Formal concepts for learning and education

Bernhard Ganter

Institut für Algebra
TU Dresden
D-01062 Dresden

AIMSA Conference, Golden Sands, Sept. 12, 2014

Outline

1. An introduction to Formal Concept Analysis
2. Formal concepts for knowledge acquisition
3. Mathematics: combinatorics of ordered sets
4. Formal concepts for skill analysis

What’s next?

1. An introduction to Formal Concept Analysis
2. Formal concepts for knowledge acquisition
3. Mathematics: combinatorics of ordered sets
4. Formal concepts for skill analysis

Formal Concept Analysis — What is it?

- A mathematical theory, invented $\approx$ 30 years ago,
- based on algebra (lattices) and ordered sets.
- With solid methodology,
- many applications,
- expressive graphics and
- powerful algorithms.

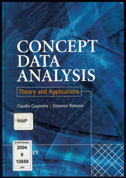
Formal Concept Analysis — What is it?

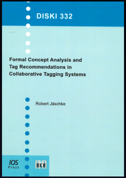
- A mathematical theory, invented $\approx$ 30 years ago,
- based on algebra (lattices) and ordered sets.
- With solid methodology,
- many applications,
- expressive graphics and
- powerful algorithms.

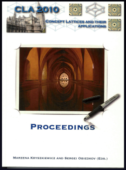
FCA it is not a “method” designed for a certain purpose, but a broad theory with many different applications.

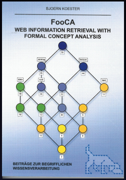
There are all kinds of variants (such as triadic, fuzzy, rough, logical, nonmonotonic, probabilistic FCA.)

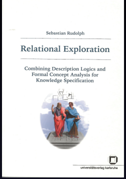
Books on Formal Concept Analysis

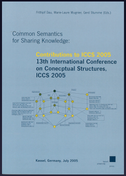


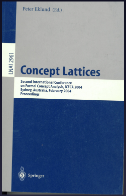


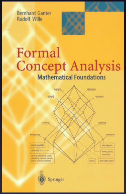


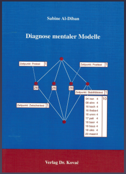


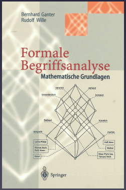


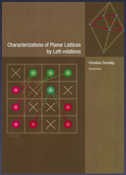












. . .

Differences to other methods

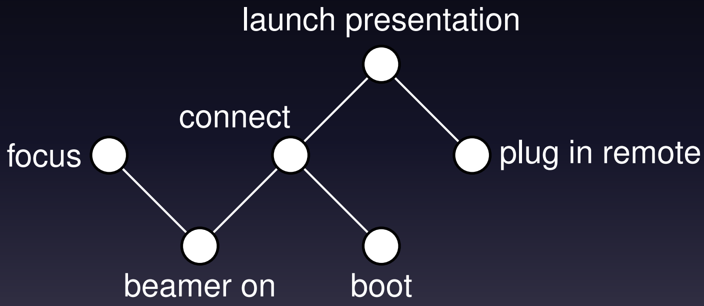
- FCA does not rely on *metric* data.
- Instead, it is an algebraic theory based on ordered sets (“polyhierachic structures”).
- It has a fundamental data type (“formal context”) which is of *relational* nature.
- It unfolds data rather than simplifying it. It emphasises *meaningfulness* and *reliability* of the analysis.
- It has different goals and purposes.
- It uses *order diagrams* for communication.

Differences to other methods

- FCA does not rely on *metric* data.
- Instead, it is an algebraic theory based on ordered sets ("polyhierachic structures").
- It has a fundamental data type ("formal context") which is of *relational* nature.
- It unfolds data rather than simplifying it. It emphasises *meaningfulness* and *reliability* of the analysis.
- It has different goals and purposes.
- It uses *order diagrams* for communication.

Please be brave: There will be no similarities, no distance functions and no trees in this lecture!

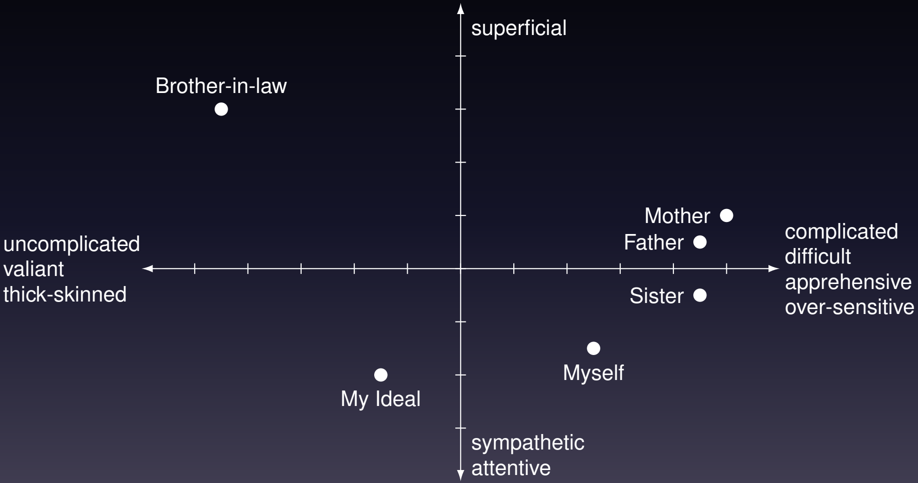
An order diagram: Operating the beamer



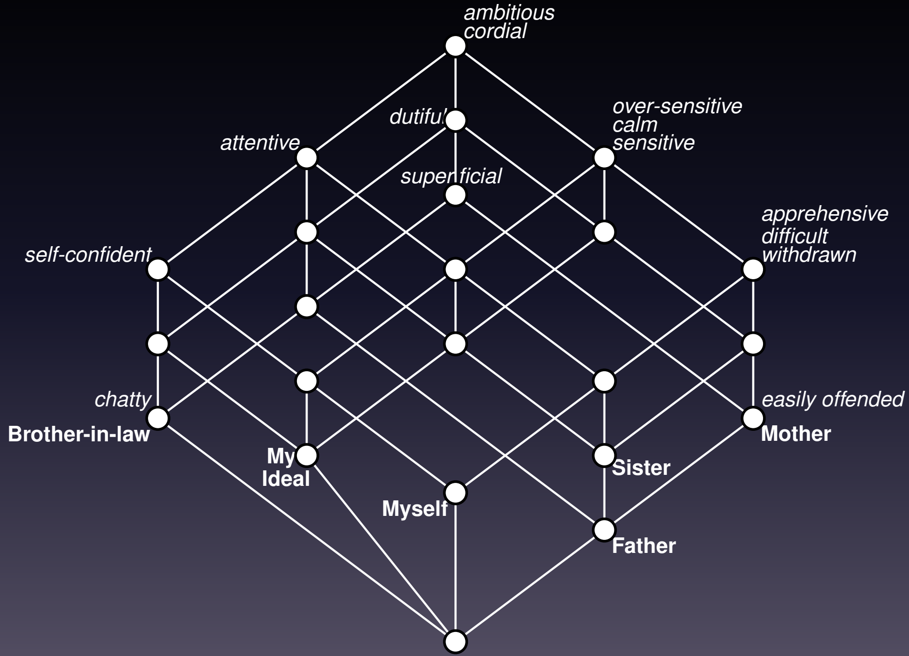
From a treatment of Anorexia nervosa

	over-sensitive	withdrawn	self-confident	dutiful	cordial	difficult	attentive	easily offended	calm	apprehensive	chatty	superficial	sensitive	ambitious
Myself	×	×	×		×	×	×		×	×			×	×
My Ideal	×		×	×	×		×		×				×	×
Father	×	×		×	×	×	×	×	×	×		×	×	×
Mother	×	×		×	×	×		×	×	×		×	×	×
Sister	×	×		×	×	×	×		×	×			×	×
Brother-in-law			×	×	×		×				×	×		×

A biplot of the interview data



A concept lattice of Anorexia data



Mathematical definitions

A **formal context** (G, M, I) consists of two sets G and M (called “objects” and “attributes”) together with a relation $I \subseteq G \times M$ (“has”).

