Integrating Trust and Competency Management to Improve Learning

Nicola Capuano†, Matteo Gaeta†, Giuseppina Rita Mangione§, Francesco Orciuoli†, Pierluigi Ritrovato*

§MOMA S.p.A.
†Department of Electronic Engineering and Information Engineering
University of Salerno, 84084 Fisciano (SA), Italy

Email: {gaeta, orciuoli}@diima.unisa.it, {ncapuano}@unisa.it, {mangione, ritrovato}@momanet.it

Abstract—Nowadays, the importance of knowledge management is well understood by managers in the organizations and, at the same time, the great significance of trust, in enabling effective knowledge sharing, is emerging. Presence or lack of trust can have serious implications for organizations with respect to the quality and of their business processes. Several scientific works have confirmed the direct correlation between social-cognitive capital, in terms of competencies and experiences, and a feeling of trust in both learning and working collaborative environments. On the other hand, Competency-Based Management allows organizations to link human resources processes to competencies in order to shape its workforce capabilities and to achieve better results. Typically, employees’ competencies (and proficiency levels) are stored and used by specific enterprise software, whereas the trust is not considered or left to managers’ feeling. This work proposes an approach to the improvement of collaborative learning activities by refining and allowing (at workers’ level) competency-finding processes through the social calculus of trust-in-competencies degree.

Keywords—corporate learning; trust; competency-based management; semantic web; social networking;

I. INTRODUCTION

A. Background and Motivations

According to [1], it is possible to define the trust as a mental state comprising expectancy (the trustee expect a specific behavior from the trustee) and belief. Trust is considered one of the main assets available for organizations to proactively manage the knowledge sharing and increase the knowledge capital. In literature, many contributions suggesting a correlation between social-cognitive capital and feel of trust in collaborative environment [2], with regard to working and learning activities [3], have been recognized. According to Ardichvili et al. [4] the lack of trust in competencies may be identified as one of the main barriers to knowledge sharing in online communities of practice discouraging any form of motivation and participation in work activities. Moreover, one’s degree of trust in the competence of a community is positively related to one’s engagement in knowledge sharing with the community. Recent studies have confirmed this hypothesis [5] and argue that social motivation in the workplace increases the growth of trustworthiness and directly impacts on employees’ performances. Although several works providing models for trust management in virtual communities and social networks are reviewed in literature [6], they are not related to the trust in competencies the workers need to perform specific jobs. Competencies are indentified, classified and managed by a set of processes, namely Competency-Based Management (CBM), ensuring that there is the right person in the right position at the right time in order to achieve better results. Typically, CBM processes are top-down organizational processes supported by enterprise software (e.g. Human Resource Information Systems, etc.) that cannot be exploited to support team building operations in non-formal collaborative learning (and working) activities. The main goal of this work is to define an approach to enrich CBM with trust information in order to support (by enabling competency finding operations) and to improve (by considering trust as a way to increase the quality of knowledge sharing) collaborative learning (and working) activities at the workplace. In order to achieve the aforementioned goal, a semantic-annotated trust-based corporate social network (where nodes are the employees’ profiles and arcs are the relationships among the employees) will be defined. The social nature of the network allows to overcome the limitation of traditional CBM by enabling bottom-up competency management operations improved with trust information. Furthermore, trust in competencies will be calculated by using a social approach, i.e. considering direct interactions between two workers and using semantic-annotated social network to propagate trust degrees.

B. A Brief Overview of the Proposed Approach

This work proposes a trust-based corporate social network model (Section II) to improve learning (and working) processes regarding human resources in organizations. The idea underlying the proposed approach consists in defining a methodology to calculate the degree of trust a worker has in his/her colleagues’ specific competencies. Let us assume the trustee’s name is a and the trustee’s name is c. The calculus of trust degree is directly (by analysing the interactions between a and c) performed if there has been at least one direct experience in which a had evidence of a specific competency of c, otherwise the trust degree is performed by propagation using the network. In the case of
II. Trust-based Corporate Social Network

In this section, a model of employees’ profiles network is described by integrating and extending existing Semantic Web schemes, based on the W3C Semantic Web stack. The main result reported in this section is the definition of the Trust in Competencies and Trust in Belief model and its correlation with the employees’ profiles in order to build the Trust-based Corporate Social Network.

