Professional Competence Identification Through Formal Concept Analysis

Paula R. Silva[1(✉)], Sérgio M. Dias[2(✉)], Wladmir C. Brandão[1(✉)], Mark A. Song[1(✉)], and Luis E. Zárate[1(✉)]

[1] Pontifical Catholic University of Minas Gerais (PUC Minas), Belo Horizonte, Brazil
paula.raissa@sga.pucminas.br,
{wladmir,song,zarate}@pucminas.br
[2] Federal Service of Data Processing (SERPRO), Belo Horizonte, Brazil
sergio.dias@serpro.gov.br

Abstract. As the job market has become increasingly competitive, people who are looking for a job placement have needed help to increase their competence to achieve a job position. The competence is defined by the set of skills that is necessary to execute an organizational function. In this case, it would be helpful to identify the sets of skills which is necessary to reach job positions. Currently, the on-line professional social networks are attracting the interest from people all around the world, whose their goals are oriented to business relationships. Through the available amount of information in this kind of networks it is possible to apply techniques to identify the competencies that people have developed in their career. In this scenario it has been fundamental the adoption of computational methods to solve this problem. The *formal concept analysis (FCA)* has been a effective technique for data analysis area, because it allows to identify conceptual structures in data sets, through conceptual lattice and implications. A specific set of implications, know as proper implications, represent the set of conditions to reach a specific goal. So, in this work, we proposed a FCA-based approach to identify and analyze the professional competence through proper implications.

Keywords: Formal concept analysis · Proper implications
Professional competence · On-line social networks

1 Introduction

Currently, the job market has become increasingly competitive. The educational and technological advancements mean that companies are demanding that professionals are prepared to take positions. So, the people that are starting or rethinking their professional career have needed some guidance to become potential candidates for job openings. This guidance can be offered based on evidences obtained through prior knowledge about the job market. The prior knowledge base can be composed by the set of competences that people have developed

S. Hammoudi et al. (Eds.): ICEIS 2017, LNBIP 321, pp. 34–56, 2018.
https://doi.org/10.1007/978-3-319-93375-7_3

in certain positions. An interesting source of information for this purpose is the on-line professional social networks, because in these networks the people have made available their professional resume.

In increasingly interconnected world, the on-line social networks attend different people's interests and address the communication and information needs of several user groups [1]. In particular, there are on-line professional social networks focused on a specific group of users interest is oriented to business. One of the largest and most popular on-line professional social network is the *LinkedIn*, which has more than 500 millions users distributed in more than 200 countries and territories [2].

LinkedIn users can create their professional profiles, and to made available informations like skills, competences, and experience. They can also interact with each other, look for jobs, and another functionalities. Thus, *LinkedIn* provides a source of professional information that can be exploited by enterprise managers in different ways, like to find people with appropriate competences to fulfill specific positions. In addition, the size and diversity of user-generated content has created the opportunity to identify behavioral trends and user communities by computational methods. In this scenario, the *formal concept analysis (FCA)* presents itself as a technique that can be applied for this purpose.

FCA presents a mathematical formulation for data analysis, which identify conceptual structures from a data set [3]. It also presents an interesting unified framework to identify dependencies among data, by understanding and computing them in a formal way [4]. It is a branch of lattice theory motivated by the need for a clear formalization of the notions of concept hierarchy.

There are two ways to extract and represent knowledge from FCA: conceptual lattice and implications. In this work, we applied a particularly type of implications, know as *proper implications* [5]. We say that a proposition P logically implies a proposition Q ($P \rightarrow Q$), if Q is true whenever P is true. The set of proper implications have a minimal left-hand side and only one item in right-hand side [5]. It has been applied when the need is to find the minimum conditions to lead a goal. In this article, the proper implications represent the set of minimum professional's skills (conditions) for achieve a job position (goal). For example, the proper implication $\{$*statistic, machine learning, databases*$\} \rightarrow \{$*data scientist*$\}$ represent a minimum set of skills which are necessary to be a data scientist.

Several authors have been applied FCA to address research problems related to *social networks analysis (SNA)*. Note that, there are other methods to retrieve knowledge from social networks. They are usually based on graph theory, clustering and frequent item-sets, which provide an approach to represent a social network through a formal way. Additionally, there are several FCA to SNA applications, such as ontology-based technique [6], social communities [7], network representation through concept lattice [8], contextual pre-filtering process and identifying user behavior through implications [9].

In this article, we proposed a FCA-based approach to identify professional competence through data from *LinkedIn*. First, we conceptually model

a *LinkedIn* user profile according to the *model of competences* [10], because this model has more acceptance in industry and academia [11]. Second, as an input data set, our approach needs an incidence table (formal context) and a subset of proper implications are expected as an output. Lastly, a proper implication represent careers trajectories, by minimum professional's skills (conditions) for achieve a position (goal). These implications were extracted using our proposed algorithm named *PropIm*. The *PropIm* algorithm was proposed to extract proper implications with support greater than zero. It is important to note that this article is an extended version of [12].

The main contributions of this paper are: a professionals data set scraped from *LinkedIn*, a FCA-based approach, and experiments set for apply FCA to professional career analysis.

The structure of the article follows as: In Sect. 2, we present the preliminary definitions related to FCA. In Sect. 3, we report related work that applied FCA to social networks analysis. In Sect. 4, we present our FCA-based approach to SNA. In Sect. 5, we report experiments and results analysis. Finally, in Sect. 6, we present the conclusions and some proposals for future works.

2 Formal Concept Analysis

In this section, we introduce concepts related of formal concept analysis (FCA) reported on literature [13].

2.1 Formal Context

Formally, a formal context is a triple (G, M, I), where G is a set of objects (rows), M is a set of attributes (columns) and I are incidences. It is defined as $I \subseteq G \times M$. If an object $g \in G$ and an attribute $m \in M$ have a relationship I, their representation is $(g, m) \in I$ or gIm, which could be read as "object g has attribute m". When an object has an attribute, the incidence is identified and represented by "x". In the formal context shown in Table 1, rows are objects representing users, and the columns are professional skills and positions.

Given a subset of objects $A \subseteq G$ of formal context (G, M, I), there is an attribute subset of M common to all objects of A, even if empty. Likewise, given a set $B \subseteq M$, there is an object subset that shares the attributes of B, even if empty. These relationships are defined by derivation operations:

$$A' := \{m \in M | gIm \forall g \in A\} \tag{1}$$

$$B' := \{g \in G | gIm \forall m \in B\} \tag{2}$$

A formal context (G, M, I) is a clarified context, when $\forall\ g, h \in G$, from $g' = h'$ it always follows that and, correspondingly $m' = n'$ implies $m = n\ \forall\ m, n \in M$. The clarification process consists in maintaining one element (objects and attributes) from a set of equal elements eliminating the others. In this process, the number of objects and attributes can be reduced while retaining lattice form [3].

