Lifting Formal Concept Analysis to System-Z and Beyond

Lucas Carr

No Institute Given

Abstract. null. **Keywrods: Non-monotonic reasoning, Formal concept analysis, preferential reasoning**

1 Introduction

In part, implications in formal concept analysis presents a method of determining correspondence between sets of attributes. A formal context describes a Galois connection between sets of attributes and sets of objects - as such, correspondence between attributes equally describes the correspondence between objects which have said attributes. That is, if `wings` $\rightarrow$ `flies` is an accepted implication then all those objects which have `wings` also `fly`.

It is immediately apparent that this kind of reasoning has limitations: *an ostritch has wings but does not fly*. Either the implication must be rejected, or amended to account for the existence of ostritches `wings` and $\neg$`ostritch` $\rightarrow$ `flies` - but what then about penguins? It is not practical to encode every exception to some conditional statement in the statement itself. Equally, discarding the rule entirely is an unappealing prospect since the connection between having wings and flying is of obvious utility.

The rest of this paper is structured as follows: Section 4 introduces the necessary definitions and concepts from FCA, Section 3 then provides an account of preferential and rational consequence relations, as well as a brief exposition of entailment - note Section 3 uses the language of propositional logic when explaining these concepts, while the remainder of the paper uses the attribute logic of FCA. **??** details our approach to implement non-monotonic entailment to FCA.

2 Classical Consequence Relations

In classical logic, the notion of *logical consequence* describes when one formula allows one to conclude another. If $\alpha, \beta \in \mathcal{L}$ are two formulae in the language, then $\alpha \vDash \beta$ iff for every valuation $u \in \mathcal{U}$ if $u \Vdash \alpha$ then $u \Vdash \beta$, equivalently that the models of α are a subset of the models of β. We can similarly define a notion of *classical entailment* which describes when a formula is a logical consequence of set of formulae, or a knowledge-base.

Definition 1. *We say that for two formulae, α, β, in the language $\mathcal{L}$, β is a* logical consequence *of α, written $\alpha \vDash \beta$ iff for every valuation $u \in \mathcal{U}$ where $u \Vdash \alpha$ then also $u \Vdash \beta$.*

Definition 2. *Given a knowledge-base $\mathcal{K}$ and a formula α, the knowledge-base entails α, written $\mathcal{K} \models \alpha$, iff every for every valuation $u \in \mathcal{U}$ where $u \Vdash \mathcal{K}$ it is also the case that $u \Vdash \alpha$.*

We introduce a consequence operator, $\mathcal{C}n$, where $\mathcal{C}n(\mathcal{K})$ describes a closed-set containing everything that can be entailed from $\mathcal{K}$. With this in mind, any notion of consequence which satisfies the condtions below is referred to as a *Tarskian operation*.

1. Monotonocity: *if $\mathcal{K} \subseteq \mathcal{K}'$ then $\mathcal{C}n(\mathcal{K}) \subseteq \mathcal{C}n(\mathcal{K}')$*
2. Idempotence: $\mathcal{C}n(\mathcal{K}) = \mathcal{C}n(\mathcal{C}n(\mathcal{K}))$

3. Inclusion: $\mathcal{K} \subseteq \mathcal{C}n(\mathcal{K})$

Consequence relations are a more removed way of describing some notion of logical consequence. These relations are a (possibly infinite) set of ordered pairs containing formulae in the language, where the first element is the antecedent and the second element is the consequent. A consequence relation might look like $\{(\alpha_0, \beta_0), \ldots, (\alpha_n, \beta_n) \ldots\}$. The relation does not give a semantic or syntactic notion of consequence. Rather, a consequence relation satisfies certain properties which describe some pattern of reasoning - as such, a consequence relation might be a characterisation of some notion of consequence. Obviously, not all sets of pairs would define a meaningful pattern of reasoning. At a minimum, a meaningful consequence relation, Γ, should satisfy

