LIN380M Semantics I Homeworks

Zeyuan Hu, iamzeyuanhu@utexas.edu

EID:zh4378 Spring 2018

1 HW1

Translate the following sentences into predicate logic. (You may choose your own letters to serve as non-logical constants. Translate "but" as if it were "and")

1. **John loves Mary, but she doesnt love him.**

   $\texttt{love}(J, M) \wedge \neg\texttt{love}(M, J)$

2. **John believes all things that Mary believes and some other things as well.**

   $(\forall y)((\texttt{things}(y) \wedge \texttt{believe}(M, y)) \rightarrow \texttt{believe}(J, y)) \wedge (\exists x)(\texttt{things}(x) \wedge \texttt{believe}(J, x) \wedge \neg\texttt{believe}(M, x))$

3. **If a cat and a mouse are in the same room and the mouse doesnt run away, then either the mouse is dead or the cat is dead.**

   $(\forall x)(\forall y)(\texttt{cat}(x) \wedge \texttt{mouse}(y) \wedge \texttt{inTheSameRoom}(x, y) \wedge \neg\texttt{runAway}(y)) \rightarrow (\texttt{dead}(x) \vee \texttt{dead}(y))$ [1]

4. **Everyone who has two jobs neglects one of them.**

   $(\forall x)(\exists y)(\exists z)(\texttt{Human}(x) \wedge \texttt{job}(y) \wedge \texttt{job}(z) \wedge \texttt{hasJob}(x, y) \wedge \texttt{hasJob}(x, z) \wedge y \neq z) \rightarrow ((\texttt{neglect}(y) \wedge \neg\texttt{neglect}(z)) \vee (\texttt{neglect}(z) \wedge \neg\texttt{neglect}(y)))$

5. **If an argument with two premises is valid and its conclusion is false, then one of the premises is false.**

   $(\forall x)(\forall y)(\forall z)(\forall k)((\texttt{argument}(x) \wedge \texttt{premise}(y) \wedge \texttt{premise}(z) \wedge \texttt{belongToArgument}(y, x) \wedge \texttt{belongToArgument}(z, x) \wedge \texttt{valid}(x) \wedge \texttt{conclusion}(k) \wedge \neg\mathrm{True}(k) \wedge \texttt{belongToArgument}(k, x)) \rightarrow (\neg True(y) \vee \neg True(z))$ [2]

[1] Here, I interpret "either ... or" as `OR` in logic term. If we treat "either ... or" as `XOR` instead, the PC of the sentence then becomes $(\forall x)(\forall y)(\texttt{cat}(x) \wedge \texttt{mouse}(y) \wedge \texttt{inTheSameRoom}(x, y) \wedge \neg\texttt{runAway}(y)) \rightarrow ((\texttt{dead}(x) \wedge \neg\texttt{dead}(y)) \vee (\texttt{dead}(y) \wedge \neg\texttt{dead}(x)))$

[2] Since an argument is consisted of premises and conclusion, $\texttt{conclusion}(k)$ may be ambiguous as it can either represent whether k is a conclusion or not in an argument (it can be premise) or it can represent the truth value of a conclusion. Thus, I think it is necessary to avoid this ambiguity by introducting another predicate `conclusionIsTrue` to indicate the truth value of a conclusion and use `conclusion` predicate to indicate whether the given variable is a conclusion or not. Quantifier cannot directly apply to variable. It has to apply to formula.

6. **There are three dishes that John doesnt like.**

   $((\exists x)(\exists y)(\exists z)(\texttt{dish}(x) \wedge \texttt{dish}(y) \wedge \texttt{dish}(z) \wedge x \neq y \wedge x \neq z \wedge y \neq z)) \rightarrow (\neg\texttt{like}(J, x) \wedge \neg\texttt{like}(J, y) \wedge \neg\texttt{like}(J, z))$

2 HW2

1. **Translate the following sentences into Predicate Logic. (You may choose your own letters to serve as non-logical constants,)**

   (a) **If one person is taller than another person and that second person is taller than a third one, then the first is taller than the third one.**

   $(\forall x)(\forall y)(\forall z)(\text{person}(x) \wedge \text{person}(y) \wedge \text{person}(z) \wedge x \neq y \wedge x \neq z \wedge y \neq z \wedge \text{taller}(x, y) \wedge \text{taller}(y, z)) \rightarrow (\text{taller}(x, z))$

   (b) **If one number is between two other numbers, then neither of the two others is between it and the third one.**

   $(\forall a)(\forall b)(\forall c)(\text{number}(a) \wedge \text{number}(b) \wedge \text{number}(c) \wedge a < b \wedge b < c) \rightarrow (\neg(b < a \wedge a < c) \wedge \neg(b < c \wedge c < a))$

