

Refining Semantic Matching for Job Recruitment: an Application of Formal Concept Analysis*

Gábor Rácz, Attila Sali, Klaus-Dieter Schewe

Alfréd Rényi Institute of Mathematics, Hungarian Academy of Sciences
Budapest, P.O.B.127, H-1364 Hungary
`gabec33@gmail.com, sali.attila@renyi.mta.hu, kd.schewe@acm.org`

Abstract. A profile describes a set of skills a person may have or a set of required for a particular job. Profile matching aims to determine how well the given profile fits the requested profile. Skills are organized into ontologies that form a lattice by the specialization relation. Matching functions were defined based on filters of the lattice generated by the profiles. In the present paper the ontology lattice is extended by additional information in form of so called extra edges that represent some kind of quantifiable relationship between skills. This allows refinement of profile matching based on these relations between skills. However, that may introduce directed cycles and lattice structure is lost. We show a construction of weighted directed acyclic graphs that gets rid of the cycles, and then present a way to use formal concept analysis to gain back the lattice structure and the ability to apply filters. We also give sharp estimates how the sizes of the original ontology lattice and our new constructions relate.

1 Introduction

A profile describes a set of properties and profile matching is concerned with the problem to determine how well a given profile fits to a requested one. Profile matching appears in many application areas such as matching applicants for job requirements, matching buyers' requirements with goods advertised such as used cars, etc.

An early idea of profile matching was considering profiles as sets of unrelated items. Then one tries to measure the similarity or distance of sets. Several ways of definition of distances of sets were introduced, such as Jaccard or Sørensen-Dice measures [13] turned out to be useful in ecological applications. However, skills or properties included in profiles are usually not totally unrelated items, implications or dependencies exist between them and need to be taken into

* The research of the first author of this paper has been partly supported by the Austrian Ministry for Transport, Innovation and Technology, the Federal Ministry of Science, Research and Economy, and the Province of Upper Austria in the frame of the COMET center SCCH

account. For example, in the human resources area several taxonomies for skills, competences and education such as DISCO [1], ISCED [2] and ISCO [3] have been set up. These taxonomies organize the individual properties into a lattice structure. Popov and Jebelean [18] proposed defining an asymmetric matching measure on the basis of filters in such lattices.

Besides the subsumption relations of the ontology lattice other “horizontal” relations between skills exist. The existence of some skills imply that the applicant may have some other skills with certain probabilities, or of some (not complete) proficiency level. For example, we may reasonably assume that knowledge of Java implies knowledge of NetBeans up to a grade of 0.7 or with probability 0.7. This kind of interdependencies were exploited in [19]. The idea is that a job application is considered better than another one for a given offer profile even if they match equally using filter methods, if the first one has more skills implied in the “fractional” way that match the offer, than the second application has. In this way we get a refinement of the matching hierarchy given by previous methods.

The subsumption hierarchy of the ontology of skills was considered as a directed graph with edge weights 1. A lattice filter generated by a profile corresponded to the set of nodes reachable from the profile’s nodes in the directed graph. Then extra edges were added with weights representing the probability/grade of the implication between skills or properties. This introduced the possibility of directed cycles. Filters of application profiles are replaced by nodes reachable in the extended graph from the profile’s nodes. For each vertex x reached a probability/grade was assigned, the largest probability/weight of a path from the profile’s nodes to x . Path probability/weight was defined as the product of probabilities of edges of the path. This process resulted in a set of nodes, which we call *derived skills*, with grades between zero and one, so it was natural interpreting it as a fuzzy set. It was proved that it’s a fuzzy filter as defined in [11, 14].

In the present paper we provide a construction that gets rid of directed cycles caused by the extra edges. In doing so we show that all matching results that can be obtained by exploiting extra edges can also be obtained from an extended lattice without such extra edges. That is, the theory of profile matching remains within the filter-based approach that we developed in [17], which underlines the power and universality of this theory. In particular, we emphasised how to obtain the lattices underlying the matching theory from knowledge bases that define concepts used in job and candidate profiles. These knowledge bases are grounded in description logics, so the lattice extensions provide also feedback for fine-tuning the knowledge representation, whereas weighted extra-edges are not supported in the knowledge bases. Furthermore, we also showed in [17] that under mild plausibility constraints on human-defined matchings appropriate weights can be defined such that the filter-based matchings preserve the human-defined rankings, which further enables linear optimization to synchronize matchings with human expertise.

The extension is done by extending the ontology lattice by new nodes and weighting of the nodes. The result is a directed acyclic graph, whose structure reflects the different possible path lengths between nodes of the ontology lattice. A directed acyclic graph naturally represents a poset, however that is not a lattice in general. In order to gain back the lattice structure formal concept analysis is used.

While extension of applications by skills derived using extra edges is natural, as employers may benefit from these skills, it is not so clear whether the offer profiles should be extended. On one hand, profiles should be handled uniformly, since a profile could represent both, application, as well as offer. On the other hand, if offers are also extended with derived skills, then it may happen that an application scores high match by having only these derived skills, not the ones in the original offer. This situation may not be so advantageous. In the present paper we discuss both scenarios, the latter one is treated by applying different weighting functions for applications and offers.

The paper is organized as follows. Section 2 introduces the basic concepts and definitions, furthermore the matching functions studied. Section 3 deals with the construction of directed extension graph and formal concept lattice. We also give node weightings that preserve the weights of fuzzy filters assuming that offers are also extended with derived skills. Section 4 discusses related extremal problems, that is how the sizes of the constructed structures relate to the size of the original ontology lattice. We show that our obtained bounds are sharp. Section 5 contains the analysis when offers are not extended by derived skills, just by those that are reachable via lattice (ontology) edges. In order to preserve the weights of fuzzy filters we have to give different node weights for offers from the weights of applications. Section 6 surveys related results, while Section 7 is a summary.

2 Semantic matching

Semantic matching has various application areas from dating applications to on-line product searching tools. We approach the problem from the field of human resources, namely we search for the best fitting application for a given job.

Formally, let $S = \{s_1, \dots, s_n\}$ be a set of skills. A job offer $O = \{o_1, \dots, o_k\}$ is a subset of S that contains the skills that are required for the job. An application $A = \{a_1, \dots, a_l\}$ is also a subset of S that represents skills of the applicant. Our task is to find the most suitable applicant for a given job. Let $match : \mathcal{P}(S) \times \mathcal{P}(S) \rightarrow [0, 1]$ be a matching function that determines how well an application fits to a job offer. If we know the matching function, then finding the most suitable applicant is a maximum search over the matching values.

Let \preceq be a specialization relation over the skills such that for all $s, s' \in S$: $s \preceq s'$ iff s is a specialization of s' or s' is more general skill than s . This relation is reflexive, antisymmetric and transitive, so it defines a partial order, a hierarchy over elements of S . Let us suppose that $\mathcal{L} = (S, \preceq)$ is an ontology lattice, i.e. for each pair of skills has infimum (greatest lower bound) and supremum (least upper

bound). Note, that we can always add a top (respectively a bottom) element to the skills that everybody (nobody) possesses.

We can extend the lattice with additional information in form of so called extra edges that represent some kind of quantifiable relationship between skills. However, these edges can form cycles in the hierarchy therefore we use directed graphs to handle them instead of the lattice structure [19].