1. Reflexivity: $(\alpha, \alpha) \in \Gamma$
2. Cut: *if* $(\alpha, \beta \wedge \gamma) \in \Gamma$ *and* $(\gamma \wedge \delta, \mu) \in \Gamma$ *then* $(\alpha \wedge \delta, \beta \wedge \mu) \in \Gamma$

3 Non-monotonic Reasoning

The discussion in Section 2 included, but did not pay any attention to the monotonic property of classical logic. In some sense, what monotonicity enforces is that if x leads to y, then x and z should continue to lead to y. More plainly, inferences that are made should never be invalidated by additional information. [3,5] In the language of propositional logic, let `mammal,platypus,laysEggs` be propositions. We might expect to be able to express the following:

$$\begin{aligned} \texttt{mammal} &\rightarrow \neg\texttt{laysEggs} && (1)\\ \texttt{platypus} &\rightarrow \texttt{mammal} && (2)\\ \texttt{platypus} &\rightarrow \texttt{laysEggs} && (3) \end{aligned}$$

The cause for concern is that in any valuation $u \in \mathcal{U}$ where $u \Vdash \texttt{platypus}$, it should also be the case $u \Vdash \texttt{laysEggs}$ and $u \Vdash \texttt{mammal}$ by $(2, 3)$. Then from (1) that $u \Vdash \neg\texttt{laysEggs}$. However, a valuation is a truth assignment to propositions - we cannot have that a proposition be both *true* and *false*. Consequently, there can be no valuations which satisfy all three formulae where `platypus` is *true*.

It is quite clear that any "system" capable of complex reasoning would not be monotonic. There are several distinct approaches which formalise notions of non-monotonic reasoning. However, the objective of this work is to develop a notion of non-monotonicity in formal concept analysis - as such, we restrict ourselves to look at rational consequence, introduced in [3,4,5]. As we progress, the reasons for this choice should become apparent.

3.1 Rational Consequence Relations

Rational consequence

4 Formal Concept Analysis

Formal concept analysis (FCA) is a mathematical framework used to study the relationships between *objects* and their *attributes*. The basic setting is a *formal context*, $\mathbb{K} = (G, M, I)$, where G is a finite set of objects, M is a finite set of attributes, and $I \subseteq G \times M$ an incidence relation describing when an object 'has' an attribute. In reasonably sized instances, a formal context can be represented as a cross table, where rows correspondence to objects, columns to attributes, and a marker $(\times)$ to the incidence relation. [1,2,6]

Two *derivation operators* are defined for sets of objects and sets of attributes, respectively. These operators - denoted with $(\uparrow, \downarrow)$ or simply $(')$ - introduce an order-reversing Galois connection between $\mathcal{P}(G)$ and $\mathcal{P}(M)$.

Definition 3. *In a formal context,* $\mathbb{K} = (G, M, I)$*, for* $A \subseteq G$ *and* $B \subseteq M$

$$A' := \{m \in M \mid \forall g \in A\ (g, m) \in I\} \tag{4}$$
$$B' := \{g \in G \mid \forall m \in B\ (g, m) \in I\} \tag{5}$$

Proposition 1. *Let* $\mathbb{K} = (G, M, I)$ *be a formal context, for* $A_1, A_2, A_3 \subseteq G$*, and* $B \subseteq M$*, we have*

$$A_1 \subseteq A_2 \text{ iff } A_2' \subseteq A_1' \tag{6}$$
$$A_1 \subseteq A_1'' \tag{7}$$
$$A_1 = A_1''' \tag{8}$$
$$(A_1 \cup A_2)' = A_1' \cap A_2' \tag{9}$$
$$A \subseteq B' \Leftrightarrow B \subseteq A' \Leftrightarrow A \times B \subseteq I \tag{10}$$

4.1 Implications

Implications in FCA are a way of describing correspondencies between sets of attributes in a given formal context. Generally, implications are restricted to the language of attribute logic - there is no good intuition for what it might mean to talk about implications between the objects of a formal context.

