Implementation and Evaluation of KLM-Style Defeasible Entailment and Explanation Algorithms

[A Work in Progress]

Chipo Hamayobe
chipo@cs.uct.ac.za

University of Cape Town
[**Supervisors**: Prof Tommie Meyer and Dr Jonas Haldimann]

Wednesday, 12 February 2025

Overview

1. **Introduction**
   1.1 Propositional Logic
   1.2 Defeasible Reasoning
   1.3 KLM Framework
   1.4 Explanations

2. **Rational Closure (RC)**

3. **Lexicographic Closure (LC)**

4. **Work In Progress**

1 of 4: Introduction

1.1. Propositional Logic

- Logic is an abstraction over natural language
- Composed of objects and their properties
- Logic is a formal language which allows us to reason about the properties of an object
- To specify the language, we use a predefined set of formation rules (syntax rules)
- Syntax rules are logic-dependent, and can therefore vary between logics
- Propositional logic lays the groundwork for many other logics e.g. DL, FOL
- Hence, the logic of choice for this research, as it is simple and expressive enough

1.2. Propositional Logic

- Propositional atoms are the simplest forms for expressing information, and cannot be further decomposed
- $\mathcal{P}$: The finite set of all possible atomic propositions e.g. $\mathcal{P} = \{p, q, r, ...\}$
- Connectives: $\land, \lor, \rightarrow, \leftrightarrow, \neg$
- Recursively combine atoms with connectives to form formulas e.g. $\alpha = (p \land q) \rightarrow r$
- $\mathcal{L}$: The finite set of all formulas (the language) e.g. $\mathcal{L} = \{\alpha, \beta, \gamma, ...\}$

1.3. Defeasible Reasoning

- Non-monotonic reasoning overcomes a shortfall of monotonic reasoning in that previous inferences can be withdrawn when explicit information becomes available
- We consider a specific type of non-monotonic reasoning known as *defeasible reasoning*
- The *KLM Framework* provides a preferential approach to defeasible reasoning
- This framework extends propositional logic, adds the defeasible implication ($\mid\!\sim$)
- A statement $\alpha \mid\!\sim \beta$ is read as *'α typically implies β'*
- A finite set of formulas containing defeasible implications is a defeasible knowledge base $\mathcal{K}$.

Definition (LM-rationalily)

Any defeasible entailment relation $\mid\!\approx$ satisfying the following KLM postulates is referred to as *LM-rational*. A defeasible entailment relation is *LM-rational* if and only if it is defined by a ranked interpretation; hence, it can be defined semantically

- Left Logical Equivalence (LLE)
- Right Weakening (RW)
- Reflexivity (Ref)
- And
- Or
- Cautious Monotonicity (CM)
- Rational Monotonicity (RM)

1.5. Entailment ($\models$)

Definition (Entailment)

For a given knowledge base $\mathcal{K}$ and some propositional formula α with $\mathcal{K} \models \alpha$ if and only if $\mathcal{K} \cup \{\neg\alpha\}$ is unsatisfiable (none of the valuations are `true`).

For defeasible entailment, $\mid\!\approx$, 3 inference operators within the KLM Framework:

1. Rational Closure, $\mid\!\approx_{RC}$
   - If inconsistency occurs in computing entailment, the whole rank is removed from $\mathcal{K}$
   - Concept definition proposed by Lehmann and Magidor
   - Algorithmic definition proposed by Casini, Meyer and Varzinczak
2. Lexicographic Closure, $\mid\!\approx_{LC}$
   - If inconsistency occurs, only remove a single statement instead of the entire rank from $\mathcal{K}$
   - Concept definition proposed by Lehmann
   - Algorithmic definition proposed by Casini, Meyer and Varzinczak
3. Relevant Closure, $\mid\!\approx_{RelC}$
   - Remove statements in lower ranks that disagree with statements in higher ranks
   - Concept definition proposed by Casini, Meyer, Moodley and Nortjé
   - Algorithmic definition proposed by Casini, Meyer, Moodley and Nortjé

1.6. Explanations

Definition (Justification)

Given a knowledge base $\mathcal{K}$ and some formula α with $\mathcal{K} \models \alpha$. $\mathcal{J}$ is said to be a justification for α in $\mathcal{K}$ if $\mathcal{J} \subseteq \mathcal{K}, \mathcal{J} \models \alpha$ and for all $\mathcal{J}' \subset \mathcal{J}$, it is the case $\mathcal{J}' \not\models \alpha$.

