知識工学（第 8 回）

二宮 崇（ninomiya@cs.ehime-u.ac.jp）

一階述語論理 (9)

§9.5 融合法

これまで紹介した一階述語論理の推論方法は必ずしもどんな場合にでも証明できる方法ではなかった。命題論理で用いた融合法は一階述語論理においても適用することができ、融合法を用いれば、伴意関係にある一階述語論理の論理式は必ず証明できる。

融合法の手順は命題論理のときと同じである。融合法はその基本戦略として背理法を用いる。知識ベースをKBとし、質問をαとしたとき、$KB \land \neg\alpha$が充足不能であることを示すことで、$KB \vDash \alpha$を示す。

1. $KB \land \neg\alpha$を連言標準形(conjunctive normal form: CNF)に変換する。連言標準形は次の形をした文である。

$$(l_{1,1} \lor l_{1,2} \lor \cdots) \land (l_{2,1} \lor l_{2,2} \lor \cdots) \land \cdots \land (l_{n,1} \lor l_{n,2} \lor \cdots)$$

ただし、$l_{i,j}$はリテラルと呼ばれ、リテラルは原子文または原子文の否定のいずれかである。原子文は正リテラル、原子文の否定は負リテラルとも呼ばれる。$(l_{i,1} \lor l_{i,2} \lor \cdots)$は節と呼ばれる。$KB \land \neg\alpha$をCNF に変換した結果の節集合を$KB'$とする。

2. 節集合KB' (知識ベース+$\neg\alpha$) に対し融合規則を適用し、$KB' \vDash R$となるRをみつけRを節集合KB'に追加する(KB'は更新される)。この操作(推論)を$KB' \vdash R$と書くことにする。

3. 追加できる新しい節がない場合、KBからαを伴意しないことがわかる

4. 偽$(= 0)$が得られたら、KBからαを伴意することがわかる

5. 2.に戻る

しかし、3 の追加できる新しい節が無い場合は伴意しないことがわかるが、このアルゴリズムは必ずしも停止するとは限らない。その場合は伴意するかどうかはわからない、ということになる。し

かし、元々の知識ベースと質問の間に伴意関係が成り立っているのなら、このアルゴリズムは必ず停止する。

§9.5.1 連言標準形(CNF)への変形

最初のステップである CNF への変換について説明する。任意の一階述語論理の論理式は CNF に変換可能である。

CNF への変換手続きは次のようになる。命題論理の場合と違いがある箇所は赤字で示している。

1. $\alpha \Leftrightarrow \beta$を$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$に置き換えることで、⇔を除去する

2. $\alpha \Rightarrow \beta$を$\neg\alpha \vee \beta$に置き換えることで⇒を除去する

3. 次の論理的同値関係を使うことで否定(¬)を内側へいれていく

$\neg\neg\alpha \equiv \alpha$ (二重否定除去)

$\neg(\alpha \wedge \beta) \equiv \neg\alpha \vee \neg\beta$ (ド・モルガンの法則)

$\neg(\alpha \vee \beta) \equiv \neg\alpha \wedge \neg\beta$ (ド・モルガンの法則)

$\neg\exists x\, P \ \equiv\ \forall x \neg P$

$\neg\forall x\, P \ \equiv\ \exists x \neg P$

4. 変数標準化: $(\forall x\, P(x)) \vee (\exists x\, P(x))$のような同じ変数を 2 回使用する文に対して、一方の変数の名称を改名する。例えば、$(\forall x\, P(x)) \vee (\exists x\, P(x))$は$(\forall x\, P(x)) \vee (\exists y\, P(y))$とすればよい。

5. スコーレム化(存在限量子の除去): 次に存在限量子の除去を行う。簡単な場合には存在具体化(EI)と同じである。例えば、$\exists x\, P(x)$は$P(A)$と変換すればよい。ただし、Aは新しい定数である。しかし、例えば、$\forall x \exists y\, Love(x, y)$に対し、$\forall x\, Love(x, A)$としてしまうと、ある$A$があって、それを全ての$x$が愛する、という解釈になってしまう。本来任意の$x$ごとにある$y$が存在していることになっているわけだから、この定数Aはxに依存しているはずである。そこで、$\forall x\, Love(x, F(x))$として、この定数は$x$に依存する関数だとする。この$F$はスコーレム関数(Skolem function)と呼ばれる。