Let $G = (V, E)$ be a directed graph where $V = S$ and $E = E_{lat} \cup E_{ext}$ is a set of lattice edges and extra edges such that for two nodes $v_i, v_j \in V$: $(v_i, v_j) \in E_{lat}$ iff $v_i \preceq v_j$ and $(v_i, v_j) \in E_{ext}$ iff there is an extra edge between v_i and v_j . Let $w_{edge} : E \rightarrow [0, 1]$ be an edge weighting function such that for all $e_{lat} \in E_{lat}$: $w_{edge}(e_{lat}) = 1$ and for all $e_{ext} \in E_{ext}$: $w_{edge}(e_{ext}) \in (0, 1)$ that represents the strength of the relationship between start and end node of the edge. Let $p_F(x, v)$ denote the set of directed paths from node x to node v using edges of a subset $F \subseteq E$ of edge set E of G .

We can define a matching function of an application A to an offer O using the graph in the following way. First, we define function ext to extend the application and the offer with all the skills that are available from them via directed path in G . For an arbitrary set of skills $X \subseteq S$ and a subset $F \subseteq E$ of edges, let $ext_F(X) = \{(v, \gamma_v) \mid v \in V \text{ and } \exists x \in X : |p_F(x, v)| \geq 1 \text{ and } \gamma_v = \max_{x' \in X, p \in p_F(x', v)} length(p)\}$ where length of a path $p = (v_1, \dots, v_n)$ is the product of the edge weights on p , i.e. $length(p) = \prod_{i=1}^{n-1} w_{edge}((v_i, v_{i+1}))$.

It was shown that the extended sets are fuzzy filters [14] in $\mathcal{L} = (S, \preceq)$, i.e. for a set of skills X and for all $t \in [0, 1]$: $ext_E(X)_t = \{x \in X \mid \gamma_x \geq t\}$ is filter in \mathcal{L} .

It perfectly makes sense to use lattice edges to extend applications and offers as lattice edges describe specialization relation between skills. Namely if an applicant possesses a special skill then he or she must possess the more general skills as well. However extra edges are used in the extension as well to get more selective matching functions that help differentiate applications.

Let us call nodes in $ext_E(X) \setminus ext_{E_{lat}}(X)$ *derived* nodes for a set $X \subseteq S$ of skills. We investigate two approaches or philosophies when extending profiles using the extra edges. The first one is *symmetric*, that is the case when offers and applications are treated in the same way. In this case we use extension function ext_E for both, offers O and applications A . The advantage is that we only have to apply one weighting function and the proof of equivalence of different representations is simpler than that of the other case. There is a disadvantage, though. If offers are also extended with derived skills, then an application may obtain high matching value just having those skills. However, it is not really advantageous for an employer, as required skills are not in the application.

The second approach called the *strict approach* is when offers are only extended with non-derived nodes, that is ext_E is used for applications but $ext_{E_{lat}}$ is used for offers. This is the approach of [19]. The disadvantage of this case is that different weighting functions have to be applied for applications and offers, consequently the proofs of equivalences are more complicated. However, the point of view of employers is better represented in the second way. An application has

to have good matching in target skills to score high, and the derived skills can be used to rank applications scoring equally otherwise. Note, that $ext_{E_{lat}}(X)$ is exactly the set of nodes contained in the lattice filter generated by X in the ontology lattice (S, \preceq) .

We adapted the profile matching function proposed by Popov et. al. [18] to fuzzy sets in [19]. We use the same function here except the different approaches in extension of offers. So, let the matching value of A to O be

$$match_{sym}(A, O) = \frac{||ext_E(A) \cap ext_E(O)||}{||ext_E(O)||} \quad (1)$$

in case of the symmetric approach, and

$$match(A, O) = \frac{||ext_E(A) \cap ext_{E_{lat}}(O)||}{||ext_{E_{lat}}(O)||} \quad (2)$$

in case of the strict approach. For two fuzzy sets f, g of S and for a skill $s \in S$ let $(f \cap g)(s) := \min\{f(s), g(s)\}$, and $||f|| := \sum_{(v, \gamma_v) \in f} \gamma_v$, i.e. $|| \cdot ||$ denotes sigma cardinality and intersection is defined as the *min* t-norm. Note, that other cardinality and intersection functions can be applied in the same way [23][11].

Let $w_{node} : V \rightarrow [0, 1]$ be a node weighting function that assigns 1 to every nodes and let $w_{fset} : \mathbb{F}_S \rightarrow [0, 1]$ be a weighting function for fuzzy sets such that for a fuzzy set f let $w_{fset}(f) = \sum_{(v, \gamma_v) \in f} \gamma_v = \sum_{(v, \gamma_v) \in f} \gamma_v \cdot w_{node}(v)$ where \mathbb{F}_S denotes all fuzzy sets of S . Note, that w_{node} is defined only to unify the notations in the rest of the paper. With this weighting functions, the matching value of A to O can be given as

$$match_{sym}(A, O) = \frac{w_{fset}(ext_E(A) \cap ext_E(O))}{w_{fset}(ext_E(O))} \quad (3)$$

and

$$match(A, O) = \frac{w_{fset}(ext_E(A) \cap ext_{E_{lat}}(O))}{w_{fset}(ext_{E_{lat}}(O))}, \quad (4)$$

respectively.

3 Lattice enlargement

In this section, we present a graph transformation method to eliminate extra edges from extended lattices preserving symmetric matching values of applications to offers, and then we use formal concept analysis to restore lattice properties in the transformed graphs.

3.1 Extension graph

Let $G = (V, E)$ be a directed graph with w_{edge} , w_{node} weighting functions as defined above and c_{ij} be the weight of the longest path from v_i to v_j where

$v_i, v_j \in V$ are two nodes. Let $v_{i_1j}, \dots, v_{i_kj}$ be the nodes from where v_j is available via directed path such that $c_{i_1j} \leq \dots \leq c_{i_kj}$. Let c^{j_1}, \dots, c^{j_l} denote the different values among $c_{i_1j}, \dots, c_{i_kj}$, i.e. $c^{j_1} < \dots < c^{j_l}$.

For all $c^{j_1} \dots c^{j_l}$, add new nodes $V_j = \{v_{j_1}, \dots, v_{j_l}\}$ (for simplicity let $v_{j_l} = v_j$) to V and add new lattice edges from v_{j_l} to $v_{j_{l-1}}, \dots$, from v_{j_2} to v_{j_1} , and from v_{j_1} to the top to E . The new edges forms a directed path from v_j to the top. Let $q_j = (v_{j_l}, \dots, v_{j_1}, \text{top})$ denote that path. Assign weight $w_{j_k} = c^{j_k} - c^{j_{k-1}}$ to v_{j_k} ($k = 1, \dots, l$) where $c^{j_0} = 0$. Note, that $\sum_{k=1}^l w_{j_k} = 1$ as it is a telescoping sum. If the length of the longest path from v_i to v_j was c^{j_k} , then add a new lattice edge from v_i to v_{j_k} . Finally, remove all extra edges from the graph. Let $G' = \text{ext}(\mathcal{L}, E_{\text{ext}}) = (V', E')$ denote the modified graph, called extension graph, and w'_{node} denote the modified node weighting function.

New nodes of V_j and new edges of q_j can be considered as an extension of v_j to a chain because there do not start edges from intermediate nodes to other chains so out-degrees of intermediate nodes are always one. We call v_j the base node of the chain. Base nodes of such chains are nodes of \mathcal{L} , and G as well.

Let q_j and q_k be two chains with base nodes v_j and v_k , respectively. Then, an edge from q_j to q_k in G' can go

- from v_j to v_k and then it represents a directed path in G from v_j to v_k containing lattice edges only;
- from v_j to an intermediate node v_i of q_k and then it represents a directed path $p_{v_j v_k}$ of G from v_j to v_k such that $\text{length}(p_{v_j v_k}) = \sum_{s=1}^i w'_{\text{node}}(v_s)$ if $q_k = (v_{k_l}, \dots, v_{s+1}, v_s, v_{s-1}, \dots, v_1, \text{top})$.