Consider the following *defeasible knowledge base*:

$$\mathcal{K} = \left\{ \begin{array}{l} \mathit{birds} \mid\sim \mathit{wings} \\ \mathit{birds} \mid\sim \mathit{fly} \\ \neg(\mathit{penguins} \rightarrow \mathit{birds}) \mid\sim \bot \\ \mathit{penguins} \mid\sim \neg \mathit{fly} \end{array} \right\}$$

- Does $\mathcal{K}$ entail that $\mathit{penguins} \mid\sim \mathit{wings}$?
- We say *YES*
- $\mathcal{J}_1 = \{\mathit{birds} \mid\sim \mathit{wings}, \mathit{penguins} \rightarrow \mathit{birds}\}$
- And there is no subset of $\mathcal{J}$ that entails $\mathit{penguins} \mid\sim \mathit{wings}$?

1.7. Explanations

Definition (Justification)

Given that a knowledge base $\mathcal{K}$ entails a query α, the justification for α is the minimum subset(s) of $\mathcal{K}$ that entails α.

It is possible to have more than one justification for an entailment. Consider the following *defeasible knowledge base*:

$$\mathcal{K} = \left\{ \begin{array}{l} \mathit{birds} \mid\!\sim \mathit{wings} \\ \mathit{birds} \mid\!\sim \mathit{fly} \\ \neg(\mathit{penguins} \rightarrow \mathit{birds}) \mid\!\sim \bot \\ \mathit{penguins} \mid\!\sim \neg \mathit{fly} \\ \mathit{penguins} \mid\!\sim \mathit{wings} \end{array} \right\}$$

- Does $\mathcal{K}$ entail that $\mathit{penguins} \mid\!\sim \mathit{wings}$? We say *YES*
- $\mathcal{J}_1 = \{\mathit{penguins} \mid\!\sim \mathit{wings}\}$, $\mathcal{J}_2 = \{\mathit{birds} \mid\!\sim \mathit{wings}, \mathit{penguins} \rightarrow \mathit{birds}\}$

2 of 4: Rational Closure (RC)

Definition

Given a defeasible knowledge base $\mathcal{K}$ and a query $\alpha \mid\!\sim \beta$ as input, `RationalClosure` returns **true** if and only if $\mathcal{K} \approx_{RC} \alpha \mid\!\sim \beta$, otherwise returns **false**.

Consider the following *defeasible knowledge base*:

$$\mathcal{K} = \left\{ \begin{array}{l} birds \mid\!\sim wings \\ birds \mid\!\sim fly \\ \neg\,(penguins \rightarrow birds) \mid\!\sim \bot \\ penguins \mid\!\sim \neg\, fly \end{array} \right\}$$

`BaseRank` **output for** $\mathcal{K}$:

$\mathcal{R}_\infty$	$\neg\,(penguins \rightarrow birds) \mid\!\sim \bot$
$\mathcal{R}_1$	$penguins \mid\!\sim \neg\, fly$
$\mathcal{R}_0$	$birds \mid\!\sim w,\ birds \mid\!\sim fly$

2.2. Rational Closure ($\approx_{RC}$)

`BaseRank` **output for** $\mathcal{K}$:

$\mathcal{R}_\infty$	$\neg(penguins \rightarrow birds) \mathrel{\mid\!\sim} \bot$
$\mathcal{R}_1$	$penguins \mathrel{\mid\!\sim} \neg fly$
$\mathcal{R}_0$	$birds \mathrel{\mid\!\sim} w,\ birds \mathrel{\mid\!\sim} fly$