   (c) **If you move a wolf, a goat and a cabbage across a river and you have a boat that can hold two but no more than two of the four of you, then there is exactly one strategy (for getting all of you safely across) . (N.B. you do not need to translate the part in parentheses.)**

   $(\forall x)(\forall y)(\forall z)(\forall k)(\forall w)(\text{wolf}(x) \wedge \text{goat}(y) \wedge \text{cabbage}(z) \wedge \text{you}(k) \wedge \text{boat}(w) \wedge (\text{canHold}(w, \{x\}) \wedge \text{canHold}(w, \{y\}) \wedge \text{canHold}(w, \{z\}) \wedge \text{canHold}(w, \{k\}) \wedge \text{canHold}(w, \{x, y\}) \wedge \text{canHold}(w, \{x, z\}) \wedge \text{canHold}(w, \{x, k\}) \wedge \text{canHold}(w, \{y, z\}) \wedge \text{canHold}(w, \{y, k\}) \wedge \text{canHold}(w, \{z, k\}) \wedge \neg\text{canHold}(w, \{x, y, z\}) \wedge \neg\text{canHold}(w, \{x, y, k\}) \wedge \neg\text{canHold}(w, \{x, z, k\}) \wedge \neg\text{canHold}(w, \{y, z, k\}) \wedge \neg\text{canHold}(w, \{x, y, z, k\}) \wedge \text{moveAcrossRiver}(x) \wedge \text{moveAcrossRiver}(y) \wedge \text{moveAcrossRiver}(z) \wedge \text{moveAcrossRiver}(k)) \rightarrow (\exists x)(\text{strategy}(x) \wedge \text{True}(x)) \wedge (\forall y)((\text{strategy}(y) \wedge y \neq x) \rightarrow (\neg\text{True}(y)))$

2. **Give as many non-equivalent translations of sentence (a) into Predicate Logic as you can think of.**

   (a) **Three girls met two boys.**

   - $(\forall x)(\forall y)(\forall z)(\forall a)(\forall b)(\text{girl}(x) \wedge \text{girl}(y) \wedge \text{girl}(z) \wedge x \neq y \wedge y \neq z \wedge x \neq z \wedge \text{boy}(a) \wedge \text{boy}(b) \wedge a \neq b \wedge \text{met}(\{x, y, z\}, \{a, b\})$ [3]

[3] "met" in $\text{met}(\{x, y, z\}, \{a, b\})$ has meaning of $\{\alpha | (\forall\alpha)(\forall\beta)((\text{function}(\alpha) \wedge \text{function}(\beta) \wedge \alpha \neq \beta) \rightarrow (\alpha \in$

- $(\forall x)(\forall y)(\forall z)(\forall a)(\forall b)(\text{girl}(x) \wedge \text{girl}(y) \wedge \text{girl}(z) \wedge x \neq y \wedge y \neq z \wedge x \neq z \wedge \text{boy}(a) \wedge \text{boy}(b) \wedge a \neq b \wedge \text{met}(x,a) \wedge \text{met}(x,b) \wedge \text{met}(y,a) \wedge \text{met}(y,b) \wedge \text{met}(z,a) \wedge \text{met}(z,b))$

3. (Note well: this exercise consists of 4 parts: a, b, c and d.)

(a) Translate the following sentences into Predicate Logic. Use the 1-place predicate P for person and the 2-place predicate R for 'met.

i. No one has met everyone.

$(\forall x)(\exists y)(P(x) \wedge P(y) \wedge x \neq y \wedge \neg R(x,y))$

ii. No one has met anyone.

$\neg((\exists x)(\exists y)(P(x) \wedge P(y) \wedge x \neq y \wedge R(x,y)))$

iii. No one has met no one.

$(\forall x)(\exists y)(P(x) \wedge P(y) \wedge x \neq y \wedge R(x,y))$

iv. If someone has met no one, then no one has met everyone.

$(\exists x)(\forall y)((P(x) \wedge P(y) \wedge x \neq y \wedge \neg R(x,y)) \rightarrow (\neg(\exists z)(P(z) \wedge z \neq y \wedge R(z,y)))$

v. If someone has met someone, then it is not the case that no one has met anyone.

$((\exists x)(\exists y)(P(x) \wedge P(y) \wedge x \neq y \wedge R(x,y)) \rightarrow (\neg(\forall x)(\forall y)(P(x) \wedge P(y) \wedge x \neq y \wedge \neg R(x,y)))$

(b) Let $M = \langle U, I \rangle$ be the following model for the language P, R:

$U = \{a, b, c, d\}$

$I(P) = U$(**so every individual in the universe of M is a person**)

$I(R) = \{< a, b >, < a, c >, < a, d >, < b, a >, < c, a >, < d, a >, < c, d >, < d, c >, \}$

i. Which of the sentences (i)-(v) are true in M and which false, assuming that the words no one, everyone, anyone, someone range over persons and that R translates the transitive verb meet?