Note, that lattice edges in G are acyclic so the corresponding edges in G' are acyclic as well, and newly added edges start from base nodes of chains only. So G' is an acyclic graph.

Figure 1 shows an example of the construction of G' . There is the original graph, called G , on the left. Blue (solid) edges represent lattice edges and orange (dashed) edges with numbers on them represent extra edges and their weights. There is the extension graph, called G' , on the right where green edges represent the newly added edges, and numbers in the top right corners of nodes are weights of the nodes.

As it can be seen, for example, node A of G has been transformed into the chain $q_A = (A, A_1, \text{Top})$ since A is available via lattice edges (i.e. via maximum length paths) from B, C, Bottom and it is available from D via the path $p_{DA} = (D, C, A)$ whose length is 0.8 and A is not available from any other nodes. Therefore A_1 got the weight 0.8 and A got the weight 0.2.

Lemma 3.1. *Let $G = (V, E)$ be a directed graph extending the lattice $\mathcal{L} = (S, \preceq)$ with extra edges, $w_{f_{\text{set}}}$ be the fuzzy set weighting function, $G' = \text{ext}(\mathcal{L}, E_{\text{ext}}) = (V', E')$ be the extension graph, and $w'_{f_{\text{set}}}$ be the modified weighting function. Let $O \subseteq S$ be an offer and $A \subseteq S$ be an application. Then,*

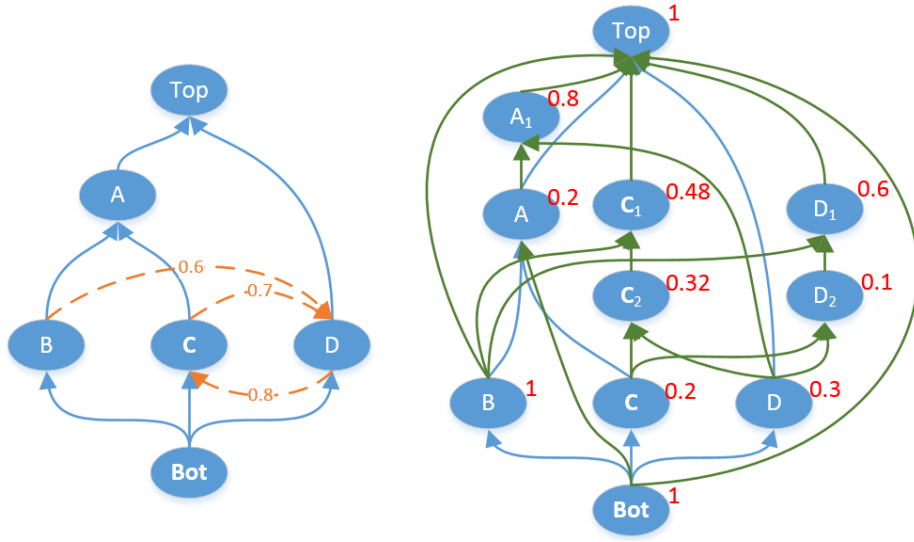


Fig. 1: Lattice with extra edges and the generated extension graph

$$match_{sym}(A, O) = \frac{w_{fset}(ext_E(A) \cap ext_E(O))}{w_{fset}(ext_E(O))} = \frac{w'_{fset}(ext_{E'}(A) \cap ext_{E'}(O))}{w'_{fset}(ext_{E'}(O))}. \quad (5)$$

Proof. Let $u \in G'$ and let $q_z = (z_l, \dots, z_1, top)$ be the node chain with base node $z \in G$ that contains u , i.e. $z_l = z$ and $u = z_i$ for some $i \in [1..l]$. First, we will show for an arbitrary $X \subseteq S$ that $u \in ext_{E'}(X)$ iff $z \in ext_E(X)$.

If $u \in ext_{E'}(X)$, then there is a node $a \in X \subseteq V'$ and a directed path $q_{au} = (x_1, \dots, x_i, x_{i+1}, \dots, x_n)$ from a to u in G' where $x_1 = a$ and $x_n = u$. If $a = z$ then $z \in ext_E(X)$. Otherwise let x_{i+1} be the first node of q_{au} that is an intermediate node of q_z as well. Such node must exist because edges between chains can start from base nodes only and we cannot reach u from a otherwise. Then for $j \in [1..i]$: x_j, x_{j+1} are nodes of G , and (x_j, x_{j+1}) edges of q_{au} represent directed paths containing lattice edges only in G . Therefore there is a $p_{az} = (x_1, \dots, x_i, z)$ path in G from $a = x_1$ to z such that $length(p_{az}) = \sum_{s=1}^k w'_{node}(z_s)$. It means $z \in ext_E(X)$ in this case as well.

On the other hand, if $z \in ext_E(X)$ with grade γ_z , then there is a node $b \in X$ and a maximal length path p_{bz} from b to z in G such that $length(p_{bz}) = \gamma_z$. In that case, there is an edge from b to z_r in G' for some $r \in [1..l]$ such that $\sum_{s=1}^r w'_{node}(z_r) = length(p_{bz})$ and $z_r, z_{r-1}, z_1 \in ext_{E'}(X)$.

Consequently, $ext_{E'}(A) \cap ext_{E'}(O)$ contains fragments of chains generated from base nodes that are available from both A and O in G . Sum of node

weights in a fragment equals to the minimum of the lengths of the maximal length paths starting from A or O ending in the base node of the chain. Thus, $w_{fset}(ext_E(A) \cap ext_E(O)) = w'_{fset}(ext_{E'}(A) \cap ext_{E'}(O))$ and $w_{fset}(ext_E(O)) = w'_{fset}(ext_{E'}(O))$, i.e. equation (5) holds. \square

Note, that G' is acyclic by its construction but does not necessarily define a lattice. Therefore, we build a concept lattice from G' in which matching values of applications to offers will also be preserved.

3.2 Concept lattice

First, we define a formal context and formal concepts based on G' . Let (V'_1, V'_2, T') be a formal *context*, where $V'_1 = V'_2 = V'$ and $(v_i, v_j) \in T'$ iff v_j is available from v_i via directed path supposing that the relation is reflexive. Consider the element of V'_1 as start points and the element of V'_2 as end points of directed paths in G' . Let $I \subseteq V'_1$ and $J \subseteq V'_2$ and let us define their dual sets I^{D_s} and J^{D_e} as follows:

$$I^{D_s} = \{b \in V'_2 \mid (a, b) \in T' \text{ for all } a \in I\}$$

$$J^{D_e} = \{a \in V'_1 \mid (a, b) \in T' \text{ for all } b \in J\}$$

A *concept* of the context (V'_1, V'_2, T') is a pair $\langle I, J \rangle$ such that $I \subseteq V'_1$, $J \subseteq V'_2$ and $I^{D_s} = J$, $J^{D_e} = I$. I is called an *extent* of $\langle I, J \rangle$, and J is called an *intent* of $\langle I, J \rangle$.

| | Bot | B | C | C1 | C2 | D | D1 | D2 | A | A1 | Top |
|-----|-----|---|---|----|----|---|----|----|---|----|-----|
| Bot | X | X | X | X | X | X | X | X | X | X | X |
| B | | X | | X | | | X | | X | X | X |
| C | | | X | X | X | | X | X | X | X | X |
| C1 | | | | X | | | | | | | X |
| C2 | | | | X | X | | | | | | X |
| D | | | | X | X | X | X | X | | X | X |
| D1 | | | | | | | X | | | | X |
| D2 | | | | | | | X | X | | | X |
| A | | | | | | | | | X | X | X |
| A1 | | | | | | | | | | X | X |
| Top | | | | | | | | | | | X |

Table 1: Formal context (V'_1, V'_2, T')

Table 1 shows the formal context (V'_1, V'_2, T') that was generated based on graph G' of Figure 1. Labels of rows and columns represent the elements of V'_1 and the elements of V'_2 , respectively. There is an X in row i column j if $(i, j) \in T'$, i.e. j is available from i via directed path in G' .