A. Competency Model

According to the Professional Learning Ontology and Competencies (PLOC) described in [7], competencies are defined as abstractions for making human Knowledge, Skills and Abilities (KSA) manageable and addressable. Before competencies can be managed (by means of CBM processes), they should be classified and their context defined. It seems to be a good choice to divide them into behavioural and technical competencies (see fig. 2). Behavioural competencies include all those soft skills and attitudes that enable a person to perform well in a specific function, such as communication, problem-solving, integrity, planning and organisation, etc. Technical competencies are specific and depending on the job. These latter could range, for instance, from the knowledge of software development methodologies to skill for Java programming. Typically, competencies are organized in a hierarchy of four levels. Main classes and properties regarding the organization of competencies in PLOC are shown in fig. 1.

B. Employees’ Profiles and Relationships

In order to model the employees’ profiles, at least personal and curriculum vitae data have to be considered. In [8] an integrated model constituted by FOAF2 (organizing basic employees’ information), ResumeRDF3 (modeling the curriculum vitae of employees) and PLOC (modeling competencies and their link with the employees) is presented. In particular, the foaf:Person class is crucial to model part of personal information about an employee. Some useful properties are foaf:firstName, foaf:surname, foaf:mbox, foaf:publications, foaf:interest, etc. Moreover, ResumeRDF is a RDF metadata vocabulary mainly used to describe work and education history of the employees. The main part of the ResumeRDF model is the cv:CV class that maintains information about a specific employee (cv:Person), his (her) education (cv:Education), his (her) work history (cv:WorkHistory), and so on. Although, ResumeRDF provides a way to model skills (cv:Skill) owned by an employee, it is preferable to use another scheme in order to adhere to a more complete definition (KSA) of competency (see paragraph II-A) provided by PLOC. The PLOC defines competence:Employee class and competence:has-competency property that are useful to link an employee with its competencies. In order to link foaf:Person with cv:Person and competence:Employee, the subclassing operation is used. In particular, competence:Employee becomes a subclass of cv:Person that becomes a subclass of foaf:Person. The effect of the application of the subclassing operation is that all individuals of type competence:Employee are also instances of cv:Person class and instances of foaf:Person class. So, all properties of foaf:Person are inherited by cv:Person and all properties of these two classes are inherited by competence:Employee class. One of the most important FOAF property is foaf:knows that links two instances of the foaf:Person class by building a semantic-annotated social network where the employees are nodes and their rela-

1 http://www.w3.org/standards/semanticweb/
2 http://www.foaf-project.org/
3 http://rdfs.org/resume-rdf/
tionships are arcs. However, in the corporate context, we need richer semantics to model relationships among employees. The RELATIONSHIP4 vocabulary represents one of the plausible solution for the aforementioned problem. In particular, relationship:worksWith (asserts that person e works for the same employer of person d) and relationship:collaboratesWith (asserts that person e has worked at least one time on the same activity with person d) are important for our purpose. The two aforementioned properties can be applied between two foaf:Person instances and they are symmetrical.

C. Trust in Competencies and Trust in Belief

The employees’ profiles network, described with Semantic Web schemes, has to be enriched with information regarding trust. In particular, in [9] the authors build a formal-semantics-based model for the calculus of trust and refer to two basic types of trust: trust in performance and trust in belief. The first one is the trust about what the trustee performs, whereas the second one is the trust about what the trustee believes. The present work starts from the results of [9] and extends them in order to deal with the concept of trust in competencies. In particular, we propose the following statements:

\[ \text{trust}_c(d, e, \text{apply}(e, x), k), \]
\[ \text{trust}_b(g, d, \text{apply}(e, x), k). \]

The statement 1 indicates that employee d (trustor) trusts employee e (trustee) regarding the capability of e to apply competency x in context k. The statement 2 indicates that employee g (trustor) trusts employee d (trustee) regarding d’s belief about the capability of e to apply competency x. Note that, in the Semantic Web-based modeling, d, e and g are individuals of type competence:Employee, x is an individual of type competence:Competency, whereas k can be interpret as an instance of a skos:Concept. SKOS5 is a Semantic Web scheme used to build taxonomies and controlled vocabularies. For the aim of this work, SKOS will be used to model a controlled vocabulary of contexts of interest in a given organization using skos:narrower and skos:broader properties to relate instances of skos:Concept.

The table in Fig. 2 shows variants of 1 and 2 for all competency types.