Mathematical definitions

A **formal context** (G, M, I) consists of two sets G and M (called "objects" and "attributes") together with a relation $I \subseteq G \times M$ ("has").

A **formal concept** of (G, M, I) is a pair (A, B) of sets $A \subseteq G$ and $B \subseteq M$ such that

- B consists of precisely those attributes that all objects in A share, and
- A consists of precisely those objects which have all attributes from B.

Mathematical definitions

A **formal context** (G, M, I) consists of two sets G and M (called "objects" and "attributes") together with a relation $I \subseteq G \times M$ ("has").

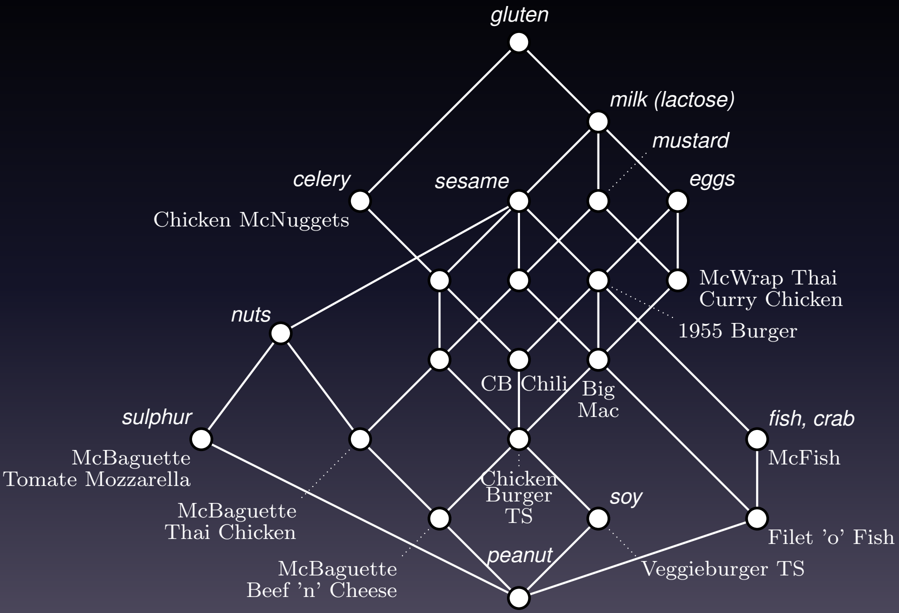
A **formal concept** of (G, M, I) is a pair (A, B) of sets $A \subseteq G$ and $B \subseteq M$ such that

- B consists of precisely those attributes that all objects in A share, and
- A consists of precisely those objects which have all attributes from B.

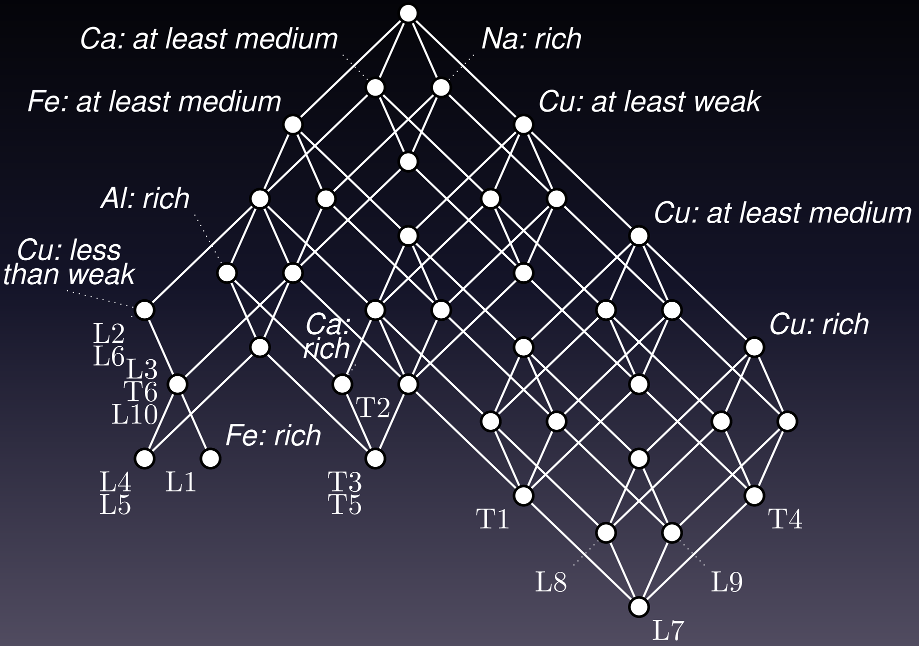
(A_1, B_1) is a **subconcept** of (A_2, B_2) iff $A_1 \subseteq A_2$.

With this order, the formal concepts of any formal context form a complete lattice, the **concept lattice**.

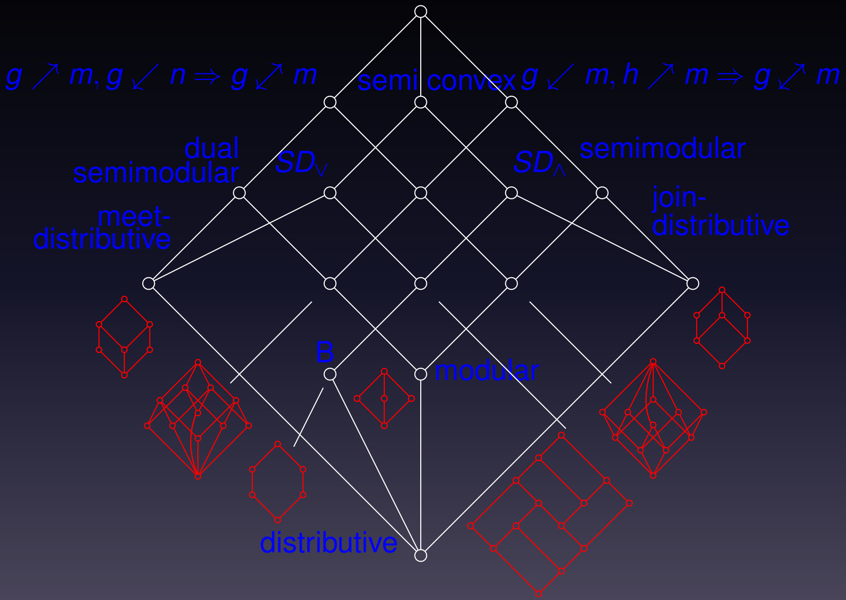
Allergenes in McDonald’s Burgers & Co.



