Contemporary Engineering Sciences, Vol. 8, 2015, no. 5, 197 - 207
HIKARI Ltd, www.m-hikari.com
http://dx.doi.org/10.12988/ces.2015.411245

A Concise Method for Modeling Profiles Using Semantic Approach

Senduru Srinivasulu

Department of IT, Sathyabama University
Chennai, Tamilnadu, India

P. Sakthivel

Department of ECE, Anna University
Chennai, Tamilnadu, India

Abstract

The web is a huge collection of unstructured data, where a large amount of data is added each and every day from multiple sources in various forms. We are actually drowning in information and starving for knowledge. The present day web (web 2.0) helps us to fetch any type of information what we wanted. But most of the information is irrelevant to the user's perspective. The user has to refine, extract knowledge from the obtained bulk of information. Hence web is not smart enough to fetch the exact information for the user. To make the web smarter, efficient and effective, semantics have been introduced to the web, which can fetch the exact information what a user wanted. To achieve smartness in web 3.0 few components like XML, RDF, Ontology, and Owl are used. One of the most important challenges faced by the web developers is personalizing a social web, it means extracting data from different sources of information. Ontology based representations are used to obtain relevant information to create meaningful relations among those data. The main aim of this paper is to generate profiles for a person, from a scientific point of view. The generated profile shows all the research made by the person, and with whom he has done the research and the time period to complete research. This model is based on FOAF (Friend of a Friend) model. The generated profiles are made for a small community which belongs to the computer science domain. The information present in the profile consists of the research activity of the person and his / her area of interests and publi-

cations of that person. The publications can be added by the author to his profile with some access policies to restrict the access to the full document. Some information's based on scientific performance extract from digital libraries, for example DBLP and ACM digital libraries. This paper demonstrates a user friendly graphical interface which allows the client to perform their actions. On one hand the client enters the name of the person and the server shows the FOAF profile of the name which is entered and it also highlights the attributes and collaborations of that person. On the other hand client can give a list of people and they can receive the articles which are common between them. The main impact of this approach has kept track of the research activities within us and also the relations and collaborations between people within their community.
The paper must have abstract.

Keywords: Semantic Web, Ontology, RDF, FOAF, OWL

1. Introduction

This Paper proposes modeling author profiles on the basis of their research activity. It also used to provide interaction between the authors and the students and with the co-authors. The profiles are split into two types one is for the Authors and other one is for general users. Each and every profile information is being converted into an RDF file which is stored in RDF data store. This makes the machine understand what kind of relationship is there between the user and the author [3].

In this system, there are two servers one is used to store the basic information of the profiles called a Database server. Another server is used to store the Publications of the profiles which is called the Admin server. Due to this performance of retrieving data can be done quicker. Admin Server has to authenticate the users and authors and the access policies given by the authors will authenticated by the admin. This is how database tier is being designed. This is how the proposed work is carried out.

2. Proposed Model

This paper aims to generate a profile for authors from a scientific point of view. Modeling a profile is based on the semantic approach in which it uses the FOAF model (i.e.) Friend of a Friend concept. [1] The modeled profiles are stored in RDFs file storage. This system consists of two servers. One server is used to store basic profile of the authors Named as a server. Another server stores the publication details of authors along with the access policies (Access Rules defined by the Authors), named as admin server. Registered users can visit the profile of the author's by entering the author's name in the user interface. The profile has been retrieved from the server by using site IDs. If the server has not found the site ID of the given author name, a message will be given to the user (Profile not found). If the server finds the site ID of the given author name, the user will be given with basic profile information. If the user's specification matching with author's specification, the user can view the publications of author's. The HIT rate of the author has been increased, if the user views the author's publication. The Download rate of the author has been increased, if the user downloads the author's publication. The Citation rate of the author has been increased, if the user included the author's publication in the reference section. System log will be maintained both in the user's log and in author's log.