Lemma 3.2. *If G' is an acyclic graph, then*

- (1) For every concept $\langle I, J \rangle$ of the context (V'_1, V'_2, T') : $I \cap J = \{v\}$ for some $v \in V'$ or $I \cap J = \emptyset$
- (2) For every $v \in V'$: there is a concept $\langle I_v, J_v \rangle$ in the context (V'_1, V'_2, T') such that $I_v \cap J_v = \{v\}$.

Proof.

- (1) Indirectly, suppose that for a concept $\langle I, J \rangle$ of (V'_1, V'_2, T') and for two different nodes $u, v \in V'$: $u, v \in (I \cap J)$ holds. In this case $(u, v) \in T'$ and $(v, u) \in T'$ hold as well. It would mean that there is a cycle in G' which is a contradiction as G' is acyclic.
- (2) For a node $v \in V'$ let $J_v = \{v\}^{D_s}$ be the set of all nodes that are available from v via directed path (including v itself). Let $I_v = J_v^{D_e}$, then $v \in I_v$. If $I_v = \{v\}$, then $\langle I_v, J_v \rangle$ is the concept we are looking for.
 Otherwise, suppose that for a node u such that $u \neq v$: $u \in I_v = J_v^{D_e} = (\{v\}^{D_s})^{D_e}$. That means $(u, v) \in T'$, i.e. v is available from u . As T' is a transitive relation $\{v\}^{D_s} \subseteq \{u, v\}^{D_s}$. However $\{u, v\}^{D_s} \subseteq \{v\}^{D_s}$ because $\{u, v\}^{D_s}$ cannot contain such node that is not available from all nodes of $\{u, v\}$. Following this construction we can get that if $J_v^{D_e} = I_v = \{u_1, \dots, u_i, v\}$, then $I_v^{D_s} = \{u_1, \dots, u_i, v\}^{D_s} = \{v\}^{D_s} = J_v$. Therefore $\langle \{u_1, \dots, u_i, v\}, \{v\}^{D_s} \rangle$ is a concept such that $\{u_1, \dots, u_i, v\} \cap \{v\}^{D_s} = \{v\}$. \square

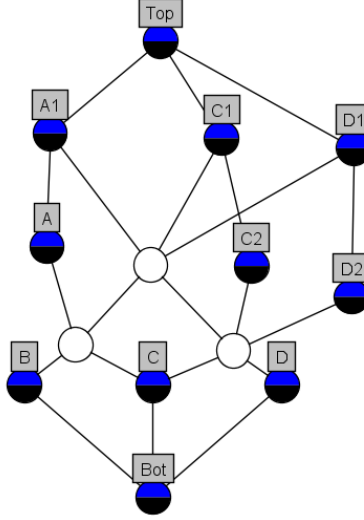
Let $\mathcal{B}(V'_1, V'_2, T')$ be the set of all formal concepts in the context, and \leq be a subconcept-superconcept order over the concepts such that for any $\langle A_1, B_1 \rangle, \langle A_2, B_2 \rangle \in \mathcal{B}(V'_1, V'_2, T')$: $\langle A_1, B_1 \rangle \leq \langle A_2, B_2 \rangle$, iff $A_1 \subseteq A_2$ (or, iff $B_2 \subseteq B_1$). $(\mathcal{B}(V'_1, V'_2, T'), \leq)$ is called *concept lattice* [10] and let $cl((\mathcal{L}, E_{ext}))$ denote the concept lattice obtained from the extension graph $ext(\mathcal{L}, E_{ext})$.

Figure 2¹ shows concept lattice of the context (V'_1, V'_2, T') from Table 1. Concepts $\langle I_v, J_v \rangle$ where $I_v \cap J_v = \{v\}$ are labeled with v . For example, $\langle I_{C_2}, J_{C_2} \rangle = \langle \{Bot, C, C_2, D\}, \{C_2, C_1, Top\} \rangle$. But, concepts $\langle I, J \rangle$ such that $I \cap J = \emptyset$ are unlabeled like the $\langle \{Bot, B, C\}, \{A, A_1, C_1, D_1, Top\} \rangle$ parent of concepts B and C . Another, larger example is the ontology on Figure 3 with added extra edges from [19].

It is worth mentioning that the concept lattice $cl((\mathcal{L}, E_{ext}))$ generated from ontology \mathcal{L} endowed with extra edges E_{ext} coincides with the Dedekind-McNeille completion [8] of the poset obtained as transitive closure of acyclic directed graph $ext(\mathcal{L}, E_{ext})$. Indeed, the collection of upper bounds of a subset S of elements of the poset is exactly the collection of the vertices reachable from the vertices of S via directed paths in the directed graph. We use the concept lattice formulation for two reasons. First, a direct construction is obtained skipping the step of constructing the poset from the directed graph $ext(\mathcal{L}, E_{ext})$. Second, the concept lattice structure allows us to define node weights properly.

An offer $O = \{o_1, \dots, o_k\} \subseteq S = V \subseteq V'$ generates a filter $F_O \subseteq \mathcal{B}(V'_1, V'_2, T')$ in the concept lattice such that $F_O = \{\langle I, J \rangle \mid \exists \langle I_o, J_o \rangle \leq \langle I, J \rangle \text{ such that } I_o \cap$

¹ The concept lattices were generated using the Concept Explorer tool. Web page: <http://conexp.sourceforge.net/>

Fig. 2: Concept lattice of context (V'_1, V'_2, T')

$J_o = \{o\}$ for some $o \in O$. Similarly, an application A generates a filter F_A in the concept lattice.

Let $w_{con} : \mathcal{B}(V'_1, V'_2, T') \rightarrow [0, 1]$ be a concept weighting function such that for a concept $\langle I, J \rangle$ of $\mathcal{B}(V'_1, V'_2, T')$:

$$w_{con}(\langle I, J \rangle) = \begin{cases} w'_{node}(v) & \text{if } I \cap J = \{v\} \text{ for some } v \in V', \\ 0 & \text{otherwise.} \end{cases}$$

Let w_{fil} be a filter weighting function such that for a filter $F \in \mathcal{P}(\mathcal{B}(V'_1, V'_2, T'))$: $w_{fil}(F) = \sum_{\langle I, J \rangle \in F} w_{con}(\langle I, J \rangle)$.

