Semantic Matchmaking for Job Recruitment: An Ontology-Based Hybrid Approach

Maryam Fazel-Zarandi

Department of Computer Science, University of Toronto

10 King's College Road, Toronto, ON, M5S 3G4, CANADA

Email: mfazel@cs.toronto.edu

Mark S. Fox

Department of Mechanical and Industrial Engineering, University of Toronto

Email: msf@eil.utoronto.ca

Abstract

Human Resources Management (HRM) is the strategic management of the employees, who individually and collectively contribute to the achievement of the strategic goals of an organization. Many HRM tasks are based on locating and matching individuals to positions. In this paper we present an ontology-based hybrid approach to effectively match job seekers and job postings. The approach uses a deductive model to determine the kind of match between a job seeker and a posting, and applies a similarity-based approach to rank applicants.

Keywords. Semantic Hybrid Matching, Recruitment, Job Search

Semantic Matchmaking for Job Recruitment: An Ontology-Based Hybrid Approach

In today's competitive business environment, companies need to accurately grasp the competency of their human resources in order to be successful. This is particularly important for organizations that engage with multiple and changing clients such as consulting firms and software development companies since these organizations need to be able to flexibly respond to internal and external demands for skills and competencies. As such, it is often necessary to reason about skills and competencies of individuals. This is the case for human resource recruiting, selecting individuals for teams based on different skills and qualifications, determining who to train and what training program to offer, and recommending the right expert to individuals for acquiring information or learning from within the organization.

In order to facilitate the management of available human resources' competencies, provide a global view of competencies available at the organizational level, and perform qualitative and quantitative reasoning about available and required skills and competencies, the development of totally or partially automated techniques has received the attention of both researchers and organizations (e.g., Colucci et al, 2003; Bizer et al, 2005; Malinowski et al 2006). In addition, the Internet has also been increasingly used for HRM purposes in recent years. For human resource recruiting, for example, the Internet is currently being mainly used to place online job advertisements, to perform resume search, and to acquire information about skills and competencies of individuals (Dafoulas et al, 2003). The International Association of Employment Web Sites[2] reports that there are more than 40,000 employment sites serving job seekers, employers and recruiters worldwide. The main reasons for the use of online resources are the opportunity to reach and attract a larger number of individuals and the ability to process and track a larger number of applications faster and more cost-effectively (Laumer and Eckhardt, 2009).

In this work, we focus on locating and matching individuals and positions, a process important for hiring and team staffing. Different matchmaking approaches exist in the literature which can be used for matching individuals to job requirements. For example, typical text-based information retrieval techniques such as database querying and similarity between weighted vectors of terms have been used in previous works (Veit et al, 2006). Techniques for ontology-based skill-profile matching have also been considered. (Lau and Sure, 2002) proposes an ontology-based skill management system for eliciting employee skills and searching for experts within an insurance company. (Liu and Dew, 2004) presents a system which integrates the accuracy of concept search with the flexibility of keyword search to match expertise within academia. (Colucci et al, 2003) proposes a semantic based approach to the problem of skills finding in an ontology supported framework. They use description logic inferences to handle the background knowledge and deal with incomplete knowledge while finding the best individual for a given task or project, based on profile descriptions sharing a common

[2] http://www.employmentwebsites.org/

ontology. Approaches for calculating the structural similarity between instances on the basis of ontologies have also been considered. (Bizer et al, 2005) and (Mochol et al, 2007), for example, present a scenario for supporting the recruitment process with semantic web technologies within the German Government which uses (Zhong et al, 2002)'s similarity measure to evaluate the degree of match between job offers and applicants.

In general, matchmaking strategies that are based on purely logic deductive facilities present high precision[4] and recall[5], but are often characterized by low flexibility (Bianchini et al, 2007). Similarity-based approaches, on the other hand, are characterized by high flexibility, but limited precision and recall (Bianchini et al, 2007). Flexibility refers to the ability to recognize the degree of similarity when an exact match does not exist. Having flexible matchmakers is of fundamental importance particularly in the context of human resources recruitment since in real world situations it is rarely the case that individuals match all the required competences for a job. Although some scholars (e.g., Bizer et al 2005) have proposed using taxonomic similarity to rank applicants, the usefulness of this technique in different contexts and environments is not clear. There may be some cases, for example, where the *is-a* relation is not sufficient to express the relation between different skills. Let us give a simple example. Assume we need someone with object-oriented programming skills. If an employee knows Smalltalk programming then we can conclude that this person qualifies, since Smalltalk is a pure object-oriented programming language and as such Smalltalk programming is a specialization of object-oriented programming. However, if we have a C++ programmer, we cannot make such a strong conclusion since although C++ supports object-oriented programming one does not have to program in such way in C++.