Table 1. Example context of an user's *LinkedIn* skills. The attributes are: a: networks, b: mobile application, c: software engineering, d: data bases, e: graphic processing, f: computer architecture, g: operational systems (Source: [12] p. 124).

	a	b	c	d	e	f	g
user 17	x	x		x	x		x
user 18			x	x			
user 19		x				x	x
user 20	x	x		x	x		
user 21			x	x			x
user 22		x				x	
user 23	x	x		x	x		x
user 24			x	x			

2.2 Formal Concept

From formal contexts we can obtain formal concepts, defined as pairs (A, B), where $A \subseteq G$ is called extension and $B \subseteq M$ and is called intention, and they must follow conditions $A = B'$ and $B = A'$ (Eqs. 1 and 2) [3].

Based on the formal context from Table 1, we generated the formal concept ({user 18, user 21, user 24}, {software engineering, data bases}), where elements of subset B are {software engineering, data bases}, that, by derivation (Eq. 2) yield subset A = {user 18, user 21, user 24}. This formal concept represents the subset of users (objects) who share skills in *software engineering* and *data bases*.

It is important to note that a formal concept corresponds to any aspect of the problem domain, represented by objects and attributes, in which exists some kind of comprehension and understanding.

2.3 Concept Lattice

With all formal concepts sorted hierarchically by order of inclusion $\subseteq$, we can build the concept lattice. Sorting must be done, so that, the concept (A_1, B_1) is considered less than or equal to (A_2, B_2) if and only if, $A_1 \subseteq A_2$ (equivalent to $B_2 \subseteq B_1$). In this case, the concept (A_1, B_1) is called sub-concept and the concept (A_2, B_2) super-concept. In Fig. 1 is shown an example of a concept lattice from the formal context in Table 1. It is important to note that concept lattice was built with the *Conexp*[1] software.

Generally the lattice is represented by a graph, in which the nodes are the formal concepts and the edges are the relationships among nodes. At the top of the graph there is a single node called *supremum*, whose extension is composed by all objects. The lower node is called *infimum*, whose intention contains the set of all attributes.

[1] Conexp: http://conexp.sourceforge.net/.

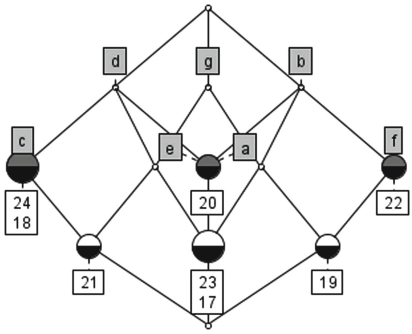

Fig. 1. Example of concept lattice (Source: [12] p. 125).

2.4 The Set of Implications

Given a formal context (G, M, I) or a concept lattice $\mathscr{B}(G, M, I)$, these can be extracted exact rules or approximate rules (rules with statistical values, for example, support and confidence) that express in a alternative way the underlying knowledge. The exact rules can be classified in implication rules and functional dependencies, while the approximation rules are divided in classification rules and association rules. It is particularly important in this work, to get the social networks users' behavior, consider exact rules. From now these rules are going to called only *implications*. Follows the definition of an implication [13]:

Definition 1. *Being a formal context whose attributes set is M. An implication is an expression $P \rightarrow Q$, which $P, Q \subseteq M$.*

An implication $P \rightarrow Q$, extracted from a formal context, or respective concept lattice, have to be such that $P' \subseteq Q'$. In other words: every object which has the attributes of P, it also have the attributes of Q.

Note that, if X is a set of attributes, then X *respects* an implication $P \rightarrow Q$ iff $P \not\subseteq X$ or $Q \subseteq X$. An implication $P \rightarrow Q$ *holds* in a set $\{X_1, \ldots, X_n\} \subseteq M$ iff each X_i respects $P \rightarrow Q$; and $P \rightarrow Q$ *is an implication of the context* (G, M, I) iff it holds in its set of object intents (an object intent is the set of its attributes). An implication $P \rightarrow Q$ *follows from* a set of implications $\mathscr{I}$, iff for every set of attributes X if X respects $\mathscr{I}$, then it respects $P \rightarrow Q$. A set of implications $\mathscr{I}$ is said to be *complete* in (G, M, I) iff every implication of (G, M, I) follows from $\mathscr{I}$. A set of implications $\mathscr{I}$ is said to be *redundant* iff it contains an implication $P \rightarrow Q$ that follows from $\mathscr{I} \backslash \{P \rightarrow Q\}$. Finally, an implication $P \rightarrow Q$ is considered *superfluous* iff $P \cap Q \neq \emptyset$.

In social networks will be convenient that each implication represent a minimum behavior. For this, we will require that the complete set of implications $\mathscr{I}$ of a formal context (G, M, I) have the following characteristics, to be used as representative of the process:

– the right hand side of each implication is unitary: if $P \rightarrow m \in \mathscr{I}$, then $m \in M$;

- superfluous implications are not allowed: if $P \rightarrow m \in \mathscr{I}$, then $m \notin P$;
- specializations are not allowed, i.e. left hand sides are minimal: if $P \rightarrow m \in \mathscr{I}$, then there is not any $Q \rightarrow m \in \mathscr{I}$ such that $Q \subset P$.

A complete set of implications in (G, M, I) with such properties is the so called set of *proper implications* [5] or *unary implication system* (UIS) [14].

Definition 2. *Let $\mathscr{J}$ be the complete closed set of implications of a formal context (G, M, I). Then the set of* proper implications *$\mathscr{I}$ for (G, M, I) is defined as: $\{P \rightarrow m \in \mathscr{J} \mid P \subseteq M$ and $m \in M \setminus P$ and $\forall Z \subset P \colon Z \rightarrow m \notin \mathscr{J}\}$.*

Table 2. Proper implications extracted from formal context in Table 1 (Source [12] p. 125).

P	→	m
{e} {b, d}	→	{a}
{a} {e} {f}	→	{b}
{d, g}	→	{c}
{a} {c} {e}	→	{d}
{a} {b, d}	→	{e}

The Table 2 shows the set of proper implications, from the formal context example (Table 1). For example, in implication $b,\ d \rightarrow a$, the set P is composed by the set of attributes $\{b,\ d\}$, $m = \{a\}$ and $\rightarrow$ symbol represents the incidence. P and m are called as premise and conclusion. So the implication $b,\ d \rightarrow a$, can be read as the premise $b,\ d$ implies in conclusion a. It is important to note that, the conclusion a can has more than one minimal premises. According to Definition 2, the premises $\{e\}$ and $\{b,\ d\}$ are minimal, and they can not be a subset of another premises that imply in a. For example, the implication $e \rightarrow a$ is a proper implication, but $e,\ g \rightarrow a$ is not a proper implication.

3 Related Works

Recently, several authors have applied FCA for social networks analysis [15–22]. These works have been motivated by the interest in understand and interpret social networks through mathematical formulation. The main subjects are social network representation as a concept lattice, community detection, concepts mining, ontology analysis and rule mining through implications.

In [23], the authors proposed knowledge-based model of influence applying FCA to compute minimal generators and closed sets directly from an implicational system, for obtain a structure of user's influence. The data was extracted from *Twitter* social network, it was transformed into a formal context and it was generated the Duquenne-Guigues basis. In [24], the author shows an approach to analyze a data base composed by internet's access logs. The authors apply the minimal set of implications and complex networks theories to identify substructures, that are not easily visualized with two-mode networks. In [25], the authors propose an FCA-based approach to build canonical models, which represents *Orkut* access' patterns. These papers resemble this work, because they also talks about how to map social networks in terms of objects and attributes, and extract knowledge through implications rules set.

