

Fair(er) and (almost) serene committee meetings with Logical and Formal Concept Analysis

Mireille Ducassé¹ and Sébastien Ferré²

¹ IRISA/INSA, Campus Universitaire de Beaulieu, 35042 Rennes Cedex, France

² IRISA/Univ. Rennes 1, Campus Universitaire de Beaulieu, 35042 Rennes Cedex,
France. Mireille.Ducasse@irisa.fr, Sebastien.Ferre@irisa.fr

Abstract. In academia, many decisions are taken in committee, for example to hire people or to allocate resources. Genuine people often leave such meetings quite frustrated. Indeed, it is intrinsically hard to make multi-criteria decisions, selection criteria are hard to express and the global picture is too large for participants to embrace it fully. In this article, we describe a recruiting process where logical concept analysis and formal concept analysis are used to address the above problems. We do not pretend to totally eliminate the arbitrary side of the decision. We claim, however, that, thanks to concept analysis, genuine people have the possibility to 1) be fair with the candidates, 2) make a decision adapted to the circumstances, 3) smoothly express the rationales of decisions, 4) be consistent in their judgements during the whole meeting, 5) vote (or be arbitrary) only when all possibilities for consensus have been exhausted, and 6) make sure that the result, in general a total order, is consistent with the partial orders resulting from the multiple criteria.

1 Introduction

There are numerous situations in academic life where decisions are taken in committee, for example to hire people or to allocate resources. The problem is to put a total order in partially ordered sets. For example, the applicants for a job have different qualities that are not necessarily comparable. Assume that a committee has to decide between two persons, if one is systematically better than the other one for all the criteria, the decision is easy to take. In general, however, the candidates are numerous (more than 100 in some cases), and some are the best with respect to some criteria and only average with respect to other criteria.

The final decisions of such committee meetings are necessarily arbitrary, at least partially. While some people may enjoy the opportunity to intrigue, our experience is that most participants have a genuine approach and try to be as honest as possible. This article is dedicated to such honest participants who want the process to be as rational as possible.

We conjecture that the frustrations felt by genuine people come mainly from the fact that the selection criteria are hard to express and that the global picture is too large for participants to embrace it fully.

In this article, we propose a decision process where Logical Concept Analysis (LCA) [4] and Formal Concept Analysis (FCA) [5] are used to address the above two problems. We do not pretend to totally eliminate the arbitrary side of the decision. After all, a committee is in general set up when it has been recognised that there is no obvious best solution. The role of the committee is therefore to make a decision and collectively take the responsibilities for it. We claim, however, that, with our approach, genuine people have the possibility to : 1) be fair with the candidates, 2) make a decision adapted to the circumstances, 3) smoothly express the rationales of a decision, 4) be consistent in their judgements during the whole meeting, 5) vote (or be arbitrary) only when all possibilities for consensus have been exhausted, and 6) make sure that the result, in general a total order, is consistent with the partial orders resulting from the multiple criteria.

In the following we illustrate our approach with an example which reconstitutes a committee meeting which had to choose among 43 job applicants. At the original meeting, the only tool which had been used was a spreadsheet. The actual arguments which had been put forward during the discussions are explicated *a posteriori* in this article.

Two tools are used, Camelis and Conexp. Camelis³[3] is a concept-based information system. It relies on concept analysis to support the organizing and browsing of a collection of objects. One specificity w.r.t other (pure) FCA based systems [7, 1, 2] is the use of logics to represent and reason on object descriptions, queries and navigation links. This allows typed attributes to be used, for instance, date intervals, string patterns, and Boolean connectives *and*, *or*, *not*. Conexp⁴, developed by Sergey A. Yevtushenko, enables, among other features, to edit a Formal Concept Analysis context and to display concept lattices.

The example illustrates how two formal concept analysis tools can help alleviate frustrations and explain a decision. We show that the taken decision has a rationale behind it and argue that had the tools been used the discussions would have been much more serene. Regarding FCA and LCA, this case study also shows that both local navigation, such as advocated by Camelis, and global formal concept lattices are needed.

2 Running example

The example which is used throughout the article reconstitutes a committee meeting which had to produce a sorted list of five candidates in order to fulfil a two-year position⁵. The application was open either to PhD or to PhD students about to defend. There were 43 candidates.

