SWARMS: A Domain Exploration Tool and its Application in FOAF Domain

Liang Bangyong, Tang Jie, Li Juanzi, Wang Kehong
Department of Computer Science and Technology
Tsinghua University
100084, Beijing, China
{liangby97, j-tang02}@mails.tsinghua.edu.cn,
{ljz, wkh}@keg.cs.tsinghua.edu.cn

Abstract

In this paper, we present a system called SWARMS, for domain knowledge exploration and its application in the domain of Friend-Of-A-Friend (FOAF). SWARMS provides methods of search, visualization, and analysis to help users browse domain data in visualization views. We collected personal information from the web, constructed personal network based on the FOAF Ontology. So far, information of more than 30,000 persons and 90,000 papers were gathered. In SWARMS, we try to address semantic search, visualized and navigation, and domain data mining. This paper describes the architecture and implementation of SWARMS in FOAF domain.

1 Introduction

Social network analysis is an important research area in Semantic Web. The Friend-Of-A-Friend (FOAF) ontology [Brickley and Miller, 2005] has been proposed to describe persons' profile information (e.g. name, email-address, etc.) and relationships among them (e.g. Knows, Publications).

Recent work on FOAF was focusing on collecting existing FOAF annotation in web pages and visualizing them in a separated mode (e.g. http://www.foafnaut.org/ views each collected FOAF annotation as a graph).

In this work, we collected personal information from the web, constructed a personal network based on the FOAF Ontology. More than 30,000 persons and their publications were gathered.

As the number of instances stored in a domain data repository can be very large (in our case, we already have more than 100,000 instances), it is impossible to visualize all the instances in a static view. Questions arise for exploring the domain knowledge in the large scaled data: 1) How to help users understand the domain knowledge? 2)How to help users easily find an instance (e.g. a person in the FOAF domain)? 3) How to help users easily navigate across the complicate relationships among instances (e.g. the person's friend and publication relationships)? 4) How to help users find an 'important' instance (e.g. expert on a topic)?

SWARMS [Liang et al., 2005] is proposed to assistant users to browse large scale domain data. Several search methods have been implemented to help users find relevant instances. Visualization and navigation methods have been developed to help users navigate the instances' detailed information. Finally, mining methods are applied to help users deep the insight of the knowledge in the domain data.

2 Features and architecture

The main features of SWARMS are as follows:

1. Semantic search. It includes three types of searches: full-text search, constraint based search, and association search. In full-text search, the user inputs keywords and the system returns person instances that related to the keywords. In constraint based search, the user can specify constraints on properties of the person and the system returns instances that satisfy the constraints. For example, the user can input a query like "*?x foaf:knows Jack*" to search for all persons who know "*Jack*". Association search returns the possible associations between any two instances, for example, a simple association between two persons can be that the two persons coauthor a paper.
2. Visualization and Navigation. It includes three views: instances' context view, person view, and paper view, which are respectively designed for an instance's detailed information, a person's network and a person's publications.
3. Domain data mining. It includes social network analysis and expert finding. In social network analysis, we calculate the *degree centrality* and *betweenness centrality* [Wasserman and Faust, 1994] between two persons. In expert finding, we try to find expert on a topic.

Figure 1 shows the processing flow of our approach.

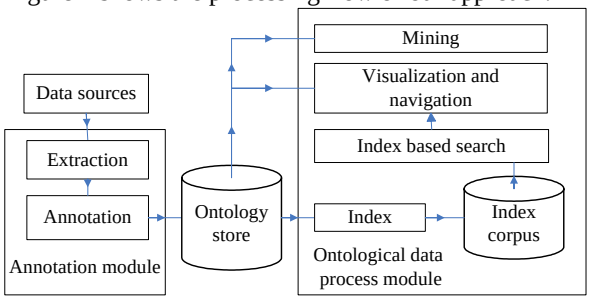

Figure 1. Processing flow of our approach

The system is composed of two main modules. One is *annotation*; the other is *ontological data process*.

The annotation module collects and extracts the personal information from the web and stores the extracted information into FOAF ontology base. The ontological data process module provides functionality for semantic search, visualization and navigation, and domain data mining.