Theorem 3.1. Let $G = (V, E)$ be a directed graph extending the lattice $\mathcal{L} = (S, \preceq)$ with extra edges and $cl((\mathcal{L}, E_{ext})) = (\mathcal{B}(V'_1, V'_2, T'), \leq)$ be the concept lattice constructed from G and w_{fil} be the filter weighting function. Let $O \subseteq S$ be an offer and $A \subseteq S$ be an application. Then,

$$match_{sym}(A, O) = \frac{w_{fil}(F_A \cap F_O)}{w_{fil}(F_O)}. \quad (6)$$

Proof. Based on Lemma 3.1 it is enough to prove that

$$\frac{w_{fil}(F_A \cap F_O)}{w_{fil}(F_O)} = \frac{w'_{fset}(ext_{E'}(A) \cap ext_{E'}(O))}{w'_{fset}(ext_{E'}(O))} \quad (7)$$

Let $\langle I_u, J_u \rangle$ and $\langle I_v, J_v \rangle$ be two concepts such that $I_u \cap J_u = \{u\}$ and $I_v \cap J_v = \{v\}$ where $u, v \in V'$, i.e. u and v are nodes of G' that is generated from G as

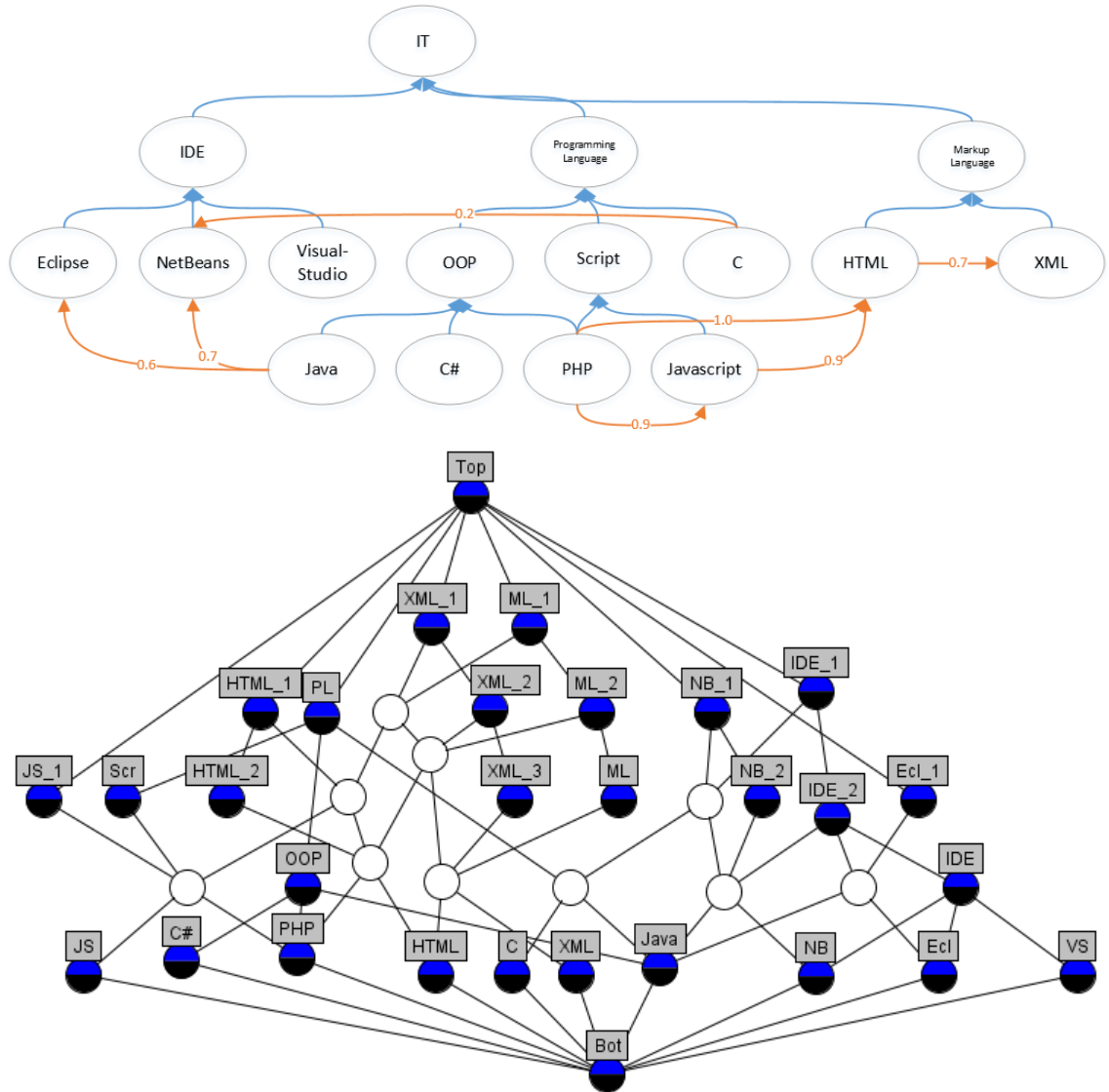


Fig. 3: Ontology with extra edges and the corresponding concept lattice

defined above. First, we will show that $\langle I_u, J_u \rangle \leq \langle I_v, J_v \rangle$ iff there is a directed path from u to v in G' .

If $\langle I_u, J_u \rangle \leq \langle I_v, J_v \rangle$, then $J_v \subseteq J_u$. But $u \in I_u$ and $v \in J_v \subseteq J_u$, and therefore $(u, v) \in T'$, i.e. there is a directed path from u to v in G' . On the other hand, if there is a directed path from u to v in G' , then $(u, v) \in T'$ therefore $v \in J_u = \{x \mid (u, x) \in T'\}$. However if v is available from u , then all nodes that are available from v , i.e. elements of J_v are also available from u as T' is a transitive relation. So $J_v \subseteq J_u$, but then $\langle I_u, J_u \rangle \leq \langle I_v, J_v \rangle$. It means that if $v \in \text{ext}_{E'}(O)$, then $\langle I_v, J_v \rangle \in F_O$ and if $\langle I_u, J_u \rangle \in F_O$, then $u \in \text{ext}_{E'}(O)$ and the same holds for $\text{ext}_{E'}(A)$ and F_A .

Since w_{con} assigns the same weights to concepts of F_A and F_O in form of $\langle I_v, J_v \rangle$ where $v \in V'$ as w'_{node} assigns to v and w_{con} assigns 0 to any other concepts therefore w_{fil} sums up the same values as w'_{set} , so equation (7) holds. \square

4 Extremal problems

It is a natural question how the size of the original ontology lattice $\mathcal{L} = (S, \preceq)$ relates to the sizes of the extension graph $\text{ext}(\mathcal{L}, E_{ext})$ and the concept lattice $cl((\mathcal{L}, E_{ext}))$ obtained from $\text{ext}((\mathcal{L}, E_{ext}))$. First, let us consider $\text{ext}(\mathcal{L}, E_{ext})$.

Proposition 4.1. *Let $\mathcal{L} = (S, \preceq)$ be an ontology lattice of $n+2$ nodes. Then for $G' = \text{ext}(\mathcal{L}, E_{ext}) = (V', E')$ we have $|V'| \leq n^2 + 2$. Furthermore, this estimate is sharp, that is for every positive integer n there exists ontology $\mathcal{L}_n = (S_n, \preceq)$ and set of extra edges E_{ext} such that $\text{ext}(\mathcal{L}_n, E_{ext})$ has $n^2 + 2$ vertices.*

Proof. Let the nodes of $\mathcal{L} = (S, \preceq)$ be $v_0, v_1, \dots, v_n, v_{n+1}$ with $v_0 = \text{bottom}$ and $v_{n+1} = \text{top}$. Then clearly there is no directed path from v_i $i > 0$ to v_0 in $\mathcal{L} \cup E_{ext}$, and the maximum weight path from any node v_i $i > 0$ to v_{n+1} is of weight 1, so no new nodes are generated from top and bottom. For v_j $0 < j < n+1$ there can be at most n distinct c^{j_1}, \dots, c^{j_l} values ($l \leq n$) that there exists a maximum weight path to v_j of weight c^{j_m} , as these paths could come from nodes v_i $i \in \{0, 1, \dots, n\} \setminus \{j\}$ only.