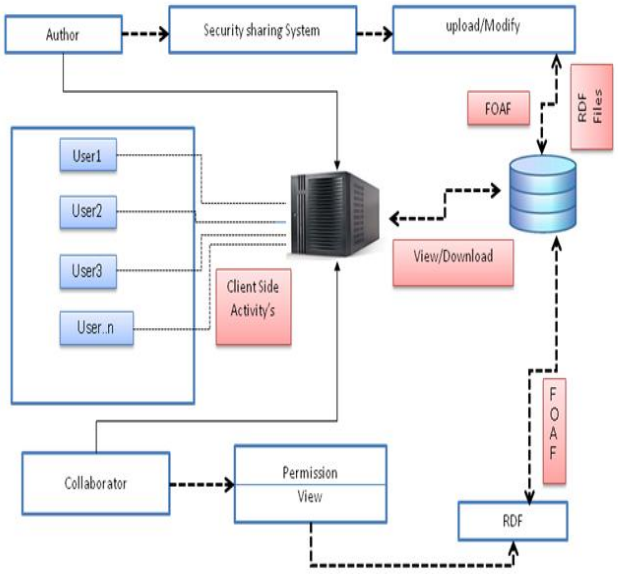
Author Profile consists of two types of data. One is the Basic information about the author and his/her research activity.[7] Another one is the Articles published by the author. Articles published by the author as two kinds of information. One is article basic content only will be viewed by the users (i.e.) Only the title and abstract will be given to the user. If the user is interested he/she can view the full article or by satisfying the authors access policies the user can download the entire article for reference. The authors can also follow authors who work on his same domain if interested.

User profile consists basic information of the user. The user has a search option to find authors. The user can search articles either by the author's name or by giving the article title. In case if the user has any queries he can communicate with the author by posting his query. This is how interaction has been provided between the user and author.

Authors can add their own publications with access policies assigned to the user. The publications from digital libraries can also be added in the profile. To access these articles from digital library the user can directly communicate with the author directly and get if it is required. If the article is viewed then hit rate will be increased for that particular article. If the article is downloaded by anyone the Download rate will be increased. If the article is added in the reference section in another, then a citation rating will be increased to that article. Using these Ratings the profiles are being rated and been ranked according to the rates [2].

The profiles are being stored as RDF files for each and everyone. The storage is being grouped into similar groups which is used to increase the performance of the system. The related profiles will be listed in the profile which is logged in with the domain match. Profile matching by the domain and area of interest of the profile which is logged in. All the profiles are being ranked according to the rating of their profile and their involvement in research activity.

3. System Architecture and System Flow Diagram



At the back end, initially a profile is created for each person. Each and every profile is an RDF file which has the information about a particular person. A site ID is created for research activity for that person. Using this site ID the information's can be obtained. The disk storage is organized in such a way that similar groups are created and stored separately. In the front end, Client sends request to the server with site ID and user name.

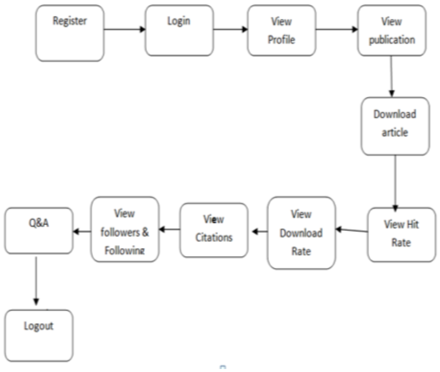


Fig. 2. System Flow Diagram

The server checks the name and the site ID in the database. If matches, the Server gives response to the client. The profile consists of two web pages. The first page contains all the person's collaborators, with their full names and site ID's. The second page is used to extract information, gives the person’s articles and URLs in the description of the articles. If the client clicks the URL, the information is displayed on the other page directly.

The above diagram represents the overall flow of the system. It also specifies the functions of the user and author. Initially the user has to register the system. Then the profiles are divided into two categories one is normal users and another one is authored. Authors can add their publications with their own with certain access policies to the users. If the user satisfies the policies only the publication can be viewed fully. Publications can be viewed, downloaded. Three types of ratings can be viewed and also the profile followers and following can be viewed.