スコーレム関数の引数は、全称的に限量化された変数のうち、対象の存在限量子をその範囲(スコープ)に含むもの全てとなる。つまり、ある存在限量子と変数$\exists y$に対し、

$$\cdots \forall x_1 \left(\cdots \forall x_2 \left(\cdots \forall x_n (\cdots \exists y\ P) \right) \right)$$

となる全ての変数$x_1, x_2, \cdots, x_n$がスコーレム関数の引数となるということである。この場合、$\exists y$を消去するには、Pに含まれる全てのyを$F(x_1, x_2, \cdots, x_n)$に置き換えれば良いということである。ただしFは今まで使ったことのない関数記号でなければいけない。例えば、$\forall x \exists y [P(y) \wedge Q(x, y)]$をスコーレム化すると、

$$\forall x \left[P\left(F(x)\right) \wedge Q\left(x, F(x)\right) \right]$$

となる。また、対象の存在限量子をスコープに含む全称限量子が全く無い場合は、存在具体化で行ったように、関数ではなく定数として良い。ただし、定数は新しい定数でないといけない。

6. 全称限量子の除去: 次の規則により全ての全称限量子を論理式の一番外側に追いだす。

$$\forall x\ P(x) \wedge Q \equiv \forall x\ (P(x) \wedge Q)$$

$$\forall x\ P(x) \vee Q \equiv \forall x\ (P(x) \vee Q)$$

また、一番外側の全称限量子は省略されることが多いので注意(以下の証明においても省略する)。

7. 次の論理的同値関係を使うことで、∨を∧に対して可能な限り分配する

$$\alpha \vee (\beta \wedge \gamma) \equiv (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$$

以上の手続きによって CNF となる。

例: 「全ての動物を愛する人は誰かに愛されている」

$$\forall x \left[\forall y \left(Animal(y) \Rightarrow Loves(x, y) \right) \Rightarrow \exists y\ Loves(y, x) \right]$$

$\forall x \left[\neg \forall y \left(Animal(y) \Rightarrow Loves(x, y) \right) \vee \exists y\ Loves(y, x) \right]$ (含意の除去)

$\forall x\ [\neg\forall y\ (\neg Animal(y) \vee Loves(x,y)) \vee \exists y\ Loves(y,x)]$ (含意の除去)

$\forall x\ [\exists y \neg\ (\neg Animal(y) \vee Loves(x,y)) \vee \exists y\ Loves(y,x)]$ (否定を内側に入れる)

$\forall x\ [\exists y\ (Animal(y) \wedge \neg Loves(x,y)) \vee \exists y\ Loves(y,x)]$ (否定を内側に入れる)

$\forall x\ [\exists y\ (Animal(y) \wedge \neg Loves(x,y)) \vee \exists z\ Loves(z,x)]$ (変数名を書き換える)

$\forall x\ [\ (Animal(F(x)) \wedge \neg Loves(x,F(x))) \vee Loves(G(x),x)]$ (スコーレム化)

$\forall x\ \left[\ (Animal(F(x)) \vee Loves(G(x),x)) \wedge \left(\neg Loves(x,F(x)) \vee Loves(G(x),x)\right)\right]$ (分配律)

§9.5.2 融合推論規則

命題論理における融合規則に対応する一階述語論理のための融合規則が存在する。

・融合規則(一般)

$$\frac{l_1 \vee \cdots \vee l_i \vee m \qquad \neg m' \vee n_1 \vee \cdots \vee\ n_j}{SUBST(\theta, l_1 \vee \cdots \vee l_i \vee n_1 \vee \cdots \vee\ n_j)}$$

ただし、$UNIFY(m,m') = \theta$となる。例えば、

$$Animal(F(x)) \vee Loves(G(x),x) \text{と} \neg Loves(u,v) \vee \neg Kills(u,v)$$

