Scalable Defeasible Reasoning

Knowledge-Representation using Logic

Propositional logic can be used to represent information (a knowledge base), e.g., the statement:

"penguins are birds"

Can be represented as: $p \rightarrow b$

We can reason about information represented in this way, e.g., if we also have the statement **"birds fly" ($b \rightarrow f$),** then we can conclude that penguins fly, i.e. **($p \rightarrow f$)**

Defeasible Reasoning

If our knowledge base also contains "penguins don't fly", this would be a contradiction, meaning penguins can't exist.

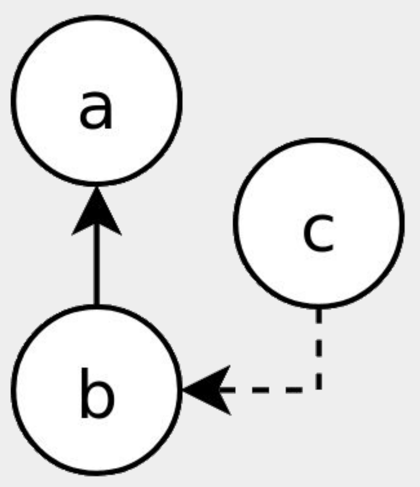
No contradiction would exist if the second statement added was instead "birds typically fly", i.e.,

$$b \mid\!\sim f$$

Reasoning about knowledge bases containing statements of this form is known as **"defeasible reasoning" (e.g., Rational Closure and Lexicographic Closure).**

Knowledge Base Generation

A subset of ranked knowledge bases adhering to parameterized structures can be represented using graphs.



0	$a \mid\!\sim d$
1	$b \mid\!\sim \neg d$
	$c \mid\!\sim b$
∞	$b \rightarrow a$

$\{a \mid\!\sim d, b \mid\!\sim \neg d, c \mid\!\sim b, b \rightarrow a\}$

Constructing these graphs allows for parameterized generation.

Key aims for Knowledge Base Generation

- Create a program to generate ranked knowledge bases
- Identify if these parameters are useful

Generation Parameters

- Number of Defeasible Ranks
- Number of Defeasible Statements
- Distribution of Defeasible Statements
- Defeasible Only Generation

Rational Closure

1. Given a knowledge base, put statements in ranks depending on how general they are. (Higher up = more general)
2. To check if a knowledge base entails "($p \mid\!\sim f$)", remove ranks one by one from the top down until its possible for p to be true in our world.
3. Return true if the statements in the remaining ranks mean that penguins are always birds.

Key aims for Rational Closure

- Create a program to compute Rational Closure.
- Test different optimisations to obtain a version of the algorithm which performs better.

Optimisations

- Use **Binary Search** to find which rank we need to remove ranks up to.
- **Store rank numbers** where certain premises become consistent with our knowledge base.

Lexicographic Closure

Unlike Rational Closure, it is not necessary to remove every statement in a rank. Lexicographic Closure removes as few statements as possible.

Key aims for Lexicographic Closure

- Create a program to compute Lexicographic Closure
- Test different optimisations to obtain a version of the algorithm which performs better.

Optimisations

- **Power Set**: The feature of power set was used to decrease the number of entailment checks required
- **Binary Search**: This optimisation is used to speed up the process of finding the rank to remove.
- **Ternary Search**: This optimisation is an improved version of Binary Search by using two keys.

Conclusions

All knowledge base generation parameters lead to various discoveries and were thus were deemed useful in one way or another.

For Rational Closure, the binary search optimisation yielded impressive results for cases where the premises of our defeasible queries become entailed at a high rank, and storing ranks yielded great results for sets of statements which mostly have the same premise.

The Lexicographic Closure implementation using Ternary Search performed faster than the other two optimisations.

See the respective project reports for more details.

Future Work

Investigate non-deterministic, pseudo-random knowledge base generation. Investigate ways to produce knowledge bases with more compounded statements using the aforementioned graphs.

For Rational Closure, further investigation should be done which focuses on the characteristics of the queries themselves, as opposed to knowledge base structure.

The Lexicographic Closure implementation using n-ary search, where n will be dependent on the number of ranks should be investigated.

Joel Hamilton, Daniel Park & Aidan Bailey
Supervised by Professor Tommie Meyer

Department of Computer Science, University of Cape Town