Forgetting and Projection

Revision: March 3, 2017; Rendered: May 25, 2021

Definitions of projection, literal forgetting, literal projection, and approximate versions of the latter two. Formalized with the *PIE* system.

1 Literal Forgetting

---

forglit$(P\text{-}\mathsf{p}, F)$

Defined as

$$\exists Q\,(G \land \forall X\,(P_X \to Q_X)),$$

where

$$\begin{aligned}
&G := F[P \mapsto Q],\\
&N := \text{arity of } P \text{ in } F,\\
&X := x_1, \ldots, x_N,\\
&Q_X := Q(X),\\
&P_X := P(X).
\end{aligned}$$

---

forglit$(P\text{-}\mathsf{n}, F)$

Defined as

$$\exists Q\,(G \land \forall X\,(Q_X \to P_X)),$$

where

$$\begin{aligned}
&G := F[P \mapsto Q],\\
&N := \text{arity of } P \text{ in } F,\\
&X := x_1, \ldots, x_N,\\
&Q_X := Q(X),\\
&P_X := P(X).
\end{aligned}$$

$\mathit{forglit}(P\text{-}\mathsf{pn}, F)$

Defined as

$$\exists P\, F.$$

$\mathit{forglit}([P|Ps], F)$

Defined as

$$\mathit{forglit}(P, \mathit{forglit}(Ps, F)).$$

$\mathit{forglit}([], F)$

Defined as

$$F.$$

1.1 Literal Forgetting: Examples

$\mathit{ex\_basic}$

Defined as

$$\forall x\, (\mathsf{a}x \rightarrow \mathsf{p}x) \land \forall x\, (\mathsf{p}x \rightarrow \mathsf{b}x).$$

Input: $\mathit{forglit}([\mathsf{p}\text{-}\mathsf{p}], \mathit{ex\_basic})$.
Result of elimination:

$$\forall x\, (\mathsf{b}x \lor (\neg \mathsf{a}x \land \neg \mathsf{p}x)).$$

Input: $\mathit{forglit}([\mathsf{p}\text{-}\mathsf{n}], \mathit{ex\_basic})$.
Result of elimination:

$$\forall x\, (\mathsf{a}x \rightarrow \mathsf{b}x \land \mathsf{p}x).$$

Input: $\mathit{forglit}([\mathsf{p}\text{-}\mathsf{p}, \mathsf{p}\text{-}\mathsf{n}], \mathit{ex\_basic})$.
Result of elimination:

$$\forall x\, (\mathsf{a}x \rightarrow \mathsf{b}x).$$

2 Projection

$proj(S, F)$

Defined as

$$\exists S_1\, F,$$

where

$$\begin{aligned} S_2 &:= \mathsf{free\_predicates}(F), \\ S_1 &:= S_2 \setminus S. \end{aligned}$$

$projlit(S, F)$

Defined as

$$forglit(S_1, F),$$

where

$$\begin{aligned} &S_2 := S \text{ (in different representation)}, \\ &S_3 := \mathsf{free\_predicates}(F) \text{ in scope representation}, \\ &S_4 := S_3 \setminus S_2, \\ &S_5 := S_4 \text{ closed under duals}, \\ &S_6 := S_5 \setminus S_2, \\ &\text{scse\_to\_scsp(S6, S1)}. \end{aligned}$$

Here we subtract, add duals and subtract again to avoid *literal* forgetting induced by occurences in the formula in just a specific polarity. Semantically we could just subtract as realized in the following version:

$projlit_s(S, F)$

Defined as

$$forglit(S_1, F),$$

where

$$\begin{aligned} &S_2 := S \text{ (in different representation)}, \\ &S_3 := \mathsf{free\_predicates}(F) \text{ in scope representation}, \\ &S_4 := S_3 \setminus S_2, \\ &\text{scse\_to\_scsp(S4, S1)}. \end{aligned}$$

3 Approximate Version of Literal Forgetting

Existentially quantifying upon all occurrences with specified polarity yields a possibly weaker formula than literal forgetting that might be simpler to process (see application in scratch_definientia). Also a version of projection, based on the weakened forgetting is specified.

$\mathit{lemma\_projlit}(S, F)$

Defined as

$$\mathit{lemma\_forglit}(S_1, F),$$

where

$S_2 := S$ (in different representation),
$S_3 :=$ $\mathsf{free\_predicates}(F)$ in scope representation,
$S_4 := S_3 \setminus S_2$,
scse_to_scsp(S4, S1).

$\mathit{lemma\_forglit}(P\text{-}\mathsf{p}, F)$

Defined as

$$\exists Q\, G,$$

where

$$G := F[P\text{-}\mathsf{p} \mapsto Q].$$

$\mathit{lemma\_forglit}(P\text{-}\mathsf{n}, F)$

Defined as

$$\exists Q\, G,$$

where

$$G := F[P\text{-}\mathsf{n} \mapsto Q].$$

$\mathit{lemma\_forglit}(P\text{-}\mathsf{pn}, F)$

Defined as

$$\exists P\, F.$$

$lemma\_forglit([P|Ps], F)$

Defined as

$$lemma\_forglit(P, lemma\_forglit(Ps, F)).$$

$lemma\_forglit([], F)$

Defined as

$$F.$$

Index