On the other hand, let $\mathcal{L}_c = (S, \preceq)$ be defined as v_1, \dots, v_n be pairwise incomparable elements, furthermore let $E_{ext}^c = \{(v_i, v_{i+1}) : i = 1, 2, \dots, n\}$ where $i+1$ is meant modulo n . Let the weight of each extra edge in E_{ext}^c be a fixed $0 < p < 1$ value. $\mathcal{L}_c \cup E_{ext}^c$ is shown on Figure 4. The maximum weight path from v_i to v_j has weight p^{j-i} if $1 \leq i < j \leq n$, while the weight is $p^{n-1-(j-i)}$ if $1 \leq j < i \leq n$, finally the weight is 1 for $i = 0 < j \leq n$. Thus, each node v_j $1 \leq j \leq n$ has exactly n different maximum weight path going into it, so $\text{ext}(\mathcal{L}_c, E_{ext}^c)$ has exactly $n^2 + 2$ nodes. \square

Our next goal is to bound the size of concept lattice $cl((\mathcal{L}, E_{ext}))$. The main question is how many “dummy” vertices are generated, that is concepts $\langle I, J \rangle$ such that $I \cap J = \emptyset$.

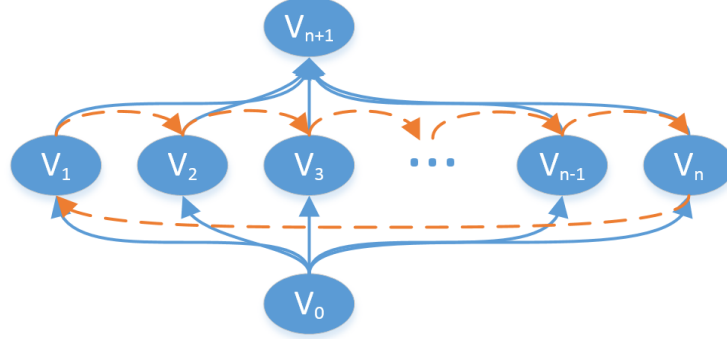


Fig. 4: Extremal example

Theorem 4.1. *Let $\mathcal{L} = (S, \preceq)$ be an ontology lattice of $n + 2$ nodes. Then for a set E_{ext} of extra edges $|cl((\mathcal{L}, E_{ext}))| \leq 2^n + n^2 - n + 1$ and this estimate is sharp, that is there exist $\mathcal{L}_n = (S_n, \preceq)$ and a set of extra edges E_{ext} such that $|cl((\mathcal{L}_n, E_{ext}))| = 2^n + n^2 - n + 1$.*

Proof. It is enough to prove that the number of concepts $\langle I, J \rangle$ such that $I \cap J = \emptyset$ is at most $2^n - n - 1$ to establish the upper bound by Lemma 3.2 and by Proposition 4.1. Indeed, Lemma 3.2 tells us that there is a concept corresponding to each element of $ext(\mathcal{L}_c, E_{ext}^c)$ and the other concepts $\langle I, J \rangle$ of $cl((\mathcal{L}, E_{ext}))$ have the property $I \cap J = \emptyset$.

Let v_{j_i} be a vertex of $ext(\mathcal{L}_c, E_{ext}^c)$ such that v_{j_i} is in the chain with base node v_j and $v_{j_i} \neq v_j$ furthermore assume that i is maximal with respect to $v_{j_i} \in I$ for some set of nodes of $ext(\mathcal{L}_c, E_{ext}^c)$. The nodes reachable from v_{j_i} via directed paths are $\{v_{j_i}, v_{j_{i-1}}, \dots, v_{j_1}, Top\}$, thus $I^{D^s} \subseteq \{v_{j_i}, v_{j_{i-1}}, \dots, v_{j_1}, Top\}$. This implies that $(I^{D^s})^{D^e} \supseteq \{v_{j_i}, v_{j_{i-1}}, \dots, v_{j_1}, Top\}^{D^e} \ni v_j$. However, $v_j \notin I$ by the maximality of i , so $(I^{D^s})^{D^e} \neq I$, that is I cannot be the extent of a concept of $cl((\mathcal{L}, E_{ext}))$. Suppose now that $\langle I, J \rangle$ is a concept and $v_{j_i} \in I$ as well as $v_{k_e} \in I$ for some $j \neq k$ so that neither v_{j_i} nor v_{k_e} is the base node of its chain. Then $I^{D^s} \subseteq \{v_{j_i}, v_{j_{i-1}}, \dots, v_{j_1}, Top\} \cap \{v_{k_e}, v_{k_{e-1}}, \dots, v_{k_1}, Top\} = \{Top\}$, so $I = \{Top\}^{D^e} = S$, that is $\langle I, J \rangle = \langle I_{Top}, J_{Top} \rangle$, i.e., $I \cap J = \{Top\}$. So we may assume that if $\langle I, J \rangle$ is a concept and $v_{j_i} \in I$ for some non-base node of a chain of $ext(\mathcal{L}_c, E_{ext}^c)$, then I does not contain non-base element of any other chain. Let i be minimal such that $v_{j_i} \in I$, where v_{j_0} is understood to be Top . We claim that $(I^{D^s}) = \{v_{j_i}, v_{j_{i-1}}, \dots, v_{j_1}, Top\}$. Indeed, we have $J = I^{D^s}$ and $I = J^{D^e}$. Let ℓ be maximal so that $v_{j_\ell} \in J$, then $J = \{v_{j_\ell}, v_{j_{\ell-1}}, \dots, v_{j_1}, Top\}$, since if there is a directed path from a node x to v_{j_ℓ} , then there is a path to any v_{j_t} for $\ell > t$, as well. Also, if $J = \{v_{j_\ell}, v_{j_{\ell-1}}, \dots, v_{j_1}, Top\}$, then for any node x , there is a directed path from x to every node in J iff there is a directed path from x to v_{j_ℓ} , since J itself forms a directed path from v_{j_ℓ} to Top . Now, by $I = J^{D^e}$ we have that $v_{j_i} = v_{j_\ell}$ and $\langle I, J \rangle = \langle I_{j_i}, J_{j_i} \rangle$.

From this we can conclude that if $\langle I, J \rangle$ is a concept such that $I \cap J = \emptyset$, then $I \subset S \setminus \{Bottom, Top\}$ and $|I| \geq 2$. The number of possible subsets I is the number of at least 2 element subsets of an n -element set, which is exactly $2^n - n - 1$.

To prove that the bound is sharp, consider again the extremal example $\mathcal{L}_c \cup E_{ext}^c$ shown on Figure 4. We have to show that for any subset I of size at least 2 of $\{v_1, \dots, v_n\}$, $\langle I, I^{D_s} \rangle$ is a concept, that is $I = (I^{D_s})^{D_e}$. Clearly, $I \subseteq (I^{D_s})^{D_e}$. Let i_j be defined as $i_j = \max\{i: v_{j_i} \in I^{D_s}\}$, that is $v_{j_{j_i}}$ is the lowest element of the j^{th} chain that is contained in I^{D_s} . Let $1 \leq j_1 < j_2 < \dots < j_t \leq n$ be such that $I = \{j_1, j_2, \dots, j_t\}$. Then it is easy to see that $n - i_j = \min\{j_k - j: j_k > j\}$ if $j < j_t$, otherwise $n - i_j = j_1 + n - j$, that is $n - i_j$ is the distance of j from the cyclically next $j_k \in I$. Let $j_0 \notin I$ and let j' be the element of $\{1, 2, \dots, n\}$ cyclically just before j_0 . Then $i_{j'} > 1$, while the only element of the j'^{th} chain that is an endvertex of a directed path from v_{j_0} is $v_{j'_1}$, so $v_{j_0} \notin (I^{D_s})^{D_e}$. \square

Another interesting question could be how the average or expected size of extension graph and the concept lattice relates to the size of the original ontology lattice. This is the topic of further investigations. The first task is finding a reasonable probability distribution for the extra edges.