As this work, in [26] the authors apply FCA to identify interaction patterns through data scraped from *LinkedIn*. They did not work with professional competencies and proper implications, like us, because their goal was classify users' behavior though their network interactions, and the knowledge was extracted from conceptual lattice.

Another authors like [27–29] have developed works related to behavior pattern mining in social networks using *LinkedIn* data. The main goals were identification of potential candidates for job positions, career path analysis and recommendation of professional skills. Even not applying FCA, these works were highlighted to show the importance of develop works related to professional social networks, as well as the solution of computational methods to optimize professional recruitment and development processes.

In general, the FCA has been applied to mining social media, because the FCA theory presents a formalism for the representation of network structure, behavior identification and knowledge extraction, through formal representation of problem domain from objects, attributes and their respective incidences.

The proposed approach in this work combines techniques of formal concept analysis, patter mining and model of professional competence. From the literature review process it was noticed that for the solution of problems related to the discovery of knowledge in on-line professional social networks, perhaps this approach is an unpublished work.

4 FCA-Based Approach

In this work, the problem of analysis and representation of professional profiles in social networks, can be grounded by building a conceptual model, that merge the social network with professional skills theory, and the transformation of this model to a formal context. After scraping and preprocess the data to a formal context, we can extract the set of implications to be analyzed. The Fig. 2 shows the methodology steps proposed to analyze *LinkedIn* social network through FCA.

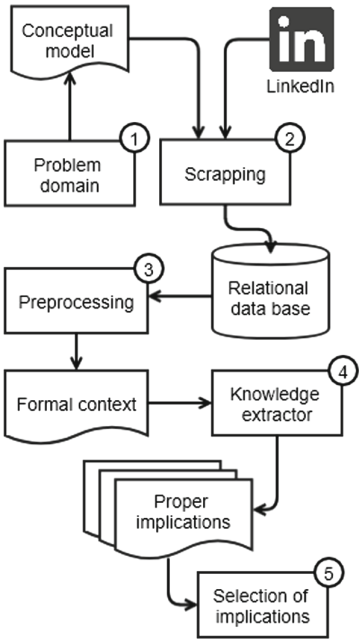

Fig. 2. Methodology based in FCA to SNA (Adapted from [12] p. 126).

4.1 Problem Domain

According to Fig. 2, the first step (1) was to construct the conceptual model according to the problem to be treated. In this case, the problem involves the characterization of a person as a professional. We adopted the *model of competence* [10], because this model has greater acceptance in both academic and business, since it seeks to integrate aspects related to this work, such as technical issues, cognition and attitudes [11]. A professional is characterized by his competence in accomplishing a certain purpose [10]. The competence is composed by three dimensions: *knowledge*, *attitudes* and *skill*. The dimensions are interdependent, because the behavior of a professional is determined not only by his *knowledge* but also by the *attitudes* and *skills* acquired over time. The *knowledge* dimension is linked to academic training and complementary courses. The *skill* dimension is related to a person's professional experiences. Finally, the *attitude* dimension is related to the way of people interact in their professional environment.

The conceptual model was built from the classification of categorical informations contained in *LinkedIn* users' pages. This classification process consisted in mapping the categories for the competency model dimensions. In this case the informations about academic education and complementary courses were attributed to the *knowledge* dimension. The information about professional experience were linked to *skill* dimension. The information related to the way of users

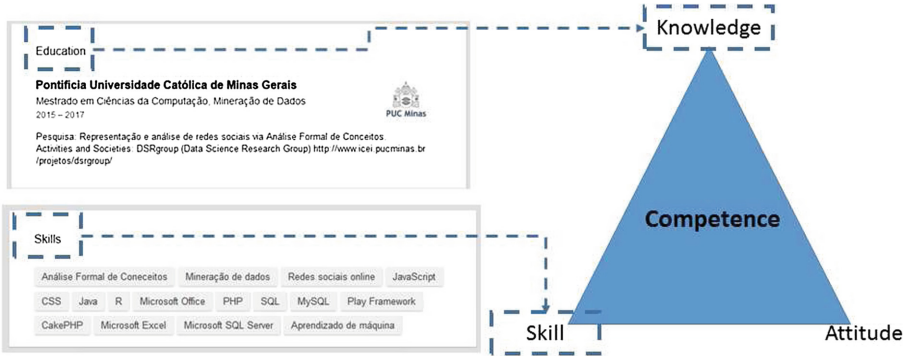

Fig. 3. Mapping LinkedIn informational categories.

interact in the social network has been attributed to the *attitudes* dimension. The Fig. 3 exemplifies how the categories were mapped into the model of competence dimensions.

As result, we obtained the conceptual model for characterize the professional profile from the model of competence (Fig. 4). In the conceptual model, the first labels' level is related to the concept of *competence*, at the second level there are the 3 dimensions related to the main concept, at the third level there are the 14 aspects that correspond to the *LinkedIn* sections, and at the fourth level there are 51 variables that correspond to fields that users can fill.

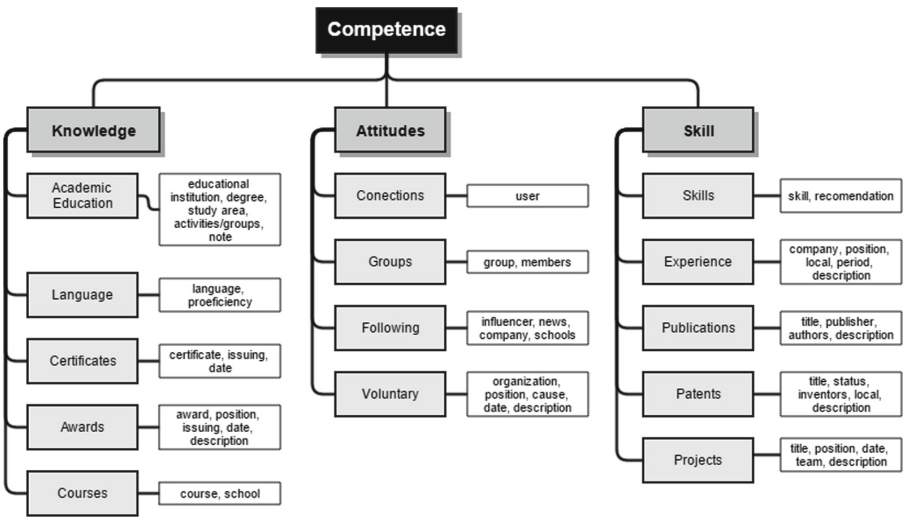

Fig. 4. Professional identification through model of competence (Source: [12] p. 127).

4.2 Scrapping

FCA techniques to social network analysis can yield insights into user behaviors, detecting popular topics, and discovering groups and communities with similar characteristics. So, a task of gathering the data on a specific subject is needed. In this case, the second step (2) of the methodology represents the Scraping component that is responsible for collecting the *LinkedIn* user data.