In the French academic system, before hiring people, reports must be written. At the computer science department of the INSA of Rennes, besides a qualitative

³ <http://www.irisa.fr/LIS/ferre/camelis>

⁴ <http://conexp.sourceforge.net/>

⁵ called “Attaché Temporaire d’Enseignement et de Recherche”

free style report, referees also fill in a spreadsheet file where a number of objective criteria are assessed. This is somehow a many valued context.

For this article we have straightforwardly used the actual spreadsheet file used in the recollected meeting as a formal context for Camelis. We have only replaced the names of the candidates (resp. the names of the research teams) by three (resp two) letter codes. The attributes are : the number of publications in international and national conferences as well as in journals, the location of the thesis, the expected date of the end of the thesis, whether the candidates have a computer science education, whether they have teaching experience, whether they have practical (programming) experience, whether there is a pedagogical project in the application file, whether they could integrate a research team of the laboratory. Two attributes, “bonus” and “malus”, were meant to capture information which had not been anticipated.

3 Advocated decision process

The decision process that we advocate has three stages : firstly an analysis of the context driven by the attributes/criteria which eliminates obviously out of scope candidates, secondly an analysis of the context driven by the candidates, thirdly a discussion to make partial orders into a total order, this includes votes.

Indeed, it is not tractable to examine all the attributes of each candidate in a detailed way. This would require at least 5mn per candidate. With more than 40 candidates, doing this for all candidates means several hours of analysis, people are not ready to do that for candidates who are obviously out of scope. Starting by an analysis driven by the attributes helps to speed up the process in a fair way and to spend time on valuable candidates.

3.1 Context analysis driven by the attributes

In the first stage the analysis of the context is driven by the attributes. The attributes are investigated in turn. For each one the committee decides how relevant the attribute is for this particular decision. In particular it is decided whether a given attribute is

selective: the committee decides that it is mandatory. The candidates who do not fulfil them are eliminated.

selective but counterbalanced: the committee decides that, in the absolute, the attribute would be mandatory, but in this context another attribute exhibited by some candidates could counterbalance the lack of this attribute. The counterbalancing attributes are specified. The candidates who do not fulfil either the selective or the counterbalancing attribute are eliminated.

relevant: the committee decides that the attribute is relevant but not mandatory. It is kept to later differentiate the candidates.

irrelevant: the committee decides that the attribute is not relevant for this particular decision.

Furthermore, new attributes may be identified, and the context is subsequently updated on the fly. Attributes that contain interesting information but which are not totally accurate can be restated.

In our example, this first stage is done under Camelis. A snapshot taken during the attribute driven stage is given Figure 1. The upper area contains the list of buttons and menus which are not detailed here.

The query area contains the selection criteria that the committee has specified so far. Namely on the figure this can be read as *the selected candidates must have a Computer science education AND (they should have an experience abroad OR a teaching experience) AND their integration in the research laboratory should be granted AND they should have at least a publication in an international conference.*

The bottom right-hand side window is the “object” window. It contains the name codes of the candidates who fulfil the criteria of the query area. One can see that, out of the initial 43 candidates, only 18 are left with the above query.

The bottom left-hand side window contains the taxonomy of all the attributes. The number on the left tells how many of the selected candidates have the attribute. One can see, for example, that only 11 candidates, out of the remaining 18, have a bonus. Starting from the top of the window, here is the information contained in the attribute window. One can read that the number of publications in journals, international conferences and national conferences are numbers (they can match “0.00e3”).

The committee has decided that having a “teaching experience” was a selective but counterbalanced attribute. Indeed, while everybody agreed that it is a very important criterion, somebody pointed out that it would not be fair that one of the candidates was eliminated because he had not enough teaching experience. He was doing a PhD partly in the USA and has not been able to teach. The group decided that having an “experience abroad” is very interesting for the department and that it can counterbalance not enough teaching experience. This is the echo of the second line of the query area. At that moment of the meeting, the context did not yet contain the information about an experience abroad. It had been easily updated on the fly. All candidates with an experience abroad were identified by their respective referee. A new attribute was added and associated to them.

The “date of the end of the thesis” has not been considered yet. The attribute is therefore not yet sorted. This also applies to whether candidates have “practical experience”, and to which “research team” the candidates might join. “Other attributes” is folded, it contains attributes that the committee has already assessed as irrelevant for this decision. The committee has decided that “Bonus”, “Malus”, “other publications” and the presence of a “pedagogical project” in the file were interesting properties but that they should not yet be used for the selection. The four attributes have therefore been put under “Relevant attributes”. The “Computer science education” and the “research integration” have been used in the query. The two attributes have therefore been put under “Selective attributes”.