Enamels in the palace of Nebuchadnezzar II

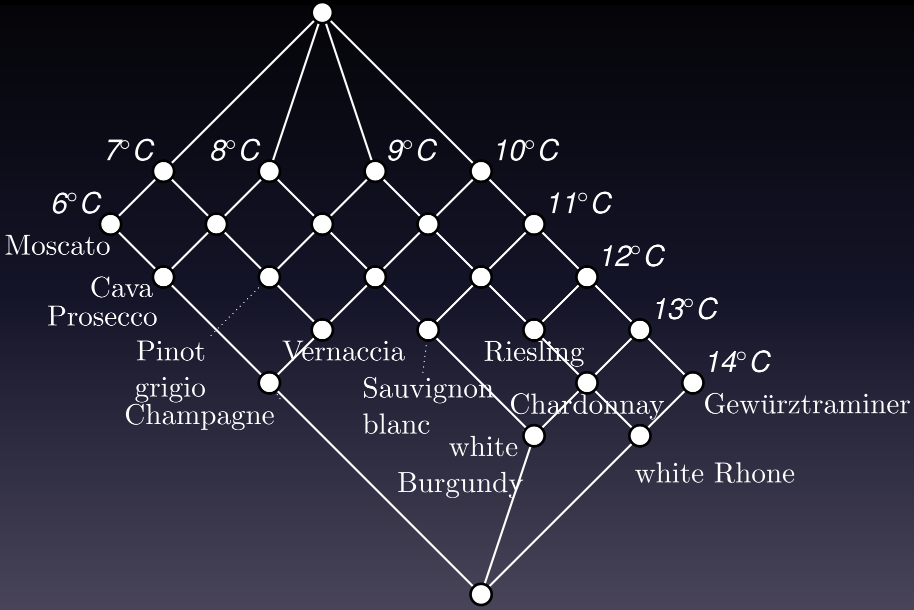


Properties of lattices

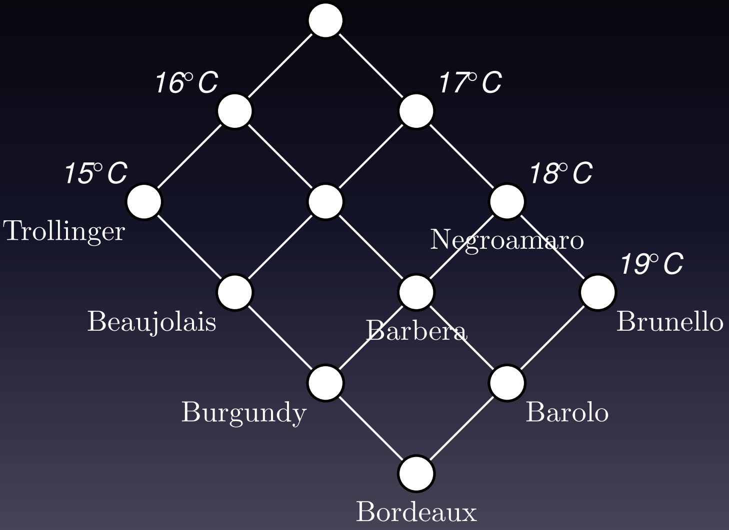


Jump forward

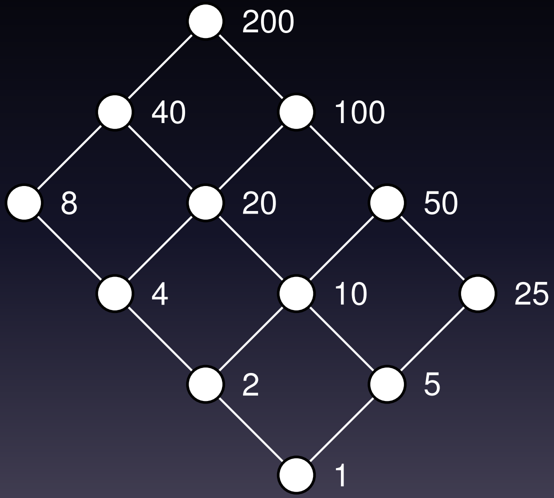
Recommended serving temperatures



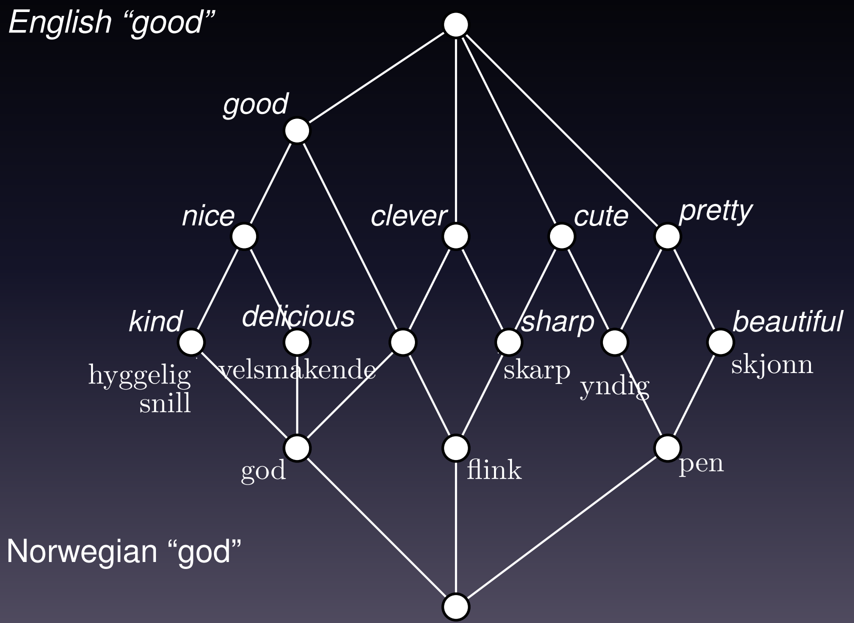
Recommended serving temperatures of red wines



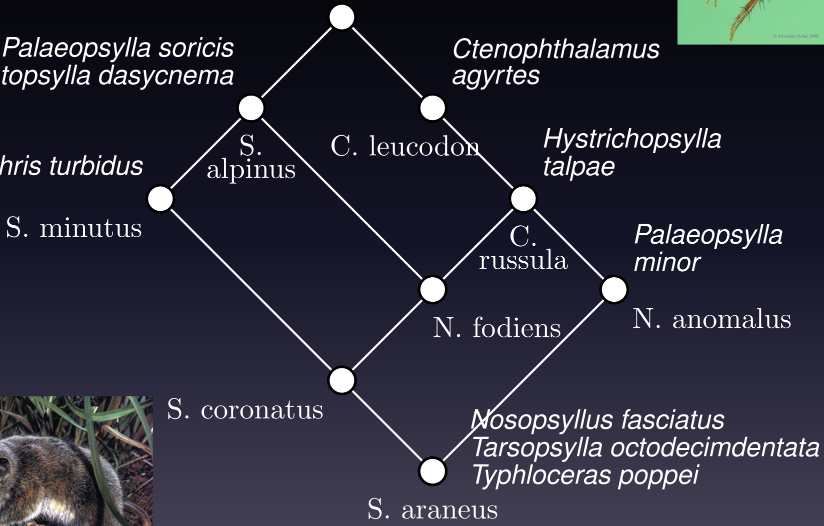
The divisor lattice of 200



Semantic Mirror: English vs. Norwegian

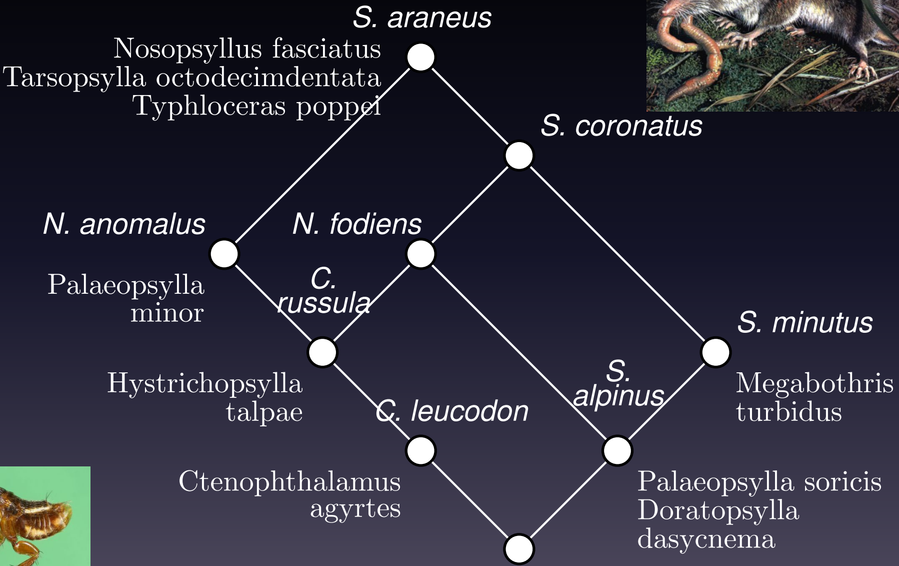


Fleas of shrews



(C. = Crocidura, N. = Neomys, S. = Sorex)

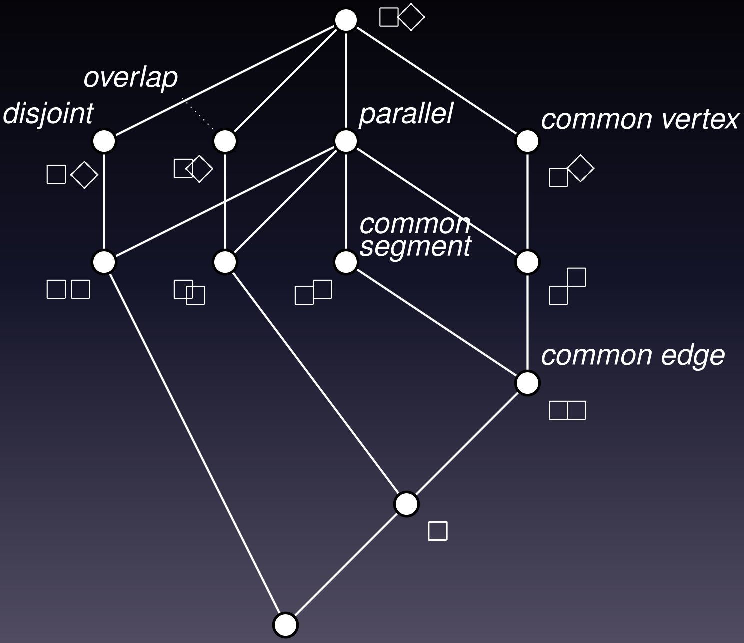
Shrews of fleas



What's next?

