On succinct representation of knowledge community taxonomies with formal concept analysis

Camille Roth*, Sergei Obiedkov†, and Derrick G. Kourie†

Electronic version of an article published as
International Journal of Foundations of Computer Science (IJFCS), Vol. 19, Issue 2, 2008, 383–404
DOI: 10.1142/S0129054108005735
© *copyright World Scientific Publishing Company*
http://www.cs.ucsb.edu/ ijfcs

Abstract

We present an application of formal concept analysis aimed at representing a meaningful structure of knowledge communities in the form of a lattice-based taxonomy. The taxonomy groups together agents (community members) who develop a set of notions. If no constraints are imposed on how it is built, a knowledge community taxonomy may become extremely complex and difficult to analyze. We consider two approaches to building a concise representation respecting the underlying structural relationships, while hiding uninteresting and/or superfluous information: a pruning strategy based on the notion of concept stability and a representational improvement based on nested line diagrams and "zooming". We illustrate the methods on two examples: a community of embryologists and a community of researchers in complex systems.

1 Introduction

A knowledge community is a group of agents who produce and exchange knowledge within a given knowledge field, achieving a widespread social cognition task in a rather decentralized, collectively interactive, and networked fashion. The study of such communities is usually a focal topic for social epistemology as well as in scientometrics and political science [27, 17, 12], *inter alia*.

In particular, a traditional concern relates to the description of the structure of knowledge communities [20] generally organized in several subcommunities. In contrast to the limited, subjective, and implicit representation that agents have of their own global community—a folk taxonomy [1]—epistemologists typically use expert-made taxonomies, which are more reliable but which still fall short in terms of precision, objectivity, and comprehensiveness.

We describe here an application of formal concept analysis (FCA) aimed at representing a meaningful structure of a given knowledge community in the form of a lattice-based taxonomy built upon groups of agents jointly manipulating some notions. Formal concepts in this case relate loosely to the sociological idea of "structural equivalence" [18], denoting groups of agents linked jointly to certain sets of terms.

This work is a development of the approach presented in [24, 25], where it was shown how to use FCA to identify the main fields in a scientific community and describe their taxonomy with several levels of detail. Sect. 2 gives an overview of the approach. In Sect. 3, we concentrate on how to make lattice-based taxonomies concise and intelligible. Concept lattices faithfully represent every trait in data, including those due to noise. Therefore, we need tools that would allow us to abstract from insignificant and noisy

*Center of Research in Social Simulation (CRESS), Department of Sociology, University of Surrey, UK *and* European Center for Living Technology, Venice, Italy, `camille.roth@polytechnique.edu`

†Department of Computer Science, University of Pretoria, Pretoria, South Africa, `sergei.obj@gmail.com` and `dkourie@cs.up.ac.za`

features. To this end, we suggest a pruning technique based on stability indices of concepts [15] and apply it on its own, as well as in combination with nested line diagrams [8] and "zoomed" diagrams. These diagrams allow for representing the community structure at various levels of precision, depending on which subcommunities are most interesting to the user of the taxonomy. The techniques described in Sect. 3 admit modifications, which are a subject for further research and experiment. Some possible directions and open questions are listed in Sect. 4.

2 A Formal Concept Analysis Approach in Applied Epistemology

2.1 Framework

Representing taxonomies of knowledge communities has usually been an issue for applied epistemology and scientometrics [17], addressed notably by describing community partitions with trees or two-dimensional maps of agents and topics. Various quantitative methods are being used, often based on categorization techniques and data describing links between authors, papers, and/or notions—such as co-citation [20], co-authorship [9], or co-occurrence data [21].

The lattice-based taxonomies discussed here allow overlapping category building, with agents possibly belonging to several communities at once, and render a finer and more accurate structure of knowledge fields by representing various kinds of interrelationships. Our notion of a community is both looser and more general than the sociological notion of structural equivalence [18] in that we identify maximal groups of agents linked jointly to various sets of notions instead of *exactly* the same notions.

A similar problem of identifying communities exists in the area of social networks. Lattices have also been used there [29, 7, 6], but in that context, groups of agents (or actors) are generally considered to be disjoint and a lattice is only a first step in their construction. Besides, social network researchers are interested in social aspects of the community structure (who the leaders are, how they influence peripheral members, how actors cooperate within their own group and between different groups, etc.), whereas we rather try to discover a structure of a scientific field (and are not particularly concerned with individuals). Because of these differences in emphasis, social network lattices are typically based on data describing interaction and relations between actors, while our data, as it will be seen later, describes actors in terms of the domain for which we want to build a taxonomy.

Before proceeding, we briefly recall the FCA terminology [8]. Given a *(formal) context* $\mathbb{K} = (G, M, I)$, where G is called a set of *objects*, M is called a set of *attributes*, and the binary relation $I \subseteq G \times M$ specifies which objects have which attributes, the derivation operators $(\cdot)^I$ are defined for $A \subseteq G$ and $B \subseteq M$ as follows:

$$A^I = \{m \in M \mid \forall g \in A : gIm\};$$
$$B^I = \{g \in G \mid \forall m \in B : gIm\}.$$

In words, A^I is the set of attributes common to all objects of A and B^I is the set of objects sharing all attributes of B.

If this does not result in ambiguity, $(\cdot)'$ is used instead of $(\cdot)^I$. The double application of $(\cdot)'$ is a closure operator, i.e., $(\cdot)''$ is extensive, idempotent, and monotonous. Therefore, sets A'' and B'' are said to be *closed*.

A *(formal) concept* of the context (G, M, I) is a pair (A, B), where $A \subseteq G$, $B \subseteq M$, $A = B'$, and $B = A'$. In this case, we also have $A = A''$ and $B = B''$. The set A is called the *extent* and B is called the *intent* of the concept (A, B).

A concept (A, B) is a *subconcept* of (C, D) if $A \subseteq C$ (equivalently, $D \subseteq B$). In this case, (C, D) is called a *superconcept* of (A, B). We write $(A, B) \leq (C, D)$ and define the relations $\geq$, $<$, and $>$ as usual. If $(A, B) < (C, D)$ and there is no (E, F) such that $(A, B) < (E, F) < (C, D)$, then (A, B) is a *lower neighbor* of (C, D) and (C, D) is an *upper neighbor* of (A, B); notation: $(A, B) \prec (C, D)$ and $(C, D) \succ (A, B)$.

The set of all concepts ordered by $\leq$ forms a lattice, which is denoted by $\underline{\mathfrak{B}}(\mathbb{K})$ and called the *concept lattice* of the context $\mathbb{K}$. The relation $\prec$ defines edges in the *covering graph* of $\underline{\mathfrak{B}}(\mathbb{K})$. The meet and join in the lattice are denoted by $\wedge$ and $\vee$, respectively.

An expression $B \to D$, where $B \subseteq M$ and $D \subseteq M$, is called an *(attribute) implication.* An implication $B \to D$ *holds* in the context (G, M, I) if all objects from G that have all attributes from B also have all attributes from D, i.e., $B' \subseteq D'$ (equivalently, $D'' \subseteq B''$). The set of all implications is summarized by the Duquenne–Guigues basis [11].

2.2 Epistemic community taxonomy

Our primary data consists of scientific papers dealing with a certain (relatively broad) topic, from which we construct a set G of authors and a set M of terms and notions used in these papers. Thus, we have a context (G, M, I), where I describes which author uses which term in one of his or her papers: gIm iff g uses m. Then, for a group of authors $A \subseteq G$, A' represents notions being used by every author $a \in A$, while, for a set of notions $B \subseteq M$, B' is the set of authors using every notion $b \in B$. Thus, we see notions as cognitive *properties* of authors who use them (skills in scientific fields).

