Enterprise People and Skill Discovery Using Tolerant Retrieval and Visualization

Jan Brunnert, Omar Alonso, and Dirk Riehle

Hasso Plattner Institut, Potsdam, Germany
jan@brunnert.de
SAP Research, Palo Alto, USA
{omar.alonso, dirk.riehle}@sap.com

Abstract. Understanding an enterprise's workforce and skill-set can be seen as the key to understanding an organization's capabilities. In today's large organizations it has become increasingly difficult to find people that have specific skills or expertise or to explore and understand the overall picture of an organization's portfolio of topic expertise. This article presents a case study of analyzing and visualizing such expertise with the goal of enabling human users to assess and quickly find people with a desired skill set. Our approach is based on techniques like n-grams, clustering, and visualization for improving the user search experience for people and skills.

Introduction

Expertise identification requires data which is usually scattered among different enterprise systems, such as groupware, address books or human resources systems. Accessing this data is often difficult and not aimed at exploring organizational capabilities by topic. Search functionality is often string-based and many searches have to be submitted until the user can put together a mental picture of how different topics relate and who the relevant employees are. Enterprise people search is of critical importance: when decisions during a business workflow require an expert, when a co-worker needs help or when assisting customers it is critical to quickly find people that have certain expertise or interests. Cohen et al. [1] pointed out that it is crucial to understand where in the company expertise resides in order to establish efficient communication. By having a visual overview of the expertise map, the knowledge quality in an organization can be assessed. As outlined by Ashrafi et al. [2] understanding the quality of knowledge in an organization can be used to sense opportunities, develop strategies and implement them effectively and efficiently.
The goal in our case study was to implement a search and visualization application designed to be powerful yet easy to use. Search and retrieval capabilities have been identified by Stein and Zwass [3] as being of importance to the success of an enterprise knowledge management. Because the spelling of names is not always apparent in large international organizations, we had to apply forgiving search strategies. The application provides a novel interface that improves the user experience. Skills can be browsed using visual interaction components like a tag

cloud and a graph visualizing connections between tags. A secondary objective of the project was to evaluate how quickly one can bootstrap such an application using only open source components.

Tolerant Retrieval and Cluster Clouds

Users frequently enter imprecise queries that contain spelling errors or, in the case of names, phonetic variations. By using n-grams the application returns good search results even when terms or names were misspelled or not clearly known. Providing a result set with similar terms, users can quickly identify the correct term even when they are uncertain what they are looking for and initiate browsing from this starting point. Employees usually provide very coarse-grained, generic keywords describing their area of expertise, while users will search for very fine-grained, specific skills [4]. Also, in international organizations like SAP, the spelling of names is not often clear. A meaningful tag cloud of skills is automatically generated as proposed by John and Seligman in [5].

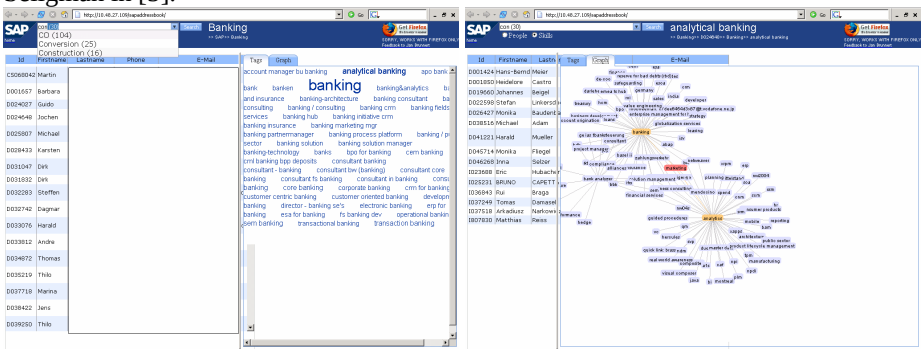

Fig. 1. AJAX auto complete-search, user list, tag clouds, and topic-graph visualization.

By clustering the tags around the most frequent terms into sets of related terms the total number of visible keywords is reduced into fewer, more meaningful entities. These skill-clusters are automatically discovered from the supplied set of keywords. For each keyword the total number of elements in a cluster determines the size of the cluster. This clustered set can now be displayed in a tag cloud that projects the size to the keywords' font-size, giving an intuitive view on the importance of a given skill. When drilling down, the application will shows terms similar to the selected tag, in effect showing the keywords that are contained in the cluster and relevant users having these skills are shown. When selecting users, their skills are shown in the visualization pane for further exploration. To support the user on their mental journey, the user's navigation path is always visible and the breadcrumbs can be used to return to previous steps.

The clustering algorithm starts by picking the most frequently used keyword and then searches for terms having a similarity value (as determined by the trigram module) within a certain range. The total number of occurrences of these words is then calculated and saved with the picked keyword. All words identified in this step are

removed from the set of available words. This process iterates until no more words can be found. The total number is cluster generated from the data set is 2880.
The application also puts terms in relation to each other in order to visualize them in a graph. We decided to generate the graph based on keyword co-occurrence as suggested by Widdows et al [6] with the edges being weighted by number of co-occurrences. Two keywords are connected when one or more person has specified both keywords in their profile (6355 pairs were identified in the production set).