1. An introduction to Formal Concept Analysis
2. Formal concepts for knowledge acquisition
3. Mathematics: combinatorics of ordered sets
4. Formal concepts for skill analysis

Knowledge acquisition: Pairs of squares



A logical base of the example set

- common edge $\to$ parallel, common vertex, common segment
- common segment $\to$ parallel
- parallel, common vertex, common segment $\to$ common edge
- overlap, common vertex $\to$ parallel, common segment, common edge
- overlap, parallel, common segment $\to$ common edge, common vertex
- overlap, parallel, common vertex $\to$ common segment, common edge
- disjoint, common vertex $\to \perp$
- disjoint, parallel, common segment $\to \perp$
- disjoint, overlap $\to \perp$

Two of the implications do not hold in general

- common edge $\rightarrow$ parallel, common vertex, common segment
- common segment $\rightarrow$ parallel
- parallel, common vertex, common segment $\rightarrow$ common edge
- overlap, common vertex $\rightarrow$ parallel, common segment, common edge
- overlap, parallel, common segment $\rightarrow$ common edge, common vertex
- overlap, parallel, common vertex $\rightarrow$ common segment, common edge
- disjoint, common vertex $\rightarrow \perp$
- disjoint, parallel, common segment $\rightarrow \perp$
- disjoint, overlap $\rightarrow \perp$

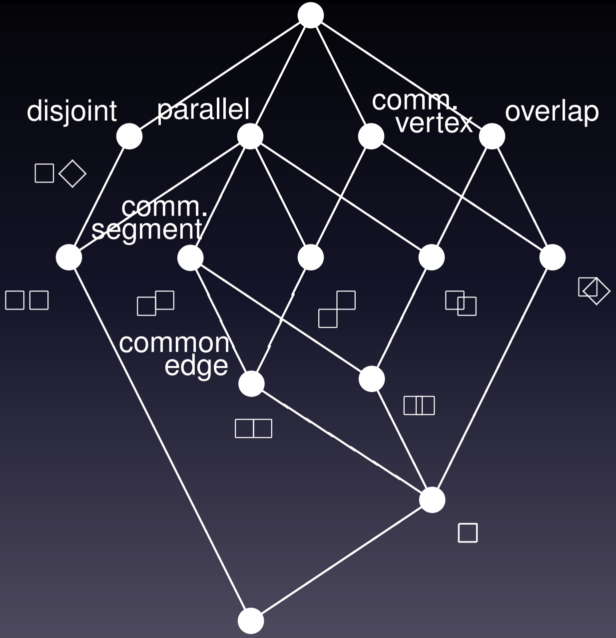
Conterexamples for the two implications

- overlap, common vertex $\rightarrow$ parallel, common segment, common edge
- Counterexample: 

Conterexamples for the two implications

- overlap, common vertex $\rightarrow$ parallel, common segment, common edge
- Counterexample: 
- overlap, parallel, common segment $\rightarrow$ common edge, common vertex
- Counterexample: 

The general case



Attribute Logic

The basic version of Attribute Exploration acquires the logic of attribute implications, i.e., expressions

$$A \to B,$$

where A and B are sets of attributes.

The meaning of an implication $A \to B$ is

> *every object which has all the attributes from A also has all the attributes from B.*

Attribute Logic

The basic version of Attribute Exploration acquires the logic of attribute implications, i.e., expressions

$$A \to B,$$

where A and B are sets of attributes.

The meaning of an implication $A \to B$ is

> *every object which has all the attributes from A also has all the attributes from B.*

The *Armstrong rules* govern implication inference. A theorem by Duquenne and Guigues provides a *canonical base* for each implication theory.

A simple method

The interactive Attribute Exploration algorithm finds the canonical base by identifying in each step an undecided implication, which then is presented to an “expert”, who can confirm the implication as valid or reject it by giving a counterexample.

A simple method

The interactive Attribute Exploration algorithm finds the canonical base by identifying in each step an undecided implication, which then is presented to an “expert”, who can confirm the implication as valid or reject it by giving a counterexample.

The algorithm is very useful, but is it effective?

A simple method

The interactive Attribute Exploration algorithm finds the canonical base by identifying in each step an undecided implication, which then is presented to an “expert”, who can confirm the implication as valid or reject it by giving a counterexample.

The algorithm is very useful, but is it effective?
No, the complexity results are rather unpleasant, at least the ones that we have so far. But even for small data sets this method my yield nontrivial results.

▸ Jump to lattices classification

Replacing the expert

Peter Kestler in his 2013 PhD dissertation has investigated the implicational logic of 70 simple groupoid identities. In order to complete this classification, he had to prove over 50 (simple) theorems confirming implications, and he had to construct over 1400 examples refuting implications. Many of these examples are infinite. The resulting lattice has more than 20 000 elements.

Replacing the expert

Peter Kestler in his 2013 PhD dissertation has investigated the implicational logic of 70 simple groupoid identities. In order to complete this classification, he had to prove over 50 (simple) theorems confirming implications, and he had to construct over 1400 examples refuting implications. Many of these examples are infinite. The resulting lattice has more than 20 000 elements.

Recently, Artem Revenko has repeated Kestler's investigation, but replaced the "expert" by a combination of a theorem prover (Prover9) and an example generator (Maze4). He could perform the complete exploration fully automatic. It took some days of computer time rather than years of manual computation.

Generalisations

Thanks to its solid mathematical foundation, Attribute Exploration could widely be generalised. The most promising recent results were obtained by Franz Baader, Felix Distel, Baris Sertkaya and, very recently, Daniel Borchmann, who extended the exploration algorithm to the Description Logic $\mathcal{EL}^{\perp}$.

Generalisations

Thanks to its solid mathematical foundation, Attribute Exploration could widely be generalised. The most promising recent results were obtained by Franz Baader, Felix Distel, Baris Sertkaya and, very recently, Daniel Borchmann, who extended the exploration algorithm to the Description Logic $\mathcal{EL}^{\perp}$.

Borchmann's thesis entitled

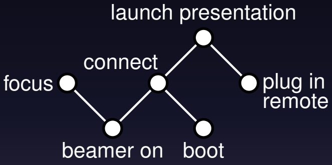
Learning Terminological Knowledge with High Confidence from Erroneous Data

develops methods for extracting "semantic" knowledge from "real world" relational data.

What’s next?

1. An introduction to Formal Concept Analysis
2. Formal concepts for knowledge acquisition
3. Mathematics: combinatorics of ordered sets
4. Formal concepts for skill analysis

The ordinal scale

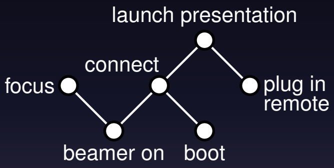


$\leq$	beamer	boot	focus	connect	remote	presentation
beamer	×		×	×		×
boot		×		×		×
focus			×			
connect				×		×
remote					×	×
presentation						×

(The concept lattice of this table is known as the Dedekind-MacNeille completion of the ordered set.)