(i), (ii), (iv) are false and (iii), (v) are true.

ii. Go through the calculation of the truth value in M of (i) by applying the truth definition for PC to the translation of (i) into PC you have given under (a).

$\{a,b\}^{\{x,y,z\}} \wedge \beta \notin \{a,b\}^{\{x,y,z\}}))\}$ (we use PC loosely here as). In words, "met" is a set of functions that maps set $\{x,y,z\}$ to set $\{a,b\}$. $\{a,b\}^{\{x,y,z\}}$ represents all possible combinations of "met" relation that can happen between boys and girls.

$$[\![(\forall x)(\exists y)(P(x) \wedge P(y) \wedge x \neq y \wedge \neg R(x,y))]\!]_{M,a} = 1$$

iff for every $a' \approx_x a$ $\quad [\![(\exists y)(P(x) \wedge P(y) \wedge x \neq y \wedge \neg R(x,y))]\!]_{M,a'} = 1$

iff for every $a' \approx_x a$ for some $a'' \approx_y a'$ $\quad [\![(P(x) \wedge P(y) \wedge x \neq y \wedge \neg R(x,y))]\!]_{M,a''} = 1$

iff for every $a' \approx_x a$ for some $a'' \approx_y a'$ $\quad \langle [\![x]\!]_{M,a''} \rangle \in I_M(P)$ *and*

$\langle [\![y]\!]_{M,a''} \rangle \in I_M(P)$ *and* $[\![x]\!]_{M,a''} \neq [\![y]\!]_{M,a''}$ *and* $[\![R(x,y)]\!]_{M,a''} = 0$

iff for every $a' \approx_x a$ for some $a'' \approx_y a'$ $\quad \langle [\![x]\!]_{M,a''} \rangle \in I_M(P)$ *and*

$\langle [\![y]\!]_{M,a''} \rangle \in I_M(P)$ *and* $[\![x]\!]_{M,a''} \neq [\![y]\!]_{M,a''}$ *and* $\langle [\![x]\!]_{M,a''}, [\![y]\!]_{M,a''} \rangle \notin I_M(R)$

iff for every $a' \approx_x a$ for some $a'' \approx_y a'$ $\quad \langle a''(x) \rangle \in I_M(P)$ *and*

$\langle a''(y) \rangle \in I_M(P)$ *and* $a''(x) \neq a''(y)$ *and* $\langle a''(x), a''(y) \rangle \notin I_M(R)$

The above statement is false because if we let $a''(x) = a$ and there is no such $a''(y)$ that can make whole PC to true. If $a''(y) = a$, then $a''(x) \neq a''(y)$ is false. If $a''(y) = b$, then $\langle a, b \rangle \in I_M(R)$, which makes $\langle a''(x), a''(y) \rangle \notin I_M(R)$ false. If $a''(y) = c$, then $\langle a, c \rangle \in I_M(R)$, which makes $\langle a''(x), a''(y) \rangle \notin I_M(R)$ false.

(c) Which of the sentences (i)-(v), if any is/are true in all models for PC?

(v)

(d) For each of the sentences (i)-(v) that is not true in M modify the interpretation of $I(M)$ in such a way that this sentence comes out true in the modified model. (But leave the Universe U as it is!)

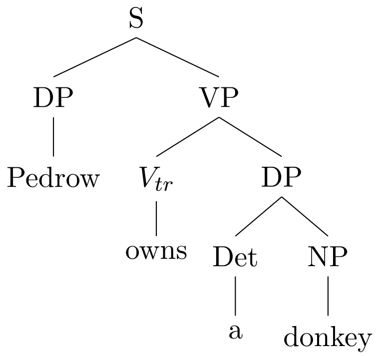
- (i) We modify the model by removing $\langle a, b \rangle$ from $I(R)$
- (ii) We modify the model by making $I(R) = \{\langle a, a \rangle\}$
- (iv) We modify the model by removing $\langle a, b \rangle, \langle a, c \rangle, \langle a, d \rangle$ from $I(R)$

