

Justifications for Defeasible Entailments

Cape-KR 24

Steve Wang

University of Cape Town
Artificial Intelligence Research Unit (AIRU)
Centre for Artificial Intelligence Research (CAIR)

Published Works

Research Conference on Metadata and Semantics Research 2022

Generating Answerable Questions from Ontologies for Educational Exercises

Proceedings of the Joint Ontology Workshops 2022

BFO Classifier: aligning domain ontologies to BFO

Southern African Conference of Artificial Intelligence Research 2022

Defeasible Justification Using the KLM Framework

Classical Reasoning

Classical Knowledge Base

- ▶ Penguins are birds
- ▶ Robins are birds
- ▶ Birds can fly
- ▶ Birds have wings
- ▶ Penguins can not fly

Represented as

$$K = \{ \text{penguin} \rightarrow \text{bird}, \text{robin} \rightarrow \text{bird}, \text{bird} \rightarrow \text{can fly}, \text{bird} \rightarrow \text{has wings}, \text{penguin} \rightarrow \neg \text{can fly} \}$$

Deduction

There are no penguins.

The KLM Framweork

KLM-style Knowledge Base

- ▶ Penguins are birds
- ▶ Robins are birds
- ▶ Birds typically fly
- ▶ Birds typically have wings
- ▶ Penguins typically can not fly

Represented as

$$K = \{ \text{penguin} \rightarrow \text{bird}, \text{robin} \rightarrow \text{bird}, \text{bird} \sim \text{fly}, \text{bird} \sim \text{wings}, \text{penguin} \sim \neg \text{fly} \}$$

Observation

Penguins are birds that cannot fly.

Rational Closure

- ▶ $K = \{ \text{penguin} \rightarrow \text{bird}, \text{bird} \rightarrow \text{animal}, \text{bird} \sim \text{fish}, \text{bird} \sim \text{wing}, \text{penguin} \sim \neg \text{fish} \}$
- ▶ Query: Do penguins typically have wings? $\text{penguin} \sim \text{wing}$

Base Rank of K

| | |
|-----------------|---|
| R_{\boxtimes} |  \rightarrow   \rightarrow  |
| R_1 |  \sim  |
| R_0 |  \sim   \sim  |











Deduction

$K \not\models \text{penguin} \sim \text{wing}$

Rational Closure

- ▶ $K = \{ \text{penguin} \rightarrow \text{bird}, \text{bird} \rightarrow \text{animal}, \text{bird} \sim \text{fish}, \text{bird} \sim \text{wing}, \text{penguin} \sim \neg \text{fish} \}$
- ▶ Query: Do penguins typically have wings? $\text{penguin} \sim \text{wing}$

Base Rank of K

| | |
|-----------------|---|
| R_{\boxtimes} |  \rightarrow   \rightarrow  |
| R_1 |  \sim  |
| R_0 |  \sim   \sim  |

Deduction

$K \not\models \text{penguin} \sim \text{wing}$

Special Penguins

Extended K

- ▶ Penguins are birds
- ▶ Robins are birds
- ▶ **Special penguins are penguins**
- ▶ Birds typically fly
- ▶ Birds typically have wings
- ▶ Penguins typically can not fly
- ▶ **Special penguins typically can fly**

Represented as

$$K = \{ \text{penguin} \rightarrow \text{bird}, \text{robin} \rightarrow \text{bird}, \text{special penguin} \rightarrow \text{penguin}, \text{bird} \sim \text{fly}, \text{bird} \sim \text{wings}, \text{penguin} \sim \text{not fly}, \text{special penguin} \sim \text{fly} \}$$

Can Special Penguins Typically Fly?

$K = \{ \text{penguin} \rightarrow \text{red}, \text{bird} \rightarrow \text{red}, \text{penguin} \rightarrow \text{black}, \text{red} \sim \text{blue}, \text{red} \sim \text{white}, \text{penguin} \sim \text{blue}, \text{penguin} \sim \text{white} \}$

Query

Query: $\text{penguin} \sim \text{blue}$

Base Rank

| | |
|--------------|--|
| R_{∞} | $\text{penguin} \rightarrow \text{red} \quad \text{bird} \rightarrow \text{red} \quad \text{penguin} \rightarrow \text{black}$ |
| R_2 | $\text{penguin} \sim \text{blue}$ |
| R_1 | $\text{penguin} \sim \text{white}$ |
| R_0 | $\text{red} \sim \text{blue} \quad \text{red} \sim \text{white}$ |

Deduction

Special penguins typically fly.

Can Special Penguins Typically Fly?

$K = \{ \text{penguin} \rightarrow \text{red}, \text{bird} \rightarrow \text{red}, \text{penguin} \rightarrow \text{black}, \text{red} \vdash \text{blue}, \text{red} \vdash \text{white}, \text{penguin} \vdash \neg \text{blue}, \text{penguin} \vdash \text{blue} \}$

Query

Query: $\text{penguin} \vdash \text{blue}$

Base Rank

| | |
|-----------------|--|
| R_{\boxtimes} | $\text{penguin} \rightarrow \text{red} \quad \text{bird} \rightarrow \text{red} \quad \text{penguin} \rightarrow \text{black}$ |
| R_2 | $\text{penguin} \vdash \text{blue}$ |
| R_1 | $\text{penguin} \vdash \neg \text{blue}$ |
| R_0 | $\text{red} \vdash \text{blue} \quad \text{red} \vdash \text{white} \quad \text{penguin} \vdash \neg \text{blue}$ |

Deduction

Special penguins typically fly.

Why can Special Penguins Typically Fly?

Defeasible Justification

$$\{ \text{🐧} \sim \text{🦋} \}$$

Invalid Justification

$$\{ \text{🐧} \rightarrow \text{🐧}, \text{🐧} \rightarrow \text{🦋}, \text{🦋} \sim \text{🦋} \}$$

Defeasible Justification Tool

The screenshot shows a software window titled "Defeasible Justification Tool". It has a standard Windows-style title bar with minimize, maximize, and close buttons. The interface is divided into several sections:

- Knowledge Base File (.txt):** A text box containing the path `resources\example\input\SpecialPenguinExample.txt` and a "Select File" button to its right.
- Query:** A text box containing the query `SpecialPenguin~>Fly`.
- Knowledge Base:** A scrollable text area containing the following text:

```
Knowledge Base:  
{ (Bird~>Wings), (Penguin=>Bird), (Robin=>Bird), (Penguin~>!Fly), (SpecialPenguin~>Fly), (SpecialPenguin=>Penguin) }  
Query:  
(SpecialPenguin~>Fly)
```
- Compute Defeasible Justification:** A large button centered below the Knowledge Base section.
- Output:** A scrollable text area containing the following text:

```
Removing rank 0 =====  
{ (Penguin=>Bird), (Robin=>Bird), (Penguin~>!Fly), (SpecialPenguin~>Fly), (SpecialPenguin=>Penguin) }  
Removing rank 1 =====  
{ (Penguin=>Bird), (Robin=>Bird), (SpecialPenguin~>Fly), (SpecialPenguin=>Penguin) }  
<<Final Justification>>  
(SpecialPenguin~>Fly)
```
- Exit:** A button located at the bottom right of the window.

Conclusions & Future work

Conclusion

- ▶ We present an algorithm that computes justifications for defeasible entailment using the KLM framework.
- ▶ Some justifications may support classical entailment but not defeasible entailment.
- ▶ We built a software tool that computes justification for defeasible entailment.

Future work

- ▶ Build defeasible justification tool for Description Logic
- ▶ Further investigation in explainable AI