The collection process was divided into two phases. The first one selects the initial seeds, randomly two user profiles were selected. This amount of initial seed was considered satisfactory for the data collection process, due to the total of profiles obtained being sufficient for the study. It was defined that, as case study, the data would be collected from people of Belo Horizonte, Minas Gerais, Brazil and they must have at least graduate courses in the information technology (IT) area.

The second phase goal is collect the public profiles. As the *LinkedIn* does not provide an API (Application Programming Interface) to extract data directly from the server, an approach, known as open collection, has been adopted to extract data from users' public pages.

The Fig. 5 exemplifies the flow of data collection process. The process starts by accessing a seed. In the public profile there is a section denoted as "People also viewed" - a list with the 10 most similar profiles related to the visited profile [28]. The collector looks up these addresses and verifies which profiles meet the scope. Valid profiles are stored and each one becomes a new seed to extract new links, restarting the collection process until it reaches the stop criterion. The stopping criterion is based on the percentage of new profiles. Each iteration checks if 65% of the profiles were already in the database.

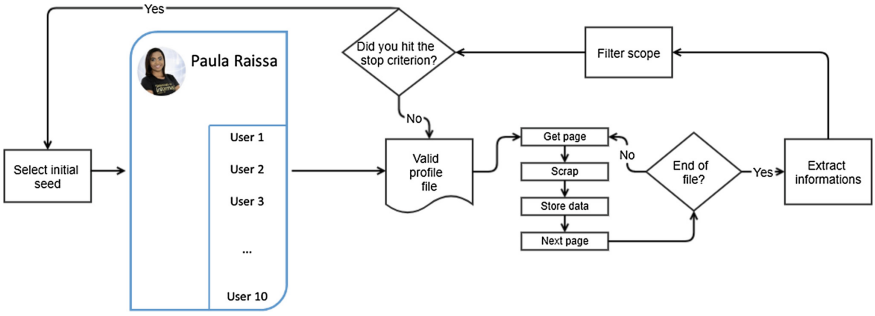

Fig. 5. Scraping *LinkedIn* profiles' process.

4.3 Preprocessing

The Preprocessing (3) component is responsible for pre-processing the data extracted in the previous step. In this work only the variables *skills* and *experience* were considered. However, in future works, the other dimensions will be included.

For the construction of the formal context, we considered, as attribute, each value of the *competence* and *experience* variables. Each user (professional of *LinkedIn*) is considered as an object. In the first version of the formal context, 4000 attributes and 1280 objects were detected. As such values are texts and *LinkedIn* allows users to fill the corresponding fields with a free text, some problems have been detected. In this case, it was necessary to create an ETL process (Extract Transform Load) to clean and transform the data, aiming to reduce the amount of attributes.

Table 3. Table of generic and specific terms.

Generic term	Specific term
Java	Java language
	JPA
	JSF
Software developer	Developer
	Programmer
	Program developer

The ETL process consists of two stages. In the first step, we applied basic procedures for string cleaning, like: UTF-8 encoding correction, accent removal, and standardization of all terms for the English language through Google Translate API[2]. In the second stage, we apply techniques to attribute reductions. Such reductions were based on terms with semantic relevance, in which the attributes *skill* and *experience* could be reduced. For example (Table 3), attributes as *JPA*, *JSF* were renamed to *java frameworks*; attributes as *developer*, *programmer*, *software developer*, *program developer* were renamed to *software developer*. The vague nouns or trademark terms, as *bachelor*, *engineer*, *accessibility*, *microsoft*, were removed, because they are not relevant to our study.

At the end of the preprocessing step, a formal context was created with 366 attributes and 970 objects, which 61 attributes are related to *experience* and 305 to *skills*.

4.4 Knowledge Extractor

The *Impec* [5] is the best-known algorithm for extracting proper implications from the formal context. This is due to the strategy that the algorithm adopts at the moment in which it finds the premises and their respective conclusions.

On the other hand, the strategy used by the algorithm is computationally inefficient and it can not be optimized or distributed. In some cases, only a few context attributes are interesting to be in the conclusion. The traditional algorithms only allow to generate the complete set, being necessary a step of filtering

[2] Translate API: https://cloud.google.com/translate/.

the rules to select to those whose attributes of interest appear as conclusion of the implications, occurring an unnecessary computational effort.

Based on the problems described above, an algorithm was proposed to generate the set of proper implications from the formal context. The algorithm allows that the set of proper implications can be extracted from the attributes of interest as the conclusion of implications. In general, the algorithm receives a formal context as input, finds the minimum premises for each conclusion by combining attributes, applies a pruning heuristic, and uses the derivation operators to validate the implications.

The pseudo-code of *PropIm* is given at Algorithm 1. The main objective is to find implications whose left side is minimal and the right side has only one attribute. The algorithm needs a formal context (G, M, I) as input, and its output is a set of proper implications. Line 1 initializes the set $\mathscr{I}$ with empty set. The following loop (Lines 2–17) looks at each attribute in the set M. We initially suppose that each attribute m can be a conclusion for a set of premises. For each m, we compute a left-hand side $P1$.

```
   Input  : Formal context (G, M, I)
   Output: Set of proper implications ℐ
 1 ℐ = ∅
 2 foreach m ∈ M do
 3 |   P = m''
 4 |   size = 1
 5 |   Pa = ∅
 6 |   while size < |P| do
 7 |   |   C = (P choose size)
 8 |   |   P_C = getCandidate(C, Pa)
 9 |   |   foreach P1 ⊂ P_C do
10 |   |   |   if P1' ≠ ∅ and P1' ⊂ m' then
11 |   |   |   |   Pa = Pa ∪ {P1}
12 |   |   |   |   ℐ = ℐ ∪ {P1 → m}
13 |   |   |   end
14 |   |   end
15 |   |   size++
16 |   end
17 end
18 return ℐ
```

Algorithm 1. PropIm algorithm (Source: [12] p. 128).

To reduce the searching space, the algorithm finds the right side P for a left side m from a set of attributes common to m objects. After, it finds sets of possible premises for m based on P. The *size* counter determines the size of each premise, as the smallest possible size is 1 (an implication of type $\{b\} \rightarrow \{a\}$), it is initialized with 1 (Line 4).

A set of auxiliary premises Pa is used, where all valid premises found leading to m conclusion are stored (at Line 5 Pa is initialized as empty). In the loop,

from Lines 6–16, the set of minimum premises is found and is bounded by $|P|$. In Line 7, the C set gets all combinations of size *size* from elements in P. In Line 8, the set of candidate premises is formed through the function *getCandidate* which will be described next.

Each candidate premise $P1 \subset P_C$ is checked to ensure if the premise $P1$ and the conclusion m results in a valid proper implication. Case $P1 \neq \emptyset$ and $P \subset P1'$, the premise $p1$ is added to the set of auxiliary premises Pa and $\mathscr{I} = \mathscr{I} \cup \{P1 \rightarrow M\}$.