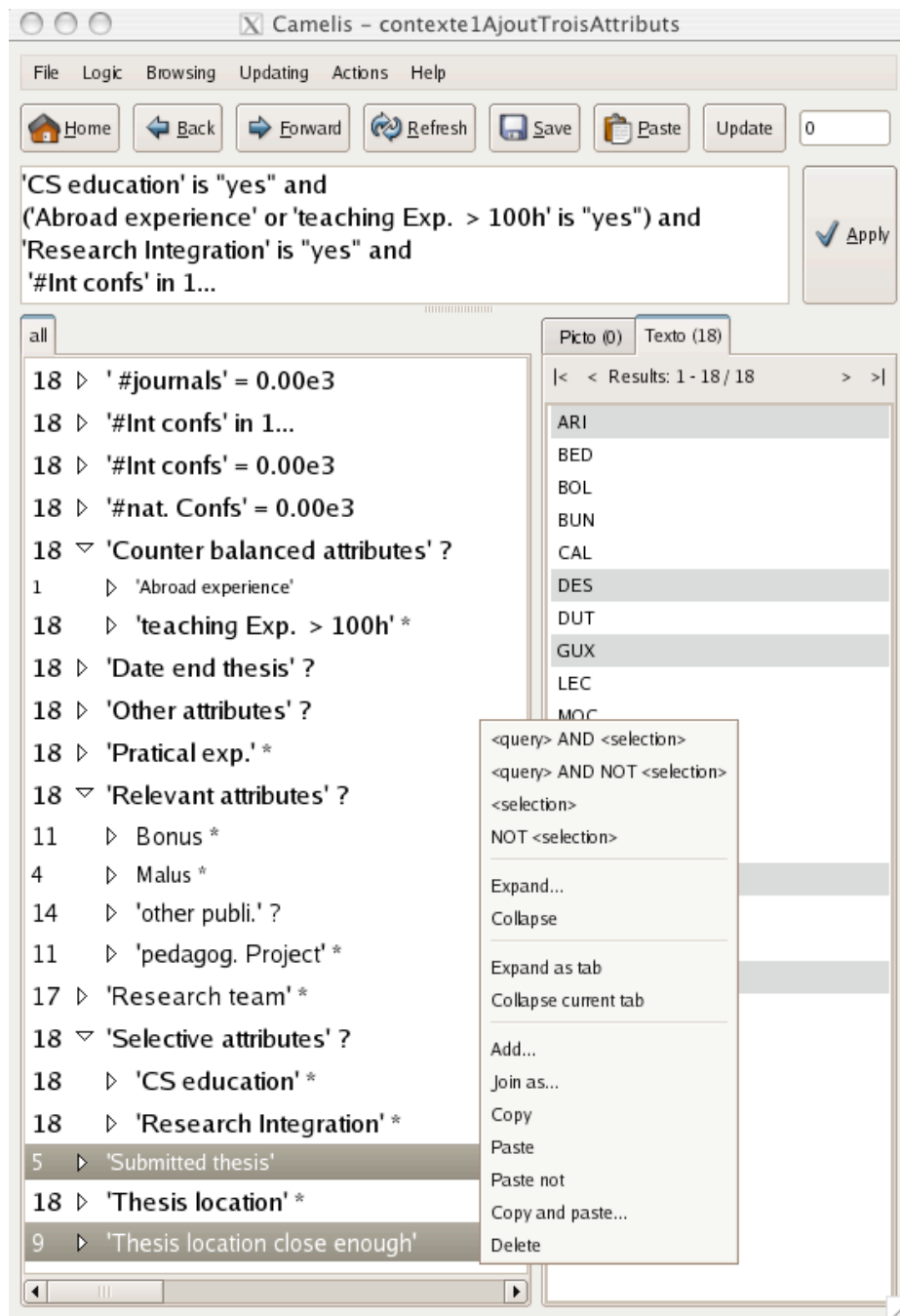


Fig. 1. Snapshot of Camelis during the attribute driven context analysis

The snapshot has been taken when the committee had just realized that “date end thesis” and “thesis location” could not be used as such. It had been decided that the candidates should either have submitted their thesis or that their thesis location should be close enough to Rennes to give them a better chance to complete their thesis. Therefore, instead of the precise thesis location, it is more accurate to know whether the thesis is done in a laboratory close enough to Rennes. With Camelis, it is easy to fix that: select all the thesis locations that are close enough to Rennes, then select the candidates associated with these locations and add a new attribute to those candidates. Similarly, instead of the estimated date of the PhD defence, it is better to know whether the thesis is already submitted. The context had therefore been extended on the fly by two new attributes, “Submitted thesis” and “Thesis location close enough”, as well as their associations to candidates.