Architecture and Implementation

The browser client follows a two-view paradigm: for every action the user takes, relevant data is displayed in all view modules. For example, when selecting a term, matching persons are shown and the tag cloud is repopulated with related tags. Likewise, selecting a person shows the person's keywords for further drilling down. The entire interface is built on the Dojo toolkit [7]. All rendering is done in these visualization components, so that the client application just has to pass on the data received from the server into the visualization modules. The result is a design where model, view and controller are nicely separated. A custom tag cloud component was developed to make the use of tag clouds easier in other applications. It automatically adjusts font-size within a given range depending on a tag's associated weight and features a fade-over effect for an increased visual browsing experience. The graph connecting keywords based on their co-occurrence in persons' profiles is displayed using the prefuse toolkit, which uses a force-based algorithm to align the graph [8].

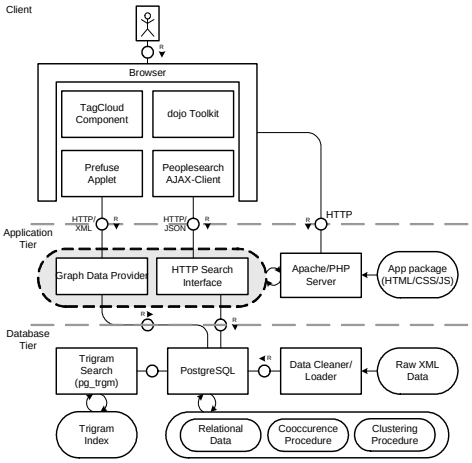

Fig. 2. Architectural Overview

A back-end had to be built to accept data from multiple data sources that are loaded into a single database. In the prototype, keywords are extracted from free-text fields in an XML dump of the corporate address book. The data is then loaded into a PostgreSQL relational database. The database schema contains trigram indexes on the person names and keywords, which is made possible by the pg_trgm [9] module. The database currently contains all of SAP's employees world-wide, around 41000 keywords (15000 distinct) showing the power of the setup on a real-life production dataset. The backend provides facilities to cluster similar keywords and determine the importance of the cluster in the organization. This is accomplished by having a stored procedure in the database. Co-occurrence is also done close to the database using a PL/pgSQL procedure.

Search queries coming from the AJAX-interface are passed on to the PostgreSQL backend via the web server application layer. The returned JSON data can be used to drive multiple visualization modules. Data is served to the prefuse applet in a similar way, providing XML data of edges and nodes in the graph, derived from the co-occurrence data.

User Evaluation

A preliminary evaluation survey was conducted among users to get feedback on the prototype. From the ten users that responded to the questionnaire, all reported that the tool was useful compared to the existing system. In a scale 1 to 5 (1=bad, 5=excellent), the people search feature averaged 3.4, the skill search 3.8, the cluster cloud 3.87, and the graph visualization 3.4. For the short development lifecycle of the prototype, the results are encouraging.

Future Work

The next step is the integration of additional data sources like documents collections with rich meta-data [10], e-mail archives [11] and publications [12]. Temporal and spatial search mechanisms are also planned.

References

[1] Cohen, W.M., and Levinthal, D.A. "Absorptive Capacity: A New Perspective on Learning and Innovation" *Administrative Science Quarterly* (35:1), 1990, 128-152
[2] Ashrafi, N. *et al.* "Boosting Enterprise Agility via IT Knowledge Management Capabilities" Proceedings of the 39th Hawaii International Conference on System Sciences, 2006
[3] Stein, E.W. and Zwass, V. "Actualizing Organizational Memory with Information Systems" Information Systems Research (6:2), 1995, 85-117
[4] Balog, K. *et al.* "Formal Models for Expert Finding in Enterprise Corpora" SIGIR'06, Seattle, WA, USA.
[5] John, A. and Seligmann, D. "Collaborative Tagging and Expertise in the Enterprise" WWW 2006, Edinburgh, UK.
[6] Widdows, D. *et al.* "Visualisation Techniques for Analysing Meaning" Fifth International Conference on Text, Speech and Dialogue, Brno, Czech Republic, 2002, 107-115.
[7] http://dojotoolkit.org/
[8] http://www.prefuse.org/
[9] http://www.sai.msu.su/~megera/postgres/gist/
[10] Reichling, T. *et al.* "Matching Human Actors based on their Texts: Design and Evaluation of an Instance of the ExpertFinding Framework" GROUP'05, Sanibel Island, FL, USA.
[11] Campbell, C.S. *et al.* "Expertise Identification using Email Communications" CKIM'03, New Orleans, LA, USA.
[12] Tho, Q.T. *et al.* "A Web Mining Approach for Finding Expertise in Research Areas" Proceedings of the 2003 International Conference on Cyberworlds (CW'03).