A neighborhood search heuristic was implemented by the *getCandidate* function (Pseudo-code in Algorithm 2). The goal is to find, in the set of C combinations, all subsets B that do not contain some attribute that already belongs to some valid premise of Pa. It receives, as parameter, the sets C and Pa, and returns a set D of proper premises.

```
1  Function getCandidate (C, Pa)
2  |  D = ∅
3  |  foreach a ∈ A|A ⊂ Pa do
4  |  |  foreach B ⊂ C do
5  |  |  |  if a ∉ B then
6  |  |  |  |  D = P_C \ B
7  |  |  |  end
8  |  |  end
9  |  end
10 |  return D
```

Algorithm 2. Function getCandidate (Source: [12] p.129).

Table 4 shows the steps of *PropIm* algorithm, on the example from Table 1. $\mathscr{I}$ contains initially $\emptyset$. The first value to m is a (first attribute from formal context) and m'' is the set of attributes $\{b, d, e, g\}$. The *size* of combination sets is 1, so $C = \{\{b\}, \{d\}, \{e\}, \{g\}\}$. Pa contains initially $\emptyset$, $C \diagdown \emptyset$, so the set of attributes returned by the function *getCandidate* is $Pc = \{\{b\}, \{d\}, \{e\}, \{g\}\}$. For each subset of Pc, only the element $\{e\}$ attends the condition in Line 10 (Algorithm 1), because $\{e\}' = \{17, 20, 23\} \subset m'$. The set $\{e\}$ is added to Pa and the pair $\{e \rightarrow a\}$ is added to $\mathscr{I}$. When Pc is $\emptyset$ and *size* is $|P|$ the loop to $m = a$ is closed. So, the same steps happens for all attributes, from formal context, imputed to m.

From the formal context (G, M, I), after to apply the *PropIm* algorithm, we identified two subset of proper implications types with the following characteristics:

- The subset of proper implications, whose premises have common conclusions: $type\_\beta = \{P \rightarrow Q, S \rightarrow Q \in \mathscr{I} \mid \forall P, S, Q \subseteq M\}$. This type of implication represents that different conditions can imply in the same goal.
- The subset of implications, in which the same premise implies in different conclusions: $type\_\delta = \{P \rightarrow Q,\ P \rightarrow R \in \mathscr{I} \mid Q \neq R;\ \forall\ P, S, Q, R \subseteq M\}$. This subset shows that the premises can be shared by different conclusions.

Table 4. Example of *PropIm* algorithm execution (Source: [12] p. 129).

m	P	$size$	C	Pc	Pa	$\mathscr{I}$
a	{b, d, e, g}	1	{{b}, {d}, {e}, {g}}	{{b},{d}, {e}, {g}}	{{e}}	{{e→a}}
a	{b, d, e, g}	2	{{bd}, {be}, {bg}, {de}, {dg}, {eg}}	{{bd},{bg}, {dg}}	{{e}, {bd}}	{{e→a}, {bd→a}}
a	{b, d, e, g}	3	{{bde}, {bdg}, {beg}, {deg}}	∅	{{e},{bd}}	{{e→a},{bd→a}}
a	{b, d, e, g}	4	{{b, d, e, g}}	∅	{{e}, {bd}}	{{e→a},{bd→a}}
b	{a, d, e, f, g}	1	{{a}, {d}, {e}, {f}, {g}}	{{a},{d}, {e},{f},{g}}	{{a}, {e}, {f}}	{{e→a}, {bd→a}, {a→b}, {e→b}, {f→b}}
b	{a, d, e, f, g}	2	{{ad}, {ae}, {af}, {ag}, {de}, {df}, {dg}, {ef}, {eg}, {fg}}	{{dg}}	{{a}, {e}, {f}}	{{e→a}, {bd→a}, {a→b}, {e→b}, {f→b}}
b	{a, d, e, f, g}	3	{{ade}, {adf}, {adg}, {aef}, {aeg}, {afg}, {def}, {deg}, {dfg}, {efg}}	∅	{{a}, {e}, {f}}	{{e→a}, {bd→a}, {a→b}, {e→b}, {f→b}}
c	{d, g}	1	{{d}, {g}}	{{d},{g}}	∅	{{e→a}, {bd→a},{a→b}, {e→b}, {f→b}}
c	{d, g}	2	{{dg}}	{{dg}}	∅	{{e→a}, {bd→a}, {a→b}, {e→b}, {f→b}}

4.5 Selection of Implications

A relevant aspect in extracting implications is the possibility of to obtain all the relations existing among the attributes of a formal context. However, a large number of implications are generated, which makes it difficult for the end user to interpret them. One of the most important steps in the knowledge discovery process is to interpret the extracted information in a way that leads to a good understanding of the problem domain. In this case, there are several approaches that can be applied to selection, interpretation, and visualization of the set of implications. Examples include: evaluation measures, oracles, clusters, networks of implications, among others.

After extracting the set of proper implications, it was necessary to define a metric to evaluate such implications. The traditional metrics like support and confidence were not effective for this study, because the FCA is especially accurate and the extraction of the implications are based on logical operations on sets, which differs FCA of the traditional approaches of frequent patterns extraction. However, it is still necessary to define a measure that allows to classify the proper implications according to the proportion of objects represented by each implication.

In this article, we decided to evaluate the proper implications according to the relative frequency. The measure is like a local support and it was calculated according to the following equation: $\mathscr{F} = \frac{F_i}{F_p}$. At the equation $\mathscr{F}$ represents the relative frequency, F_i is the number of objects that respect the implication, and F_p is the number og objects that have the attribute of implication conclusion

as incidence. For example: the implication $\{C\ language\} \rightarrow \{software\ engineer\}$ represents 31 objects (Fi), among 59 objects that have the attribute *software engineer* (implication conclusion) as incidence (Fp). So, the relative frequency of this implication is 0.52. To obtain the result in percentage we multiply the $\mathscr{F}$ value by 100, it generates value of 52% for the example mentioned above.

It is important to note that the relative frequency measure was adopted, because in this case the frequency represents the number of objects represented by the implication according to its class (implication conclusion). Thus, each job title is considered as a class and the sets of skills are evaluated separately for each job position. It makes that the relative frequency our local support be more relevant to evaluate this type of implication than the traditional support measure, which has to be calculated from the complete set of proper implications. For example: the implication $\{java\ frameworks\} \rightarrow \{software\ engineer\}$ has $\mathscr{F} = 75.56\%$, and global support equals 2.47%.

5 Experiments and Result Analysis

This section shows the procedures, adopted for running the experiments, and the analyses of results obtained based in proposed FCA-based approach. The experiments and results analysis were structured according to the two types of implications sets which were defined in Sect. 4.4. The interpretation of implications types β and δ shows how the competences developed by the professionals are related to job positions and how this information can help the people who need a specific position.

5.1 Obtaining Implications Type β

The implications subset of type β have the notation $type\_\beta = \{P \rightarrow Q, S \rightarrow Q \in \mathscr{I} \mid \forall P, S, Q \subseteq M\}$ and express different competence patterns (premises) which share the same job position (conclusion).