Now, a Semantic Web-based scheme is needed in order to model trust statements of Fig. 2. The Trust Ontology6. The Trust Ontology is a small ontology, written on OWL, that allows people to say how much they trust in other people. It is defined as an extension of the FOAF scheme. The most important class of the Trust Ontology is trust:TopicalTrust. The classes trust:TrustInCompetency and TrustInBelief have been defined by subclassing trust:TopicalTrust in order to respectively model trust in competencies and trust in belief. Subclasses of trust:TrustInCompetency could be defined in order to model all types of competencies illustrated in Fig. 2. The individuals of trust:TrustInCompetency allow to link an instance of competence:Employee (trustor) with another instance of competence:Employee (trustee) and with the specific target of the trust, i.e. an individual of type competence:Competency. The data property trust:trustValue is used to assign a value to the trust in the target competency of the trustee.

It is important to note that in origin the class trust:TopicalTrust did not support values for distrust (that is the opposite of trust) and untrust (that is the status in which the degree of confidence is insufficient to trust) in order to solve some reasoning problems. We have extended the Trust

---

4http://vocab.org/relationship.html
5www.w3.org/2004/02/skos/
6http://trust.mindswap.org/trustOnt.shtml
Ontology with the new data properties trust: distrustValue and trust: untrustValue. In the extended ontology, trust, distrust and untrust have values in [0, 1]. The graph in fig. 3 states that Paolo trusts Mario, with value 0.6, to apply the Project Management competency (for sake of simplicity the graph does not report the competency level) in the context ResearchProject. The modelling of a statement for trust in belief is similar to that for trust in competencies. In this case of trust in belief, the property trust: trustSubject does not link a competency, it links a context (an individual of skos: Concept).

III. WORK ACTIVITIES

In order to calculate trust, distrust and untrust degrees it is important to understand how organizational activities are modelled. In particular, the PLOC provides the competence: OrganizationalEntity class. The competence: Task subclass of competence: OrganizationalEntity can be used as a basis to define a work activity in which an employee (worker) executes a specific task by means of the competence: isIn property (with domain competence: OrganizationalEntity and range competence: Employee). Furthermore, the competence: requires competency (with domain competence: OrganizationalEntity and range competence: Competency) is needed to indicate the competencies that are required in order to successfully carry out a specific task. In order to complete the Work Activities model we need to include a supervisor and the possibility for the supervisor to review the performance of the worker. So, the trust: supervises property and the RDF Review Vocabulary\(^7\) have to be introduced. The first one is used in order to relate instances of competence: Task class (supervised tasks) with instances of competence: Employee (supervisors). The second one provides a set of classes and properties useful to attach a review (rev: Review class) to a resource (in this case the resource will be an individual of type competence: Task). The underlying idea is that the supervisor assigns a rate (rev: rating) is a numeric property with domain rev: Review) to the performance of the employee involved in the task. The rate can be interpreted as a value indicating the how the required competencies have been applied during the execution of the task. Supervisor and worker are both individuals of competence: Employee class.

IV. CALCULUS OF TRUST, DISTRUST AND UNTRUST

In this section, a method to calculate degrees for trust, distrust and untrust is proposed. For this aim, the probabilistic approach illustrated by the authors in [9] has been adapted for context and purpose of this work. The aforementioned contextualisation and the use of semantic-annotated data are advancements with respect to the results of [9]. Trust is interpreted as a probability distribution over three mental states (believing, disbelieving and undecidable) corresponding to trust, distrust and untrust. Intuitively, the trust α of e (trustor) for d (trustee) in successfully applying a competency x in context k can be approximated with the probability that e will successfully apply x in context k. Distrust β is the opposite of trust. Instead, untrust γ, corresponding to the third state of the probability distribution, can be calculated by using the formula \(\alpha + \beta + \gamma = 1\). Authors in [9] propose to calculate probability measures in terms of frequencies of positive experiences between the trustor and the trustee. More in details, in the case of trust in competencies the following formula (3) can be used.

\[
\alpha = td_c(d, e, apply(e, x), k) = n/m,
\]  

where m is the total number of situations, experienced by d (trustor), in which e (trustee) has applied competency x in the context k, and n is the number of the trustor’s positive experiences in that set, \(\beta\) (distrust) can be calculated with the same formula (3) of \(\alpha\) by replacing n with l and \(td_c\), with \(dt_c\), where l is the number of negative experiences in the same set. Values for n and l can be calculated with the following:

\[
n = \sum_{i=1}^{m} e_p(i), l = \sum_{m} e_n(i),
\]  

where \(e_p(i), e_n(i) \in [0, 1]\), representing the degree of the experience \(i\) being positive or negative separately, and \(e_p(i) + e_n(i) \leq 1\). In the context of this work, a positive (negative) experience of a trustor with a trustee is represented by a task, that requires competency \(x\), where the supervisor (trustor) \(d\) has provided a positive (negative) review rate \(e_p(i) e_n(i)\) to the worker (trustee) \(e\) for the application of competency \(x\) in the context \(k\). Moreover, in the case of trust in belief the following equations can be used.