The contraordinal scale

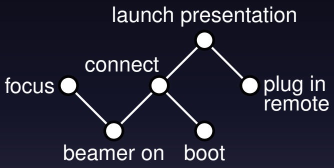
$\ngeq$	beamer	boot	focus	connect	remote	presentation
beamer		×	×	×	×	×
boot	×		×	×	×	×
focus		×		×	×	×
connect			×		×	×
remote	×	×	×	×		×
presentation			×			



A pair (A, B) is a formal concept of the contraordinal scale of $(J, \leq)$ if and only if

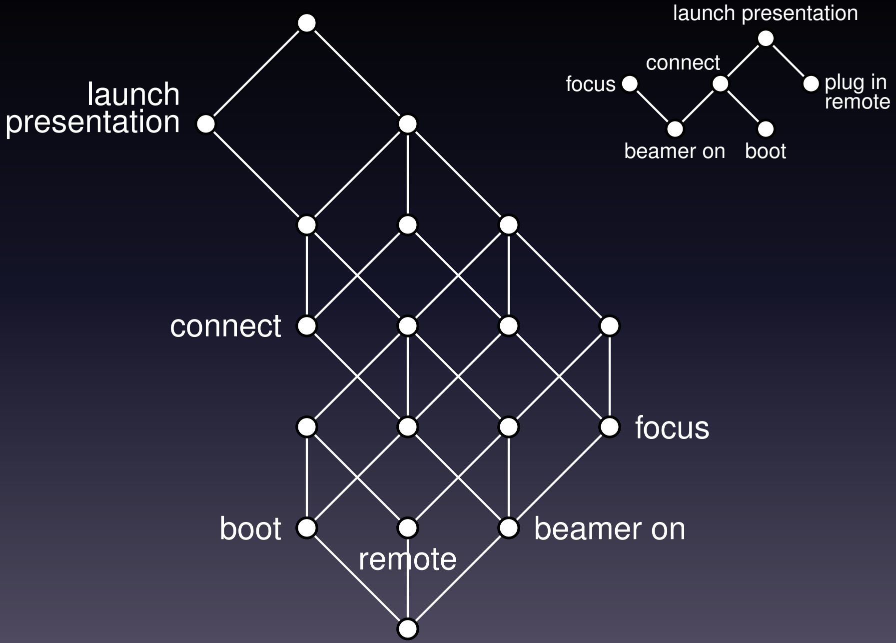
The contraordinal scale

$\ngeq$	beamer	boot	focus	connect	remote	presentation
beamer		$\times$	$\times$	$\times$	$\times$	$\times$
boot	$\times$		$\times$	$\times$	$\times$	$\times$
focus		$\times$		$\times$	$\times$	$\times$
connect			$\times$		$\times$	$\times$
remote	$\times$	$\times$	$\times$	$\times$		$\times$
presentation			$\times$			

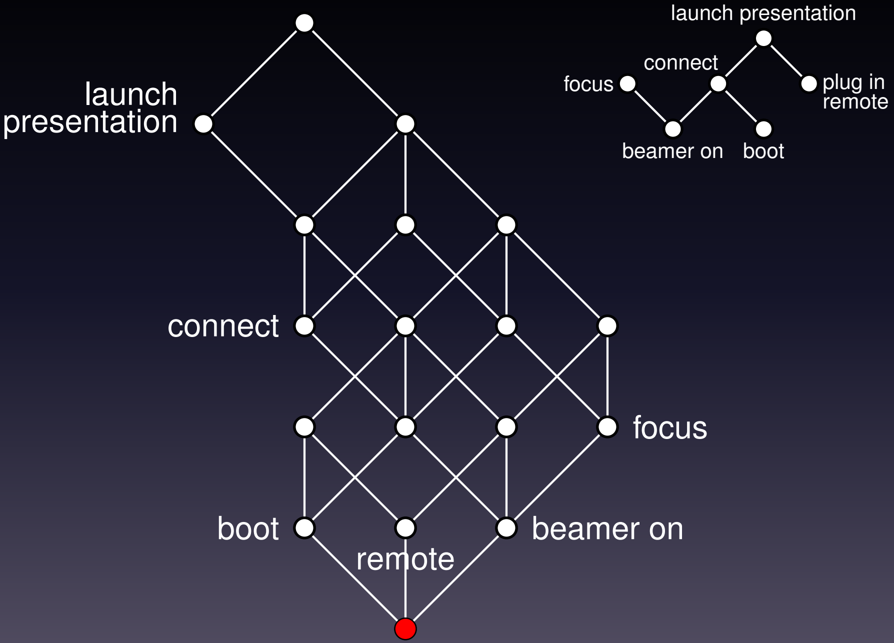


A pair (A, B) is a formal concept of the contraordinal scale of $(J, \leq)$ if and only if A is an order ideal, B is an order filter, and A and B are complements of each other.