To improve the matching process and provide an adaptive, flexible and efficient job offering and discovery environment, we combine different matchmaking strategies. We propose to first use a deductive model to determine the kind of match between an individual and a job posting, and then based on the kind of match determine the similarity measure to use in order to rank the applicants with partial match.

The remainder of this paper is organized as follows: Section 1 presents the underlying ontology. Section 2 describes the matchmaking model, and Section 3 presents the ranking algorithm. Finally, Section 4 concludes the paper with a discussion of contributions made and areas of future work.

Ontological Framework

In human resource recruiting, two perspectives are distinguished. A job seeker creates an application by specifying his/her academic background, previous work experience, and set of

---

[4] Precision is a measure of exactness or fidelity. In information retrieval, it is the *number of relevant documents retrieved* by a search divided by the *total number of documents retrieved* by that search.

[5] Recall is a measure of completeness. In information retrieval, it is the *number of relevant documents retrieved* by a search divided by the *total number of existing relevant documents*.

competences. A recruiter, on the other hand, creates a job posting in the form of a set of requirements in terms of job related descriptions and constraints on skills, proficiency levels, and/or degrees.

We use description logics (DL) with rules to represent and reason about applications and job postings. The expressions can be represented in OWL-DL, corresponding to the *SHOIN(D)* family of description logics. For simplicity when writing rules we use *p* to denote skilled person, *c* denote competence, *s* denote skill, *j* denote job posting, *fl* denote formal learning activity, *nfl* denote formal learning activity, *d* denote degree program learning activity, and *e* denote experience.

Skill

There are several definitions of competency present in the literature (De Coi et al, 2007). The definition we assume is the one given by the HR-XML Consortium work group[6]: “a specific, identifiable, definable, and measurable knowledge, skill, ability and/or other deployment-related characteristic (e.g. attitude, behavior, physical ability) which a human resource may posses and which is necessary for, or material to, the performance of an activity within a specific business context.”

We adopt this definition for its emphasis on measurable knowledge and skills and the connection between competencies and activity performance. Hereafter, we focus on measurable skills possessed by human resources and may use skill instead of competency[7]. We use the term *competence* to refer to a skill along with a level of proficiency.

We assume skills in a specific domain of interest. For our reference ontology, skills are semantically organized in a skill taxonomy (i.e., skill specialization/generalization with an *is-a* semantics). We do not include skills related to specific tools and technologies in this taxonomy. This is due to the fact that tools and technologies used may provide different set of functionalities and capabilities. The Smalltalk-C++ example given in the introduction falls into this category. As another example, consider the skill of working with Microsoft Office Excel. This may suggest competency in working with spreadsheets, plotting graphs, and/or macro programming. To include skills related to specific tools and technology, we extend our simple skill taxonomy with the *part-of* relation. For example, in the above situation object-oriented programming would be defined as *part-of* C++ programming.

In addition to *is-a* and *part-of* relations, we define the symmetric *alternative-for* relation between two tools or technologies. Two skills related to tools and technologies can be thought of as alternatives if at least one skill exists that is *part-of* both of them. For example, working with Java Servlet is an *alternative-for* working with JSP, or programming in Java is an *alternative-for* working with C++.

[6] http://hr-xml.org
[7] The reader should note that skill is not a synonym for competency, as it only covers part of its scope.