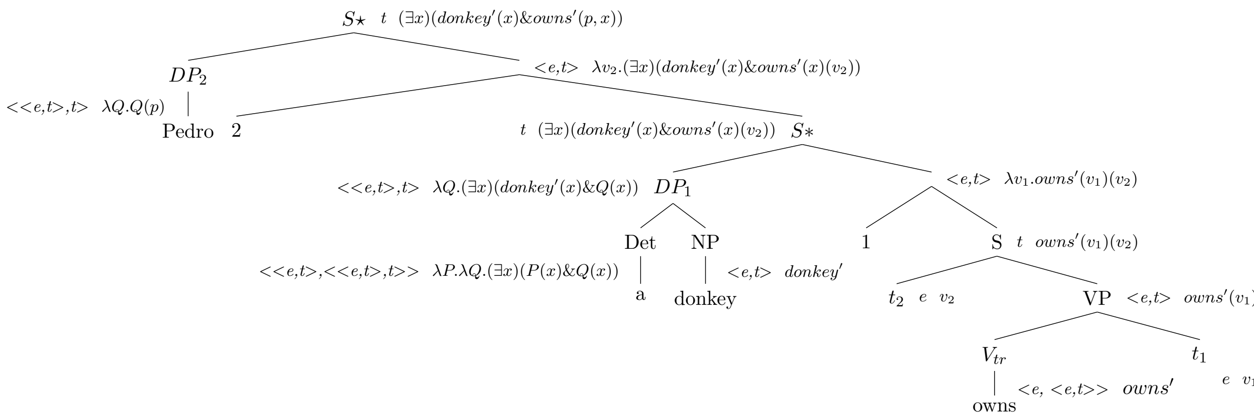
3 HW3

For each of the sentences below give an LF and derive the semantics of the sentence from this LF in the way we have been doing for the exercises on the 3d practice homework (that is, by assigning Lambda Calculus terms to the nodes of the LF tree).

In (2) and (5) only choose an LF for what you take to be the most plausible scope ordering.

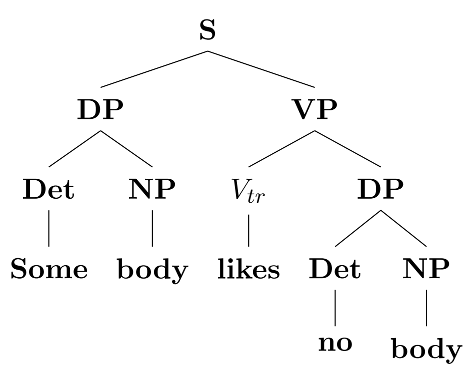
1. Pedro owns a donkey.



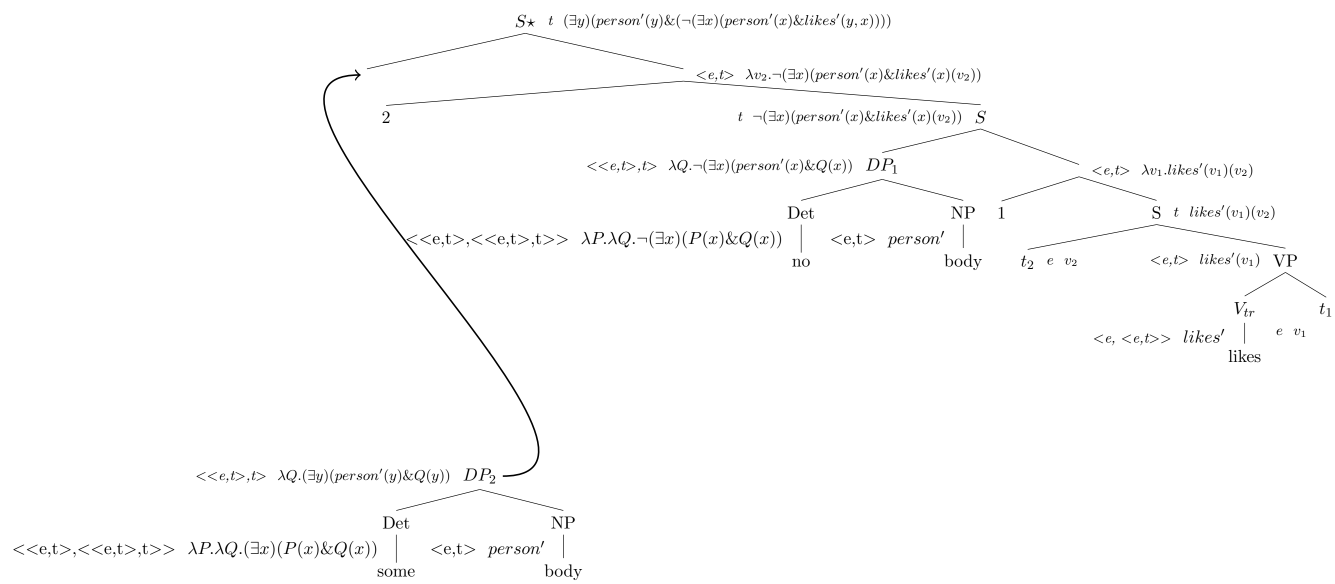


$$
\begin{aligned}
* &= \lambda Q.(\exists x)(donkey'(x)\&Q(x))(\lambda v_1 owns'(v_1)(v_2)) \\
&= (\exists x)(donkey'(x)\&(\lambda v_1 owns'(v_1)(v_2))(x)) \\
&= (\exists x)(donkey'(x)\&owns'(x)(v_2))
\end{aligned}
$$