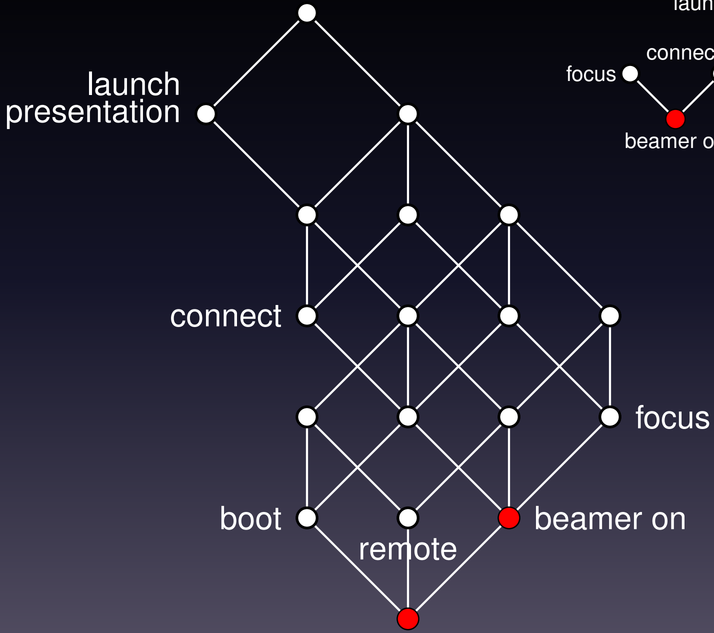
Lattice of order ideals

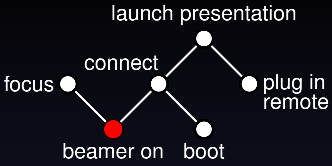


Linear extensions and maximal chains

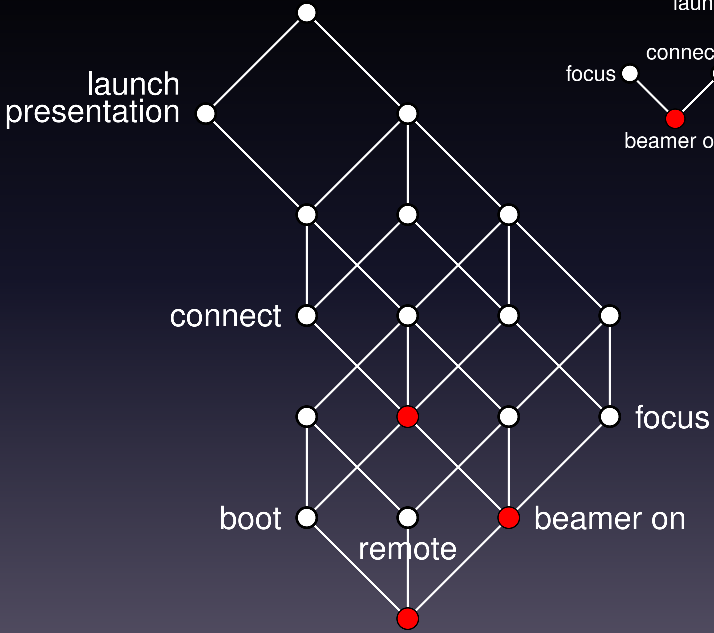


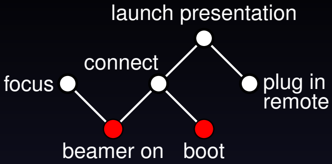
Linear extensions and maximal chains



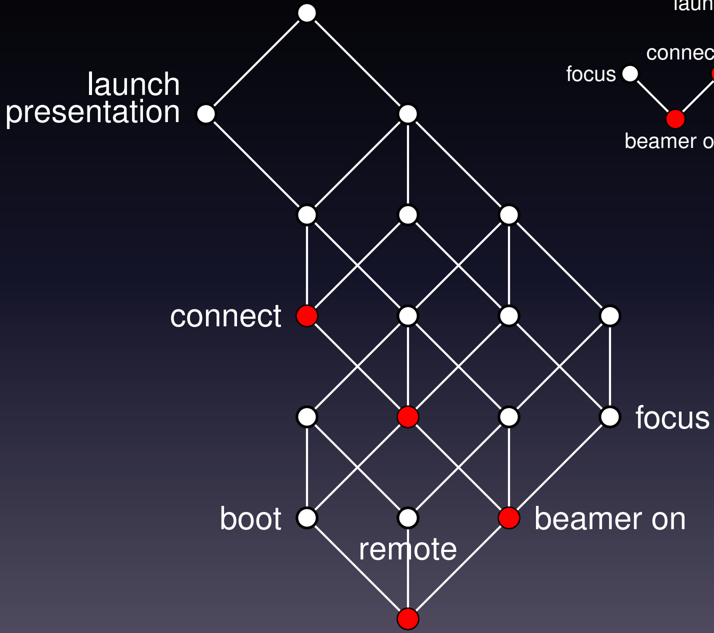


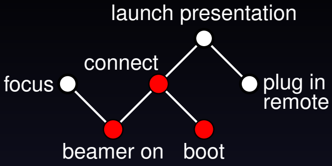
Linear extensions and maximal chains



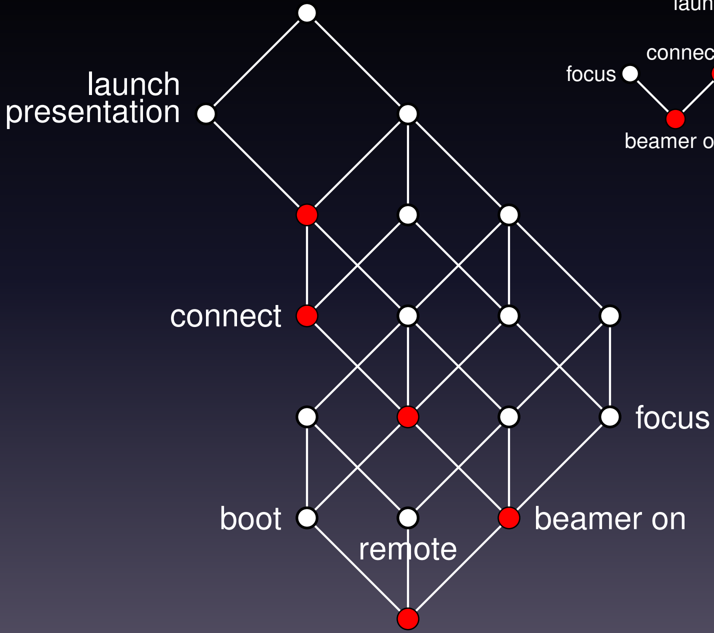


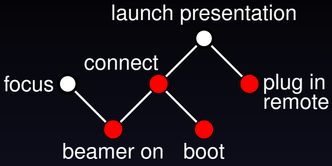
Linear extensions and maximal chains



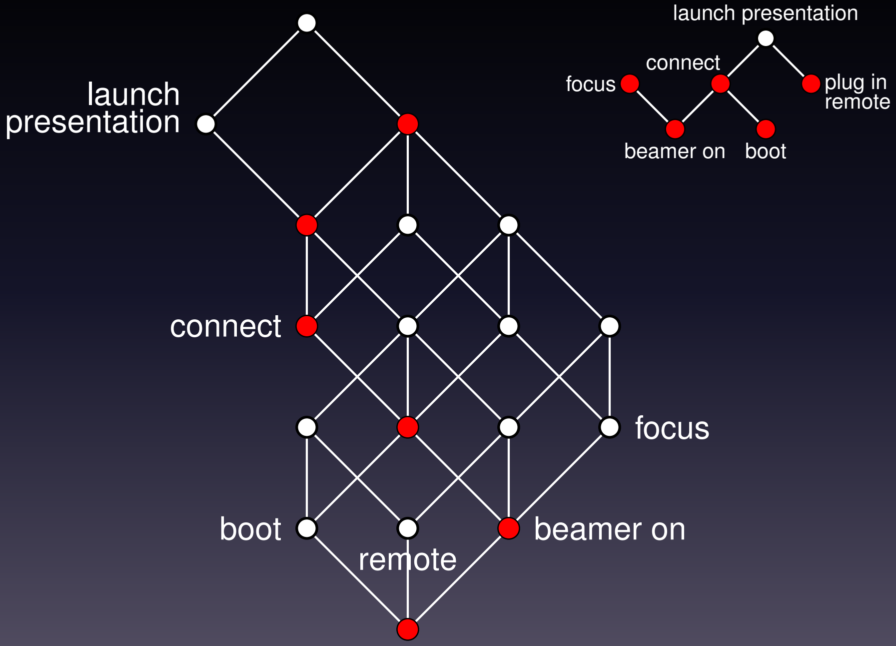


Linear extensions and maximal chains

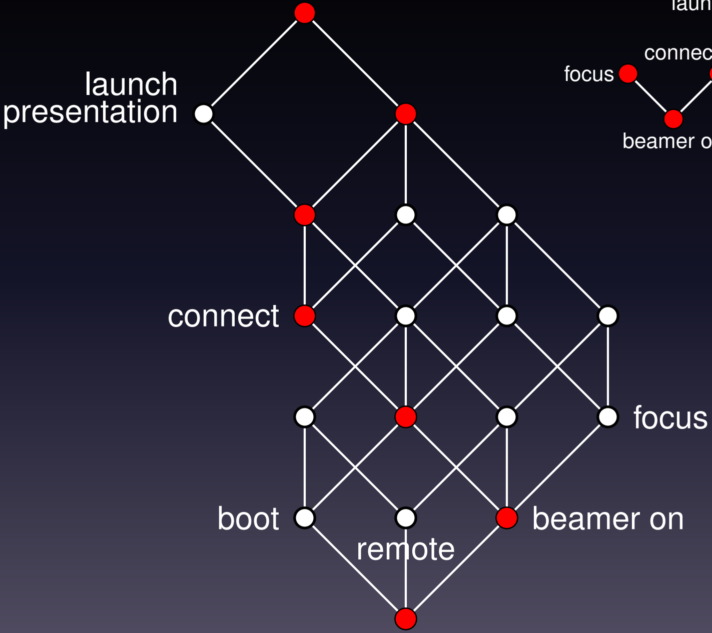


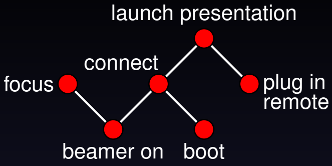


Linear extensions and maximal chains

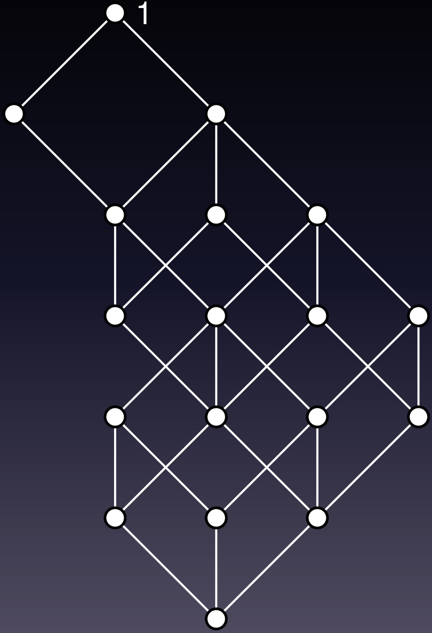


Linear extensions and maximal chains

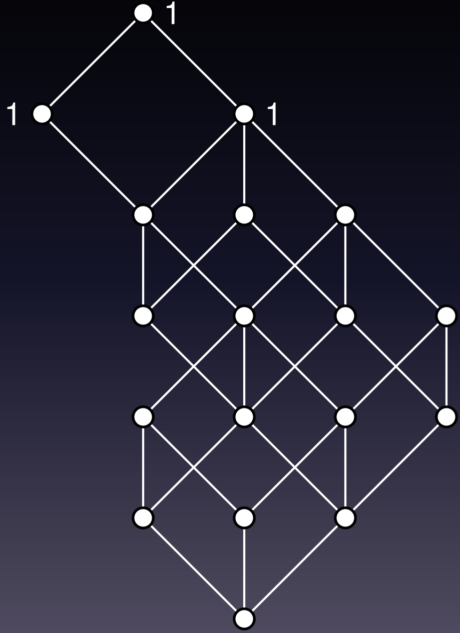




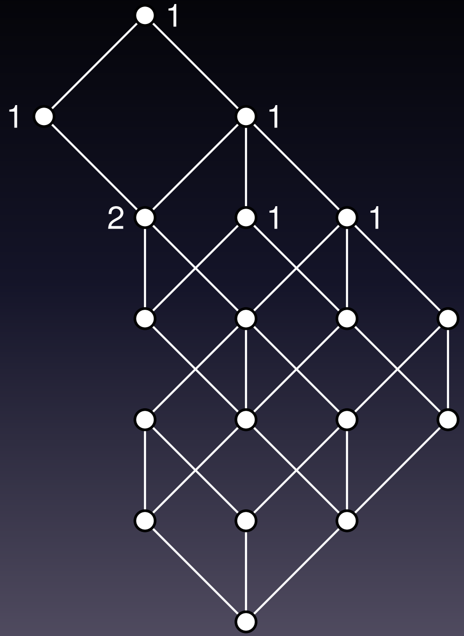
Counting maximal chains



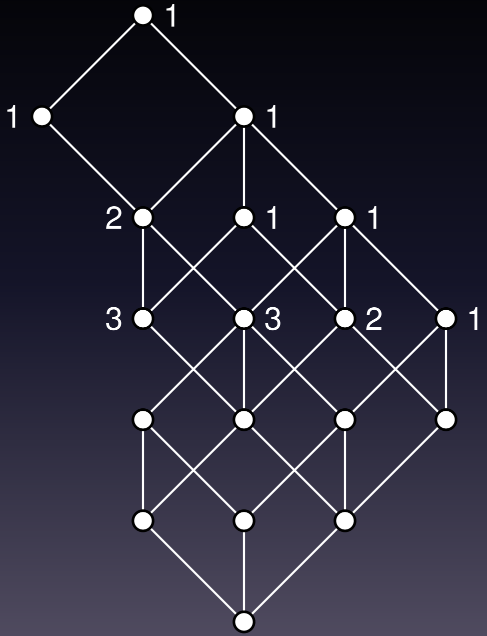
Counting maximal chains



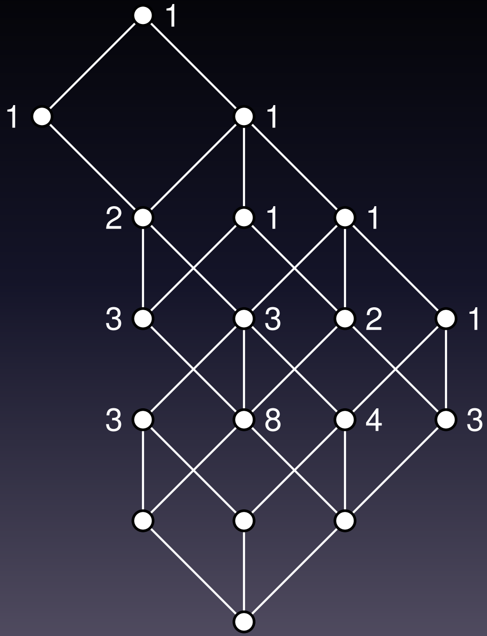
Counting maximal chains



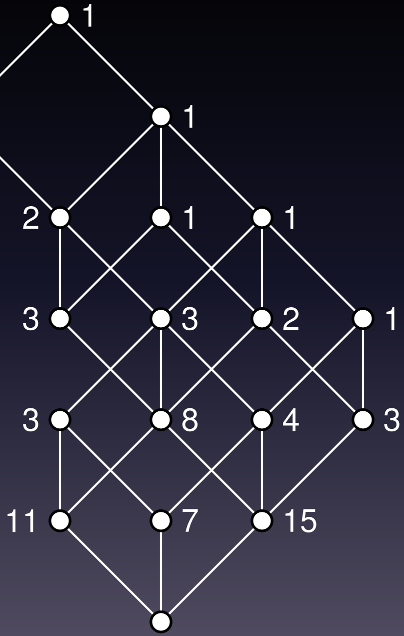
Counting maximal chains



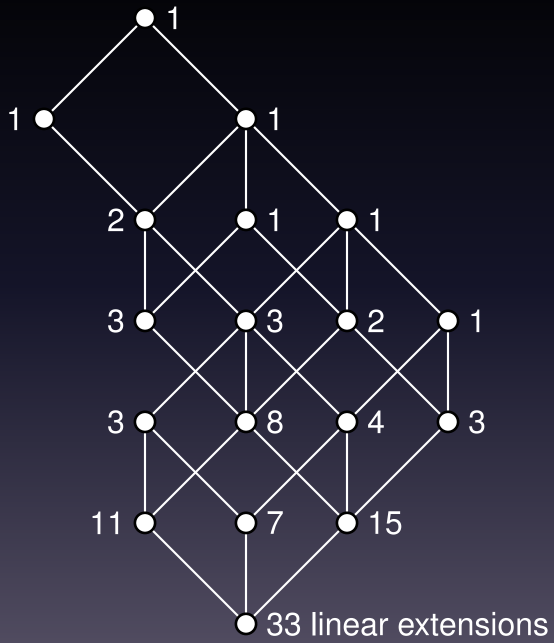
Counting maximal chains



Counting maximal chains



Counting maximal chains



What’s next?

1. An introduction to Formal Concept Analysis
2. Formal concepts for knowledge acquisition
3. Mathematics: combinatorics of ordered sets
4. Formal concepts for skill analysis

Knowledge Spaces

The notion of a knowledge space was introduced by J.C. Falmagne and J.P. Doignon in the 1980es. Knowledge spaces provide a simple formal model for assessment and learning (mainly applied to the learning of school mathematics).

Knowledge Spaces