Competence

In our model, competences are general descriptions, independent of specific individuals or job descriptions. A competence statement refers to a skill along with a proficiency level. Different quantitative and qualitative measurement scales exist for evaluating an individual against a skill. Examples include Rating scales, Behaviorally Anchored Rating scales, and Threshold scales (Moyer, 2001). Rating scales are the most popular and typically consist of a numeric scale with a brief description of each number's corresponding meaning. The disadvantage of these scales, however, lies in their inconsistent interpretations across users of a scale (Moyer, 2001). To overcome the disadvantage of rating scales, we define a proficiency level in terms of the required level of knowledge and years of experience. We distinguish between four levels of knowledge: basic, intermediate, advanced, and expert knowledge (expert subsumes advanced which in turn subsumes intermediate which in turn subsumes basic). The years of experience is specified as the minimum number of years required.

$$\texttt{Competence} \equiv \exists_{=1}\texttt{refers-to} \sqcap \exists_{=1}\texttt{has-knowledge-level} \sqcap \exists_{=1}\texttt{has-years-of-experience} \qquad \text{(D-1)}$$

Using the HR-XML definition, having a particular skill becomes tightly bound to the evidence that suggests one has the certain skill at a particular level of proficiency. The evidence also helps in understanding how a skill can be achieved, which is especially useful for arranging training programs. In this regard, we distinguish between learning activity and demonstration of a skill. A learning activity[8] (LA) is an activity that has one or more *learning outcomes* associated with it and occurs within a particular context[9] (Gráinne and Fill, 2005). A learning outcome is what the learners should know or be able to do after completing the LA. Demonstration of a skill, on the other hand, indicates the experience one has in performing the tasks that require the particular skill. A demonstration of a skill is represented by the concept:

$$\texttt{WorkExperience} \equiv \exists_{=1}\texttt{hasPosition} \sqcap \exists_{=1}\texttt{atOrganization} \sqcap \exists_{=1}\texttt{has-start-date} \sqcap \exists_{=1}\texttt{has-end-date} \sqcap \exists\texttt{requires.Competence} \qquad \text{(D-2)}$$

A learning activity can either be formal or non-formal. Figure 2 illustrates the learning activity taxonomy. Formal learning occurs as a result of instructor-led programs within the curricula of educational institutions or the courses or workshops offered by different agencies (Schugurensky, 2000). Non-formal learning, on the other hand, involves the pursuit of knowledge or skills outside such settings, for example, learning achieved through reading books, engaging in self-study programs, or collaborating with communities of practice.

---

[8] The definition of a learning activity can be extended to include learning and teaching approaches adopted and the tasks undertaken.

[5] The validity and reliability of the assessments of the outcome of the learning activities are outside the scope of this paper.

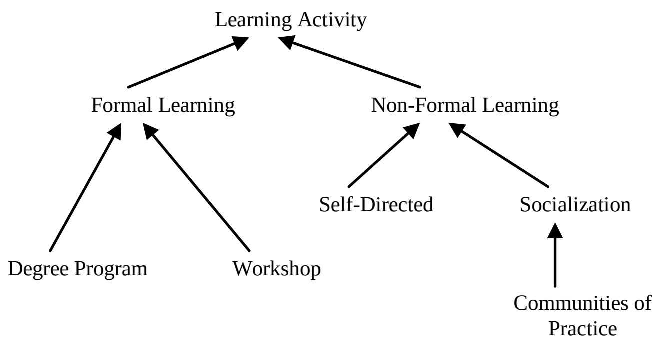

Figure 2. Learning activity classification

Formal learning activities can have a set of competences as required preconditions (*has-precondition*), but must have at least one competence statement as outcome (*has-outcome*). For non-formal learning activities the set of preconditions and outcomes may not be so clear. For these activities we define the relation *covers* indicating that a non-formal learning activity covers topics related to a certain skill. For brevity we only include the definition for degree program with will be used later for skills-requirements matching:

$$\texttt{DegreeProgram} \equiv \texttt{FormalLearning} \sqcap \exists_{=1}\texttt{has-degree-title} \sqcap \exists_{=1}\texttt{has-study-field} \sqcap \exists_{=1}\texttt{from-insitution} \sqcap \exists_{=1}\texttt{has-start-date} \sqcap \exists_{=1}\texttt{has-end-date.Date} \qquad \text{(D-3)}$$

Having these definitions, we can define a skilled person as a person who has taken some learning activities, has some work experiences, and has a set of competence statements:

$$\texttt{SkilledPerson} \equiv \texttt{Person} \sqcap \exists_{\geq 0}\texttt{has-taken} \sqcap \exists_{\geq 0}\texttt{has-experience} \sqcap \exists_{\geq 1}\texttt{has-competence} \qquad \text{(D-4)}$$