4. Methodology

A. Registration and Login

Initially client has to register to access the application. Registration consists of two roles. They are

- Author
- User
- Collaborator

Author and user have to register the basic information into his/her profile. The Profile information is stored as an RDF file in an RDF Data store. After the registration author / user has to login for further interaction with the application. There are two main pages describing a person.

- The first page contains all the person’s collaborators, with their full names and their unique IDs.
- The second page used to extract information, gives the person’s articles, or more specifically, URLs in the description of the articles.

User’s pages are being structured in JSON (i.e) JavaScript Object Notation representation. User and Author, has to login after registration. Admin server is used to authenticate the username and password. When the user/author is logged into the page the basic profile information is being shown initially. This is how the registration and login is implemented.

In this registration is being used to model the profile. In Registration the form is created in PHP it contains some fields. The data's which are entered in text field are stored in a database. The database consists of a two tables one is stored the Author Information and another one is to store User Information. During registration there is a field called role to split the clients. The tables are named as the Author and another one is the wiser. After registration the data's are stored in respect tables. The passwords are encrypted using Md5 and then stored in a database.

At login, username and password are authenticated and verified before the providing the access to the profiles. The Password is verified by decrypting the password which is stored in the database. The profiles can be managed by the users and they can edit the profile information. In database, id of the profiles is set as Primary Key, so that the data's can be retrieved using the id. The profile information is stored in RDF so that the relationship between the profiles can be identified.

B. Add Publications

In this module author has a privilege to their articles in their Profiles. Author specifies certain Access Policies to provide access to the full Content of the article/publication [2]. Publications consists of a basic information of the article (i.e) Article Title and Article Abstract is being shown to the clients initially. To provide full access to the article the user as to satisfy the access policies this has been given by the author when it has been added. After satisfying the access policies user can download the full publication of the reference. These publications are stored in a Database server, which is controlled by the Administrator. Author specifies the domain of the article. The article is being grouped into similar groups according to the domains and stored in a database. Publications are retrieved using Publication Id. Initial basic information of the publication is shown to the user. If interested, he/she can view the entire document.

This module is used to add publications in which the publications are stored into a table Publications. Initially, the input obtained to add publications are the name of the publication, Domain of the publication, Abstract of the publication and Co-authors of the publication these fields are stored in the Database.[4] ID for publication is created in a session which is used to retrieve the publications. Access policies are given to each and every article these policies are verified by the administrator. Publications are stored in a different database with access policies.

Publications are displayed in the newest added publication in ascending order. Retrieving the information newest added is being separated in script logic. Publications from digital libraries are crawled using Regular expression and those data is displayed in profiles. Ratings such as Download, Hit, and Citation for each article are calculated. Data's are retrieved using the IDs in the data store. [8] These publications are grouped into similar domains and stored into the data store separately, so that it can be retrieved quickly. All the publications are created as RDF files and relationship are identified and grouped into similar items.

C. Search, View and Download Publications

In this module, article is searched in two ways as one using the Title and the other using the Author Names. After the search, the page navigates to the publication page of that particular author. Initially the publication is viewed by either a user or an author. If they have an interest in that publication they can access the full publications by satisfying access policies of the owner which is predefined. When user gets full access to the publication, he/she can download that publication of their Reference. When the search is made by the client, the server processes the request to the database server. And the server fetches the content and return to the requestor.

1) Search for a person: In this client has to use the person's name. If the client gives the correct name, then the particular information for that login is displayed and the client will be redirected to the requested page.If there is no such person's name in the data store the client will be notified that the requested page is not available.

2) Search for a collaborator: To find collaborators the client should specify the list of people separated by "," into the text box which is available on the application. Using that URL the user can click on it and gather information about that particular article. If in case one of the requested people does not have a FOAF profile a message is given to the Client. There are no profiles as such. If it is successful all the articles in common are listed out in the application.