Definition 4. *Given two sets of attributes* $A, B \subseteq M$*, the implication* $A \to B$ *is* respected *by another set of attributes* $C \subseteq M$ *iff* $A \not\subseteq C$ *or* $B \subseteq C$*. In this case we say* $C \vDash A \to B$*.*

Each object in a formal context has an *object intent* - the set of attributes belonging to that object. This notion is used to define what it means for a formal context to respect an implication.

Definition 5. *Given a formal context* $\mathbb{K} = (G, M, I)$ *and an implication* $A \to B$ *over* M*, the implication is* valid *in* $\mathbb{K}$ *iff for every* $g \in G$*,* $g' \vdash A \to B$*. Then we say* $\mathbb{K} \models A \to B$*. This is equivalent to:*

$$A' \subseteq B' \Leftrightarrow B \subseteq A''$$

At this point the reason for looking into preferential reasoning might be clearer. Implications are analagous to the logical consequence in Definition 1. Then, a formal context respecting an implication continues the analogy, mirroring classical entailment defined in Definition 2. To this end, we view object intents as valuations. We should, however, abandon the intuition of valuations as *possible worlds*. Rather, they represent existing objects.

5 Preferential Reasoning in FCA

How do we get the order? Put this on hold and show that this is not completely non-monotonic. Then we introduce ranking function which IS non-monotonic and gives us an order.

6 Implications and Rankings

It is not immediately obvious how the pairing of non-monotonic reasoning and FCA may manifest. FCA generally involves a starting point of a *formal context*, and discovering implications (*rules*) which “agree” with the context. In a sense, we have a single *model*. In contrast, a more classical approach to non-monotonic reasoning begins with a collection of rules—a knowledge base—which may have many “models”, with the aim of determining what else might follow.

We suggest a, perhaps blunt, approach to introduce non-monotonicity into FCA where the formal context is supplied with an additional collection of (defeasible) rules.

6.1 Implications

Implications in FCA express total correspondence between different sets of attributes. Going forward, whenever an implication is discussed within the setting of FCA, the reader should assume that the antecedent and consequent are only ever sets of attributes.

Definition 6. *In a formal context $\mathbb{K} = (G, M, I)$, an implication $A \to B$ is respected by a set of attributes C iff $A \not\subseteq C$ or $B \subseteq C$, this is denoted $C \Vdash A \to B$. Then, the implication is respected by the formal context, $\mathbb{K} \models A \to B$, iff for all objects $g \in G$ it is the case that $g' \models A \to B$. This latter condition is equivalent to:*

$$\text{(i) } A' \subseteq B' \\ \text{(ii) } B \subseteq A''$$

If object intents are regarded as a kind of interpretation (we will make this notion more concrete later on), then it becomes apparent that FCA-style implications are similar to the notion of *logical consequence*. In fact, they satisfy the requirements to be a characterisation of a Tarskian logical consequence. [3] When a formal context respects an implication, it is really a notion of *entailment*. The formal context encodes all the information we know about the object-level, and implications are discovered relationships which are consistent with this information. As such, all that an implication tells us is that the relationship between attribute sets described by $A \to B$ (with an understanding of the semantics of $\to$) is consistent with the information we have in our formal context. We are interested in a restricted form of these implications, where *vacuous agreement* is done away with.

Our aim is to formalise a conditional relation between sets of attributes which should be interpreted as saying, attributes A *usually* correspond to attributes B - or, that objects with A usually also have B. We denote this conditional by $(B \mid A)$.