For testing and analyzing this implication type we selected 20 positions and their respective 180 skills, among the 61 positions identified in Sect. 4.3. In this case, the *PropIm* algorithm extracted 895 proper implications related to this reduced formal context. Figure 6, shows these proper implications as a graph representation (proper implications network). The central nodes are the positions and the edges represents the implications between premises (set of skills) and their conclusion (position). In this study case, the graph representation helps us to analyze the distribution among sets of skills and their respective positions.

The central nodes density represents the diversification of minimum sets of skills. The denser nodes represents positions which have more diversification of minimum sets of skills. For example, the highlighted node AD related to *administrative director* position have 163 minimal sets of skills. Generally, the *administrative director* function is manage the company resources, like human, technologies and financial resources. The specific skills of this professional can be different according to the company industry, because he have to develop

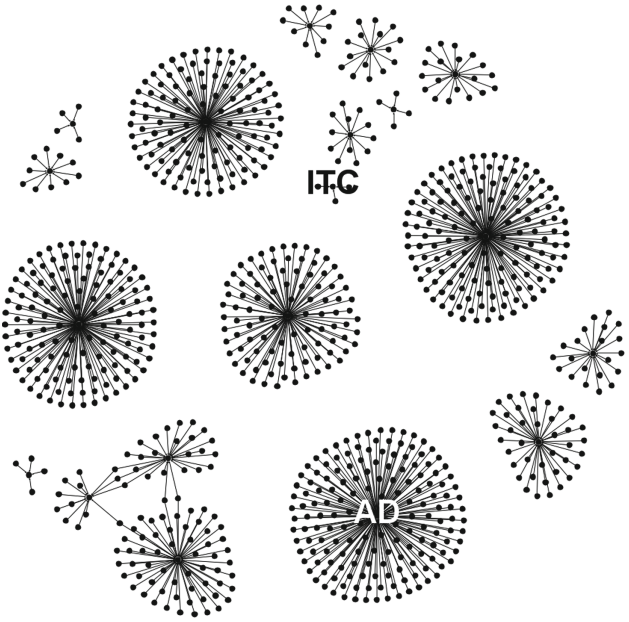

Fig. 6. Proper implications network (Source [12] p. 130).

business skills and know how about the company resources. It is expected that *administrative director* develop skills related to leadership, management, technology and communication. One of the proper implications which represents this professional profile is {*entrepreneurship, human resources, information management*} → {*administrative director*}. Another implication as {*assets management, it governance, leadership development, software development*} → {*administrative director*}, can represent a specific *administrative director* from companies focused on software development.

The central nodes with lower degree of incidence represent jobs positions that demand more specific sets of skills. This also indicates that there is not much variation among the requirements of the companies. For example the *ITC (IT consultant)* node have only 3 sets of skills related to it. An *IT consultant* duties can vary depending on the nature of company's project and client desires. However, in general, this professional has skills which combine IT and business knowledge. So, the proper implication {*ABAP*[3], *agile methodology, BI*[4]} → {*it consultant*} shows the common set of skills that the IT consultants have.

[3] ABAP: Advanced Business Application Programming.
[4] BI: Business Intelligence.

5.2 Obtaining Implications Type δ

The implications type δ have the notation $type\_\delta = \{P \rightarrow Q,\ P \rightarrow R \in \mathscr{I} \mid Q \neq R; \forall\ P, S, Q, R \subseteq M\}$ and represent competence patterns which the same set of skills (premise) result in two or more professional positions (conclusion). This characteristic can also be called as intersection among minimum sets of skills.

According to *Career Cast* research [30], the top 3 best jobs in *Information Technology* area is: data scientist, information security analyst and software engineer. From the set of proper implications, generated by *PropIm* algorithm, we filtered the *top 3* jobs positions, for analyze these jobs and identify the intersection among their skills.

Figure 7 shows the top 3 job positions and the intersections among their skills. The central nodes are the top 3 positions, according to *Career Cast* ranking [30]: P_1 (*data scientist*), P_2 (*information security analyst*) and P_3 (*software engineer*). The edges weight are the implication relative frequency. For example: the implication $\{C\ language\} \rightarrow \{software\ engineer\}$ represents 31 objects (Fi) among 59 objects that have *software engineer* as incidence (Fp). So, this implication relative frequency is 0.52. For obtain the result as percentage is only multiply by 100, generating the relative frequency percentage of 52% to this proper implication in *software engineer* set of implications. Therefore, the thicker edges represents implications with greater relative frequency. It is important to note that, we applied the relative frequency measure, because in this case, the frequency represents the significance of an implication inside its class. And the local significance is more relevant than the implication support in the complete proper implications set. For example, the implication $\{java\ frameworks\} \rightarrow \{software\ engineer\}$ has relative frequency of 75.56%, and its support is 2.47%. In this example, to analyze the relative frequency is more important than the implication support, because the specific objective is identify the conditions to reach a *software engineer* job. Is important to note that, in both cases the confidence is 100%.

Figure 7 shows the intersections between the minimal sets of skills, considering the top 3 jobs positions described above. In this case, the nodes P_1 and P_3 share four set of skills like $\{BI\} \rightarrow \{security\ analyst\}$ and $\{BI\} \rightarrow \{software\ engineer\}$. The nodes P_1 and P_2 share two set of skills, like the proper implications show: $\{agile\ methodology\} \rightarrow \{data\ scientist\}$ and $\{agile\ methodology\} \rightarrow \{information\ security\ analyst\}$. And, the nodes P_1 and P_3 share only one set of skills, on $\{active\ directory\}$[5] $\rightarrow \{data\ scientist\}$ and $\{active\ directory\} \rightarrow \{software\ engineer\}$.

From these intersections, we observed that the greater the intersections amount between skills sets, more similar are the requirements to achieve a position. It would indicated possibilities to professional mobility among positions, when the set of skills (premises) implies in several different positions (conclusions). So a professional could have competence to assume different positions,

[5] Active directory: Microsoft tool kit for store and control information about network configurations.

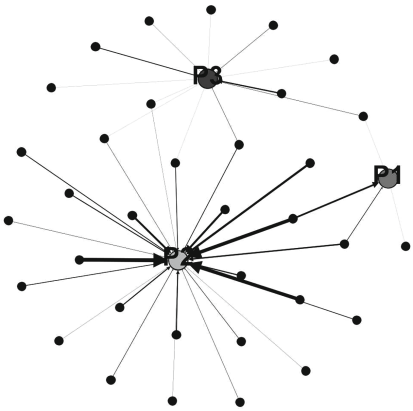

Fig. 7. Top 3 jobs and their skills, where P_1 is *data scientist*, P_2 is *information security analyst* and P_3 is *software engineer* job position (Source: [12] p. 131).

because his skills could be applied to different jobs. For example, in recruitment and selection hiring process, this professional could be compatible with several job vacancy, therefore he could be more jobs opportunities. Another example is the case when a professional needs change jobs, his skills allow greater career mobility.