D. Publication Ratings

In this module the Ratings are calculated for the publications. The Ratings are given in three types: they are Hit Rate, Download Rate and Citation Rate. The HIT rate of the author gets increased, if the user views the author's publication. The Download rate of the author gets increased, if the user downloads the author's publication. The Citation rate of the author gets increased, if the user included the author's publication in the reference section. The quality of the paper also can be determined with the number of citations of a particular paper. The higher the citation the high is the quality and reach of the paper [10].

E. Clustering

Clustering is the procedure used to group a number of similar things in order to achieve a common goal.[5] This goal may be either a complex computation, which requires more computing power than every single node can offer, or collecting data (sensor data, for example) from a wide geographic area or theoretical data as such[6]. The concept of similarity may be specified using many different parameters related to the specific problem we are trying to solve, such as domains, networks, profiles or combination of the above parameters. Here we have done two clustering as domain based clustering and author profile based clustering. Publications in a same domain's are grouped into a cluster [11]. The various domains are networking, data mining, wireless sensor networks and so on. Under the domain sub clusters are formed for profile and publication. Here FCM is used to cluster the publications and author profiles. FCM is an extension of K-means Algorithm FCM allows a data point to be assigned to more than one cluster. For example, but he may be a collaborator in his friend's work in Network

Security at that time FCM is needed. The Author must be placed in both Data Mining and Network Security Clusters [12]. This is how the Publications and Profiles of authors is grouped and formed as a Cluster.

In this module fuzzy, c means algorithm is implemented. This algorithm is used to group the similar items based on the domain and area of interest of the author's and users. Initially the cluster head is identified with the highest impact ratio of the profile. The following steps are given below:

Step 1: Randomly initializes Cluster Centre

Step 2: Create a distance matrix from a point Xi to each of the cluster centers by calculating the Euclidean distance between the data points and the cluster centre, using the formula

$$d_{ij} = \sqrt{\sum_{i=1}^{n} (x_i - c_j)^2}$$

Step 3: Fuzzy partitioning is carried out through an iterative optimization of the objective function shown above, with the update of membership d_{ij} and the cluster centers c_j.

F. Server Creation

In this work two servers are created one is a database server and another one application server. Database server is used to store the basic information about the profiles. The Application server is used to store the publications of authors with access policies. Data are stored as RDF files in the database. Each person has his or her own RDF. The main class consists of certain attributes of a person. The attributes are name of the person, the people he or she knows and a list of documents that represents his or her research. Parsing HTML pages from Java classes provides the information within the RDFs [7]. There is a mediator, who gathers the entire information and uses it to create the RDFs.

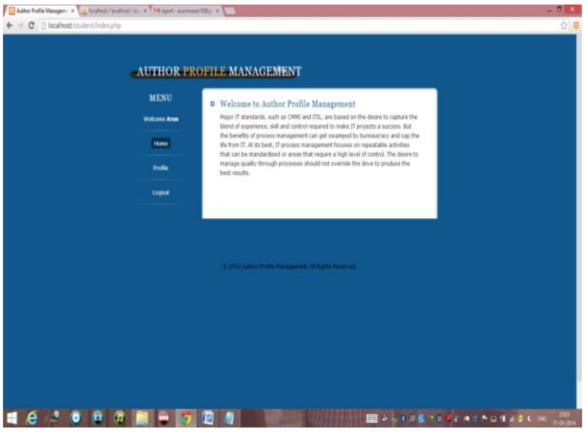
In this proposed methodology SQL Database is used to store and manage datasets. All the fields entered are stored into the Database. The database consists of four tables they are Author, User, files and publications. The Author table consists of the details entered while registering to the application ID of the author is created and then it is retrieved by using the Author ID stored in the database. User table consists of the user information. If the user edits the profile the data's in the dataset is updated automatically. These two databases are being stored in a different server which is called as application server.