Considering learning activities, we can infer that an individual has a skill at a level of proficiency if the individual has completed a formal learning activity and the skill is either a precondition or outcome of the learning activity:

$$\texttt{has-taken(p,fl)} \wedge \texttt{has-precondition(fl,c)} \supset \texttt{has-competence(p,c)} \qquad \text{(R-1)}$$

$$\texttt{has-taken(p,fl)} \wedge \texttt{has-outcome(fl,c)} \supset \texttt{has-competence(p,c)} \qquad \text{(R-2)}$$

If, however, the individual has participated in a non-formal learning activity, then it can only be suggested that the individual may have the desired skill:

$$\texttt{has-taken(p,nfl)} \wedge \texttt{covers(nfl,s)} \supset \texttt{may-have-skill(p,s)} \qquad \text{(R-3)}$$

Considering demonstration of a skill, we can infer that an individual has a competence if s/he has an experience which requires the related skill at a particular level of proficiency:

$$\texttt{has-experience(p,e)} \wedge \texttt{requires(e,c)} \supset \texttt{has-competence(p,c)} \qquad \text{(R-4)}$$

If the experience requires a skill related to the use of a tool or technology, then the use of the tool can suggest having the skills that are part of it:

$$\texttt{has-experience(p,e)} \wedge \texttt{requires(e,c)} \wedge \texttt{refers-to(e,s)} \wedge \texttt{part-of(s',s)} \supset \texttt{may-have-skill(p,s')} \quad \texttt{(R-5)}$$

Job Posting

We define a job posting as a set of requirements in terms of job related descriptions and constraints on competences. Every job posting is represented using the DL formalism as the conjunction of:

- A concept in the form $\exists\texttt{has-description.JobDescription}$, where

  $$\texttt{JobDescription} \equiv \exists_{=1}\texttt{has-position-title} \sqcap \exists_{=1}\texttt{has-brief-description} \sqcap \exists_{=1}\texttt{has-category} \sqcap \exists_{=1}\texttt{at-company} \sqcap \exists\texttt{has-function.JobFunction} \quad \texttt{(D-5)}$$

  Example categories include administrative, engineering, and customer care.
- One or more concepts in the form $\exists\texttt{has-requirement.Competence}$ representing the set of required competences for the job.
- Zero or more concepts in the form $\exists\texttt{has-degree-requirement.DegreeRequirement}$ representing required degree program learning activities;

  $$\texttt{DegreeRequirement} \equiv \exists_{=1}\texttt{requiresDegree} \sqcap \exists_{=1}\texttt{requiresField} \quad \texttt{(D-6)}$$

- Zero or more concepts in the form $\exists\texttt{has-nice-to-have-requirement.Desire}$, where

  $$\texttt{Desire} \equiv \texttt{Competence} \sqcap 6\ \exists_{=1}\texttt{hasDesireLevel} \quad \texttt{(D-7)}$$

  where, `hasDesireLevel` can take an integer value in the range [1, 10].

Skills-Requirements Matchmaking

When searching for jobs (or applicants), a job seeker (or recruiter) can ask for all job postings (or applications) that match her application. In skills-requirements matchmaking, we are interested in determining whether or not an individual satisfies a set of requirements. We distinguish between must-have and nice-to-have requirements when matching. Must-have requirements are hard constraints whereas nice-to-have requirements are soft constraints (or preferences) that are taken into account when ranking.

We propose to first use a deductive model to determine the kind of match between an individual and a job description, and then based on the kind of match determine the similarity measure to use in order to rank the applicants with partial match.

Logic-Based Matching

Let $\boldsymbol{P}$ be a job posting with a set of requirements $\{ d\_req_P^i, c\_req_P^k \}$, where $d\_req_P^i$ is the i-th degree requirement, and $c\_req_P^k$ is the k-th competence requirement of $\boldsymbol{P}$. Let D be the conjunction of the following terms:

- For each $d\_req_P^i$, requiring degree d_i in field f_i,

$$\mathit{term}_i = \exists\texttt{has-taken.}(\exists\texttt{has-degree-title.}d_i \sqcap \exists\texttt{has-study-field.}f_i)$$

A *qualified* match denotes that an individual satisfies all the required competence and degree requirements of $\boldsymbol{P}$. In order to determine a qualified match, we create a new concept C_1 as a conjunction of D and the following terms:

- For each $c\_req_P^k$, $\mathit{term}_k = \exists\texttt{has-competence.}c\_req_P^k$

All instances of C_1 are qualified matches for $\boldsymbol{P}$.