The extraction is conducted automatically. Details of the extraction process will be reported elsewhere. The extracted information is stored in the ontology base. Index component constructs an indexing for the domain data. The indexing is the foundational model to support full-text search.

3 Implementation

We extracted personal information from the web and publication information from DBLP. The extracted personal information by the annotation module are used to create instances of the concept "*foaf:Person*". The extracted publication information are used to create instances of the concept "*foaf:Document*". A "*foaf:knows*" relation is created between two persons if they are co-authors of one paper. The "*foaf:publications*" and "*foaf:maker*" relation are created between a paper and its authors.

The domain data is indexed by Lucene (http://lucene.apache.org) and the index is used for full-text search. Users can use full-text search to find persons and then use visualization view to inspect the person's information in a interactive way. Figure 2 is an example for full-text based search.

Figure 2. Full-text search

In visualization view, the system visually displays information of the selected instance(s) (ref. Figure 3). The system also shows the relationships between persons. Figure 4 indicates the person's view in which the persons' social network analysis results are displayed.

In our current ontology store, there are more than 30,000 persons and 90,000 papers. Their relations were created. We use MySQL to store the ontology data, and Jena API is used to store and retrieve the data. SPARQL [Prud'hommeaux and Seaborne, 2006] is employed for constraint based search. We have extended JUNG (http://jung.sourceforge.net) in visualizing the ontology data.

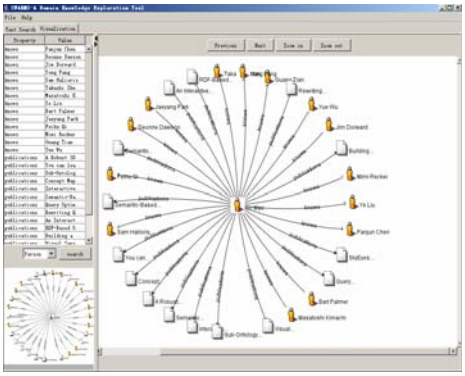

Figure 3. Visualization view of an instance

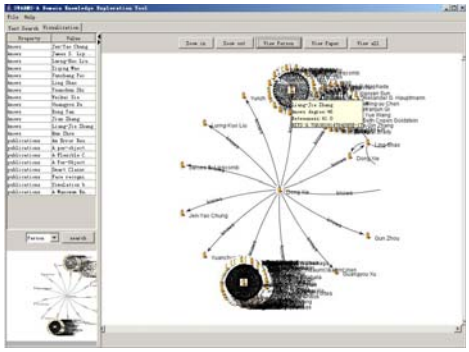

Figure 4. Person view and social network analysis

In this paper, we have developed a prototype system called SWARMS and applied it to domain of Friend-Of-A-Friend. Three main features: search, visualization and navigation, and domain data mining have been proposed and implemented in SWARMS.

Acknowledgments

The work is supported by the Natural Science Foundation of China with the grant No. 90604025.

References

[Brickley and Miller, 2005] Dan Brickley and Libby Miller. FOAF vocabulary specification, 2005.

[Prud'hommeaux and Seaborne, 2006] Eric Prud'hommeaux and Andy Seaborne, SPARQL Query Language for RDF, 2006.

[Liang et al., 2005] Bangyong Liang, Jie Tang, Gang Wu, et al. SWARMS: a tool for exploring domain knowledge in semantic web. In *Proceedings of AAAI'05 workshop: Context and Ontologies: Theory, Practice and Applications*, pages 120–124, July, 2005.

[Wasserman and Faust, 1994] Stanley Wasserman and Katherine Faust. *Social Network Analysis: Methods and Applications.* Cambridge University Press, 1994.

Demonstration Overview

A live software demonstration will be provided, consisting of interaction with the SWARMS tool in FOAF domain.

In the demonstration, three types of searches in SWARMS will be introduced. Specifically, full-text search, constraint based search, and association search. All the visualization views will be described in detail.

(1) In the demonstration of full-text search, users input keywords, and the system returns persons that related to the keywords. An example search result is shown in Figure 5.