Publication table consists of the details about the articles and the ratings of the publications. Publications are retrieved using the ID and the latest publication first. These publications are stored in a different database called as a DB server which consists of the access policies which is given by the authors when uploading. The policies are verified when there is a request to view the entire Article. This is how the server and database is managed.

5. Experimental Results

A. Screen Shots

Home page of the site is given below.



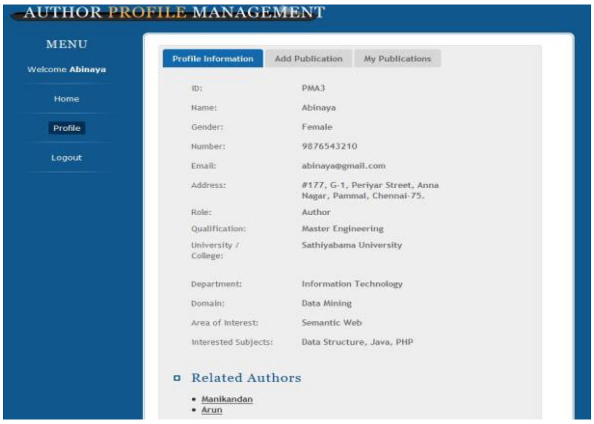
AUTHOR PROFILE MANAGEMENT

Welcome to Author Profile Management

Major IT standards, such as CMMI and ITIL, are based on the desire to capture the blend of experience, skill and control required to make IT projects a success. But the benefits of process management can get swamped by bureaucracy and sap the life from IT. At its best, IT process management focuses on repeatable activities that can be standardized or areas that require a high level of control. The desire to manage quality through processes should not override the drive to produce the best results.

© 2014 Author Profile Management. All Rights Reserved.

Fig.3.HomePage



AUTHOR PROFILE MANAGEMENT

ID:	PMA3
Name:	Abinaya
Gender:	Female
Number:	9876543210
Email:	abinaya@gmail.com
Address:	#177, G-1, Periyar Street, Anna Nagar, Pammal, Chennai-75.
Role:	Author
Qualification:	Master Engineering
University / College:	Sathiyabama University
Department:	Information Technology
Domain:	Data Mining
Area of Interest:	Semantic Web
Interested Subjects:	Data Structure, Java, PHP

Related Authors

- Manikandan
- Arun

Fig. 4. View Profile Page

View the profile of the author, if there is any change in the profile information it can be edited. View of the related author profile, the basic profile is viewed initially.

B. Experimental Results

In experimental results two types of analysis are conducted, one is to analyze each and every article and another one is to analyze the profile ratings.

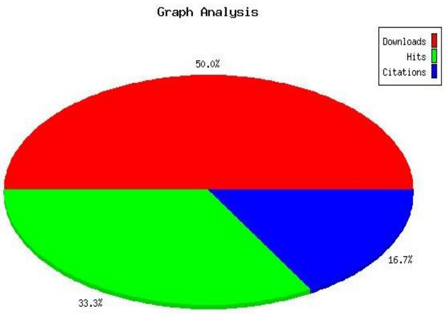


Fig. 5. Publication Analysis

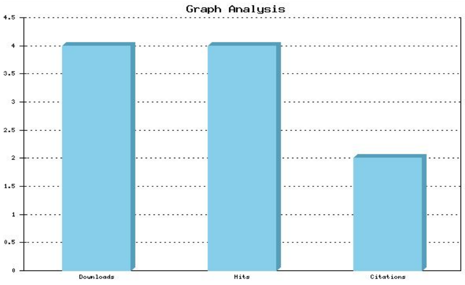


Fig. 6. Profile Ratings

The above graph represents the ratings of the article. It consists of three ratings Hit rate, Download rate and Citation rate. The above graph is used to analyze the profile rating. It is done by summing the hit rate, download rate and citation rate of all the publications which is present in his/her profile. Using these ratings profile is rated and ranked in the list.