$$
\begin{aligned}
\star &= (\lambda Q.Q(j))(\lambda v_2.(\exists x)(donkey'(x)\&owns'(x)(v_2))) \\
&= (\lambda v_2.(\exists x)(donkey'(x)\&owns'(x)(v_2)))(p) \\
&= (\exists x)(donkey'(x)\&owns'(x)(p)) \\
&= (\exists x)(donkey'(x)\&owns'(p,x))
\end{aligned}
$$

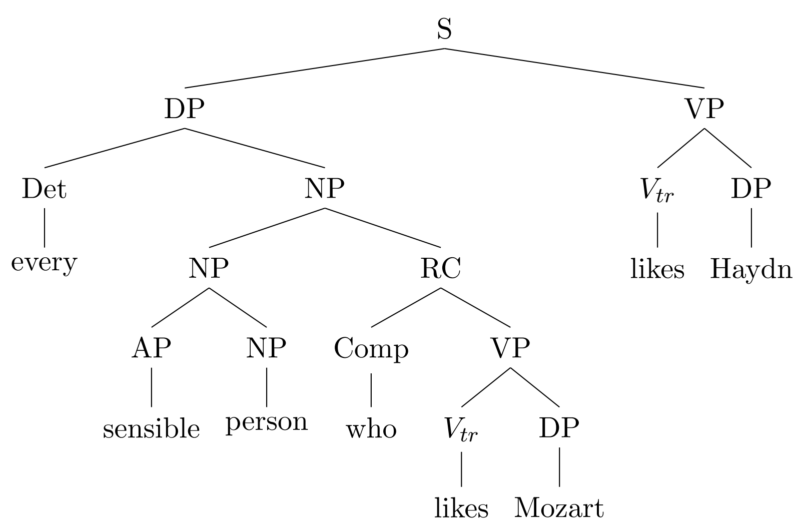


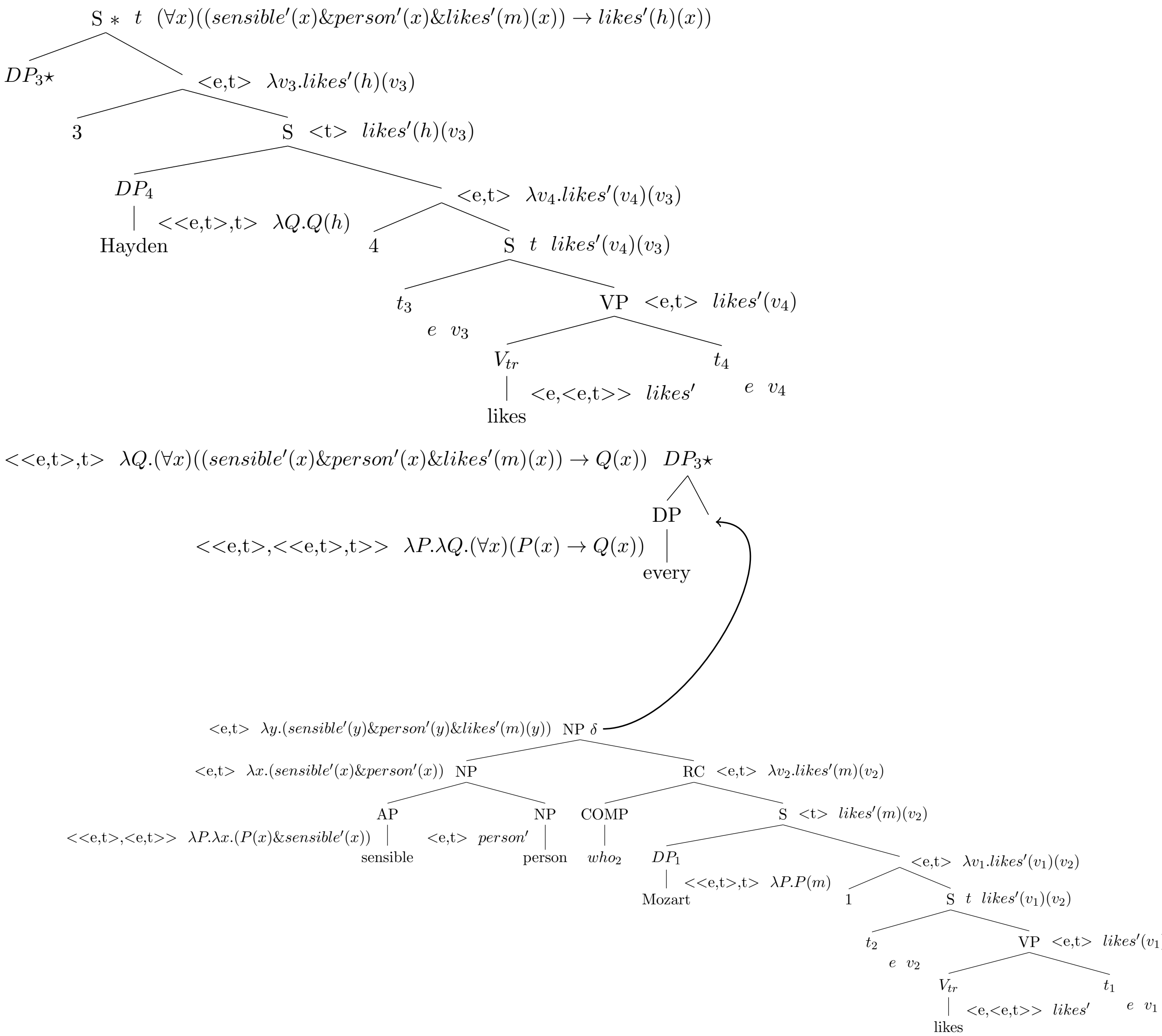
2. Somebody likes nobody.