を融合すると$\theta = \{u/G(x), v/x\}$で単一化し、

$$Animal(F(x)) \vee \neg Kills(G(x),x)$$

を得る。

§9.5.3 証明の例題

例:「アメリカ人が敵対国に武器を売るのは犯罪であると法律は述べている。Nono という国はアメリカの敵国であり、ミサイルをもっていて、そのすべてがアメリカ人の Colonel West により販売された。」

知識ベースKB:

$\forall x,y,z\ [American(x) \wedge Weapon(y) \wedge Sells(x,y,z) \wedge Hostile(z) \Rightarrow Criminal(x)]$

$\exists x\ [Owns(Nono, x) \land Missile(x)]$

$\forall x\ [Missile(x) \land Owns(Nono, x) \Rightarrow Sells(West, x, Nono)]$

$\forall x\ [Missile(x) \Rightarrow Weapon(x)]$

$\forall x\ [Enemy(x, America) \Rightarrow Hostile(x)]$

$American(West)$

$Enemy(Nono, America)$

質問Q: $Criminal(West)$

$KB \land \neg Q$の連言標準形(CNF):

R_1: $\neg American(x) \lor \neg Weapon(y) \lor \neg Sells(x, y, z) \lor \neg Hostile(z) \lor Criminal(x)$

R_2: $Owns(Nono, M)$

R_3: $Missile(M)$

R_4: $\neg Missile(x) \lor \neg Owns(Nono, x) \lor Sells(West, x, Nono)$

R_5: $\neg Missile(x) \lor Weapon(x)$

R_6: $\neg Enemy(x, America) \lor Hostile(x)$

R_7: $American(West)$

R_8: $Enemy(Nono, America)$

R_9: $\neg Criminal(West)$

証明:

R_{10}: $\neg American(West) \lor \neg Weapon(y) \lor \neg Sells(West, y, z) \lor \neg Hostile(z)$ (R_1とR_9より)

R_{11}: $\neg Weapon(y) \lor \neg Sells(West, y, z) \lor \neg Hostile(z)$ (R_7とR_{10}より)

R_{12}: $\neg Missile(y) \lor \neg Sells(West, y, z) \lor \neg Hostile(z)$ (R_5とR_{11}より)

R_{13}: $\neg Sells(West, M, z) \lor \neg Hostile(z)$ (R_3とR_{12}より)

R_{14}: $\neg Missile(M) \lor \neg Owns(Nono, M) \lor \neg Hostile(Nono)$ (R_4とR_{13}より)

R_{15}: $\neg Owns(Nono, M) \lor \neg Hostile(Nono)$ (R_3とR_{14}より)

R_{16}: $\neg Hostile(Nono)$ (R_2とR_{15}より)

R_{17}: $\neg Enemy(Nono, America)$ (R_6とR_{16}より)

R_{18}: 0 (R_8とR_{17}より)

従って、$KB \land \neg Q$が充足不能であることが示せたため、KBがQを伴意することを示せた。

例: 「動物が好きな人は誰かに愛される。動物を殺す人は誰にも愛されない。Jack は全ての動物を愛する。Jack か Curiosity のどちらかが、Tuna という猫を殺した。Curiosity はその猫を殺したか？」

知識ベースKB:

$\forall x \left[\forall y \left(Animal(y) \Rightarrow Loves(x, y)\right) \Rightarrow \exists y\ Loves(y, x)\right]$

$\forall x \left[\exists y \left(Animal(y) \land Kills(x, y)\right) \Rightarrow \forall z\ \neg Loves(z, x)\right]$

$\forall x\ [Animal(x) \Rightarrow Loves(Jack, x)]$

$Kills(Jack, Tuna) \lor Kills(Curiosity, Tuna)$

$Cat(Tuna)$

$\forall x\ [Cat(x) \Rightarrow Animal(x)]$

質問Q: $Kills(Curiosity, Tuna)$

$KB \land \neg Q$の連言標準形(CNF):

R_1: $Animal\left(F(x)\right) \lor Loves(G(x), x)$

R_2: $\neg Loves\left(x, F(x)\right) \lor Loves(G(x), x)$

R_3: $\neg Animal(y) \lor \neg Kills(x, y) \lor \neg Loves(z, x)$

R_4: $\neg Animal(x) \lor Loves(Jack, x)$

R_5: $Kills(Jack, Tuna) \lor Kills(Curiosity, Tuna)$

R_6: $Cat(Tuna)$

R_7: $\neg Cat(x) \lor Animal(x)$

R_8: $\neg Kills(Curiosity, Tuna)$

証明:

R_9: $Animal(Tuna)$ (R_6とR_7より)

R_{10}: $\neg Kills(x, Tuna) \lor \neg Loves(z, x)$ (R_3とR_9より)

R_{11}: $Kills(Jack, Tuna)$ (R_5とR_8より)

R_{12}: $\neg Loves(z, Jack)$ (R_{10}とR_{11}より)

R_{13}: $\neg Animal(F(Jack)) \lor Loves(G(Jack), Jack)$ (R_2とR_4より)

R_{14}: $Loves(G(Jack), Jack)$ (R_1とR_{13}より)

R_{14}: 0 (R_{12}とR_{14}より)

従って、$KB \land \neg Q$が充足不能であることが示せたため、KBがQを伴意することを示せた。

この証明は次の言葉で言い表すことができる。

「Curiosity が Tuna を殺さなかったとしよう。Jack か Curiosity がやったことはわかっている。従って、Jack がやったに違いない。さて、Tuna は猫であり、猫は動物であるので、Tuna は動物である。動物を殺すものは誰にも愛されないので、誰も Jack を愛さないことを知る。一方、Jack はすべての動物を愛するので、誰かが Jack を愛している。そこで、矛盾に達する。従って、Curiosity が猫を殺した。」

これらの推論は伴意関係が成り立つか成り立たないかの yes/no を返してくれるだけだが、一般には「誰が猫を殺したか？」という質問に対する答えを知りたいと思うこともあるだろう。その場合は、質問Qは$\exists w\ Kills(w, Tuna)$となり、$KB \land \neg Q$に対して、$R_8$が

R_8: $\neg Kills(w, Tuna)$

となる。これに対する推論は、

R_{11}: $Kills(Jack, Tuna)$ (R_5とR_8より)

を得る過程において、$\{w/Curiosity\}$とすればあとは同じとなる。従って、wは$Curiosity$だとわかる。このように質問における変数の束縛を記憶しておけば誰が？どれが？どこで？といった質問に対する答えを知ることができる。

　しかし、このように存在を問う質問に対して融合法は構成的でない証明(nonconstructive proof)を作ることがある。例えば、

R_8: $\neg Kills(w, Tuna)$

に対し、次の証明が与えられうる。

R_9: $Kills(Jack, Tuna)$　　　(R_5とR_8より)

R_{10}: 0　　　(R_8とR_9より)

これはwに対して 2 回束縛が行われており、$w/Curiosity$か$w/Jack$であるかのどちらかであるということを言っていることになる。つまり、融合法は猫を殺したのは Curiosity か Jack のどちらかである、と告げてくれていることになる。しかし、もともと、Curiosity か Jack が猫を殺していることはわかっているのだから、このような解が得られることにはあまり意味がない。

　この問題に対する一つの解法は、与えられた証明中で質問変数が一度しか束縛されないように融合の制限をかけることである。そして、次の可能な解を求めるためにバックトラック(証明を巻き戻して別の束縛を与えて証明をやり直すこと)をする必要がある。

　この問題に対するもう一つの解法は、否定質問に特別な回答リテラル(answer literal)を加えて、$\neg Kills(w, Tuna) \vee Answer(w)$とすることである。このようにすると、上記の非構成的証明に対しては、$Answer(Curiosity) \vee Answer(Jack)$を生成するが、これは解を構成しない。つまり、

R_8: $\neg Kills(w, Tuna) \vee Answer(w)$

に対し、

R_9: $Kills(Jack, Tuna) \vee Answer(Curiosity)$　　　(R_5とR_8より)

R_{10}: $Answer(Curiosity) \vee Answer(Jack)$　　　(R_8とR_9より)