The notion of a knowledge space was introduced by J.C. Falmagne and J.P. Doignon in the 1980es. Knowledge spaces provide a simple formal model for assessment and learning (mainly applied to the learning of school mathematics).

Though simple, Knowledge Space Theory (KST) is remarkably successful. Its commercial software has many users, and is sometimes even used as a subsitute when mathematics teachers are missing.

Knowledge Spaces

The notion of a knowledge space was introduced by J.C. Falmagne and J.P. Doingnon in the 1980es. Knowledge spaces provide a simple formal model for assessment and learning (mainly applied to the learning of school mathematics).

Though simple, Knowledge Space Theory (KST) is remarkably successful. Its commercial software has many users, and is sometimes even used as a subsitute when mathematics teachers are missing.

A short while ago Doignon and Falmagne have presented a refinement of their approach, called *Learning Spaces*.

Tasks

Doignon and Falmagne focus on the ability of *mastering tasks* (solving problems, answering questions, . . .).

So they fix a set of tasks, and observe how learners do or do not master these tasks.

Tasks

Doignon and Falmagne focus on the ability of *mastering tasks* (solving problems, answering questions, …).

So they fix a set of tasks, and observe how learners do or do not master these tasks. They allow for “lucky guesses and careless errors”, but apart from that it is assumed that a learner usually masters a task if he has the ability to master it.

Tasks

Doignon and Falmagne focus on the ability of *mastering tasks* (solving problems, answering questions, …).

So they fix a set of tasks, and observe how learners do or do not master these tasks. They allow for “lucky guesses and careless errors”, but apart from that it is assumed that a learner usually masters a task if he has the ability to master it.