The intent of a concept in this context is a subtopic and the extent is the set of all authors active in this subtopic. Thus, formal concepts provide a solid formalization of the notion of *epistemic community* (EC) traditionally defined as a group of agents dealing with a common set of issues and aiming towards a common goal of knowledge creation [12]. By EC, we understand henceforth a field or a subdiscipline within a given knowledge community together with authors working in this subdiscipline irrespective of their affiliation or personal interactions, i.e., neither a department nor a research project. The concept lattice represents the structure of a given knowledge community as a taxonomy of ECs, with more populated and less specific subtopics closer to the top [24].

2.3 Empirical examples and protocol

We focus on two databases. One of them is a bibliographical database of `MedLine` abstracts coming from the fast-growing community of embryologists working on the zebrafish during the period 1998–2003.[1] According to experts in the field [22, 10, 5], three major subfields are to be distinguished. First, an important part of the community focuses on biochemical signaling mechanisms, involving pathways and receptors, which are crucial in understanding embryo growth processes. A second field includes comparative studies: the zebrafish, as a model animal, may show similarities with other close vertebrate species, in particular, with mice and humans. Finally, another significant area of interest relates to the brain and the nervous system, notably in association with signaling in brain development.

From the bibliographical database, we build up a context describing which author used which notion during the whole period, where the notion set is made of a limited dictionary of about 70 lemmatized words selected by the expert [22] among the most frequent yet significant words of the community, i.e., excluding rhetorical and paradigmatic words such as "*is*", "*with*", "*study*", "*biology*", "*develop*", etc. At first, we thus should say that each term appearing in an article is a notion, which is a classical assumption in scientometrics [21, 19]. In other words, we extract the semantics from article contents rather than from their metadata. As such, scientific fields will be defined by notion sets describing EC intents. Then, we extract a random sample context of 250 agents and 18 notions, which we use to illustrate the techniques described in the paper. The concept lattice of this context consists of 1146 concepts. Taking a smaller data sample (25 authors and the same 18 notions), we obtain the concept lattice shown in Fig. 1 (only notion/attribute labels are shown); it contains 69 formal concepts or epistemic communities[2].

Our second example is of similar nature. It comes from the database of all papers submitted to the Second European Conference on Complex Systems in 2006[3]. From this data, we build a context made of authors and terms mentioned in article titles and abstracts. The resulting context contains 401 authors and 109 terms, of which we have decided to keep only terms referring to topics (22 terms such as "*dynamics*" and "*transmission*") and those somehow characterizing methods used in the study of these topics (nine terms

[1] Data is obtained from a query on article abstracts containing the term "*zebrafish*" at http://www.pubmed.com. Using a precise term is likely to delimit properly the community, in contrast to global terms such as "*molecular biology*".

[2] Diagrams are produced with ConExp (http://sourceforge.net/projects/conexp) and ToscanaJ (http://sourceforge.net/projects/toscanaj).

[3] http://complexsystems.lri.fr/Portal/tiki-index.php?page=ECCS'06

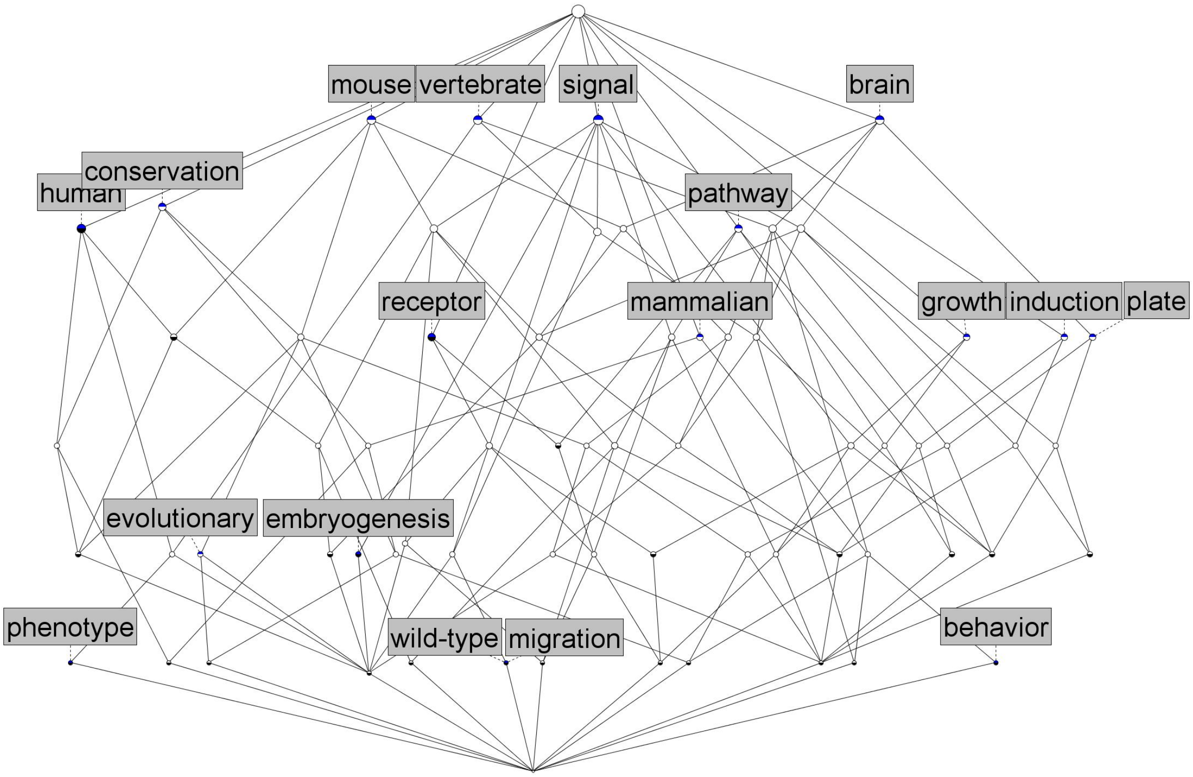

Figure 1: The concept lattice of a sample zebrafish context (with 25 authors).

such as "*interaction*" and "*physical*").[4] The concept lattice built from these terms contains 549 concepts, which is still too many for analysis and interpretation and thus calls for a more concise representation, which is the subject of the next section.

3 Concise Representation

3.1 Rationale

Even if the concept lattice appears to adequately identify and organize epistemic communities, the number of ECs is likely to be large. Although derived from a small context, the diagram in Fig. 1 is indeed rather complicated. This is a well-known risk when using concept lattices. To quote [8], "even carefully constructed line diagrams lose their readability from a certain size up, as a rule from around 50 elements up". Unfortunately, there is no hope that lattices built from real-size data will be limited to anything close to fifty elements.

Moreover, some ECs are likely to be irrelevant for the purposes of deriving a practical taxonomies of knowledge fields. One solution is to compute only an upper part of the lattice (an order filter), e.g., concepts covering at least $n\%$ of authors (in this case, we get an "iceberg lattice" [28]). Here, one should be careful not to overlook small but interesting groups, for example, groups that represent a new research trend not yet supported by a large number of followers or groups that contain individuals who are not members of

[4] Note that our selected topics are mostly *theoretical* topics and therefore do not cover specific case studies, such as "social systems", even if a sizeable part of the complex system community is concerned with these notions; by topics, we mean various properties of and processes related to complex systems, rather than particular types of complex systems. Other notions fall under more than one category and have thus been excluded (networks as topics, formal structures, or methods).

any other group. To take account of such groups, one should also compute all lower neighbors (proper subgroups) of "large" ECs (satisfying the $n\%$ threshold). Top-down lattice construction algorithms are particularly suitable for this approach [16]; alternatively, one may look at algorithms designed specifically for constructing iceberg lattices [28] and other algorithms from the frequent itemset mining community [3]. The reduction in the number of concepts can be considerable; however, though computationally feasible, this would still be unsatisfying from the standpoint of manual analysis.