The figure illustrates how the two new attributes will be taken into account in the query. Namely, the committee would like to select only candidates with a “submitted thesis” or a “thesis location close enough” to Rennes. The two attributes have been clicked. Camelis has greyed them. It has also greyed the candidates not fulfilling one or the other. The committee can therefore see who is going to be eliminated if the disjunction of the two attributes is judged selective. “ARI”, “DES”, “GUX” and two other candidates hidden by the pop-up window might disappear. Each referee has a chance to tell if a candidate that he considers valuable might be lost. At the actual meeting, the committee had decided that the selection was fine. A right-click opens the pop-up window. The user is about to click on “<query> AND <selection>” which will add “AND (Submitted thesis’ OR ’Thesis location close enough’)” to the query.

3.2 Context analysis driven by the objects/candidates

At some point, the number of remaining candidates becomes small enough so that it becomes tractable to examine candidates in a detailed way. The committee analyzes all the attributes of each candidate in turn. In so doing, the committee can, of course, still decide that an attribute should be “selective” or “selective but counterbalanced”.

During the first, attribute-driven, stage the committee checks that the candidates who are about to be eliminated indeed miss a required selective attribute. During the second, candidate-driven, stage the committee checks that the remaining candidates indeed have the attributes that their referees associated to them, in particular the selective ones. The committee also checks that no important attribute association is missing. It is most likely that new attributes emerge.

Figure 2 illustrates the investigation of a candidate assessment. The “CAL” candidate has been clicked in the right-hand side “object” window. His attribute values are shown, in two different ways, in the left-hand side attribute window as well as in the query window. The committee has just detected that the research team associated to this candidate is not “ic+tx” but “ic”. It is about to select “Paste not” in the pop-up window to remove the attribute from the intent of

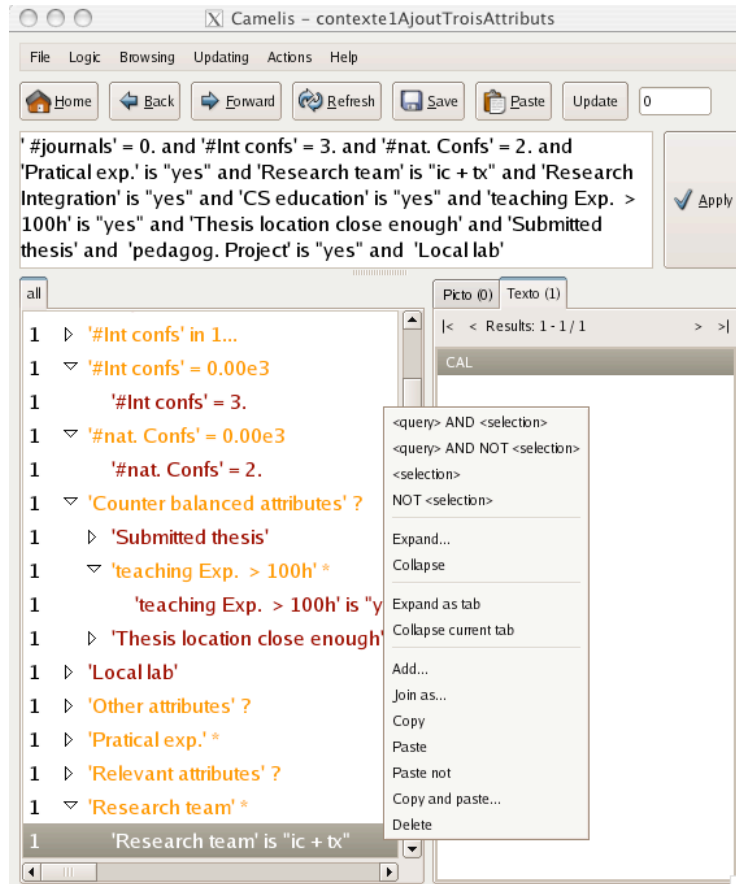


Fig. 2. Snapshot of Camelis during the candidate driven context analysis

“CAL”. The next step will then be to “Paste” candidate “CAL” to attribute “Research team is ”ic””.