6.2 Rankings

Definition 7. *Given a defeasible implication $(B \mid A)$, a set of attributes C* verifies *the implication iff $A \cup B \subseteq C$. A set C is said to falsify the implication iff $A \subseteq C$ and $B \not\subseteq C$.*

Definition 8. *Given a formal context $\mathbb{K} = (G, M, I)$, a set of defeasible implications Δ, and an implication $\delta \in \Delta$, Δ tolerates δ, denoted $T(\delta \mid \Delta)$, iff there exists some $g \in G$ such that g' verifies δ and $g' \models \Delta$. T defines a toleration relation.*

Definition 9. *A set Δ of defeasible implications is* consistent *if, for every non-empty subset $\Delta' \subseteq \Delta$, there is at least one $\delta \in \Delta'$ that is tolerated by Δ'. That is,*

$$\forall \Delta' \subseteq \Delta,\ \exists \delta \in \Delta' \textit{ such that } T(\delta \mid \Delta' \setminus \delta)$$

Pearl

The following may, at times, be entirely plagiarised from [5]. It is for my own understanding.

Consider a set of rules $R = \{r : A_r \to B_r\}$ where A_r and B_r are sets of attributes and $\to$ is the normal attribute implication. In a classical sense, this implication is respected by another set of attributes, Y, in case $A \not\subseteq Y$ or $B \subseteq Y$. A stronger notion is that Y *verifies* $A \to B$ when $A \cup B \subseteq Y$. This is enforcing an intuitive understanding of conditionals, where the antecedent *must* be true - we avoid vacuous truths of implications. [5] Conversely, Y is said to falsify $A \to B$ when $A \subseteq Y$ and $B \not\subseteq Y$.

A new notion of *toleration* is introduced in the form of a *toleration relation*:

Definition 10. *A set of rules $R' \subseteq R$ tolerates an individual rule r, denoted $T(r \mid R')$, if*

$$\bigcup_{r' \in R'} (A'_r \cup B'_r) \cup \{A_r \cup B_r\}$$

is satisfiable.

What it means for an individual rule, r, to be tolerated by a set of rules R' is that there should be a model of R' which verifies r and does not falsify any $r' \in R'$. Shifting into the world of formal concept analysis: an implication, i, is tolerated by a set of implications I if there is an object g such that g' respects I and g' verifies i.

The next notion to be introduced is *consistency*,

Definition 11. *A set R of rules is consistent if in every non-empty subset $R' \subseteq R$ there exists an r' such that R' tolerates r'.*

$$\forall\, R' \subseteq R,\ \exists\, r' \in R',\ \text{such that}\ \ T(r' \mid R' - r') \tag{11}$$

Consistency is stronger than satisfiability: $\alpha \to \beta$ and $\alpha \to \neg\beta$ is satisfiable by $\neg\alpha$, although it is not consistent. Any ω that verifies $\alpha \to \neg\beta$ necessarily falsifies $\alpha \to \beta$ and vice versa. Implicit in the notion of consistency is that implications which are only ever true through negation of the antecedent do not align with our understanding of conditonals. [5]

Consistency gives rise to a natural ordering of the rules in R. Given a consistent R, identify every rule that is tolerated by R, and assign this rule a rank of 0.

6.3 Z-ordering on Formal Contexts

The algorithm to determine the Z-order of a set of implications *(rules)* can be translated to a formal concept analysis setting.

Algorithm 1 Partitioning of G using Defeasible Implications

Require: A formal context $K = (G, M, I)$ and a set of (defeasible) implications Δ
Ensure: A partition on G

$P_0 \leftarrow G$;
$i \leftarrow 0$
while $P_{i-1} \neq P_i$ **do**
  $P_{i+1} \leftarrow \{g \in P_i \mid \exists\delta \in \Delta \text{ s.t. } g' \not\Vdash \delta\}$; *These are the objects with rank $> i$*
  $P_i \leftarrow P_i \setminus P_{i+1}$; *Now P_i contains correct objects*
  $\Delta_i \leftarrow \{(A \to B) \in \Delta \mid \exists g \in P_i \text{ s.t. } A \subseteq g'\}$; *$\Delta_i$ contains the rules dealt with on rank i*
  $\Delta \leftarrow \Delta \setminus \Delta_i$; *Future partitions don't need to satisfy Δ_i*
  $i \leftarrow i + 1$;
if $P_{i-1} = \emptyset$ **then**
  $n \leftarrow i - 1$;
else
  $n \leftarrow i$;
return $(P_0, \ldots, P_n)$