5 Strict approach

As it was mentioned above, extra edges can be used based on different philosophies when extending offers. In this section, we show that strict matching values of applications to offers can also be preserved in the extension graph and in the concept lattice.

5.1 Preserving strict matching in extension graph

The main problem of preserving strict matching values in the extension graph is if extra edges are used to extend the offer, then extra nodes might appear in the extended offer whose weights are greater than 0. However, to address this problem, special node weighting functions can be defined depending on the offers.

For an offer O let w_{node}^O be a node weighting function that preserves the weights of the nodes that are available from O via lattice edges in G , and the nodes that were generated from such nodes in G' , and it assigns 0 to the other nodes, i.e. for a node $v \in V'$ let

$$w_{node}^O(v) = \begin{cases} w'_{node}(v) & \text{if } \exists v_j \in ext_{E_{lat}}(O) : v \in V_j, \\ 0 & \text{otherwise.} \end{cases}$$

Let w_{fset}^O be a fuzzy set weighting function that uses w_{node}^O , so for a fuzzy set f let $w_{fset}^O(f) = \sum_{(v, \gamma_v) \in f} \gamma_v \cdot w_{node}^O(v)$. Note, that computing w_{node}^O is a preprocessing step that has to be done once for all offers, and then w_{node}^O can be reused to calculate matching values of applications to the given offer.

With these weighting function a similar result can be shown as in Lemma 3.1.

Lemma 5.1. *Let $G = (V, E)$ be a directed graph extending the lattice $\mathcal{L} = (S, \preceq)$ with extra edges, w_{fset} be the fuzzy set weighting function, $G' = ext(\mathcal{L}, E_{ext}) = (V', E')$ be the extension graph, and w'_{fset} be the modified weighting function. Let $O \subseteq S$ be an offer with w_{node}^O and w_{fset}^O node and fuzzy set weighting functions, respectively and let $A \subseteq S$ be an application. Then,*

$$match(A, O) = \frac{w_{fset}(ext_E(A) \cap ext_{E_{lat}}(O))}{w_{fset}(ext_{E_{lat}}(O))} = \frac{w_{fset}^O(ext_{E'}(A) \cap ext_{E'}(O))}{w_{fset}^O(ext_{E'}(O))} \quad (8)$$

Proof. The proof is analogous to Lemma 3.1's. However, $ext_{E'}(A) \cap ext_{E'}(O)$ may contain chain fragment $(v_{y_k}, \dots, v_{y_1})$ of a chain $q_y = \{v_{y_l}, \dots, v_{y_1}, top\}$ with base node v_y where v_y is only available from O via extra edges in G , i.e. $v_y \in ext_E(O) \setminus ext_{E_{lat}}(O)$. But w_{node}^O assigns 0 to such v_{y_k}, \dots, v_{y_1} nodes by definition. In addition, G' contains lattice edges only, so $ext_{E'}(A)$ and $ext_{E'}(O)$ are crisps sets, so grades of their elements are always 1. Therefore $w_{fset}^O(ext_{E'}(A) \cap ext_{E'}(O)) = \sum_{u \in ext_{E'}(A) \cap ext_{E'}(O)} w_{node}^O(u) = w_{fset}(ext_E(A) \cap ext_{E_{lat}}(O))$ and analogously, $w_{fset}(ext_{E_{lat}}(O)) = w_{fset}^O(ext_{E'}(O))$. Thus equation (8) holds as well. \square

5.2 Preserving string matching in concept lattice

The same issue appears if we want to preserve strict matching values of applications to offers in the concept lattice as we solved in case of the extension graph, namely extended offer might contain new nodes with weight greater than 0. However, the offer specific weighting functions solve this issue as well.

We extend w_{node}^O to be able to use it for concepts. So, let w_{con}^O be a concept weighting function generated by an offer O such that for a concept $\langle I, J \rangle$:

$$w_{con}^O(\langle I, J \rangle) = \begin{cases} w_{con}(\langle I, J \rangle) & \text{if } I \cap J = \{v\} \text{ such that } \exists v_j \in ext_{E_{lat}}(O) : v \in V_j, \\ 0 & \text{otherwise.} \end{cases}$$

Let w_{fil}^O be the filter weighting function based on w_{con}^O , i.e for a filter $F \in \mathcal{P}(\mathcal{B}(V'_1, V'_2, T'))$: $w_{fil}^O(F) = \sum_{\langle I, J \rangle \in F} w_{con}^O(\langle I, J \rangle)$.

With these weighting functions, we can prove the following theorem similarly to Theorem 3.1.

Theorem 5.1. *Let $G = (V, E)$ be a directed graph extending the lattice $\mathcal{L} = (S, \preceq)$ with extra edges and $cl((\mathcal{L}, E_{ext})) = (\mathcal{B}(V'_1, V'_2, T'), \leq)$ be the concept lattice constructed from G and w_{fil} be the filter weighting function. Let $O \subseteq S$ be an offer with w_{con}^O and w_{fil}^O concept and filter weighting functions, respectively and let $A \subseteq S$ be an application. Then,*

$$match(A, O) = \frac{w_{fil}^O(F_A \cap F_O)}{w_{fil}^O(F_O)} \quad (9)$$

Proof. Analogously to Theorem 3.1’s proof and based on Lemma 3.1 it is enough to prove that

$$\frac{w_{fil}^O(F_A \cap F_O)}{w_{fil}^O(F_O)} = \frac{w_{fset}^O(ext_{E'}(A) \cap ext_{E'}(O))}{w_{fset}^O(ext_{E'}(O))}. \quad (10)$$

However, F_A and F_O contain concepts for all nodes of $ext_{E'}(A)$ and $ext_{E'}(O)$ respectively. But w_{con}^O assigns 0 to such $\langle I_v, J_v \rangle$ concepts where $v \in V'$ is not contained in any chain whose base was available from O in G using lattice edges only. Therefore w_{fil}^O sums up the same values as w_{fset}^O , i.e. equation (10) holds as well. \square

6 Related work

The aim of profile matching is to find the most fitting candidates to given profiles. Due to its various applications areas, it has become a widely investigated topic recently. Profiles can be represented as sets of elements and then numerous set similarity measures [5], such as Jaccard or Sørensen-Dice, are applicable to compute matching values.

There exist methods assuming that elements of profiles are organized into a hierarchy or ontology. For example, Lau and Sure [12] proposed an ontology based skill management system for eliciting employee skills and searching for experts within an insurance company. Ragone et al. [20] investigated peer-to-peer e-market place of used cars and presented a fuzzy extension of Datalog to match sellers and buyers based on required and offered properties of cars.

Di Noia et al. [7] placed matchmaking on a consistent theoretical foundation using description logic. They defined matchmaking as information retrieval task where demands and supplies are expressed using the same semi-structured data in form of advertisement and task results are ranked lists of those supplies best fulfilling the demands. Popov et al. [18] used filters in the ontology hierarchy lattice to represent profiles and defined matching function based on the filters.