Figure 8 shows 4 positions that represents different hierarchical levels of IT career. The central nodes represent these 4 positions: P_1 (IT analyst), P_2 (IT coordinator), P_3 (IT manager) and P_4 (IT director). The other nodes represent minimal sets of skills, and edges represent implications. It is important to note that, edges weight was calculated using the relative frequency, previously described. From the figure we could observe that there are disjoint sets and there are not any intersections among positions. According to this hierarchy, P_1 and P_2 are positions related to the early career, while P_3 and P_4 are positions hierarchically superior. So, for P_1 was expected technical skills like in the proper implication $\{.NET,\ automation\ systems\} \rightarrow \{IT\ analyst\}$. P_2 involves skills that represent the transition between technical and managerial level, like in the proper implication $\{.NET,\ data\ base,\ ERP,\ it\ governance\} \rightarrow IT\ coordinator$. P_3 also involves skills related to hierarchical transition, but it was expected more managerial than technical skills, it could be expressed by the implication $\{BPM$[6], *cloud computing,* CRM[7]$\} \rightarrow IT\ manager$. Finally, an IT director (P_4) have to develop managerial skills like was identified in implication $\{assets\ management,\ BI,\ business\ management,\ consulting\} \rightarrow IT\ director$. Then, for the professional get a career advancement, he have to develop skills of different natures.

[6] BPM: Business process management.
[7] CRM: Customer relationship management.

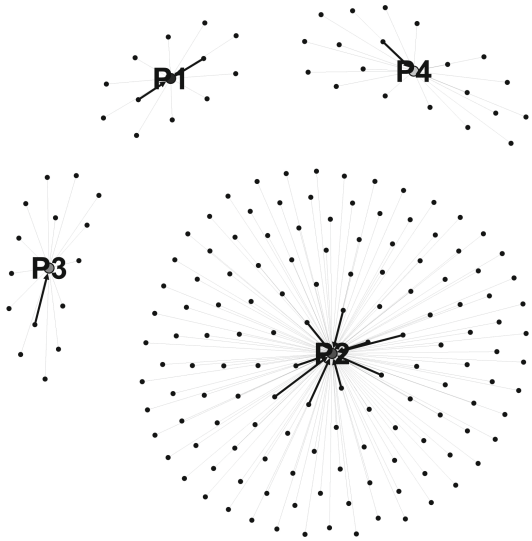

Fig. 8. IT career hierarchical levels, where P_1 to P_4 represents the following job positions: P_1 is *IT analyst*, P_2 is *IT manager*, P_3 is *IT coordinator* and P_4 is *IT director* (Source [12] p. 131).

6 Conclusion and Future Works

In this work our FCA-based approach was presented with the objective of identifying professional competencies. Specifically, one's own implications have been applied to identify the minimum sets of skills that are necessary to achieve a position. In this case, in the first place, the problem domain model was constructed in order to identify the variables that characterize a person as a professional, according to model of competence. Data were extracted from users of *LinkedIn* and techniques of data processing and transformation were applied to the formal context. Then, the *PropIm* algorithm was applied to extract the set of proper implications from formal context. Finally, graphs were constructed based on their own implications, and the relationships between premises and their respective conclusions were analyzed.

The main contributions of this article were:

- a FCA-based approach to obtain the set of proper implications, capable of representing the minimum conditions that imply in specific objectives;
- a relational database of profiles extracted from *LinkedIn*;
- two types of subsets of proper implications were defined and analyzed. These subsets of implications represent the diversity of competences required by the labor market ($type\_\beta$), and the shared competences by different positions ($type\_\delta$).

As part of FCA-based approach, we propose the *PropIm* algorithm. The goal of *PropIm* algorithm is extract the set of proper implications. It was implemented

applying pruning heuristics and scalable strategy. In future works the algorithm will be modified to run as a distributed application. The problem's order complexity to extract proper implications from formal context is $O(|M||\mathscr{I}|(|G||M| + |\mathscr{I}||M|))$. The proposed algorithm has exponential complexity, but the implemented strategy reduce the computational effort computing only implications with support greater than zero, and the pruning heuristic reduce the possibilities of attributes combinations into premises. Moreover, each conclusion from formal context can be processed separately without causing loss of information in the final set of proper implications.

Regarding the context of the work, the experiments answered the following questions:

- How to identify and represent professional competence?
- What are the relationships between competences and positions?

For the first question, the experiments the experiments allowed professional skills to be identified through proper implications. These implications represent the minimum set of characteristics that a professional could increase, when his goal is to achieve a job position. It is important to note that, when adopting the technique of minimum sets, the interest is to find the smallest set of objects that attend some condition. The number of objects for each condition, defines the relevance of the implication itself, by calculating the relative frequency, which was also called local support. By means of this calculation, the conditions could be classified as primary (the highest support, therefore most relevant), secondary (randomness) and specific (particularities).

For the second question, the relationships between competences and positions were defined by the subset of proper implications types (β and δ). These relationships were represented through a directed graph, which the nodes are sets of attributes and the edges represent the implications. The attribute sets are divided into two types, in which the source node is the premise and the destination node is the conclusion. It is important to note that we were only used graphs as an alternative representation, since they facilitate the visualization and analysis of the types of implications mentioned above. In general, the analyzes were based on the calculation of the degree of input of the nodes of the conclusion type (central nodes), and the degree of output of the nodes of the premise type. Through the in-degree, we identified the subset of proper implications named β, which represent the diversification of professional profiles required by the job market. Through the out-degree of nodes that represent the premises we could identify the intersections (implications type δ) which were shared among the skills sets and positions. These shared sets shows that the same skill set is a requirement for different positions, it allows the professional to be a potential candidate for different job opportunities. This also allows for greater job mobility if the person is looking for re-enter at the job market.

Despite the positive results obtained with the present work, it is also important to discuss the limitations of the proposed approach, ranging from problem modeling to the selection of implications. Such limitations and possible ways of solving them were discussed below.

As model of problem domain was built by mapping the sections of the *LinkedIn* for the *model of competence* [10], it is limited to the data provided by this social network, and by the interpretation given to the sections during the construction process. In future works, if there is any change in the *LinkedIn* sections, it will be necessary to adapt the model according to the platform changes. We also intend expand the experiments for all dimensions from the professional model of competence.

The data extraction process was performed from two initial seeds and the criterion of collector's stopping was based on the percentage of return of exclusive profiles. This percentage was chosen empirically, as well as the number of profiles used in the work, because it was necessary to define a set of data to analyze and extract the results. It is still necessary to improve the data extraction process in order to achieve, if possible, all profiles that belong to a problem scope.

At the data processing stage, the biggest limitation is that there is dependence on the domain expert. Therefore, such processes are susceptible to different interpretations, which would lead to different forms of attributes reduction, which could have an impact on results. In the future works, statistical or computational methods will be adopted to reduce the expert's dependence on the domain.

The algorithm was implemented sequentially and uses a traditional approach to the attributes combination. This causes an impact on the processing time of proper implications, which limits the amount of objects and attributes inserted in the formal context. As future work, another algorithms could be exploited, particularly those capable of obtaining the set of implications from concept or the subset of formal concepts as proposed by [31]. Moreover, we intend to implement the *PropIm* algorithm as a distributed application and compare several algorithms to extract proper implications from formal context. We also intent improve the selection of implications, and another measures will be tested and analyzed.