$$
\begin{aligned}
\star &= (\lambda Q.(\exists y)(person'(y)\&Q(y)))(\lambda v_2.\neg(\exists x)(person'(x)\&likes'(x)(v_2)))\\
&= (\exists y)(person'(y)\&(\lambda v_2.\neg(\exists x)(person'(x)\&likes'(x)(v_2))(y)))\\
&= (\exists y)(person'(y)\&(\neg(\exists x)(person'(x)\&likes'(x)(y))))\\
&= (\exists y)(person'(y)\&(\neg(\exists x)(person'(x)\&likes'(y,x))))
\end{aligned}
$$

3. Every sensible person who likes Mozart likes Haydn.





$$\delta = \lambda y.((\lambda x.(sensible'(x) \& person'(x)))(y) \& (\lambda v_2.likes'(m)(v_2))(y))$$

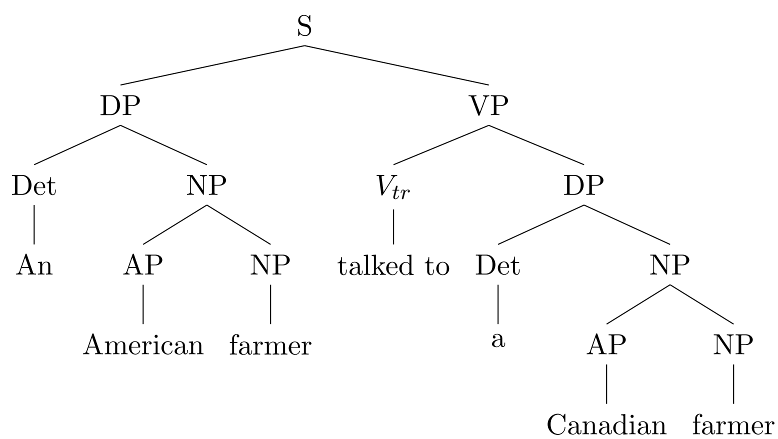
$$= \lambda y.((sensible'(y) \& person'(y)) \& (likes'(m)(y)))$$