Clearly, the size of the concept lattice is not only a computational problem. The lattice may contain nodes that are just too similar to each other because of noise in data or real minor differences yet irrelevant to our purposes. In this case, taking an upper part of the lattice does not solve the problem, since this part may well contain such similar nodes. Besides, it should also be interesting to distinguish major trends from minor subfields, perhaps, with a representation allowing for different levels of precision.

In this section, we consider two approaches to improve the readability of line diagrams: pruning on one side, and nesting and zooming on the other side. When pruning, we assume that some concepts are irrelevant: we filter out those that do not satisfy specified constraints of a certain kind. In a previous attempt to use concept lattices to represent EC taxonomies [24, 25], heuristics combining various criteria—such as extent size, the shortest distance from the top, the number of lower neighbors, etc.—were used to score ECs and keep only the n best ones. The resulting pictures were meaningful taxonomies, but required a posteriori manual analysis, while it was unclear whether it could be possible to go further than a rough representation. Here, we focus on a particular pruning strategy based on the notion of the stability of a concept [15].

Nested line diagrams [8], on the other hand, provide no reduction and, hence, do not incur any loss of information. Rather, they rearrange the concepts in such a way that the entire structure becomes more readable; they provide the user with a partial view, which can then be extended to a full view if so desired. Thus, nested line diagrams offer a useful technique for representing complex structures. Yet, because they preserve all details of the lattice, in order to remove (many) irrelevant details we combine nesting and pruning in Sect. 3.4. We thus try to get a representation that respects the original taxonomy while hiding at the same time uninteresting and superfluous information; our aim is a trade-off between the noise level, the number of details, and readability. In this respect, as nested line diagrams improve readability but may render the examination a more complex task, we suggest a compromise that organizes concepts while providing local diagrams for which interpretation is straightforward—we call this latter approach "zooming" and present it in Sect. 3.5.

3.2 Stability-Based Pruning

Our structures are complex, but, in fact, they are more complex than they should be, since our data is fairly noisy: for instance, an author might use a term accidentally (e.g., discussing related work), or there may be different names for the same thing (e.g., "Galois lattice" and "concept lattice"), in which case it is not obvious that people preferring one terminology should be grouped exactly under the same field as people preferring another one, but, at least, there ought to be a super-field uniting them (especially since we are interested in the taxonomy of knowledge fields rather than in social networks within academia). As a result, many concepts do not correspond to real communities, and some pruning seems unavoidable.

The pruning technique we describe here is based on the notion of stability first introduced in [14] in relation to hypotheses generated from positive and negative examples; it can be easily extended to formal concepts of a context [15]. The definition we use is slightly different from the original one, but the difference is irrelevant to our discussion.

Definition 1. *Let $\mathbb{K} = (G, M, I)$ be a formal context and (A, B) be a formal concept of $\mathbb{K}$. The* stability index*, σ, of (A, B) is defined as follows:*

$$\sigma(A, B) = \frac{|\{C \subseteq A \mid C' = B\}|}{2^{|A|}}. \tag{1}$$

The stability index of a concept indicates how much the concept intent depends on particular objects of the extent. A stable intent is probably "real" even if the description of some objects is "noisy". In application to our data, the stability index shows how likely we are to still observe a field if we ignore several authors. Apart from noise-resistance, a stable field does not collapse (e.g., merge with a different field, split into several independent subfields) when a few members stop being active or switch to another topic. The following proposition describing the stability index of a concept (A, B) as a ratio between the number of subcontexts of $\mathbb{K}$ where B is an intent and the total number of subcontexts of $\mathbb{K}$ makes the idea behind stability more explicit:

Proposition 1. *Let $\mathbb{K} = (G, M, I)$ be a formal context and (A, B) be a formal concept of $\mathbb{K}$. For a set $H \subseteq G$, let $I_H = I \cap (H \times M)$ and $\mathbb{K}_H = (H, M, I_H)$. Then,*

$$\sigma(A, B) = \frac{|\{\mathbb{K}_H \mid H \subseteq G \text{ and } B = B^{I_H I_H}\}|}{2^{|G|}}. \tag{2}$$

Proof. Every $C \subseteq A$ defines a family of contexts:

$$\mathfrak{F}_C(\mathbb{K}) = \{\mathbb{K}_H \mid C \subseteq H \subseteq G \text{ and } A \cap H = C\}. \tag{3}$$

Obviously, $\mathfrak{F}_C(\mathbb{K}) \cap \mathfrak{F}_D(\mathbb{K}) = \emptyset$ if $C \neq D$. In fact, the sets $\mathfrak{F}_C(\mathbb{K})$ form a partition of subcontexts of $\mathbb{K}$ (with the same attribute set M). It is easy to see that all sets $\mathfrak{F}_C(\mathbb{K})$ (with $C \subseteq A$) have the same size: $|\mathfrak{F}_C(\mathbb{K})| = 2^{|G|-|A|}$. Note also that, for $\mathbb{K}_H \in \mathfrak{F}_C(\mathbb{K})$, we have $B^{I_H I_H} = C^{I_H} = C'$; hence, B is closed in the context $\mathbb{K}_H \in \mathfrak{F}_C(\mathbb{K})$ if and only if $C' = B$. Therefore,

$$|\{\mathbb{K}_H \mid H \subseteq G \text{ and } B = B^{I_H I_H}\}| = \frac{2^{|G|}|\{C \subseteq A \mid C' = B\}|}{2^{|A|}}, \tag{4}$$

which proves the proposition.

In other words, the stability of a concept is the probability of preserving its intent after leaving out an arbitrary subset of objects from the context. This is the idea of cross-validation [13] carried to its extreme: stable intents are those generated by a large number of subsets of the data. In the case of cross-validation, it is more common to consider only (some) subsets of a fixed size. Indeed, one may argue that subcontexts of different sizes should have different effect on the stability: the smaller the subcontext is, the further it is from the initial—observed—context, and, hence, the smaller should be its contribution to the instability of a concept. However, we leave these matters for further research and use the definition of the stability given above.

3.2.1 Computing stability

In [15], it is shown that, given a formal context and one of its concepts, the problem of computing the stability index of this concept is #P-complete. Below, we present a simple algorithm that takes the covering graph of a concept lattice $\underline{\mathfrak{B}}(\mathbb{K})$ and computes the stability indices for every concept of the lattice. The algorithm is meant only as an illustration of a general strategy for computing the stability; therefore, we leave out any possible optimizations.

```
Algorithm ComputeStability
  Concepts := $\underline{\mathfrak{B}}(\mathbb{K})$
  for each (A, B) in Concepts
    Count[(A, B)] := the number of lower neighbors of (A, B)
    Subsets[(A, B)] := $2^{|A|}$
  end for
  while Concepts is not empty
    let (C, D) be any concept from Concepts with Count[(C, D)] = 0
    Stability[(C, D)] := Subsets[(C, D)] / $2^{|C|}$
    remove (C, D) from Concepts
    for each (A, B) $>$ (C, D)
      Subsets[(A, B)] := Subsets[(A, B)] $-$ Subsets[(C, D)]
      if (A, B) $\succ$ (C, D)
        Count[(A, B)] := Count[(A, B)] $-$ 1
      end if
    end for
  end while
  return Stability
```

To determine the stability index $\sigma(A, B)$, we compute the number of subsets $E \subseteq A$ that generate the intersection B (i.e., for which $E' = B$) and store it in `Subsets`. The index $\sigma(A, B)$ is simply the number of such subsets divided by the number of all subsets of A, that is, by $2^{|A|}$. Once computed, $\sigma(A, B)$ is stored in `Stability`, which is the output of the algorithm.

The algorithm traverses the covering graph from the bottom concept upwards. A concept is processed only after the stability indices of all its subconcepts have been computed; the `Count` variable is used to keep track of concepts that become eligible for processing. In the beginning of the algorithm, `Count[(A, B)]` is initialized to the number of lower neighbors of (A, B). When the stability index is computed for some lower neighbor of (A, B), we decrement `Count[(A, B)]`. By the time `Count[(A, B)]` reaches zero, we have computed the stability indices for all lower neighbors of (A, B) and, consequently, for all subconcepts of (A, B). Then, it is possible to determine the stability index of (A, B).