In the actual session, the remaining of the second, candidate-driven, stage went as follows. Fourteen candidates were still in competition at the beginning of the second stage. While examining each candidate in turn, the committee questioned the potential integration into the research laboratory of two candidates. After discussion, it was agreed that the referees may have been a bit overoptimistic. The two candidates were said not to be easily integrated in the lab. The context was therefore updated and there were 12 candidates left.

Investigating the “malus” attribute, the committee decided that one of the candidates, currently working in the laboratory, might never complete his thesis. His long term integration into the research laboratory was therefore questioned and the related attribute negated.

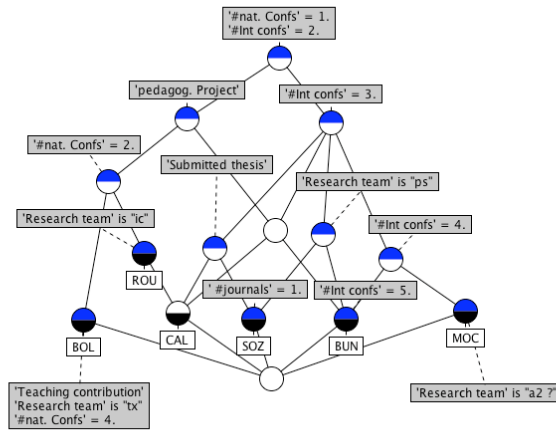


Fig. 3. The concept lattice with six remaining candidates and relevant attributes

Among the remaining candidates, somebody pointed out that one of them was having a “major contribution in teaching”. This would be interesting to keep. An attribute was added.

As there were still numerous good candidates in the list, the committee tested whether there would be enough good candidates to reinforce research teams already present at INSA. A new attribute had been introduced and six good candidates fulfil it. The committee decided that the attribute could be selective.

3.3 Making partial orders into a total order

At some point, the committee is confident that the context is valid and that the selection query is relevant for the current decision process. Furthermore, no more objective selection can be done with general consensus. We conjecture that, at this point, the formal concept lattice could be useful as it gives a global picture of all the partial orders. The task of the committee is to rank 5 candidates, hence making the partial orders into a total order.

Figure 3 shows the lattice related to the six candidates coming out of the second stage. From Camelis, we exported into Conexp a context containing the remaining 6 candidates and attributes which had been identified “relevant”. In order to see the partial orders on numerical attributes, the context had been completed. Namely a candidate exhibiting “#int. confs=3” has also be credited by “#int. confs=2” and by “#int. confs=1”. We can see that research teams “ic” and “ps” still have two candidates. From the informal information given by the two teams, the committee decides that, for “ic”, “ROU” is better than “CAL”, and it, therefore, keeps “ROU” and eliminates “CAL”. Similarly, for “ps”, “BUN” is kept and “SOZ” is eliminated. These decisions use information

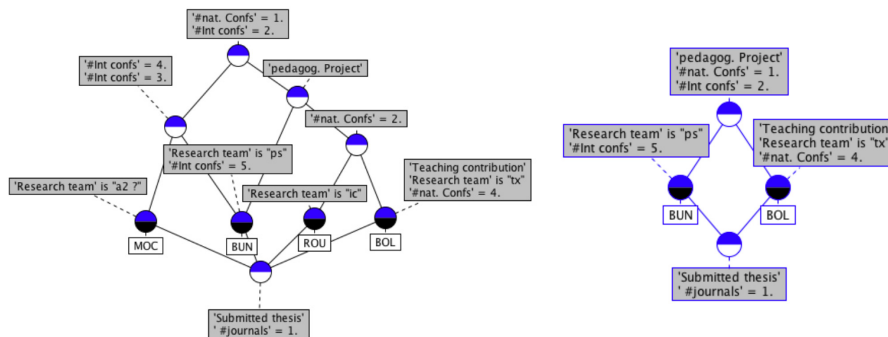


Fig. 4. The concept lattices with four remaining candidates and with the two finalists

not yet in the formal context. At this stage of the process, there are few remaining candidates, the decisions start to become arbitrary, it is not so crucial to update the context. The eliminated candidates are simply removed from the display.