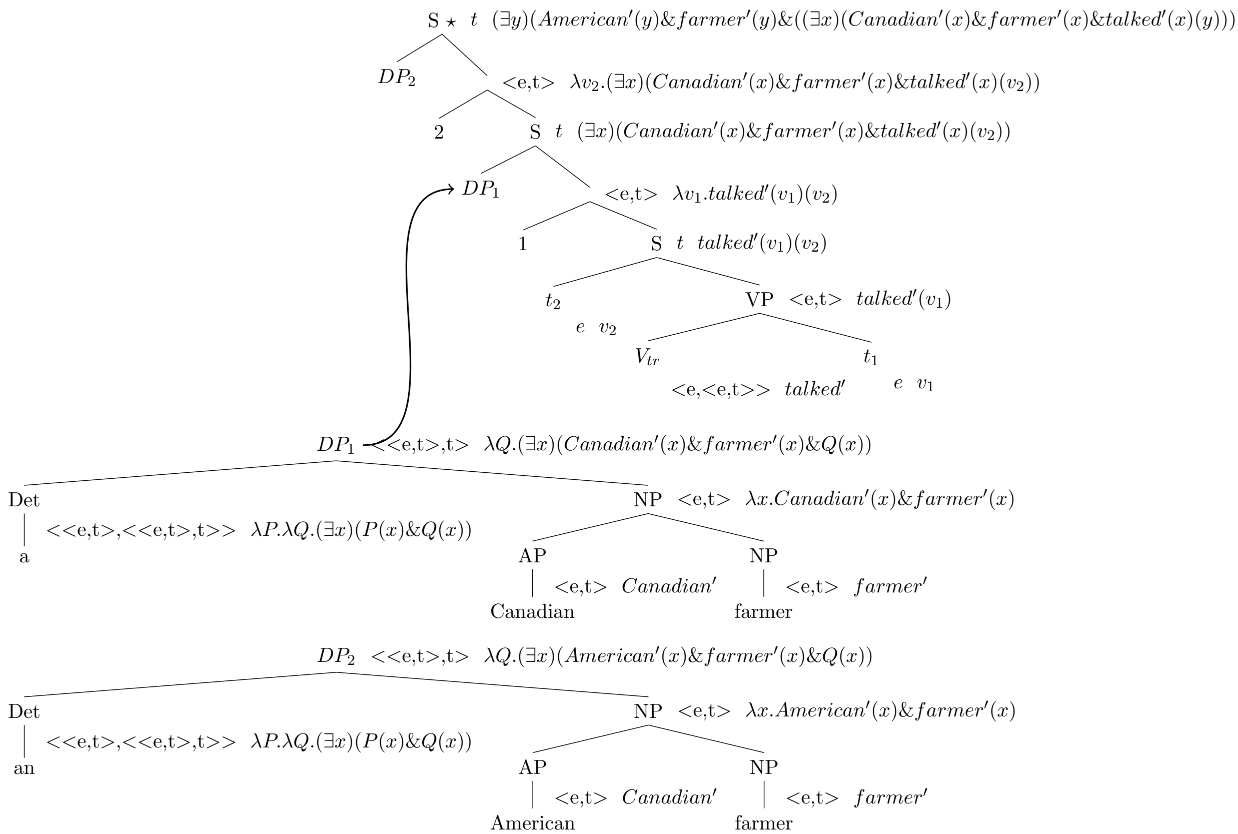
$$= \lambda y.(sensible'(y) \& person'(y) \& likes'(m)(y))$$

$$\star = \lambda P.\lambda Q.(\forall x)(P(x) \rightarrow Q(x)))(\lambda x.sensible'(x)\&person'(x)\&likes'(m)(x))$$
$$= \lambda Q.(\forall x)((\lambda x.sensible'(x)\&person'(x)\&likes'(m)(x))(x) \rightarrow Q(x))$$
$$= \lambda Q.(\forall x)((sensible'(x)\&person'(x)\&likes'(m)(x)) \rightarrow Q(x))$$

$$* = \lambda Q.(\forall x)((sensible'(x)\&person'(x)\&likes'(m)(x)) \rightarrow Q(x))(\lambda v_3.likes'(h)(v_3))$$
$$= (\forall x)((sensible'(x)\&person'(x)\&likes'(m)(x)) \rightarrow likes'(h)(x))$$

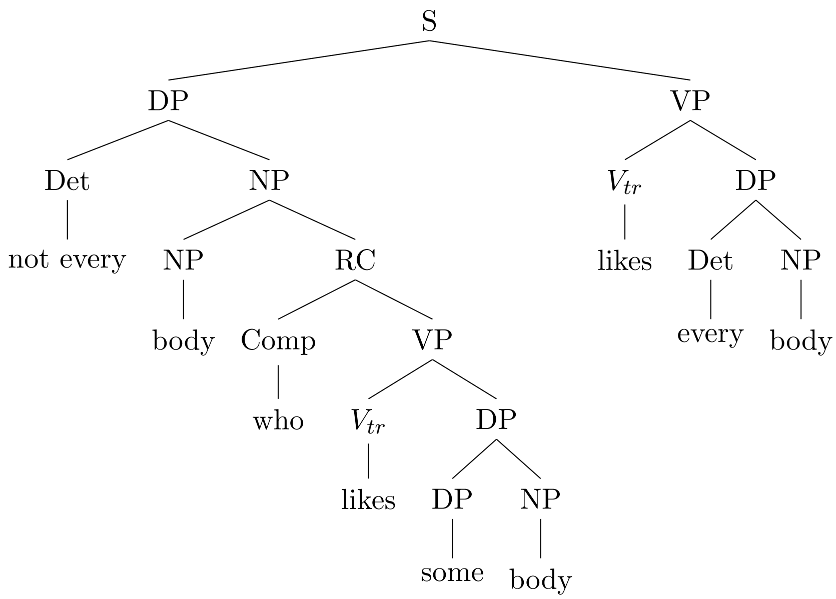
4. An American farmer talked to a Canadian farmer. (Treat talk to as a transitive verb.)