Initially, `Subsets[(A, B)]` is set to the number of all subsets of A, that is, $2^{|A|}$. Before processing (A, B), we process all subconcepts (C, D) of (A, B) and decrement `Subsets[(A, B)]` by the number of subsets of C generating the intersection D. By doing so, we actually subtract from $2^{|A|}$ the number of subsets of A which do not generate B: indeed, every subset of A generates either B or the intent of a subconcept of (A, B). Thus, the value of `Subsets[(A, B)]` eventually becomes equal to the number of subsets of A generating B.

3.2.2 Applying stability

The basic stability-based pruning method is to remove all concepts with stability below a fixed threshold. We computed the stability indices for concepts of our zebrafish example (250 authors and 18 words). Coincidentally, the 16 most stable concepts are closed under intersection of intents. Hence, they form a lattice, which is shown in Fig. 2.

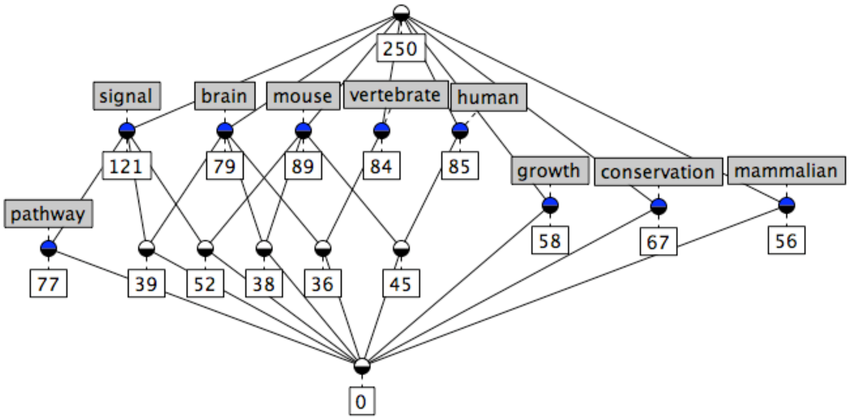

Figure 2: The lattice of the 16 most stable concepts of a context built from 250 zebrafish researchers and 18 notions they used in their papers. Here and below, attributes that are not contained in any stable intent other than the intent of the bottom concept are not shown on the diagram; numbers indicate the sizes of the corresponding extents.

Of course, stable concepts (i.e., satisfying the chosen stability threshold) do not always form a lattice. This may or may not be a problem. If all we need is a directly observable taxonomy of scientific fields, there seems to be no reason to require that this taxonomy should be a lattice. In other contexts, however, a lattice may be desired in order to apply lattice-based analysis techniques. This issue is beyond the scope of the present paper; nonetheless, we suggest some possible strategies in Sect. 4.2.

Apart from the obvious compression—we keep 16 concepts out of 1146—Fig. 2 presents a readable epistemic taxonomy representation, displaying the major fields of the community along with some meaningful joint communities (such as "*mouse*" and "*human*"). It is interesting to note, for example, that "*pathway*" occurs as a subconcept of "*signal*", which certainly makes sense from the domain point of view ("*pathway*" on its own is still a concept intent, but it is not sufficiently stable). Some less important communities, like "*mouse, conservation*" or "*signal, pathway, mouse*" are missing from Fig. 2. Instead, the taxonomy focuses on more solid associations ignoring some particularities, which can be reintroduced by increasing the number of stable concepts included in the taxonomy. Alternatively, we could have a multi-level representation with these communities rendered at a deeper level. Something similar applies, e.g., to the community "*signal, receptor, growth, pathway*", which is missing from Fig. 2, but is interesting according to the expert-based description of the field (see Sect. 2.3). In this respect, nested line diagrams would appear to provide a handy representation by distinguishing various levels of importance of notions.

The reduced substructure corresponding to the ECCS dataset (with 401 authors and 33 words) featuring the 17 most stable concepts is presented in Fig. 3. Again, note that the set of all stable concepts (for an arbitrary threshold) does not have to be a lattice, even if it is in the examples used in this paper.

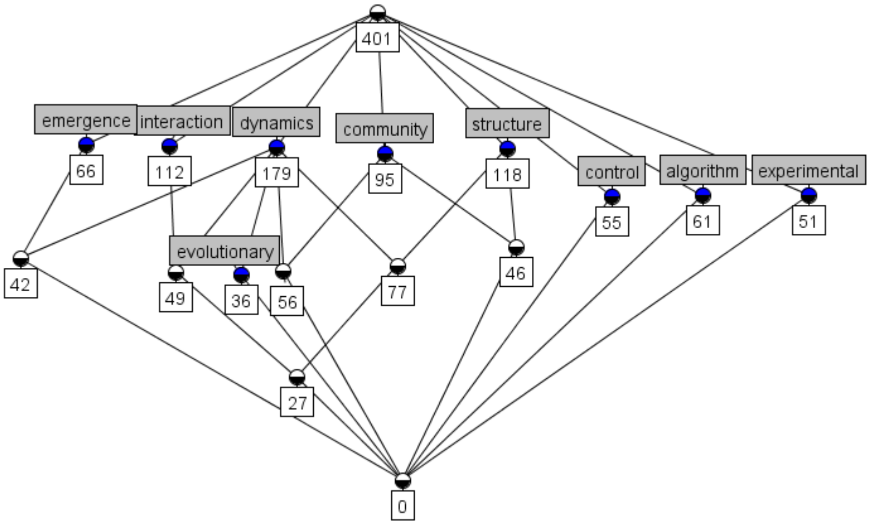

Figure 3: The 17 most stable concepts in the ECCS dataset.

From this lattice, it is possible to provide the following description of the community who attended the ECCS in 2006:

- Most notions present at the top-level should be typical of major issues in the community: "*dynamics*", "*community*" and "*structure*", "*algorithm*", "*emergence*", and "*interaction*". In particular, interactions, which are often modeled within networks, are as such a central methodological tool for complex systems.

- Besides, ECs focused on the *dynamics* of complex systems are strongly structuring the field, as they form many subcommunities with other ECs, at various levels; including for instance "*dynamics, emergence*" and "*dynamics, interaction*".

- More specifically, "*interaction*" as a method is often used jointly with "*dynamics*" and "*structure*", suggesting a particular interest in the structural evolution of networks and interactions. Additionally, "*evolutionary*" as a methodological viewpoint is mostly involved in the studies of "*dynamics*".

- There is no method that is specific or prevailing in the topic of "*control*"—i.e., the control of systems and optimization of their behavior towards some task.
- On the other hand, algorithmic methods and experimental techniques do not seem to be specific to any topic, and "*algorithm*" and "*experimental*" thus make isolated ECs in this lattice.

3.3 Nested Line Diagrams

Nested line diagrams are a well-established tool in formal concept analysis that makes it possible to distribute representation details across several levels [8]. The main idea is to divide the attribute set of the context into two (or more) parts, construct the concept lattices for the generated subcontexts, and draw the diagram of one lattice inside each node of the other lattice. In the case of two parts, an inner concept (A, B) enclosed within an outer concept (C, D) corresponds to a pair $(A \cap C, B \cup D)$. Not every such pair is a concept of the original context. Only inner nodes that correspond to concepts are represented by circles; such nodes are said to be "realized". The outer diagram structures the data along one attribute subset, while the diagram inside an outer concept describes its structure in terms of the remaining attributes. For more details, see [8].

The first step in constructing a nested line diagram is to split the attribute set into several parts. These parts do not have to be disjoint, but they will be in our case; hence, we are looking for a partition of the attribute set. As we seek to improve readability, we should display foremost the most significant attributes; therefore, we should assign major notions to higher levels, while leaving minor distinctions for lower levels. To this end, the words should be partitioned according to a "preference function", which could range from the simplest (e.g., word frequency within the corpus) to more complicated designs.