6. Conclusion

The work paves the way to find similar people in the same domain, and also makes people to interact with each other those who belong to the same group. It also aims to make the author’s publication Secure. This enables resource sharing and it uses semantic FOAF model to provide interactions between the author’s and collaborators. This also ranks the author profiles and provides rankings based on the profile rate and hence improves the performance of the system. Thus the aim of enhancing the people interaction and secure resource sharing is achieved and the performance of the system is increased.

References

[1] Ana-Cristina Surdu, Florin Pop. 2013. Semantic Approach for Modeling Profiles and Interactions based on Digital Content. In 19th International Conference on Control Systems and Computer Science.
http://dx.doi.org/10.1109/cscs.2013.23

[2] Li Ding, Lina Zhou, Tim Finin, and Anupam Joshi. 2005. How the Semantic Web is Being Used: An Analysis of FOAF Documents. In Proceedings of the Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05) - Track 4 - Volume 04 (HICSS '05), Vol. 4. IEEE Computer Society, Washington, DC, USA, 113.3-.
http://dx.doi.org/10.1109/hicss.2005.299

[3] Shoaib, M.; Basharat, A., "Ontology based knowledge representation and semantic profiling in personalized semantic social networking framework," Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on on Computer Science and Information Technology, vol.5, no., pp. 95-99, 9-11 July 2010.
http://dx.doi.org/10.1109/iccsit.2010.5564449

[4] Uldis Bojārs, John G. Breslin, Vassilios Peristeras, Giovanni Tummarello, and Stefan Decker. 2008. Interlinking the Social Web with Semantics. IEEE Intelligent Systems 23, 3 (May 2008), 29-40.
http://dx.doi.org/10.1109/mis.2008.50

[5] Key-Sun Choi, Natural Language Processing an Knowledge Engineering (NLP-KE), 2011 7th International Conference, 27-29 Nov., 2011.

[6] YounHee Kim; ByungGon Kim; HaeChull Lim, "The index organizations for RDF and RDF schema," Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference, vol.3, no., pp.4 pp., 1874, 20-22 Feb. 2006. http://dx.doi.org/10.1109/icact.2006.206357

[7] Nicole Alexander and Siva Ravada. 2006. RDF Object Type and Reification in the Database. In Proceedings of the 22nd International Conference on Data Engineering (ICDE '06). IEEE Computer Society, Washington, DC, USA, 93-. http://dx.doi.org/10.1109/icde.2006.126

[8] Xi S. Guo, Mark Chaudhary, Christopher Dozier, Yogi Arumainayagam, and Venkatesan Subramanian. 2004. A web application using RDF/RDFS for metadata navigation. In Proceeedings of the Workshop on NLP and XML (NLPXML-2004): RDF/RDFS and OWL in Language Technology (NLPXML '04), Nancy Ide and Laurent Romary (Eds.). Association for Computational Linguistics, Stroudsburg, PA, USA, 17-24. http://dx.doi.org/10.3115/1621066.1621069

[9] Tim Berners-Lee and Mark Fischetti. 2001. Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by its Inventor. DIANE Publishing Company.

[10] Edlira Kalemi and Edlira Martiri. 2011. FOAF-Academic Ontology: A Vocabulary for the Academic Community. In Proceedings of the 2011 Third Int. Conf. on Intelligent Networking and Collaborative Systems (INCOS '11). IEEE Computer Society, Washington, DC, USA, 440-445. http://dx.doi.org/10.1109/incos.2011.94

[11] FOAF Software, From Wikipedia, the free encyclopaedia, http://en.wikipedia.org/wiki/FOAF (software), Accessed on March 30th, 2013.

[12] S. Gowri; G. S. Anandha Mala; “Improving Intelligent IR Effectiveness in Forensic Analysis” Institution of Computer Science Informatics and Telecommunication Engineering 2012, Page(s): 451. http://dx.doi.org/10.1007/978-3-642-35615-5_73

Received: December 8, 2014; Published: February 25, 2015