- Does $\mathcal{K} \mathrel{|\!\!\approx}_{RC} penguins \mathrel{\mid\!\sim} wings$?
- `RationalClosure` algorithm returns **false**

Reason:

- Is the antecedent of the query consistent with $\mathcal{K}$: $\mathcal{K} \models \neg penguins$
- Returns **true** because $\{penguins \mathrel{\mid\!\sim} \neg fly\}$ and $\{birds \mathrel{\mid\!\sim} fly, penguins \rightarrow birds\}$
- Hence *penguins* is inconsistent with $\mathcal{K}$, and the entire $\mathcal{R}_0$ is discarded
- The new $\mathcal{K} = \{penguins \mathrel{\mid\!\sim} \neg fly, penguins \rightarrow birds\} \not\mathrel{|\!\!\approx}_{RC} penguins \mathrel{\mid\!\sim} wings$?

3 of 4: Lexicographic Closure (LC)

3.1. Lexicographic Closure ($\approx_{LC}$)

Definition

Given a defeasible knowledge base $\mathcal{K}$ and a query $\alpha \mid\!\sim \beta$ as input, `LexicographicClosure` returns **true** if and only if $\mathcal{K} \approx_{LC} \alpha \mid\!\sim \beta$, otherwise **false**.

Consider the following *defeasible knowledge base*:

$$\mathcal{K} = \left\{ \begin{array}{l} birds \mid\!\sim wings \\ birds \mid\!\sim fly \\ \neg\,(penguins \rightarrow birds) \mid\!\sim \bot \\ penguins \mid\!\sim \neg\, fly \end{array} \right\}$$

`BaseRank` **output for** $\mathcal{K}$:

$\mathcal{R}_\infty$	$\neg\,(penguins \rightarrow birds) \mid\!\sim \bot$
$\mathcal{R}_1$	$penguins \mid\!\sim \neg\, fly$
$\mathcal{R}_0$	$birds \mid\!\sim w,\; birds \mid\!\sim fly$

3.2. Lexicographic Closure ($\approx_{LC}$)

`BaseRank` **output for** $\mathcal{K}$:

$\mathcal{R}_\infty$	$\neg(penguins \rightarrow birds) \mid\!\sim \bot$
$\mathcal{R}_1$	$penguins \mid\!\sim \neg fly$
$\mathcal{R}_0$	$birds \mid\!\sim wings,\ birds \mid\!\sim fly$

- Does $\mathcal{K} \approx_{LC} penguins \mid\!\sim wings$?
- The `LexicographicClosure` algorithm returns **true**

Reason:

- Is the antecedent of the query consistent with $\mathcal{K}$: $\mathcal{K} \models \neg penguins$
- Returns **true** because $\{penguins \mid\!\sim \neg fly\}$ and $\{birds \mid\!\sim fly, penguins \rightarrow birds\}$
- Only statements causing inconsistencies are removed from $\mathcal{R}_0$: $\{birds \mid\!\sim fly\}$
- The new $\mathcal{K} = \{birds \mid\!\sim wings, penguins \mid\!\sim \neg fly, penguins \rightarrow birds\} \approx_{LC} penguins \mid\!\sim wings$ with $\mathcal{J}_1 = \{birds \mid\!\sim wings, penguins \rightarrow birds\}$

4 of 4: Work In Progress

4.1. Work In Progress (Next $\sim$3 Months)

- For $\approx_{RC}, \approx_{LC}, \approx_{RelC}$
- Implementation and Testing of Relevant Closure Entailment Algorithm
- Implementation and Testing of Relevant Closure Justification Algorithm
- Implementation of an automated Defeasible Knowledge Base Generator
- Evaluation of the 3 Entailment Algorithms
- Evaluation of the 3 Justification Algorithms

Questions

The End