One could consider a minimal set of notions covering all authors, i.e., find an irredundant cover set, as words from such a set could be expected to play a key role in describing the community (even if this set is, of course, not unique in general). This is what we have done in [26], where we used the algorithm from [2] to compute such a cover set for the small zebrafish context (with 25 authors), whose concept lattice is shown in Fig. 1. We apply it iteratively: the first subcontext contains notions forming an irredundant cover set for the whole author set; the second subcontext includes notions not occurring in the first subcontext, while covering the set of authors excluding those that use only notions from the first subcontext, etc. The last level contains the remaining notions. For more details and the resulting structures see [26].

In this paper, we focus on the ECCS data. To partition attributes, we divide the set of terms into several groups depending on their meaning. We identify two groups of terms: those referring to topics and those referring to methods. In nesting, we can draw the line diagram of methods inside the line diagram of topics to see which methods are used for which topics. Yet, while nesting makes it possible to distinguish between various levels of precision, both the outer and inner diagrams are still too large (240 nodes for topics and 53 nodes for methods). Stability-based pruning will address this problem; combining both procedures should yield a concise hierarchical representation.

3.4 Combining Nesting and Stability-Based Pruning

After partitioning the set of words and building lattices for individual parts, we prune each lattice using the stability criterion. We can use different thresholds for different parts depending on the number of concepts we are comfortable to work with. For our ECCS example, we get the two diagrams shown in Fig. 4. Many attributes are not shown in the picture, as they are not contained in any stable intent but the bottom ones.

We proceed by drawing one diagram inside the other and interpret the picture as usual. Again, only "realized" inner nodes, i.e., those corresponding to concepts of the full context are represented by circles. Figure 5 shows the resulting structure for our context.

This approach may also help in reducing the computational complexity. Generally, computing inner concepts is the same as computing the lattice for the whole context, but, combining pruning and nesting, we compute inner nodes only for relevant (that is, non-pruned) outer nodes.

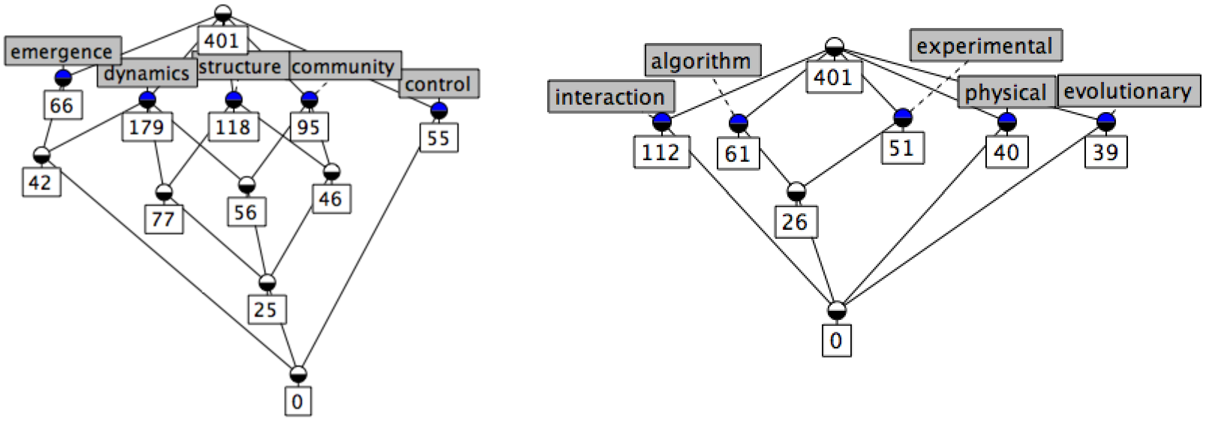

Figure 4: The pruned lattices of ECCS topics and methods.

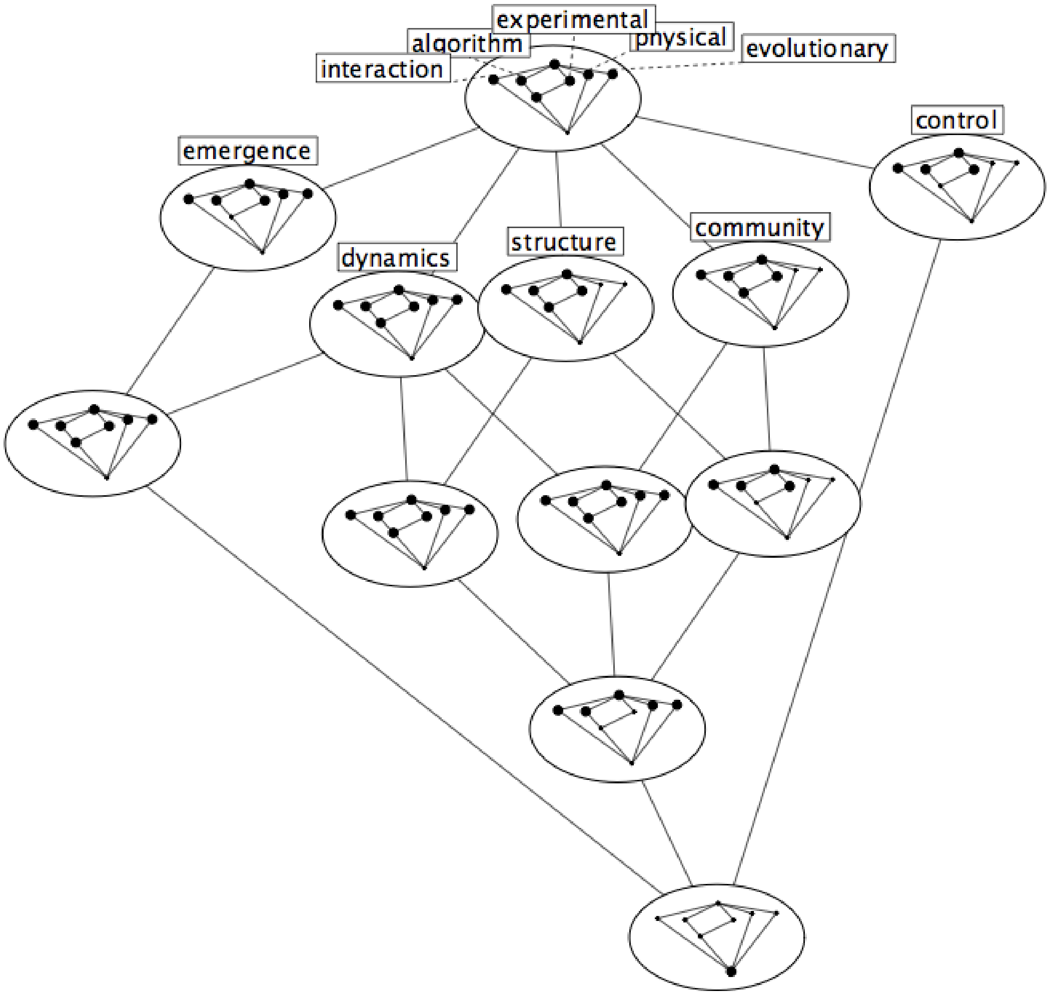

Figure 5: Nested line diagram of pruned lattices from Fig. 4

Let us denote by $\underline{\mathfrak{B}}_p(\mathbb{K})$ the set of concepts of $\mathbb{K}$ satisfying the chosen pruning criteria and ordered in the usual way (one may regard p as an indicator of a specific pruning strategy). Assume that contexts $\mathbb{K}_1 = (G, M_1, I_1)$ and $\mathbb{K}_2 = (G, M_2, I_2)$ are subcontexts of $\mathbb{K} = (G, M, I)$ such that $M = M_1 \cup M_2$ and $I = I_1 \cup I_2$. We define the set of concepts corresponding to nodes of the nested line diagram of the pruned concept sets $\underline{\mathfrak{B}}_p(\mathbb{K}_1)$ and $\underline{\mathfrak{B}}_p(\mathbb{K}_2)$:

$$\underline{\mathfrak{B}}_p(G, M_1, M_2, I) = \{(A, B) \in \underline{\mathfrak{B}}(\mathbb{K}) \mid \forall i \in \{1, 2\} : ((B \cap M_i)', B \cap M_i) \in \underline{\mathfrak{B}}_p(\mathbb{K}_i)\}.$$