となり、ここで偽$(= 0)$となることはないためである。

第 8 回のまとめ

○一階述語論理の証明（融合法）

1. $KB \wedge \neg\alpha$を連言標準形(conjunctive normal form: CNF)に変換する。連言標準形は次の形をした文である。

$$(l_{1,1} \vee l_{1,2} \vee \cdots) \wedge (l_{2,1} \vee l_{2,2} \vee \cdots) \wedge \cdots \wedge (l_{n,1} \vee l_{n,2} \vee \cdots)$$

ただし、$l_{i,j}$は**リテラル**と呼ばれ、リテラルは原子文または原子文の否定のいずれかである。原子文は**正リテラル**、原子文の否定は**負リテラル**とも呼ばれる。$(l_{i,1} \vee l_{i,2} \vee \cdots)$は**節**と呼ばれる。$KB \wedge \neg\alpha$を CNF に変換した結果の節集合を$KB'$とする。

2. 節集合KB'に対し融合規則を適用し、$KB' \vDash R$となるRをみつけRを節集合KB'に追加する(KB'は更新される)。この操作(推論)を$KB' \vdash R$と書くことにする。

3. 追加できる新しい節がない場合、KBからαを伴意しないことがわかる

4. 偽$(= 0)$が得られたら、KBからαを伴意することがわかる

5. 2.に戻る

○融合法での推論規則（$KB \vdash R$）

融合法で用いる推論規則は融合規則ただ一つのみ

$$\frac{l_1 \vee \cdots \vee l_i \vee m \qquad \neg m' \vee n_1 \vee \cdots \vee n_j}{SUBST(\theta, l_1 \vee \cdots \vee l_i \vee n_1 \vee \cdots \vee n_j)} \qquad \text{(融合規則(一般))}$$

ただし、$UNIFY(m, m') = \theta$。

○連言標準形(CNF)への変換

1. $\alpha \Leftrightarrow \beta$を$(\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$に置き換えることで、$\Leftrightarrow$を除去する

2. $\alpha \Rightarrow \beta$を$\neg\alpha \vee \beta$に置き換えることで$\Rightarrow$を除去する

3. 次の論理的同値関係を使うことで否定($\neg$)を内側へいれていく

$\neg\neg\alpha \equiv \alpha$　　(二重否定除去)

$\neg(\alpha \wedge \beta) \equiv \neg\alpha \vee \neg\beta$　　(ド・モルガンの法則)

$\neg(\alpha \vee \beta) \equiv \neg\alpha \wedge \neg\beta$　　(ド・モルガンの法則)

$\neg\exists x\, P \;\equiv\; \forall x \neg\, P$

$\neg\forall x\, P \;\equiv\; \exists x \neg\, P$

4. 変数標準化: $(\forall x\, P(x)) \vee (\exists x\, P(x))$のような同じ変数を 2 回使用する文に対して、一方の変数の名称を改名する。例、$(\forall x\, P(x)) \vee (\exists x\, P(x))$を$(\forall x\, P(x)) \vee (\exists y\, P(y))$とする。

5. スコーレム化(存在限量子の除去): 存在限量子の除去を行う。ある存在限量子$\exists y$に対し、

$$\cdots \forall x_1 \left(\cdots \forall x_2 (\cdots \forall x_n (\cdots \exists y\, P))\right)$$

となる全ての変数$x_1, x_2, \cdots, x_n$を見つけ、Pに出現する全てのyを$F(x_1, x_2, \cdots, x_n)$と置き換えて$\exists y$を消去する。Fのことをスコーレム関数といい、Fは今まで使ったことのない新しい関数記号でなければいけない。また、存在限量子が全称限量子に囲まれていない場合は関数ではなく定数として良い。ただし、定数は新しい定数でなければいけない。

6. 全称限量子の除去: 次の規則により全ての全称限量子を論理式の一番外側に追いだす。

$$\forall x\, P(x) \wedge Q \equiv \forall x\, (P(x) \wedge Q)$$

$$\forall x\, P(x) \vee Q \equiv \forall x\, (P(x) \vee Q)$$

また、一番外側の全称限量子は省略されることが多いので注意。

7. 次の論理的同値関係を使うことで、$\vee$を$\wedge$に対して可能な限り分配する

$$\alpha \vee (\beta \wedge \gamma) \equiv (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$$