Notes on Default Reasoning

Stuart C. Shapiro

November 30, 2004

1 Introduction

These notes comment on, and are, in part, derived from Brachman & Levesque, *Knowledge Representation and Reasoning*, Chapter 11.

This chapter goes beyond the previous two, to discuss *default* (or *defeasible*) reasoning, in a general logical framework. (Note that there is a large literature on defeasible reasoning that is not mentioned in this chapter.)

One issue is the difference between generics and true universals.

A theme is *nonmonotonic* reasoning. Standard logic is **monotonic**:

$$\begin{array}{ll} \text{If} & \mathcal{A} \vdash \phi \\ \text{then} & \mathcal{A} \cup \mathcal{B} \vdash \phi \end{array}$$

But default reasoning is not:

1. Birds fly.
2. Penguins are birds.
3. Chilly is a penguin.
4. Does Chilly fly?
5. BTW, peguins don't fly.
6. Does Chilly fly?

(Aside: Last semester, we had a discussion about the meaning of *but*.
Notice: *Penguins are birds, but they don't fly.*)

2 Closed-World Reasoning

Incorporate the Closed-World Assumption (CWA) into the standard approach by defining

$\mathrm{KB} \models_C \alpha$ iff $\mathrm{KB}^+ \models \alpha$, where
$\mathrm{KB}^+ = \mathrm{KB} \cup \{\neg p \mid p \text{ is atomic and } \mathrm{KB} \not\models p\}$

Problem: if $\mathrm{KB} = \{(p \vee q)\}$, then KB^+ contains $\{(p \vee q), \neg p, \neg q\}$, and is inconsistent.
Solution: generalized closed-world assumption (GCWA)

$\mathrm{KB} \models_{GC} \alpha$ iff $\mathrm{KB}^* \models \alpha$, where
$$\mathrm{KB}^* = \mathrm{KB} \cup \{\neg p \mid \text{for all collections of atoms } p, q_1, \ldots, q_n, \text{ if } \mathrm{KB} \models (p \vee q_1 \vee \ldots \vee q_n), \text{ then } \mathrm{KB} \models (q_1 \vee \ldots \vee q_n)\}$$

So, if $\mathrm{KB} = \{(p \vee q)\}$, then $\neg p \notin \mathrm{KB}^*$ and $\neg q \notin \mathrm{KB}^*$, but $\neg r \in \mathrm{KB}^*$.
Example: In Clue, Tom is asked if he has Miss Scarlett, the Kitchen, or the knife, and shows one card. Later, he's asked if he has Prof. Plum, the Kitchen, or the knife, and shows one card. However, you have Prof. Plum. GCWA allows you to conclude, by default, that Tom doesn't have Miss Scarlett.

Domain Closure: What does $\{\neg p \mid p \text{ is atomic}\}$ include? Form all atomic sentences by giving every predicate symbol every possible combination of arguments. It's a finite collection as long as there's no function symbol. (The text says even then, but I don't think so.) Consider a KB with: no quantified formulas; one unary predicate, P; one binary predicate, R; two individual constants, a, b; one individual constant, c, such that KB has $P(c)$, but c doesn't occur as either argument of R.

$\{\neg p \mid p \text{ is atomic}\}$, and therefore KB$^+$, includes

$$\{\neg R(a,c), \neg R(c,a), \neg R(b,c), \neg R(c,b), \neg R(c,c)\}$$

However, $\mathrm{KB}^+ \not\models \neg\exists x R(x,c)$ if you assume there could be some individual other than a, b, or c, i.e. an *unnamed* individual.

This raises the issue of how semantics is done:

1. (The way I like): $[\![\exists x R(x,c)]\!]$ is True if there is some ground term, t such that $[\![R(t,c)]\!]$ is True; otherwise, it is False. By this, $\mathrm{KB}^+ \models \neg\exists x R(x,c)$.

2. (The usual way): $[\![\exists x R(x,c)]\!]$ is True if there is some individual i in the domain such that $\langle i, [\![c]\!]\rangle \in [\![R]\!]$; otherwise, it is False. By this, there might be no ground term denoting i, and $\mathrm{KB}^+ \not\models \neg\exists x R(x,c)$.

Closed-world assumption with domain closure uses (2), but uses:

KB $\models_{CD} \alpha$ iff KB$^\diamond \models \alpha$, where
KB$^\diamond$ = KB$^+ \cup \{\forall x[x = c_1 \vee \ldots \vee c_n]\}$
where $c_1, \ldots, c_n$ are all the individual constants in KB.