Proposition 2. *If $\underline{\mathfrak{B}}_p(\mathbb{K}_1)$ and $\underline{\mathfrak{B}}_p(\mathbb{K}_2)$ are $\bigvee$-subsemilattices of $\underline{\mathfrak{B}}(\mathbb{K}_1)$ and $\underline{\mathfrak{B}}(\mathbb{K}_2)$, respectively, then $\underline{\mathfrak{B}}_p(G, M_1, M_2, I)$ is a $\bigvee$-subsemilattice of $\underline{\mathfrak{B}}(\mathbb{K})$ and the map*

$$(A, B) \mapsto (((B \cap M_1)', B \cap M_1), ((B \cap M_2)', B \cap M_2)) \tag{5}$$

is a $\bigvee$-preserving order embedding of $\underline{\mathfrak{B}}_p(G, M_1, M_2, I)$ in the direct product of $\underline{\mathfrak{B}}_p(\mathbb{K}_1)$ and $\underline{\mathfrak{B}}_p(\mathbb{K}_2)$.

Proof. Let $(A, B), (C, D) \in \underline{\mathfrak{B}}_p(G, M_1, M_2, I)$. Then, we have $(A, B) \vee (C, D) = ((B \cap D)', B \cap D) \in \underline{\mathfrak{B}}_p(G, M_1, M_2, I)$, since $B \cap D \cap M_i = (B \cap M_i) \cap (D \cap M_i)$ is the intent of a concept in $\underline{\mathfrak{B}}_p(\mathbb{K}_i)$ for $i \in \{1, 2\}$. Hence, $\underline{\mathfrak{B}}_p(G, M_1, M_2, I)$ is indeed a $\bigvee$-subsemilattice of $\underline{\mathfrak{B}}(\mathbb{K})$. To see that the above-mentioned mapping is $\bigvee$-preserving, note that it maps the intent $B \cap D$ to the pair of intents $(B \cap D \cap M_1, B \cap D \cap M_2)$, and $B \cap D \cap M_i$ is the intent of the join of concepts with intents $B \cap M_i$ and $D \cap M_i$ for $i \in \{1, 2\}$.

Unlike in standard nesting [8], the component maps $(A, B) \mapsto ((B \cap M_i)', B \cap M_i)$ are not necessarily surjective on $\underline{\mathfrak{B}}(\mathbb{K}_i)$. Hence, some outer nodes in our nested line diagram may be empty, i.e., contain no realized inner nodes, and some nodes of the inner diagram may never be realized.

Back to our example, the pruned outer diagram on Fig. 5 embraces most major topics, while the stabilized inner diagram covers most major methods, as detailed above in Sect. 3.2.2. We can thus discern "*community*", "*structure*" and "*dynamics*" which altogether form pairs of joint communities, while "*control*" and "*emergence*" are also present as main issues: "*emergence*", in particular, is a single-topic EC, and it also and makes a joint community with "*dynamics*". Looking at realized nodes in inner diagrams, we notice that "*interaction*" and "*algorithm*" are everywhere, and this plausibly indicates a very strong core of methods for complex systems topics. However, on the whole, most nodes are actually realized, which is unsurprising given that we selected major methods, likely to be realized for all major topics. In this respect, studying non-realized nodes may yield some more precise insight: for example, apart from "*interaction*", "*algorithm*", and "*experimental*" no node is realized in the "*control*" and "*community, structure*" nodes, while a lack of white nodes in ECs related to dynamics of communities seems to indicate that more diverse methods are used in relation to these topics than in studying other issues related to communities. One can also notice that "*physical*" and "*evolutionary*" nodes are unrealized in the outer "*community*" node, but become realized in its child "*dynamics, community*" node. This indicates implications in the data: "*community, physical*" → "*dynamics*" and "*community, evolutionary*" → "*dynamics*", suggesting that physical and evolutionary approaches to the study of communities are applied exclusively—or, given possible noise in our data, *mostly*—in the studies of community dynamics. The same inner nodes—"*physical*" and "*evolutionary*"—are unrealized for "*control*", but, then, the "*control*" node has no descendants in the "stabilized" outer lattice. Thus, even if physical or evolutionary approaches might be common in the study of some subtopics of "*control*", these subtopics are not relevant to the general description of the ECCS community at the chosen level of detail.

Yet, even with these finer distinctions, it remains substantially hard to distinguish which are the most important methods for each topic; in this regard, it might be preferable to have a proper subdiagram for each outer node, i.e., *zoom* on each topic to get its proper structure.

3.5 Zooming

Nested line diagrams allow one to uncover connections between inner nodes located in different outer nodes (see the discussion on implications above). The drawback of this representation is that pruning is possible only on the global level: in the case of Fig. 5, we select major combinations of topics and major combinations of methods and then use nesting to show which major methods are used in which major topics. However, different methods are used in studying different topics, and some methods that are not

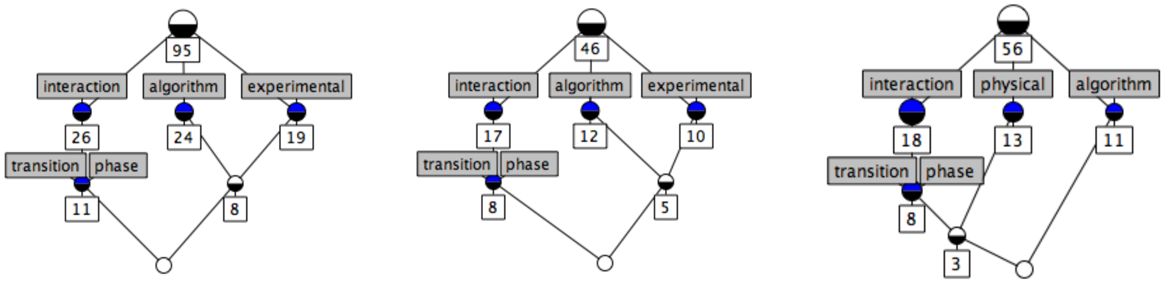

Figure 6: Zoomed lattices for “*community*” (*left*), “*community, structure*” (*middle*), and “*community, dynamics*” (*right*).

that important “globally” (and hence missing from our pruned diagram) might be essential in studying a particular topic. Therefore, it would be better in some cases to prune each “inner” lattice individually. With this approach, we will not be able to directly see the links between method combinations across different topics, but we will have a more adequate representation of methods used for each topic.

We call this approach “zooming”. The idea is simple: by zooming into a concept (C, D) from the lattice of topics, we mean the construction of the lattice of the subcontext of methods formed by restricting the set of objects to C (and, of course, by restricting the set of attributes to those referring to methods).

Zoomed lattices reveal differences between topics in a much clearer and more sensible manner than would be by interpreting inner nodes of a nested line diagram: significant methods are described in the zoomed lattice of each topic without requiring a comparison with the rest of the structure in order to understand which methods are indeed present or not, major or minor. Applying this to our ECCS example, we consider a selection of a few topics from the lattice of topics (see Fig. 4) and show the corresponding zoomed lattices of methods in Figs. 6 and 7. For each topic, we use various stability thresholds such that lattice sizes remain in the same range of 6-8 nodes, while including all nodes with the same stability index. It is subsequently straightforward to choose various levels of detail for each diagram rather than work with the same lattice for all topics. Besides, whereas having more nodes would obviously have provided more details, it would however have been at the cost of including nodes whose significance might have been questionable—such as nodes covering fewer than 5 agents, for instance.