Figure 4 shows the lattice related to the remaining four candidates on the left-hand side. Using, again, informal information, the committee reckoned that “MOC” and “BUN” were stronger with respect to research and that “ROU” and “BOL” were stronger with respect to teaching. Furthermore, it had been decided that “BUN” was stronger than “MOC” considering the number and the quality of the publications. It had also been decided that “BOL” was stronger than “ROU” considering the teaching contribution. This resulted in the lattice shown on the right-hand side of Figure 4. The lattice shows the attributes common to the two candidates at the top. The attributes that neither of them has are displayed at the bottom. The specific attributes are attached to each candidate. At that point, the committee can vote.

4 Discussion

In this section we discuss the benefits of our method to make decisions. We argue 1) that the process is fair to the candidates, 2) that the decision is adapted to the circumstances, 3) that committee members can be (relatively) serene, 4) that the process requires LCA/FCA tools and 5) that using a fully automatic tools would be unwise.

4.1 The process is fair to the candidates

This process is fair to the candidates. Until the last “political” stage, no candidate can be eliminated without an explicit reason and the reason is applied to all candidates. The selection criteria are explicitly specified. All candidates fulfilling a given criterion are treated equally. When the committee thinks that a candidate quality could counterbalance a required criterion, *all* candidates exhibiting this very quality will be considered equally. In general, every time the context is

updated, all candidates are concerned. Reports of candidates eliminated by the current selection are also updated. Indeed, if the committee has second thoughts and relaxes some of the selective attributes, some candidates may no longer be eliminated by the relaxed attributes. They may therefore re-appear in the selection. As their attributes have been updated, they will benefit from all the decisions that have been taken after their initial elimination.

4.2 The decision is adapted to the circumstances

The process allows the decision to be adapted to the circumstances. Indeed, even if a committee uses more or less the same criteria for different meetings, the context is every time different, even if only slightly. For example, for a given recruitment there may be very few candidates, the committee can decide either not to fulfil a position or to adapt the selective criteria. It can also happen that the set of candidates is especially strong and that for a given recruitment the selective criteria can be tightened. It can also happen that some candidates exhibit special qualities not yet identified. For all the cases, the formal context can be easily updated, the selection query can be easily refined, constraints can be easily relaxed. The approach has all the required flexibility to adapt to the situation.

4.3 Committee members can be (relatively) serene

It is always hard to take multi-criteria decisions. The discussion below takes the point of view of honest committee members who sincerely want that the decision benefits to the institution.

The process is consistent, flexible and backtrackable. The underlying concept lattice ensures that the process is consistent through out the meeting and that the result, in general a total order, is consistent with the, easy to express, partial orders. Flexibility and consistency are the basis of the fairness discussed in the above section. For example, with our approach, it cannot happen that a candidate is eliminated whereas he has an attribute that enabled another candidate to be given a second chance. Furthermore, without support it is easy to be inconsistent even genuinely. With our approach, if a criterion is said crucial at the beginning of the discussion and if the chosen candidates do not fulfil it, then at the very least it will be visible and the committee can discuss whether this is acceptable on the spot. Last but not least the result of the selection is independent of the order in which the atomic decisions are taken. As a result, every partial decision can always be questioned, the process is backtrackable. There is no need to be always on edge, no fatal decision is taken until the last minute.

The process is transparent and traceable. Some committee meetings sometimes feel like a “Three card trick” game. Our process is transparent and traceable.

The context can be easily extended on the fly with new attributes and their associations to candidates. Every member of the committee can follow the updates and can raise an objection at any time. The new context is easy to check. The selection decisions are visible in the query area. It expresses the rationale of the final decision.

The end result comes from many small and (relatively) easy decisions. Anybody can propose a new attribute or a new association, or suggest that an attribute should be selective. If the committee agrees by consensus, it is fine. If no consensus emerges the committee can vote to decide whether a new attribute is relevant or selective. It can also vote to decide whether a given candidate has an attribute. Those are small decisions, relatively easy to take. Furthermore, if the committee decides that the attribute is not selective but only relevant, it is fine that the context is anyway updated. It will be taken into account later when people will vote. It is also fine that a criterion is labeled relevant even if only one person in the committee judges it so. Consensus is only mandatory for selective criteria. This can save a lot of fruitless discussions.