So $\mathrm{KB}^\diamond \models \neg\exists x R(x,c)$, even using semantics version (2). Moreover,

KB $\models_{CD} \forall x\alpha$ iff KB $\models_{CD} \alpha\{c/x\}$, for every c appearing in KB;
KB $\models_{CD} \exists x\alpha$ iff KB $\models_{CD} \alpha\{c/x\}$, for some c appearing in KB;

Notice that the domain closure axiom means that every individual in the domain has an *individual constant* that denotes it. In fact, if the KB includes function symbols, then, by the domain closure axiom, every functional term is also equal to some individual constant. This is more restrictive than what I use for semantics, which is just the assumption that every individual is denoted by some ground term.

The text's ultimate KB for CWA is KB$^\diamond$ with the further addition of the *unique name assumption*, that $c_1 \neq c_2$ for every pair of distinct individual constants, c_1 and c_2.

With this version of KB$^\diamond$, KB $\models_{CD} \alpha$ or KB $\models_{CD} \neg\alpha$ for any formula, α.

Warning: Still, I may need to modify my approach, and KB$^\diamond$ may be inconsistent. Consider KB = $P(a) \wedge (\forall x \neg R(x,x)) \wedge (\exists x R(x,a))$. This is only consistent if there is an "unnamed" individual—one other than $[\![a]\!]$.

In fact, one can require an infinite number of individuals with

$$N(a) \wedge (\forall x B(x, f(x))) \wedge (\forall x, y, z (B(x,y) \wedge B(y,z) \Rightarrow B(x,z))) \wedge (\forall x, y B(x,y) \Leftrightarrow \neg B(y,x))$$

and this would make the domain closure axiom inconsistent. The text says

> But these examples are somewhat far-fetched; they look more like formulas that might appear in axiomatizations of set theory than in *databases*. For "normal" applications, domain closure is much less of a problem. [B&L, p. 215, italics added]

But what about KBs that include $\forall x[Duck(motherOf(x)) \Rightarrow Duck(x)]$ or $\forall p[Believe(p) \Rightarrow Believe(Believe(p))]$? These require an infinite number of individuals, and seem useful for commonsense reasoning.
[Modification of my approach: Add constants and function symbols as would be required when Skolemizing all formulas in the KB.**]**

3 Circumscription

3.1 Idea 1: Circumscribing Predicates

Can view CWA as minimizing the extension of all predicates. I.e. $P(a)$ is True only where it has to be; for all other individuals, b, $\neg P(b)$.

In circumscription, we minimize (circumscribe) the extension of selected predicates.

3.2 Idea 2: Ab

We can't say $\forall x[Bird(x) \Rightarrow Flies(x)]$, because there are birds that don't fly for various reasons: they're penguins, they're emus, they're immature, they're dead, etc. We could think of doing

$$\forall x[Bird(x) \wedge \neg Penguin(x) \wedge \neg Emu(x) \wedge \neg Immature(x) \wedge \neg Dead(x) \Rightarrow Flies(x)]$$

but whenever we come upon a new exceptional class, we'd have to delete this rule, and replace it with a more inclusive one. Instead, once and for all state

$$\forall x[Bird(x) \wedge \neg Ab_{Flies}(x) \Rightarrow Flies(x)]$$

and then for each exceptional case, add

$\forall x Penguin(x) \Rightarrow Ab_{Flies}(x)$
$\forall x Emu(x) \Rightarrow Ab_{Flies}(x)$
$\forall x Immature(x) \Rightarrow Ab_{Flies}(x)$
$\forall x Dead(x) \Rightarrow Ab_{Flies}(x)$
etc.

3.3 Together

So say we've done that, and have

$$Bird(chilly), Bird(tweety), (tweety \neq chilly), Pengin(chilly)$$

Can we conclude Flies(tweety)? No! Because can't infer $\neg Ab_{Flies}(tweety)$.

So circumscribe Ab_{Flies}.

3.4 Minimal Entailment

$\mathrm{KB} \models_{\leq} \alpha$ iff every interpretation (situation) that satisfies KB and minimizes the extension of each Ab predicate also satisfies α.

Above KB $\models_{\leq}$ $Flies(tweety)$ because every interpretation that satisfies the KB and minimizes Ab_{Flies} also satisfies $\neg Ab_{Flies}(tweety)$ and therefore also satisfies $Flies(tweety)$.

3.5 Problems

Sections 11.3.2 and 11.3.3 of the text suggest additional modifications of circumscription. Unfortunately, it all ends with "there is a serious limitation in using circumscription for default reasoning" [B & L, p. 222].