Our set includes nodes corresponding to “*community*”-related topics which appear to be very similar to each other, at various scales, indicating that the main methods in use in these subfields are consistent across subtopics. Most interestingly, zoomed lattices for “*community*” and “*community, structure*” are strictly identical, with the exception of population figures (see Fig. 6), while “*community, dynamics*” differs only slightly by the fact that “*physical*” replaces “*experiment*” as a method and is jointly linked to “*transition phase, interaction*”—this suggests that notions from statistical physics are more prominent in the dynamical study of communities, with less empiricism.

To the contrary, zoomed nodes for “*dynamics*” and “*control*” are significantly different from each other and from the “*community*”-related lattices: the subtopic “*dynamics*” indeed has a rather flat zoomed lattice, where all methods seem to hold equal positions, although their populations range from 17 agents for “*deterministic*” to 49 for “*interaction*”. This confirms the previous observation that dynamics are a strongly structuring topic for complex systems spanning almost all major methodological notions. On the other hand, the zoomed lattice for “*control*” exhibits a more hierarchical structure where “*experimental*” and “*interaction*” appear as major methods, possibly representative of the relationships of these studies with empirical approaches and network design problems—“*evolutionary*” and “*algorithm*” are somewhat less important, possibly related to optimization methods *per se.* Note that “*control, evolutionary*”, indistinguishable in the “global” nested line diagram, becomes visible when we zoom into “control”. Finally, the fact that “*algorithm*” is described as a submethod of “*experimental*” could emphasize how helpful algorithms are in implementing and experimenting normative control prescriptions.

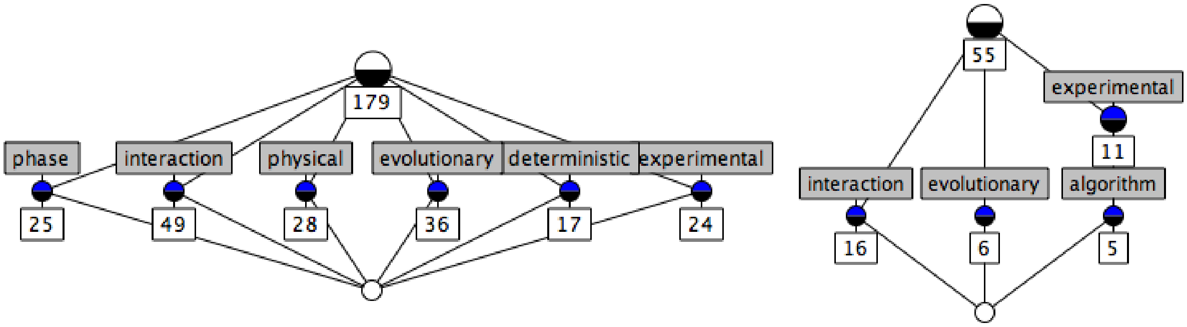

Figure 7: Zoomed lattices for "*dynamics*" (*left*) and "*control*" (*right*).

4 Further Work

4.1 Variants of Stability

The stability index σ as in Definition 1 and [15] refers to the stability of an intent; we may call it *intensional*. The *extensional stability index* of a concept (A, B) can be defined similarly:

$$\sigma_e(A, B) = \frac{|\{D \subseteq B \mid D' = A\}|}{2^{|B|}}. \tag{6}$$

The extensional stability of a concept is the probability of preserving its extent after leaving out an arbitrary subset of attributes from the context. A proposition similar to Proposition 1 holds. Extensional stability relates to the social aspect of the concept, measuring how much the community as a group of people depends on a particular topic. It may also be useful for fighting noisy words: a community based on a noisy word (or, e.g., a homograph used differently within different communities) is likely be extensionally unstable.

Proposition 1 suggests how the *general stability index* of a concept (A, B) should be defined—as the ratio between the number of subcontexts of $\mathbb{K} = (G, M, I)$ preserving the concept up to the omitted objects and attributes and the total number of subcontexts, or, more formally:

$$\frac{|\{(H, N, J) \mid H \subseteq G, N \subseteq M, J = I \cap (H \times N), A_H^J = B_N, B_N^J = A_H\}|}{2^{|G|+|M|}} \tag{7}$$

where $A_H = A \cap H$ and $B_N = B \cap N$. As of now, we are not aware of any realistic method for computing this general stability; thus, it seems to be mainly of theoretical interest.

On the other hand, limited versions of stability (e.g., computed over subsets of a certain size only), as well as various combinations of extensional and intensional stability, are worth trying.

4.2 Strategies for Pruning

Other techniques aiming at reducing the number of concepts should be tested and, perhaps, some of them can be combined with stability for better results—notably pruning based on monotonous criteria like extent/intent size. Another method is given by attribute-dependency formulas [4], involving an expert-specified hierarchy on the attribute set (e.g., "*human*" and "*mouse*" are subtypes of "*vertebrate*").

As noticed in Sect. 3.2.2, pruning may not necessarily yield a lattice. We can handle this situation in several ways: for example, enlarging the resulting structure by including all intersections of stable intents or reducing it by eliminating some stable intents. Regarding the latter option, we may prefer to merge an unstable concept (A, B) with its subconcept (C, D) rather than simply drop (A, B). It is not immediately clear how to choose (C, D)—only that it should be somehow close to (A, B). Merging can be done by assuming that all objects from A have all attributes from D and replacing the context relation I

by $I \cup A \times D$ (cf. [23]). However, the modified context may have intents that are absent from the initial context, which is probably undesirable. Alternatively, one could add $B \to D$ to the implication system of the context. The lattice of attribute subsets generated by this augmented implication system will be different from the original lattice only in that B and possibly some of its previously closed supersets are not in the new lattice.

A different approach would involve merging based on partial implications (or association rules): compute all partial implications for the given confidence threshold and add them to the implication system of the context. It is a matter of further experiments to see which strategies are suitable for our goals.

4.3 Nesting and zooming

Nested line and zoomed diagrams are not limited to two levels, although it still has to be investigated whether multi-level diagrams remain readable and interpretable. Various techniques for partitioning attribute sets should be explored. One strategy specific to our application is to partition words according to their type: as a verb, noun, adjective; or as a method, object, property, etc., which is what we have done for the ECCS data in this paper. Such a method can be combined with other feature selection algorithms.

It should be noted that nesting and zooming seem to have more potential if used in interactive software tools that allow the user to zoom in and out on particular communities instead of having to deal with the entire picture. Such a tool could even allow the user to interchange between nesting and zooming, since these two representations should be seen as complementary. The fact that one need not compute everything at once provides an additional computational advantage.

4.4 Dynamic Monitoring

Modeling changes of the community structure should be particularly useful to describe the evolution of fields historically, either longitudinally or dynamically. The longitudinal approach means establishing a relation between community structures corresponding to different time points, e.g., identifying cases when several communities have merged into one or a community has divided into several sub-communities. FCA offers some methods for comparing two lattices built from identical objects and/or attributes (e.g., see [30]). Yet the relevance of such methods is likely to be application-dependent, and they should certainly be adapted for the reduced lattice-based structures we work with. One possibility in line with the static approach is to use nested line diagrams by nesting diagrams of contexts corresponding to successive time points. It will also be worth exploring what temporal concept analysis [31] has to offer in this regard.

A more ambitious dynamic approach to modeling changes assumes that any elementary change in the database (any modification of G, M, or I: a new author, a new word, an author using a particular word for the first time, or removal of authors due to their inactivity, etc.) should correspond to a concrete change in the representation of communities. Although not every such change will have effect on the structure of the communities, it should always be possible to trace a change in the structure to a sequence of elementary changes in the database.