7 Example

For a (perhaps) lengthy example, consider a dataset of animal species and some of their attributes. We obviously have existing expert domain knowledge, which we represent as the set of defeasible rules $\varDelta$ - we could in future investigate including non-defeasible rules as a method of reducing noise. Given the formal context $\mathbb{K}$ and the set of rules $\varDelta$, we use the Object Rank algorithm to determine an order for objects.

$$\varDelta := \begin{cases} \texttt{Aves} & \rightsquigarrow \texttt{Aerial} \\ \texttt{Mammal} & \rightsquigarrow \neg\texttt{Eggs} \\ \texttt{Reptile} & \rightsquigarrow \texttt{Solitary} \\ \texttt{Aquatic} & \rightsquigarrow \texttt{Eggs} \\ \texttt{Aerial} & \rightsquigarrow \texttt{Migratory} \end{cases}$$

	Aves	Mammal	Reptile	Aerial	Aquatic	Terrestrial	Migratory	Solitary	Carnivore	Eggs
Bat		×		×					×	
Crocodile			×		×	×			×	×
Crow	×			×						×
Hippo		×			×	×		×		
Ostrich	×					×		×		×
Penguin	×				×	×	×		×	×
Platypus		×			×	×		×	×	×
Snake			×			×		×	×	×
Swallow	×			×			×	×	×	×
Whale		×			×		×	×	×	

Table 1. Cross-table of $\mathbb{K}$

The algorithm partitions the objects into:

Rank	Objects
3	Platypus
2	Hippo, Whale
1	Bat, Crocodile, Crow, Ostrich, Penguin
0	Snake, Swallow

What if there is a rule $\delta : (a \rightarrow b) \in \varDelta$ such that there is no object $g \in G$ with $a \subseteq g$? - Well, this rule tells us nothing, and we can discard it. If typical objects that have x also have y but there are no objects which have x, then this rule means nothing to us.

What do we do with the the objects that arent

	Republican	Quaker	Pacifist
Nixon	×	×	×
Bush	×		
Fox		×	×

Table 2. Formal Context for Nixon Diamond

$$\varDelta := \begin{cases} \texttt{Republican} & \rightsquigarrow \neg\texttt{Pacifist} \\ \texttt{Quaker} & \rightsquigarrow \texttt{Pacifist} \end{cases}$$

Rank	Objects
1	`Nixon`
0	`Bush, Fox`

References

1. Bernhard Ganter and Sergei Obiedkov. Implications in Triadic Formal Contexts. In Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Dough Tygar, Moshe Y. Vardi, Gerhard Weikum, Karl Erich Wolff, Heather D. Pfeiffer, and Harry S. Delugach, editors, *Conceptual Structures at Work*, volume 3127, pages 186–195. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004. Series Title: Lecture Notes in Computer Science.
2. Bernhard Ganter, Sergei Obiedkov, Sebastian Rudolph, and Gerd Stumme. *Conceptual exploration*. Springer, 2016.
3. Adam Kaliski. An overview of klm-style defeasible entailment. 2020.
4. Daniel Lehmann and Menachem Magidor. What does a conditional knowledge base entail?, 2002.
5. Judea Pearl. System z: a natural ordering of defaults with tractable applications to nonmonotonic reasoning. In *Proceedings of the 3rd Conference on Theoretical Aspects of Reasoning about Knowledge*, TARK '90, page 121–135, San Francisco, CA, USA, 1990. Morgan Kaufmann Publishers Inc.
6. Rudolf Wille and Bernhard Ganter. Formal concept analysis, 1996.