Finally, we intend to implement a web platform with this approach, to help professionals to increase their competencies in their professional resume and define career plans. In this platform the person can indicate the positions of interest and it will be returned the necessary competences to reach such position. It will also include a temporal analysis of career development, which may allow such plans to be traced in a dynamic approach.

Acknowledgement. The authors acknowledge the financial support received from the Foundation for Research Support of Minas Gerais state, FAPEMIG; the National Council for Scientific and Technological Development, CNPq; Coordination for the Improvement of Higher Education Personnel, CAPES. We would also express gratitude to the Federal Service of Data Processing, SERPRO.

References

1. Russell, M.A.: Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More. O'Reilly Media Inc., Russell (2013)
2. LinkedIn: About LinkedIn (2017). Accessed 16 Aug 2017. https://press.linkedin.com/about-linkedin
3. Ganter, B., Stumme, G., Wille, R.: Formal Concept Analysis: Foundations and Applications, vol. 3626. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-31881-1
4. Codocedo, V., Baixeries, J., Kaytoue, M., Napoli, A.: Contributions to the formalization of order-like dependencies using FCA. In: Proceedings of the 5th International Workshop What can FCA do for Artificial Intelligence, CEUR-WS (2016)
5. Taouil, R., Bastide, Y.: Computing proper implications. In: Proceedings of the International Conference on Conceptual Structures - ICCS, Stanford, pp. 46–61 (2001)
6. Kontopoulos, E., Berberidis, C., Dergiades, T., Bassiliades, N.: Ontology-based sentiment analysis of Twitter posts. Expert Syst. Appl. **40**, 4065–4074 (2013)
7. Ali, S.S., Bentayeb, F., Missaoui, R., Boussaid, O.: An efficient method for community detection based on formal concept analysis. In: Andreasen, T., Christiansen, H., Cubero, J.-C., Raś, Z.W. (eds.) ISMIS 2014. LNCS (LNAI), vol. 8502, pp. 61–72. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08326-1_7
8. Cuvelier, E., Aufaure, M.-A.: A buzz and e-reputation monitoring tool for twitter based on Galois Lattices. In: Andrews, S., Polovina, S., Hill, R., Akhgar, B. (eds.) ICCS 2011. LNCS (LNAI), vol. 6828, pp. 91–103. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-22688-5_7
9. Neto, S.M., Song, M., Dias, S., et al.: Minimal cover of implication rules to represent two mode networks. In: IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), vol. 1, pp. 211–218. IEEE (2015)
10. Durand, T.: Forms of incompetence. In: Proceedings Fourth International Conference on Competence-Based Management. Norwegian School of Management, Oslo (1998)
11. Brandão, H.P., Guimarães, T.A.: Gestão de competências e gestão de desempenho: tecnologias distintas ou instrumentos de um mesmo construto? Revista de Administração de empresas **41**, 8–15 (2001)
12. Silva, P., Dias, S., Brandão, W., Song, M., Zárate, L.: Formal concept analysis applied to professional social networks analysis. In: Proceedings of the 19th International Conference on Enterprise Information Systems, vol. 1, pp. 123–134. INSTICC, ScitePress (2017)
13. Ganter, B., Wille, R.: Formal Concept Analysis: Mathematical Foundations. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-59830-2
14. Bertet, K., Monjardet, B.: The multiple facets of the canonical direct unit implicational basis. Theor. Comput. Sci. **411**, 2155–2166 (2010)
15. Rome, J.E., Haralick, R.M.: Towards a formal concept analysis approach to exploring communities on the world wide web. In: Ganter, B., Godin, R. (eds.) ICFCA 2005. LNCS (LNAI), vol. 3403, pp. 33–48. Springer, Heidelberg (2005). https://doi.org/10.1007/978-3-540-32262-7_3
16. Snasel, V., Horak, Z., Kocibova, J., Abraham, A.: Analyzing social networks using FCA: complexity aspects. In: IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technologies, WI-IAT 2009, vol. 3, pp. 38–41 (2009)

17. Stattner, E., Collard, M.: Social-based conceptual links: conceptual analysis applied to social networks. In: IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 25–29 (2012)
18. Pedrycz, W., Chen, S.-M. (eds.): Social Networks: A Framework of Computational Intelligence, vol. 526. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-02993-1
19. Krajči, S.: Social network and formal concept analysis. In: Pedrycz, W., Chen, S.-M. (eds.) Social Networks: A Framework of Computational Intelligence. SCI, vol. 526, pp. 41–61. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-02993-1_3
20. Atzmueller, M.: Subgroup and community analytics on attributed graphs. In: Proceedings of the Workshop on Social Network Analysis using Formal Concept Analysis (2015)
21. Neznanov, A., Parinov, A.: Analyzing social networks services using formal concept analysis research toolbox. In: Proceedings of the Workshop on Social Network Analysis using Formal Concept Analysis (2015)
22. Soldano, H., Santini, G., Bouthinon, D.: Abstract and local concepts in attributed networks. In: Proceedings of the Workshop on Social Network Analysis using Formal Concept Analysis (2015)
23. Cordero, P., Enciso, M., Mora, A., Ojeda-Aciego, M., Rossi, C.: Knowledge discovery in social networks by using a logic-based treatment of implications. Knowl. Based Syst. **87**, 16–25 (2015)
24. Neto, S.M., Song, M.A., Dias, S.M., Zárate, L.E.: Using implications from FCA to represent a two mode network data. Int. J. Softw. Eng. Knowl. Eng. (IJSEKE) **1**, 211–218 (2015)
25. Jota Resende, G., De Moraes, N.R., Dias, S.M., Marques Neto, H.T., Zarate, L.E.: Canonical computational models based on formal concept analysis for social network analysis and representation. In: IEEE International Conference on Web Services (ICWS), pp. 717–720. IEEE (2015)
26. Barysheva, A., Golubtsova, A., Yavorskiy, R.: Profiling less active users in online communities. In: Proceedings of the Workshop on Social Network Analysis using Formal Concept Analysis (2015)
27. Li, L., Zheng, G., Peltsverger, S., Zhang, C.: Career trajectory analysis of information technology alumni: a LinkedIn perspective. In: Proceedings of the 17th Annual Conference on Information Technology Education. SIGITE 2016, New York, pp. 2–6. ACM (2016)
28. Xu, Y., Li, Z., Gupta, A., Bugdayci, A., Bhasin, A.: Modeling professional similarity by mining professional career trajectories. In: Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1945–1954. ACM (2014)
29. Lorenzo, E.R., Cordero, P., Enciso, M., Missaoui, R., Mora, A.: CAISL: simplification logic for conditional attribute implications. In: CLA (2016)
30. CareerCast: Jobs Rated Report 2016: Ranking 200 Jobs (2016). Accessed 12 Dec 2016. http://www.careercast.com/jobs-rated/jobs-rated-report-2016-ranking-200-jobs
31. Dias, S.M.: Redução de Reticulados Conceituais (Concept Lattice Reduction). Ph.D. thesis, Department of Computer Science of Federal University of Minas Gerais (UFMG), Belo Horizonte, Minas Gerais, Brazil (2016). (in Portuguese)