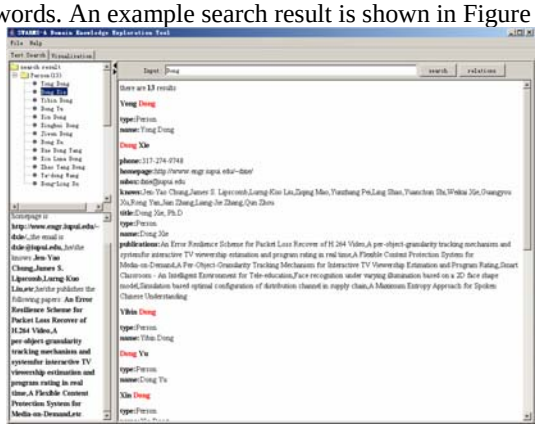

Figure 5. Full-text search

In Figure 5, the left-upper window lists persons found; the right window displays the detailed information of the selected person; and the left-bottom window shows the summary in natural language of the selected person.

(2) By double-clicking one person, the system will switch to the visualization view of the person. The visualization view gives the detailed information of the selected persons in graph. The nodes in the graph are instances and the edges are relations among them. Figure 6 is an example of the visualization and navigation view.

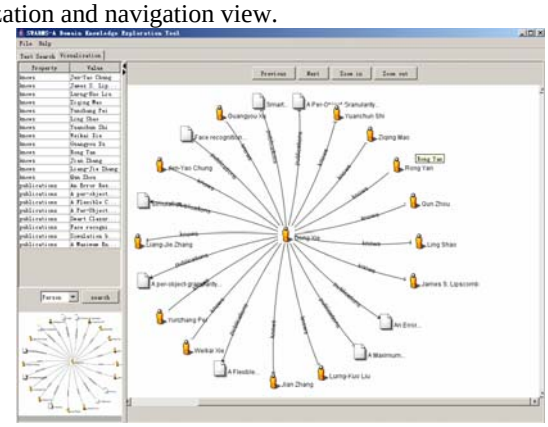

Figure 6. The visualization and navigation view

The left-upper window here displays property-value of the selected person. The right window is the visualized results of the selected person and his friends and his publications. The left-bottom window is the satellite view of right window. It is useful when the graph of the selected person is very large.

(3) In visualization, users can change to the person view, paper view, and instance's context view. In person view, the social network analysis results can be displayed by placing the Mouse cursor on the person's icon. Also, users can click nodes in the visualization to navigate to other instance's view.

(4) In the demonstration of constraint based search, users input the properties' values in the property-value window in the visualization view to conduct the constraint based search. Instances that satisfy the specified criteria will be found and shown in the visualization view.

(5) Users can use the association search to find possible associations between any two persons. Their associations will be shown in the visualization view (ref. Figure 7).

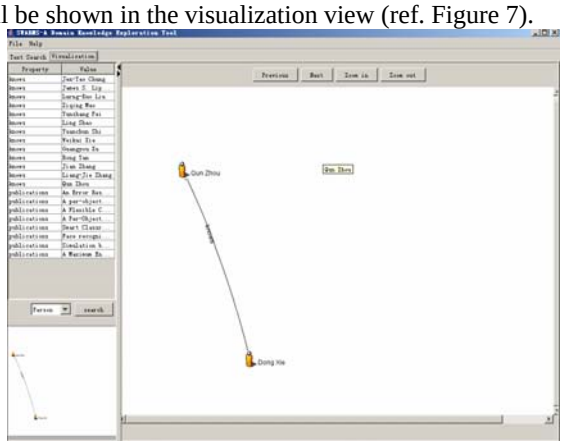

Figure 7. Association search

(6) In expert finding, users input keywords (called topic) to find experts on the topic. We make use of publications and documents a person involved to calculate his 'expertise degree' on a topic. The basic idea for the feature is that if a person has authored many documents on a topic, then it is very likely that he is an expert on the topic, or if the person's name co-occurs in many times with the topic, then it is likely that he is an expert on the topic.

(7) We will introduce the method we use to calculate the social network analysis results among the persons in research community.