation engine needs to be completed.

Acknowledgments Special thanks to our colleagues C. Thovex, F. Arámburo, E. Castillo and D. Báez-López for their significant contributions in this work. This project was partially supported by CONACYT-PROINNOVA No.198881, CONACYT-OSEO No.192321, and China's Foreign Experts Recruitment Program.

References

1. Anand, D., Mampilli, B.S.: Folksonomy-based fuzzy user profiling for improved recommendations. In: Expert Systems with Applications 41, 2424–2436 (2014)
2. Bobadilla, J., Hernando, A., Ortega, F., Bernal, J.: A framework for collaborative filtering recommender systems. Expert Systems with Applications 38, 14609–14623 (2011)
3. Bobadilla, J., Ortega, F., Hernando, A., Bernal, J.: A Collaborative Filtering Approach to Mitigate the New User Cold Start Problem. Know.-Based Syst. 26, 225–238 (2012)
4. Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Recommender systems survey. In: Knowledge-Based Systems. ACM 46, 109-132 (2013)
5. Cantador, I., Fernández, M., Vallet, D., Castells, P., Picault, J., Ribière, M.: A Multi-Purpose Ontology-Based Approach for Personalised Content Filtering and Retrieval, in: Wallace, D.M., Angelides, P.M.C., Mylonas, D.P. (Eds.), Advances in Semantic Media Adaptation and Personalization, Studies in Computational Intelligence. Springer Berlin Heidelberg, pp. 25–51 (2008)
6. Cervantes, O., Thovex, C., Trichet, F.: Dynamic recommendations for smart citizens based on socio-semantic network analysis.: In: International Conference on Digital Intelligence. (2014)
7. Chang, C.-C., Chu, K.-H.: A Recommender System Combining Social Networks for Tourist Attractions. In: Proceedings of the 2013 Fifth International Conference on Computational Intelligence, Communication Systems and Networks, CICSYN '13. IEEE Computer Society. Washington, DC, USA, pp. 42–47 (2013)
8. De Campos, L.M., Fernández-Luna, J.M., Huete, J.F., Rueda-Morales, M.A.: Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks. International Journal of Approximate Reasoning 51, 785–799 (2010)

9. Le Merrer, E., Trédan, G.: Centralities: Capturing the Fuzzy Notion of Importance in Social Graphs, in: Proceedings of the Second ACM EuroSys Workshop on Social Network Systems, SNS '09. ACM, New York, NY, USA, pp. 33–38 (2009)
10. Eyharabide, V., Amandi, A.: Ontology-based user profile learning. Appl Intell 36, 857–869 (2011)
11. Freeman, L.C., Borgatti, S.P., White, D.R.: Centrality in valued graphs: A measure of betweenness based on network flow. Social Networks 13, 141–154 (1991)
12. Gutiérrez, E., Cervantes, O., Báez-López, D., Sánchez, J.A.: Sentiment Groups as Features of a Classification Model Using a Spanish Sentiment Lexicon: A Hybrid Approach. In: Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F., Sossa-Azuela, J.H., López, J.A.O., Famili, F. (Eds.), Pattern Recognition. LNCS, vol. 9116, pp.258–268. Springer International Publishing (2015)
13. Hwang, C.-S., Su, Y.-C., Tseng, K.-C.: Using Genetic Algorithms for Personalized Recommendation, in: Pan, J.-S., Chen, S.-M., Nguyen, N.T. (Eds.), Computational Collective Intelligence. Technologies and Applications. LNCS, vol. 6422, pp.104-112. Springer Berlin Heidelberg, (2010)
14. Movahedian, H., Khayyambashi, M.R.: Folksonomy-based User Interest and Disinterest Profiling for Improved Recommendations: An Ontological Approach. J. Inf. Sci. 40, 594–610 (2014)
15. M. Newman.: A measure of betweenness centrality based on random walks. Social Networks, vol. 27, no. 1, pp. 39 – 54, (2005)
16. Pedraza, D.: RoX: Sistema de recomendación basado en preferencias. MSc Thesis. Universidad de las Américas Puebla (2015)
17. Robertson, S., Zaragoza, H.: The Probabilistic Relevance framework: BM25 and Beyond. Found. Trends Inf. Retr. 3, 333–389 (2009)
18. Saiph Savage, N., Baranski, M., Elva Chavez, N., Höllerer, T.: I'm feeling LoCo: A Location Based Context Aware Recommendation System.In: Gartner, G., Ortag, F. (Eds.), Advances in Location-Based Services. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 37–54 (2012)
19. Schiaffino, S., Amandi, A.: Intelligent User Profiling. In: Bramer, M. (Ed.), Artificial Intelligence An International Perspective. LNCS, vol. 5640, pp.193–216. Springer Berlin Heidelberg, (2009)
20. Shinde, S.K., Kulkarni, U.: Hybrid personalized recommender system using centering-bunching based clustering algorithm. Expert Systems with Applications 39, 1381–1387 (2012)
21. Thovex, C., Trichet, F.: Semantic social networks analysis. Soc. Netw. Anal. Min. 3, 35–49 (2012)
22. Wang, H., Terrovitis, M., Mamoulis, N.: Location Recommendation in Location-based Social Networks Using User Check-in Data. In: Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, SIGSPATIAL'13. ACM, New York, NY, USA, pp. 374–383 (2013)
23. Ye, M., Yin, P., Lee, W.-C.: Location Recommendation for Location-based Social Networks, in: Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS '10. ACM, New York, NY, USA, pp. 458–461 (2010)
24. Zanker, M., Jessenitschnig, M.: Case-studies on exploiting explicit customer requirements in recommender systems. User Model User-Adap Inter, Jour. Pers. Res., Springer 19, 133–166 (2009)