5 Conclusion

The approach discussed in this paper is based on the assumption that community structure in knowledge-based social networks should be dealt with more deeply than by simply relying on single-mode characterizations, as is often the case. In previous work [24, 25], it was shown how concept lattices can be used to build knowledge taxonomies from data describing authors by sets of terms they use in their papers. As frequently happens with concept lattices derived from real data, such taxonomies tend to be huge and, therefore, hard to compute and analyze. The computational complexity can be partially addressed by reducing the number of agents, since a taxonomy centered on knowledge fields rather than individuals justifies the use of a random representative sample of authors.

However, the interpretability of results requires a more serious effort. In this paper, we proposed a pruning method based on the stability indices of formal concepts [15]. We think that this method does not simply reduce the concept lattice to a somewhat rougher structure, but also helps to combat noise in data, so that the resulting structure might even be more accurate in describing the knowledge community than the original lattice is.

We suggested that this method could also be applied to constituent parts of a nested line diagram or of a zoomed diagram in order to achieve an optimal relationship between the readability of the taxonomy and the level of detail in it. This is beneficial from the viewpoint of computational complexity, too: it is easier to compute the lattices of subcontexts used in nesting or zooming and then prune each of them individually than to compute the lattice of the entire context and prune it. Besides, such diagrams admit "lazy" computation: if nesting or zooming is implemented within an interactive software tool, the user should be allowed to choose which outer nodes to explore; if the user is not interested in some outer nodes, the corresponding inner diagrams will never be computed.

We have illustrated the proposed techniques with a few examples. Of course, wider experiments are needed to see how this works. There are quite a few open questions: how to efficiently compute stability, how exactly stability-based criteria should be formulated and applied (e.g., dropping unstable nodes vs. merging them with stable nodes), what other compression techniques exist and how they perform against stability-based pruning, etc. Also, better linguistic processing is certainly need at the stage when a context is constructed from a database: smart methods to identify notions would be able to deal with homonyms and synonyms, take into account the context and domains-specific associations between words. We have summarized some of the possible research directions in Sect. 4. Thus, this paper is only a step towards a consistent methodology for creating concise knowledge taxonomies based on concept lattices.

References

[1] Scott Atran. Folk biology and the anthropology of science: Cognitive universals and cognitive particulars. *Behavioral and Brain Sciences*, 21:547–609, 1998.

[2] Ramachendra P. Batni, Jeffrey D. Russell, and Charles R. Kime. An efficient algorithm for finding an irredundant set cover. *J. Ass. for Comp. Machinery*, 21(3):351–355, 1974.

[3] R. Bayardo, Jr., B. Goethals, and M.J. Zaki, editors. *Proc. of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations (FIMI 2004)*. CEUR-WS.org, 2004.

[4] Radim Belohlávek and Vladimir Sklenar. Formal concept analysis constrained by attribute-dependency formulas. In Bernhard Ganter and Robert Godin, editors, *ICFCA 2005*, volume 3403 of *LNAI*, pages 176–191, 2005.

[5] Jane Bradbury. Small fish, big science. *PLoS Biology*, 2(5):568–572, 2004.

[6] Lucia Falzon. Determining groups from the clique structure in large social networks. *Social Networks*, 22:159–172, 2000.

[7] L.C. Freeman. Cliques, Galois lattices, and the structure of human social groups. *Social Networks*, 18:173–187, 1996.

[8] B. Ganter and R. Wille. *Formal Concept Analysis: Mathematical Foundations*. Springer, Berlin, 1999.

[9] Michelle Girvan and Mark E. J. Newman. Community structure in social and biological networks. *PNAS*, 99:7821–7826, 2002.

[10] D. J. Grunwald and J. S. Eisen. Headwaters of the zebrafish – emergence of a new model vertebrate. *Nature Rev. Genetics*, 3(9):717–724, 2002.

[11] J.-L. Guigues and V. Duquenne. Familles minimales d'implications informatives resultant d'un tableau de données binaires. *Math. Sci. Humaines*, 95:5–18, 1986.

[12] P. Haas. Introduction: epistemic communities and international policy coordination. *International Organization*, 46(1):1–35, winter 1992.

[13] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *IJCAI*, pages 1137–1145, 1995.

[14] Sergei O. Kuznetsov. Stability as an estimate of the degree of substantiation of hypotheses derived on the basis of operational similarity. *Nauchn. Tekh. Inf., Ser.2 (Automat. Document. Math. Linguist.)*, (12):21–29, 1990.

[15] Sergei O. Kuznetsov. On stability of a formal concept. In Eric SanJuan, editor, *JIM*, Metz, France, September 2003.

[16] Sergei O. Kuznetsov and Sergei Obiedkov. Comparing performance of algorithms for generating concept lattices. *J. Expt. Theor. Artif. Intell.*, 14(2/3):189–216, 2002.

[17] Loet Leydesdorff. In search of epistemic networks. *Social Studies of Science*, 21:75–110, 1991.

[18] F. Lorrain and H. C. White. Structural equivalence of individuals in social networks. *Journal of Mathematical Sociology*, 1(49–80), 1971.

[19] K. W. McCain, J. M. Verner, G. W. Hislop, W. Evanco, and V. Cole. The use of bibliometric and Knowledge Elicitation techniques to map a knowledge domain: Software Engineering in the 1990s. *Scientometrics*, 65(1):131–144, 2005.

[20] Katherine W. McCain. Cocited author mapping as a valid representation of intellectual structure. *J. Am. Society for Information Science*, 37(3):111–122, 1986.

[21] E. C. M. Noyons and A. F. J. van Raan. Monitoring scientific developments from a dynamic perspective: self-organized structuring to map neural network research. *Journal of the American Society for Information Science*, 49(1):68–81, 1998.

[22] Nadine Peyriéras. Personal communication., 2005.

[23] Jayson E. Rome and Robert M. Haralick. Towards a formal concept analysis approach to exploring communities on the world wide web. In Bernhard Ganter and Robert Godin, editors, *ICFCA 2005*, volume 3403 of *LNAI*, pages 33–48, 2005.

[24] Camille Roth and Paul Bourgine. Epistemic communities: Description and hierarchic categorization. *Mathematical Population Studies*, 12(2):107–130, 2005.

[25] Camille Roth and Paul Bourgine. Lattice-based dynamic and overlapping taxonomies: the case of epistemic communities. *Scientometrics*, 69(2), 2006.

[26] Camille Roth, Sergei Obiedkov, and Derrick G. Kourie. Towards concise representation for taxonomies of epistemic communities. In Sadok Ben Yahia and Engelbert Mephu Nguifo, editors, *CLA 4th International Conference on Concept Lattices and their Applications*. Springer, 2006.

[27] F. Schmitt, editor. *Socializing Epistemology: The Social Dimensions of Knowledge*. Lanham, MD: Rowman & Littlefield, 1995.

[28] Gerd Stumme, Rafik Taouil, Yves Bastide, Nicolas Pasquier, and Lotfi Lakhal. Computing iceberg concept lattices with TITANIC. *Data & Knowledge Engineering*, 42:189–222, 2002.

[29] D. R. White and Vincent Duquenne. Social network & discrete structure analysis: Introduction to a special issue. *Social Networks*, 18:169–172, 1996.

[30] R. Wille. Conceptual structures of multicontexts. In P.W. Eklund, G. Ellis, and G. Mann, editors, *Conceptual Structures: Knowledge Representation as Interlingua*, volume 1115 of *LNAI*, pages 23–29, Heidelberg-Berlin-New York, 1996. Springer.

[31] K. E. Wolff. Temporal concept analysis. In E. Mephu Nguifo and et al., editors, *ICCS-2001 Intl. Workshop on Concept Lattices-Based Theory, Methods and Tools for Knowledge Discovery in Databases*, pages 91–107, Palo Alto (CA), July 2001. Stanford Univ.