\[
\alpha = dt_b(g, d, k) = n'/m',
\]  

where \(dt_b(g, d, k)\) approximates \(dt_b(g, d, apply(e, x), k), m'\) is the total number of experiences observed by g (trustor), in which d (trustee) has applied any competency in context k, and \(n'\) is the number of positive experiences in that set. \(\beta\) (distrust) can be calculated by using the same formula (5) and replacing \(n'\) with \(l'\), where \(l'\) is the number of negative experiences in the same set.

\[
n' = \sum_{m'} e'_p(i), l' = \sum_{m'} e'_n(i),
\]  

\(^7\)http://danja.talis.com/xmlns/rev_20071109/index.html
where $e_p(i), e_n(i) \in [0, 1]$, representing the degree of the experience $i$ being positive or negative separately, and $e_p(i) + e_n(i) \leq 1$. In this case, a positive (negative) experience of a trustor with a trustee is represented by a task where the supervisor (trustor) $g$ has provided a positive (negative) review rate $e_p(i)$ ($e_n(i)$) to the worker (trustee) $d$ for the application of any competency in the context $k$.

V. PROPAGATION OF TRUST

In the previous section (IV) a method to calculate trust, distrust and untrust, between two nodes (employees) in the network, is provided. Now, a model of propagation of trust degrees in social networks [10] has to be exploited in order to calculate the trust in competencies between employee $e$ and employee $d$ also if $e$ has never supervised a task performed by $d$ in which $d$ has used the required competency $x$ in context $k$. In this case, it is possible to use the corporate social network and trust in belief to propagate trust in competencies (and also trust in belief) by contextualising the approach presented in [9]. In fig. ??, a piece of the semantic-annotated corporate social network is depicted. In particular, for employee $e$ there are not values for trust, distrust and untrust for competency $x$ of employee $g$ in context $k$. But, it is possible to obtain the aforementioned values by exploiting trust in belief (in context $k$) of $e$ for $d$ and trust in competency $x$ of $d$ for $g$ in context $k$. In ?? only degrees for trust and distrust are indicated. In particular $td_e$ and $td_d$ are values for $td_e(d, g, apply(x, k))$ and $td_d(e, g, apply(x, k))$, $td_b$ and $td_{d_b}$ are values for $td_b(e, g, k)$ and $td_b(e, g, k)$. $ptd_e$ and $ptd_{d_e}$ are values calculated by using the formulas 7 and 8.

$$
ptd_e = td_b(e, d, k) \cdot td_e(d, g, apply(x, k)) + td_b(e, d, k) \cdot td_d(d, g, apply(x, k)), 
$$

$$
ptd_{d_e} = td_b(e, d, k) \cdot td_e(d, g, apply(x, k)) + td_b(e, d, k) \cdot td_d(d, g, apply(x, k)).
$$

The formulas 7 and 8 can be used to propagate trust, distrust and untrust by considering paths of any length over the corporate social network. Trust in belief is propagated by replacing, in 7, $td_e(d, g, apply(x, k))$ and $td_d(d, g, apply(x, k))$ with $td_b(e, g, k)$ and $td_b(e, g, k)$. The propagation of distrust degrees is obtained in the same way by modifying formula 8.

VI. FINAL REMARKS

The proposed approach enables several scenarios in which competency-finding is executed by workers that, when introduced in project teams, can exploit feelings of trust in their peers’ capabilities in order to activate help seeking processes. The competency search operation is refined by calculating trust degrees. For instance, a non-formal learning activity, like peer-coaching, is improved (trust improves learning as asserted in Section I) when the coach is selected by the worker by using, not only the required competencies as search parameters, but also the trust degree of the learner for the potential coach. The approach will be experimented and exploited in ARISTOTELE project to develop a trust-based Decision Support System for team building.

ACKNOWLEDGMENT

This research is partially supported by the European Commission under the Collaborative Project ARISTOTELE "Personalised Learning & Collaborative Working Environments Fostering Social Creativity and Innovations Inside the Organisations", VII Framework Program, Theme ICT-2009.4.2 (Technology-Enhanced Learning), Grant Agreement n. 257886

REFERENCES