Intriguing is not so easy. At such meetings, there are often people who are only there to “push” their own candidate regardless of the means. For example, we have seen situations where a candidate who had never been discussed comes out of the votes because a sub-part of the committee had plotted beforehand. With our approach the candidates who are voted upon must have been examined in depth, their attributes must have been validated. They must satisfy a number of required properties. It is not so easy to manipulate a group on small concrete decisions. It is therefore most likely that candidates who do not fulfil the required properties will have been eliminated. Sometimes it also happens that a new criterion comes out at the end of the meeting and that it gives a decisive advantage to the very candidate who is supported by the referee who expresses the criterion. As this is bad practice, genuine referees can refrain from specifying an important criterion only because it is somehow too late. With our approach if somebody wants to defend a candidate, it is fine at any time because all candidates with the same attributes will be treated equally. Last but not least, voting occurs only when all possibilities for consensus have been exhausted. At that moment, any result is fair as the whole rational is fulfilled.

4.4 LCA/FCA tools are relevant

The overall process is very hard to achieve without appropriate tools. When we started using a spreadsheet it was already a big improvement over oral or written reports even if structured. However, selecting criteria and candidates in the spreadsheet is very error prone. It is hard to be 100% sure that the process is consistent. Furthermore adding attributes and stating in the table who has them is very tedious and again error prone. Even worse, keeping track of the selection process is almost impossible, especially when it is a combination of a logical AND, OR and NOT connectors. In Camelis, everything that was so hard to do

with a spreadsheet can be done naturally. Furthermore the global consistency is guaranteed and the query keeps the history of decisions.

At the actual meeting, neither Camelis nor the lattices were available. The decisions were, nevertheless, taken mostly with respect to the criteria displayed in the previous section. People had to keep the picture in their mind and it took a lot of time. The same arguments had to be repeated again and again, every time we needed them somebody had forgotten them.

It is not straightforward to get the initial context, especially when no formal reports are required. However, our experience shows that people are more keen to write them once they understand the potential gain. We are actually considering to use Camelis to fill in the initial reports, skipping the spreadsheet altogether. Indeed, updating the associations attributes-candidates is very easy. If the attributes already exist it is just a matter of drag and drop. Adding a new attribute is also straightforward. A verbose report would be easy to generate automatically from the final context. Furthermore, the criteria resulting from a meeting could be used to initialize the next recruitment.

4.5 Using a fully automatic tools would be unwise

The context must be updated and validated. It is important to note that even with a large experience of committee sessions there is no way the formal context can be filled and not been questioned. The formal context has to be updated and validated. Firstly, it is impossible to guarantee that all the important criteria have been foreseen and anticipated. Secondly, the context depends of the set of candidates: it is possible that not enough candidates fulfil the selection criteria that have been used in previous meetings. Thirdly, some of the candidates can show interesting features that had not been previously identified. Fourthly, referees may make mistake while filling in the context. Lastly, referees make judgements that the remaining of the committee may want to question. Some people are too kind and other are too strict, it is also easy to miss an important feature in a CV. As a consequence, it is out of question that the decision is taken automatically using the context as it is at the beginning of the meeting. One of the objectives of the process described so far is, on the contrary, that the members of committee collectively agree both on a set of attributes/criteria and on who satisfy them. Namely the context is revised and updated during the process and is as much a result of the meeting as the resulting ranked list of candidates.

No magical number. Once the context is agreed upon by the committee, we still do not advocate to build a program that would compute magical numbers. Numbers have the nice feature to be naturally ordered. However, the committee has to take full responsibility for the final decision. With our process, all the attributes are identified. All the selective attributes have been agreed upon. The weights and priority among the attributes which have been labelled relevant depend of each committee member who takes them into account while voting. This makes it easier for committee members to shoulder the final decision.