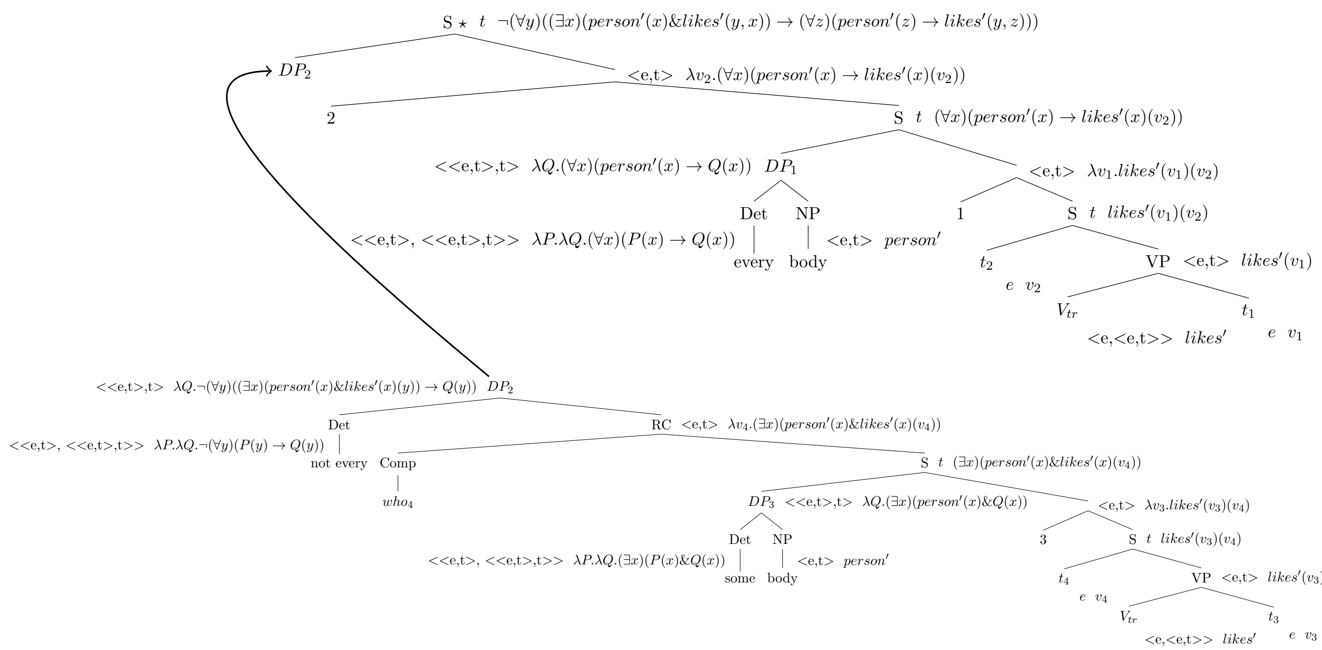




$$
\begin{aligned}
\star &= \lambda Q.(\exists y)(American'(y)\&farmer'(y)\&Q(y))(\lambda v_2(\exists x)(Canadian'(x)\&farmer'(x)\&talked'(x)(v_2)))\\
&= (\exists y)(American'(y)\&farmer'(y)\&(\lambda v_2(\exists x)(Canadian'(x)\&farmer'(x)\&talked'(x)(v_2)))(y))\\
&= (\exists y)(American'(y)\&farmer'(y)\&((\exists x)(Canadian'(x)\&farmer'(x)\&talked'(x)(y))))
\end{aligned}
$$

5. Not everybody who likes somebody likes everybody. (N.B. In (5) treat 'not everybody' as consisting of the complex determiner 'not every' and the noun 'body'. Treat 'not every' as an unanalyzed determiner with the semantics: $\lambda P.\lambda Q.\neg(\forall x)(P(x) \rightarrow Q(x))$) and treat 'body' as an NP to which the relative clause 'who likes somebody' can be adjoined.)





$$
\begin{aligned}
\star &= (\lambda Q.\neg(\forall y)((\exists x)(person'(x)\&likes'(x)(y)) \to Q(y)))(\lambda v_2.(\forall z)(person'(z) \to likes'(z)(v_2))) \\
&= \neg(\forall y)((\exists x)(person'(x)\&likes'(x)(y)) \to (\forall z)(person'(z) \to likes'(z)(y))) \\
&= \neg(\forall y)((\exists x)(person'(x)\&likes'(y,x)) \to (\forall z)(person'(z) \to likes'(y,z)))
\end{aligned}
$$