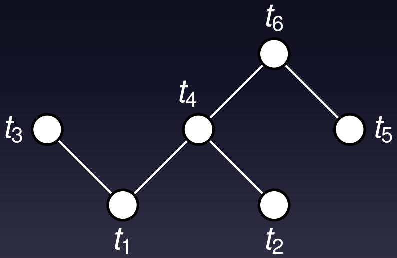
Some tasks are easier to master than others, and a learner who fails for an easy task will lack the ability to master more difficult ones. Or equivalently: a learner who is able to solve a difficult problems is also able to solve all problems which are easier.

An ordered set of tasks

To have a toy example, assume that we consider six tasks t_1, t_2,...,t_6, and assume that t_1 is easier than t_3, t_4 and t_6, t_2 is easier than t_4 and t_6, and that t_4 and t_5 are easier than t_6.

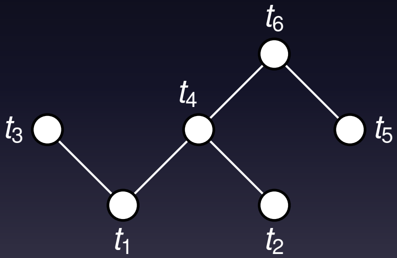
An ordered set of tasks

To have a toy example, assume that we consider six tasks t_1, t_2,...,t_6, and assume that t_1 is easier than t_3, t_4 and t_6, t_2 is easier than t_4 and t_6, and that t_4 and t_5 are easier than t_6. This can be represented as a diagram:



An ordered set of tasks

To have a toy example, assume that we consider six tasks t_1, $t_2, \ldots, t_6$, and assume that t_1 is easier than t_3, t_4 and t_6, t_2 is easier than t_4 and t_6, and that t_4 and t_5 are easier than t_6. This can be represented as a diagram:



The admissible *knowledge states* that a learner can have are precisely the order ideals (downsets) of this ordered set. And the *learning paths* are precisely its linear extensions.

The set of all admissible knowledge states forms the *knowledge space* (over the given set of tasks).

Typical applications of KST include

- determining the knowledge state of a learner efficiently,
- finding the “outer fringe”, meaning the set of tasks which a learner is ready to learn,
- design learning paths etc. from the given difficulty order of the tasks.

There is extensive literature on these questions and successful software (ALEKS).

Competences

Some researchers extend KST by asking *why* a learner is able to master a task. To explain learner behaviour, they introduce abstract *competences*.

In their approach (which is called *Competence based Knowledge Space Theory, CbKST*) learner master a task if and only if they hava a competence that enables them to do so.

Formalisation

Define three formal contexts as follows:

- $(L, Q, \Box)$, where L is the set of *learners*, Q is the set of *questions* (or *tasks*), and

$$l \mathrel{\Box} q :\iff \text{learner } l \text{ masters question } q.$$

- $(L, \mathcal{C}, \circ)$, where $\mathcal{C}$ is the set of competences and

$$l \circ C :\iff \text{learner } l \text{ has competence } C.$$

- $(\mathcal{C}, Q, \models)$ where

$$C \models q :\iff \text{competence } C \text{ enables mastering question } q.$$

Boolean factorisation

The model assumption

$$I \,\Box\, q \iff \exists_C \,(\, I \circ C \text{ and } C \models q \,)$$

can be expressed in the form

$$(L, Q, \Box) = (L, \mathcal{C}, \circ) \cdot (\mathcal{C}, Q, \models),$$

Boolean factorisation

The model assumption

$$l \,\Box\, q \iff \exists_C \,(\, l \circ C \text{ and } C \models q \,)$$

can be expressed in the form

$$(L, Q, \Box) = (L, \mathcal{C}, \circ) \cdot (\mathcal{C}, Q, \models),$$

It is easy to see that this is just the same as writing $(L, Q, \Box)$ as a Boolean matrix product $(L, \mathcal{C}, \circ) \cdot (\mathcal{C}, Q, \models)$.

Boolean factorisation

The model assumption

$$I \,\Box\, q \iff \exists_C \,(\, I \circ C \text{ and } C \models q \,)$$

can be expressed in the form

$$(L, Q, \Box) = (L, \mathcal{C}, \circ) \cdot (\mathcal{C}, Q, \models),$$

It is easy to see that this is just the same as writing $(L, Q, \Box)$ as a Boolean matrix product $(L, \mathcal{C}, \circ) \cdot (\mathcal{C}, Q, \models)$.

The problem of finding Boolean factorisations was studied by many authors. Belohlavek and others have discovered that it cooresponds to the problem of covering a formal context by formal concepts.

No math details, please!

OK, let us avoid the nitty-gritty part.

Finding abstract competences that explain observed learner-question data turns out to be a mathematical problem that can be solved by an algorithm, though not efficiently.

No math details, please!

OK, let us avoid the nitty-gritty part.

Finding abstract competences that explain observed learner-question data turns out to be a mathematical problem that can be solved by an algorithm, though not efficiently.

This is somewhat disappointing for educational theory. What can you do with abstract competences, which mathematicians generate by magic tricks?

No math details, please!

OK, let us avoid the nitty-gritty part.

Finding abstract competences that explain observed learner-question data turns out to be a mathematical problem that can be solved by an algorithm, though not efficiently.

This is somewhat disappointing for educational theory. What can you do with abstract competences, which mathematicians generate by magic tricks?

Fortunately, it is not that bad.
The factorisation is not unique. Unless you impose additional restrictons, there will be very many possible factorisations, and room enough for educational theory.

No math details, please!

OK, let us avoid the nitty-gritty part.

Finding abstract competences that explain observed learner-question data turns out to be a mathematical problem that can be solved by an algorithm, though not efficiently.

This is somewhat disappointing for educational theory. What can you do with abstract competences, which mathematicians generate by magic tricks?

Fortunately, it is not that bad.
The factorisation is not unique. Unless you impose additional restrictons, there will be very many possible factorisations, and room enough for educational theory.

Skills

But more interesting is that the approach can support further theory.

Some authors assume that the competences themselves are composed from simpler units, which they call *skills*. In the simplest version, each competence is just a skill combination.

Skills

But more interesting is that the approach can support further theory.

Some authors assume that the competences themselves are composed from simpler units, which they call *skills*. In the simplest version, each competence is just a skill combination.

Our theory then has to find Boolean factorisations that can be interpretetd by the skill model.

Skills

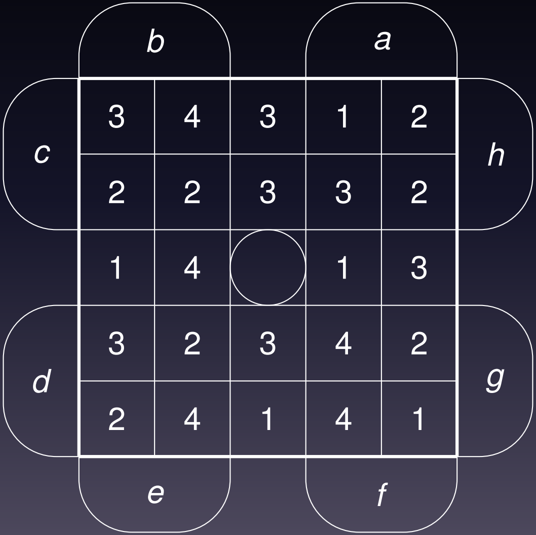
But more interesting is that the approach can support further theory.

Some authors assume that the competences themselves are composed from simpler units, which they call *skills*. In the simplest version, each competence is just a skill combination.

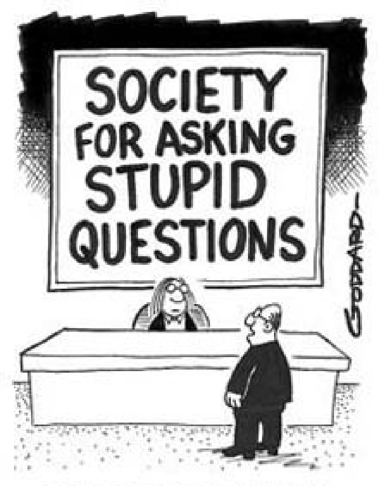
Our theory then has to find Boolean factorisations that can be interpretetd by the skill model.

This is very promising work in progress.

A concluding example



Questions!



"Excuse me, is this the Society for Asking Stupid Questions?"