4 Default Logic

The general problem we're considering is that generic statements may have exceptions—they're not actually strict universals. The Default Logic approach is to represent strict universals in the usual Predicate Logic way, but to represent generic statements as a special kind of *default rule.*

A default rule is of the form $\frac{\alpha:\beta}{\delta}$, and may be glossed as "If α is true and β is consistent, then conclude δ."

Default rules are not sentences of the object language. That is, a default rule cannot be a subexpression of another expression. A default rule, instead, is like a rule of inference of Predicate Logic.

A KB using default logic has two parts

1. A set $\mathcal{F}$ of sentences of Predicate Logic.

2. A set $\mathcal{D}$ of default rules.

If you view $\mathcal{F}$ as a set of beliefs that can increase as reasoning is performed, then you can add to $\mathcal{F}$ any belief that is logically implied (according to the usual rules of inference) by the beliefs already in $\mathcal{F}$, and in addition, any belief that is justified by any of the default rules in $\mathcal{D}$. More formally, a set of beliefs implied by a default theory KB = $(\mathcal{F}, \mathcal{D})$ is an *extension* $\mathcal{E}$ such that

$$\pi \in \mathcal{E} \text{ iff } \mathcal{F} \cup \{\delta \mid \frac{\alpha : \beta}{\delta} \in \mathcal{D}, \alpha \in \mathcal{E}, \neg\beta \notin \mathcal{E}\} \models \pi$$

Example:

$\mathcal{F} = \{Bird(Tweety), Penguin(Chilly),$
$\quad \forall x[Penguin(x) \Rightarrow Bird(x)], \forall x[Penguin(x) \Rightarrow \neg Flies(x)]\}$

$\mathcal{D} = \{\frac{Bird(x):Flies(x)}{Flies(x)}\}$

$Flies(Tweety)$ may be inferred, but not $Flies(Chilly)$.

Although, in general, in default rules, $\frac{\alpha:\beta}{\delta}$, β need have no syntactic relation to δ, the most useful default rules are *normal default rules,* where β is the same as δ. Normal default rules also avoid the problematic $\frac{True:p}{\neg p}$.

What if we allow for flying penguins? Such a default KB might look like:

$\mathcal{F} = \{Bird(Tweety), Penguin(Chilly), \forall x[Penguin(x) \Rightarrow Bird(x)]\}$

$\mathcal{D} = \{\frac{Bird(x):Flies(x)}{Flies(x)}, \frac{Penguin(x):\neg Flies(x)}{\neg Flies(x)}\}$

We may conclude that Chilly flies or not, depending on which default rule is used (first). Thus, we are again in the realm of *multiple extensions,* and can consider credulous reasoners *vs.* skeptical reasoners.

We could try to patch the first default rule, and replace it with $\frac{Bird(x):Flies(x)\wedge\neg Penguin(x)}{Flies(x)}$ Now we are back to the problem that circumscription was designed to deal with, so how about $\frac{Bird(x):Flies(x)\wedge\neg Ab_{Flies}(x)}{Flies(x)}$ and $\frac{True:\neg Ab_{Flies}(x)}{\neg Ab_{Flies}(x)}$? Would you then add $\frac{Penguin(x):Ab_{Flies}(x)}{Ab_{Flies}(x)}$?

5 Autoepistemic Logic

Put default rules into the object language by introducing a modal, or higher-level, operator $\mathbf{B}$, where the meaning of $\mathbf{B}\alpha$ is "α is believed." Then the default flying rule becomes $\forall x[Bird(x) \wedge \neg\mathbf{B}\neg Flies(x) \Rightarrow Flies(x)]$.

Now a set of beliefs justified by an autoepistemic KB is a *stable extension*, $\mathcal{E}$ such that

$$\pi \in \mathcal{E} \text{ iff } \mathrm{KB} \cup \{\mathbf{B}\alpha \mid \alpha \in \mathcal{E}\} \cup \{\neg\mathbf{B}\alpha \mid \alpha \notin \mathcal{E}\} \models \pi$$

$\mathcal{E}$ has three significant properties:

1. Closure under entailment: if $\mathcal{E} \models \alpha$ then $\alpha \in \mathcal{E}$.
2. Positive introspection: if $\alpha \in \mathcal{E}$ then $\mathcal{B}\alpha \in \mathcal{E}$
3. Negative introspection: if $\alpha \notin \mathcal{E}$ then $\neg\mathcal{B}\alpha \in \mathcal{E}$

Question: Are these cognitively reasonable?

6 Conclusion

B&L conclude, "Getting a logical account of default reasoning that is simple, broadly applicable, and intuitively correct remains an open problem...it is perhaps *the* open problem in the whole area of knowledge representation." [p. 232–3]