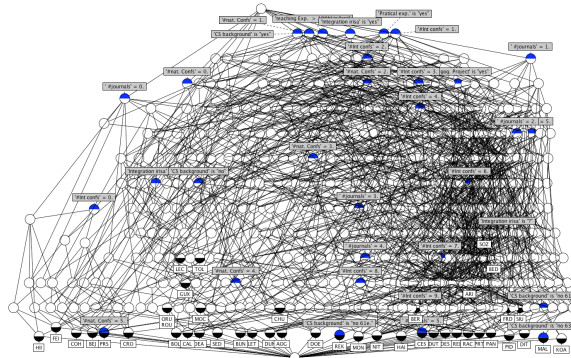


Fig. 5. Concept lattice for the 43 candidates and 40 attributes out of 62

5 Related work

Concept analysis has been applied to numerous social contexts, such as social networks [8] and computer-mediated communication [6]. Most of those applications are intended to be applied *a posteriori*, in order to get some understanding of the studied social phenomena. On the contrary, we propose to use LCA and FCA in the course and as a support of the phenomena itself. In our case, the purpose is to support a social/committee decision process. Our approach is to other social applications, what information retrieval is to data-mining. Whereas data-mining automatically computes a global and static view on *a posteriori* data, information retrieval (i.e. navigation in the concept lattice) presents the user with a local and dynamic view on *live* data, and only guides users in their choice.

A reason for not showing the global concept lattice is that it is too large to be managed by hand. Figure 5 shows the resulting concept lattice. Even in the case of our small context reduced to the 43 candidates and 40 attributes (out of 62), the number of concepts is 1239. Indeed, that formal context is dense. All candidates have many attributes, hence the large number of concepts. Local views such as proposed by Camelis or other FCA-based tools in the domain of information retrieval [2] are better suited for the first two stages described in this article than global lattices.

A specificity of Camelis is the use of logics. This has consequences both on the queries that can be expressed, i.e. on the set of candidates that can be selected, and on the attribute taxonomy, i.e. on the presentation of the criteria occurring in the selected candidates. The use of logics allows to express inequalities on numerical attributes (e.g., number of publications), disjunctions and negations in queries. In pure FCA, only conjunctions of Boolean attributes can be expressed. Previous sections have shown how disjunction is important to take into account counterbalanced selection criteria. In the taxonomy, criteria are organized according to the logical subsumption relation between them (e.g., “nb.

papers = 2” is placed under “nb. papers 1..”). In pure FCA, criteria would be presented as a long flat list. Logics helps to make the taxonomy more concise and readable by grouping and hierarchizing together similar criteria. The taxonomy can be updated dynamically, making it possible to group together all irrelevant attributes. In this way, irrelevant attributes are displayed in one line, instead of many, but they are still accessible, and can be moved back as relevant attributes.

6 Conclusion

There are numerous situations similar to the one recollected in this article. Every time a scarce resource has to be assigned by a group which wants to put rationale into its decision, our approach could be used. We have illustrated that with our approach genuine people have the possibility to smoothly express the rationales of a decision. The resulting query gives an explanation of the selection. The committee can take the responsibility of its decision. The committee can be consistent in their judgements during the whole meeting and can be fair with the candidates. It can make a decision adapted to the circumstances. Voting can be postponed to the moment when all possibilities for consensus have been exhausted. The result, in general a total order, is consistent with the expressed partial orders.

Acknowledgements The authors thank Daniel Herman for fruitful comments.

References

1. R. Cole and G. Stumme. CEM - a conceptual email manager. In G. Mineau and B. Ganter, editors, *Int. Conf. Conceptual Structures*, LNCS 1867. Springer, 2000.
2. J. Ducrou, B. Vormbrock, and P. W. Eklund. FCA-based browsing and searching of a collection of images. *Int. Conf. Conceptual Structures*, LNCS 4068. 2006.
3. S. Ferré. CAMELIS: Organizing and browsing a personal photo collection with a logical information system. In J. Diatta, P. Eklund, and M. Liquière, editors, *Int. Conf. Concept Lattices and Their Applications*, pages 112–123, 2007.
4. S. Ferré and O. Ridoux. An introduction to logical information systems. *Information Processing & Management*, 40(3):383–419, 2004.
5. Bernhard Ganter and Rudolf Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, Heidelberg, 1999.
6. N. Hara. Analysis of computer-mediated communication: Using formal concept analysis as a visualizing methodology. *Journal of Educational Computing Research*, 26(1):25–49, 2002.
7. C. Lindig. Concept-based component retrieval. In *IJCAI95 Workshop on Formal Approaches to the Reuse of Plans, Proofs, and Programs*, 1995.
8. Camille Roth. Binding social and semantic networks. In *European Conf. on Complex Systems*. ECCS, <http://ecss.csregistry.org/>, September 2006.