In real world situations, however, it rarely happens that applications match all the requirements specified in a job posting. A gap between the set of requirements and the set of competences of an individual may exist for different reasons. It might be the case that an individual is not proficient enough in a specific skill or in worst case does not satisfy a competence requirement at all. For this, in addition to the qualified match, we consider different types of under-qualified matches.

For the first type of under-qualified match, we relax the required proficiency level constraints. In this case, an application is considered to be *proficiency-under-qualified* match for job posting $\boldsymbol{P}$ if and only if[11] the required proficiency level for one or more skills is not satisfied. To determine such a match, we create a new concept C_2 as a conjunction of D and the following terms:

- For each $c\_req_P^k$ referring to skill s_k,

$$\mathit{term}_k = \exists\texttt{has-competence.}c\_req_P^k \sqcup 7\ \exists\texttt{has-competence.}(\exists\texttt{refers-to.}s_k)$$

All instances of C_2 are proficiency-under-qualified matches for $\boldsymbol{P}$.

The second type of under-qualified match, *competency-under-qualified* match, takes into account the fact that it is not always the case that all the required skills are present in an application. For this type of match we first consider individuals who may have the missing skill(s). To determine such a match, we create a new concept C_3 as a conjunction of D and the following terms:

- For each $c\_req_P^k$ referring to skill s_k,

$$\mathit{term}_k = \exists\texttt{has-competence.}c\_req_P^k \sqcup \exists\texttt{may-have-skill.}s_k$$

All instances of C_3 are competency-under-qualified-case-1 matches for $\boldsymbol{P}$.

[11] For now we consider all degree requirements to be hard constraints.

Finally, we consider all individuals who satisfy a subset of the required competences. In order to reduce the search space, we first find all individuals who satisfy at least one of the required competences. Next, for each application found we solve a Concept Abduction Problem (CAP) (Colucci et al, 2007) to find the missing skills. The solution of a CAP can be inte rpreted as what has to be hypothesized in an application $\boldsymbol{A_j}$ and added to it in order to make it a match for $\boldsymbol{P}$. To do this, we solve a CAP for each $\boldsymbol{A_j}$ and a new concept C_5 which is a conjunction of the following terms:

- For each $c\_req_P^k$ referring to skill s_k, $term_k$ = ∃has-competence.(∃refers-to.s_k)

Having the solution to the CAP for each $\boldsymbol{A_j}$, we can consider only those that have fewer missing skills. To achieve a better match it is possible to iterate through all the requirements that are not satisfied, replace a skill at a time with its parent (which is a more general skill) and check to see if $\boldsymbol{A_j}$ satisfies this new requirement.

Similarity-Based Ranking

In order to rank the applications matched to a job description, we need to consider two scenarios. The first scenario involves ranking the set of under-qualified applications. The second scenario involves considering nice-to-have requirements or preferences for finding the most suitable applicants in the set of all applications.

Ranking Under-qualified Applicants

To rank applications that are proficiency-under-qualified, we define a dissimilarity measure and rank applicants accordingly.

$$dissimilarity(P, j) = \sqrt{\sum_i \left\{[(k_P^i > k_j^i)(k_P^i - k_j^i)]^2 + [(e_P^i > e_j^i)(e_P^i - e_j^i)]^2\right\}}$$

where, k_P^i (e_P^i) is the normalized required knowledge level (experience) of skill requirement i of $\boldsymbol{P}$, and k_j^i (e_j^i) is the normalized knowledge level (experience) of application $\boldsymbol{A_j}$ for the matching skill. In case one criterion is more important than the other, it is possible to consider a weighted sum of knowledge level and experience.

To rank applications that are competency-under-qualified-case-1, we simply count the number of may-have skills and rank applicants accordingly. To rank applicants that are missing one or more skills, we consider the size of the set of their missing skills. We then use the dissimilarity measure to rank those applicants that have the same number of missing skills.