We also assumed a structure among elements of profiles. We supposed this structure is an ontology that fulfills lattice properties as well and similarly to Popov’s proposal we also represented profiles with filters. However, we extended the ontology lattice with extra edges to capture such relationships that subsumptions cannot express. Then we showed how these edges are usable to refine the ontology.

There are several methodologies to learn ontologies from unstructured texts or semi-structured data [4][21]. Besides identifying concepts, discovering relationships between the concepts is a crucial part of ontology construction and refinement. Text-To-Onto [16] uses statistical, data mining, and pattern-based approaches over text corpus to extract taxonomic and non-taxonomic relations. In [22], various similarity measures were introduced between semi-structured Wikipedia infoboxes and then SVMs and Markov Logic Networks were used to detect subsumptions between infobox-classes.

We presented a method to refine ontology based on extra edges that represent some sort of quantifiable relationship between skills. These relationships can be given by domain experts, computed from statistics, or resulted by data mining techniques. For example, in [24] the authors used association rules and latent semantic indexing over job offers to detect relationships between competencies. In our method we defined profile extensions and weighting functions as well to preserve matching values of profiles computed from edge weights.

Formal concept analysis (FCA) [9] is also used to build and maintain formal ontologies. For example, Cimiano et al. [6] presented a method of automatic acquisition of concept hierarchies from a text corpus based on FCA. In [15], the authors used FCA to revise ontology when new knowledge was added to it.

In our method we used FCA to restore lattice properties after added new nodes and edges to it based on extra edges. However as we focused on preserving matching values of profiles during the transformations, we adapted our profile weighting functions to the modified ontology lattice as well.

7 Summary

In this paper we investigated how ontology lattices can be extended by additional information and used for semantic matching. We focused on the field of human resources and defined matching functions to find the most suitable applicant to a job offer, however, our results are applicable in other fields as well.

First, profiles of job applications and offers were represented as filters in an ontology lattice of skills that was built based on specialization relations between skills. Then, the ontology lattice got extended by additional information in form of extra edges describing quantifiable relations between the skills. A directed graph was built from the lattice endowed with extra edges to handle directed cycles that the new edges might have introduced and matching functions were defined based on reachable, or derived, nodes from profiles' nodes.

Two approaches were presented to extend profiles with derived nodes. In the first one, the offer and the applications were all extended, since the same profile can describe an application and an offer as well and these cases should be handled uniformly. In the second approach, only the applications were extended to help the employer differentiate better among the applicants.

We presented a method that eliminates directed cycles from the graph. It constructed an extension graph by adding node chains to the original lattice based on directed paths between nodes in the directed graph and node weights got also modified as part of the construction. An extension graph is a directed acyclic graph and therefore a poset but it is not necessary a lattice. Formal concept analysis was used to extend the poset into a concept lattice so that filters of this lattice could be used to calculate matching values. Different node weightings were used to preserve the original matching values in the two approaches.

Comparisons of the sizes of the ontology lattice and the generated acyclic directed graph, as well as the concept lattice were given.

References

1. European dictionary of skills and competences. <http://www.disco-tools.eu>.
2. International standard classification of education. <http://www.uis.unesco.org/Education/Pages/international-standard-classification-of-education.aspx>.
3. International standard classification of occupations, 2008.
4. Paul Buitelaar, Philipp Cimiano, and Bernardo Magnini. Ontology learning from text: An overview. *Ontology learning from text: Methods, evaluation and applications*, 123:3–12, 2005.
5. Seung-Seok Choi, Sung-Hyuk Cha, and Charles C Tappert. A survey of binary similarity and distance measures. *Journal of Systemics, Cybernetics and Informatics*, 8(1):43–48, 2010.
6. Philipp Cimiano, Andreas Hotho, and Steffen Staab. Learning concept hierarchies from text corpora using formal concept analysis. *J. Artif. Intell. Res.(JAIR)*, 24(1):305–339, 2005.
7. Tommaso Di Noia, Eugenio Di Sciascio, and Francesco M Donini. Semantic match-making as non-monotonic reasoning: A description logic approach. *J. Artif. Intell. Res.(JAIR)*, 29:269–307, 2007.
8. Bernhard Ganter and Sergei O. Kuznetsov. Stepwise construction of the dedekind-macneille completion. In Marie-Laure Mugnier and Michel Chein, editors, *Conceptual Structures: Theory, Tools and Applications*, pages 295–302, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.
9. Bernhard Ganter, Gerd Stumme, and Rudolf Wille. *Formal concept analysis: foundations and applications*, volume 3626. springer, 2005.
10. Bernhard Ganter and Rudolf Wille. *Formal concept analysis: mathematical foundations*. Springer Science & Business Media, 2012.
11. Peter Hájek. *Mathematics of Fuzzy Logic*. Kluwer Academic Publishers, Dordrecht, 1998.
12. Thorsten Lau and York Sure. Introducing ontology-based skills management at a large insurance company. In *Proceedings of the Modellierung*, pages 123–134, 2002.
13. M. Levandowsky and D. Winter. Distance between sets. *Nature*, 234(5):34–35, 1971.
14. Lianzhen Liu and Kaitai Li. Fuzzy filters of bl-algebras. *Information Sciences*, 173(1):141–154, 2005.
15. Dominic Looser, Hui Ma, and Klaus-Dieter Schewe. Using formal concept analysis for ontology maintenance in human resource recruitment. In *Proceedings of the Ninth Asia-Pacific Conference on Conceptual Modelling-Volume 143*, pages 61–68. Australian Computer Society, Inc., 2013.
16. Alexander Maedche and Raphael Volz. The ontology extraction & maintenance framework text-to-onto. In *Proc. Workshop on Integrating Data Mining and Knowledge Management, USA*, pages 1–12, 2001.
17. Jorge Martínez Gil, Alejandra Lorena Paoletti, Gábor Rácz, Attila Sali, and Klaus-Dieter Schewe. Accurate and efficient profile matching in knowledge bases, 2017. submitted for publication.
18. Nikolaj Popov and Tudor Jebelean. Semantic matching for job search engines – a logical approach. Technical Report 13-02, Research Institute for Symbolic Computation, JKU Linz, 2013.
19. Gábor Rácz, Attila Sali, and Klaus-Dieter Schewe. Semantic matching strategies for job recruitment: A comparison of new and known approaches. In *International*

- Symposium on Foundations of Information and Knowledge Systems*, volume 9616 of *LNCS*, pages 149–168. Springer, 2016.
20. Azzurra Ragone, Umberto Straccia, Tommaso Di Noia, Eugenio Di Sciascio, and Francesco M Donini. Fuzzy matchmaking in e-marketplaces of peer entities using datalog. *Fuzzy Sets and Systems*, 160(2):251–268, 2009.
 21. Mehrnoush Shamsfard and Ahmad Abdollahzadeh Barforoush. The state of the art in ontology learning: a framework for comparison. *The Knowledge Engineering Review*, 18(4):293–316, 2003.
 22. Fei Wu and Daniel S Weld. Automatically refining the wikipedia infobox ontology. In *Proceedings of the 17th international conference on World Wide Web*, pages 635–644. ACM, 2008.
 23. Maciej Wygralak. *Cardinalities of fuzzy sets*. Springer, 2003.
 24. Sabrina Ziebarth, Nils Malzahn, and Heinz Ulrich Hoppe. Using data mining techniques to support the creation of competence ontologies. In *AIED*, pages 223–230, 2009.