Considering Nice-To-Have Requirements for Ranking Applicants

To find the most suitable applications in the set of all matched applications (both qualified and under-qualified), we take into account the desire level values, $u(ds_i)$, assigned to each nice-to-have requirement (desire) by the recruiter and normalize them to 1 (i.e., $\Sigma\, u(ds_i) = 1$). We can write the global match degree as the sum of the desire levels of the satisfied desired skills:

$$sim(P, j) = \sum x_{ji} \times u(ds_i)$$

where, x_{ji} is the Boolean variable indicating whether desire i is satisfied by applicant $\boldsymbol{A_j}$ in the set of all qualified applications. To calculate x_{ji}, for each desire a term similar to $term_k$ is created and then instance checking is done to see if $\boldsymbol{A_j}$ is an instance of this term. Note that this function is used to rank applications that are considered equally good with respect to the previous measures.

Empirical Results

We have collected data on individual's skills from an e-retail company and tested our approach with this data to compare the different matching and ranking criteria. We will send you the results once they are finalized.

Conclusions and Future Directions

This paper presented an approach to matching job seekers and job postings which takes advantage of the benefits of both logic-based and similarity-based matching. In other words, this hybrid approach presents high precision and recall while being flexible. The approach first uses a deductive model to determine the kind of match between an individual and a job posting, and then based on the kind of match determines the similarity measure to use in order to rank the applicants with partial match.

In addition to satisfying advertised job requirements, other factors such as recommendations, cultural fit, ability to adapt to the company's marketplace and ability to grow with the organization play an important part in selecting employees. Furthermore, when considering individuals for teams, complexities may arise due to fitness between an individual and other team members. It would be interesting to see how these complexities can be supported by automated techniques.

The basis for HRM is the accurate grasp of the competency of human resources. Currently the approach relies on self declarations of learning activities and experiences which can be inaccurate or insufficient. In addition, assessments need to be valid and reliable. Assessment results often lack validity and reliability due cognitive biases and inability to adequately gather human resource growth information among other things (Seta et al, 2005). It would be interesting to use mechanisms to automatically discover up-to-date competency information from secondary sources such as codes, documents, and forums. For this the domain ontology can be used to automatically annotate existing information resources and to perform automated reasoning to improve the detection and extraction of indicators of expertise (Fazel-Zarandi and Yu, 2008). Another useful ontology in this regard is the organization ontology (Fox et al, 1997) which formalizes the organizational structure and can be used to infer skills and expertise based on the roles that the agents play and the communications that occur among them. The knowledge provenance and trust ontologies presented in (Huang, 2008) are other examples of ontologies which can prove to be useful in this context. These ontologies can be used to formally define the semantics of information sources, information dependencies, relationships between information sources and experts, and trust relationships to improve competency recognition and extraction and reduce fluctuation in competency evaluation.

Acknowledgements

This research is supported, in part, by the Natural Science and Engineering Research Council of Canada.

References

Bianchini, D.; De Antonellis, V.; Melchiori, M. (2007). Flexible Semantic-Based Service Matchmaking and Discovery, *World Wide Web Journal*.

Badica, C.; Popescu, E.; Frackowiak, G.; Ganzha, M.; Paprzycki, M.; Szymczak, M.; Park, M.W. (2008) On Human Resource Adaptability in an Agent-Based Virtual Organization; *Studies in Computational Intelligence*, Vol. 134, pp. 111-120.

Biesalski, E.; Abecker, A. (2005) Human Resource Management with Ontologies, *Wissensmanagement. Professional Knowledge Management, Third Biennial Conference*, LNAI Vol. 3782, 499-507.

Bizer, C., Heese R., Mochol, M., Oldakowski, R., Tolksdorf, R., Eckstein, R. (2005) The Impact of Semantic Web Technologies on Job Recruitment Processes; in *Proceedings of the 7th International Conference Wirtschaftsinformatik*.

Colucci, S.; Di Noia, T.; Di Sciascio, E.; Donini, F.M.; Mongiello, M.; Mottola, M. (2003) A Formal Approach to Ontology-Based Semantic Match for Skills Descriptions, *Journal of Universal Computer Science*, Vol. 9/12, 1437-1454.

Colucci, S.; Di Noia, T.; Di Sciascio, E.; Donini, F.M.; Ragone, A. (2007). Semantic-based Skill Management for Automated Task Assignment and Courseware Composition, *Journal of Universal Computer Science*, Vol. 13/9, 1184-1212.

De Coi, J.; Herder, E.; Koesling, A.; Lofi, C.; Olmedilla, D.; Papatreou, O.; Siberski, W. (2007) A Model for Competence Gap Analysis, in *Proceedings of 3rd International Conference on Web Information Systems and Technologies (WEBIST).*

Dittmann, L. (2003) Towards Ontology-based Skill Management, Projektbericht zum Verbundprojekt KOWIEN, *Universität Duisburg-Essen*.

Dorn, J.; Naz, T.; Pichlmair, M. (2007) Ontology Development for Human Resource Management, in *Proceedings of the 3rd International Conference on Knowledge Managements*.

Fazel-Zarandi, M.; Yu, E. (2008) Ontology-Based Expertise Finding, in *Proceedings of the 7th International Conference on Practical Aspects of Knowledge Management* (PAKM 2008), Lecture Notes in Computer Science, Vol, 5345, pp. 232-243.

Fox, M.S., Barbuceanu, M., Gruninger, M., Lin, J. (1997) An Organization Ontology for Enterprise Modelling. In: Prietula, M., Carley, K., Gasser, L. (eds.) *Simulating Organizations: Computational Models of Institutions and Groups*, AAAI/MIT Press, Menlo Park, pp. 131–152.

Gomez-Perez, A.; Ramirez, J.; Villazon-Terrazas, B. (2007) Reusing Human Resources Management Standards for Employment Services, in *Proceedings of the Workshop on the First Industrial Results of Semantic Technologies* (FIRST'2007) at *ISWC/ASWC*, pp. 28-41.

Gráinne, C.; Fill, K. (2005) A Learning Design Toolkit to Create Pedagogically Effective Learning Activities, *Journal of Interactive Media in Education*, Vol. 8.

Huang, J. (2008) Knowledge Provenance, *An Approach to Modeling and Maintaining the Evolution and Validity of Knowledge*, PhD Thesis, Department of Mechanical and Industrial Engineering, University of Toronto, Canada.

Lau, T.; Sure, Y. (2002) Introducing Ontology-Based Skills Management at a Large Insurance Company, in *Proceedings of the Modellierung*, 123-134.

Liu, P.; Dew, P. (2004) Using Semantic Web Technologies to Improve expertise Matching within Academia, in *Proceedings of I-KNOW'04*, 370-378.

Mochol, M.; Wache, H.; Nixon, L. (2007) Improving the Accuracy of Job Search with Semantic Techniques, in *Proceedings of the 10th International Conference on Business Information Systems* (BIS'2007), Lecture Notes in Computer Science, Vol. 4439, pp. 301-313.

Moyer, R. (2001) Defining and Measuring Competencies: An Overview of Approaches, Click2learn Inc.

Razmerita, L.; Angehrn, A.; Maedche, A. (2003) Ontology-based User Modeling for Knowledge Management Systems; in *User Modeling*, Pennsylvania, 213-217.

Reich, J.R.; Brockhausen, P.; Lau, T.; Reimer, U. (2002) Ontology-Based Skills Management: Goals, Opportunities and Challenges, *Journal of Universal Computer Science*, Vol. 8/5, 506-515.

Saip, H.B.; Lucchesi, C. (1993). Matching Algorithms for Bipartite Graph, *Technical Report DCC-0393*, Universidade Estudal de Campinas.

Schmidt, A.; Kunzmann, C. (2006) Towards a Human Resource Development Ontology for Combining Competence Management and Technology-Enhanced Workplace Learning, in *Lecture Notes in Computer Science* Vol. 4278, Springer, 1078-1087.

Schugurensky, D. (2000) The Forms of Informal Learning: Towards a Conceptualization of the Field, NALL Working Paper No. 19.

Seta, K.; Ikeda, M.; Hirata, K.; Hayashi, Y.; Kuriyama, K. (2005) A Human Resource Model and Evidence Based Evaluation--Ontology for IT Skill Standards--. In *Proceeding of the 2005 Conference on Towards Sustainable and Scalable Educational Innovations Informed by the Learning Sciences*, 388-395.

Veit, D.; Muller, J.; Schnider, M; Fiehn, B. (2001). Matchmaking for Autonomous Agents in Electronic Marketplaces. In *Proceedings of the International Conference on Autonomous Agents '01*, 65-66.

Zhong, J.; Zhu, H.; Li, J.; Yu, Y. (2002). Conceptual Graph Matching for Semantic Search. In *Proceedings of the 10th International Conference on Conceptual Structures*, 92-105.