## Small Model [Ravi Kini]

## Model Source

curtain

Burning Down The Chippy (Derry Girls S01E02)

```
Domain
D_{+} = \{0, 1\}
D<sub>e</sub> = {Erin, Orla, Clare, Michelle, James, Fionnula, Mary, Sarah, Colm, alcohol, curtains, fire extinguisher, scented
candle, radiator}
Interpretation Function
I = [
p_{e}
                     Erin
                     0rla
                     Clare
p_c
                     Michelle
                     James
                     Fionnula
                     Mary
                     Sarah
                     Colm
i_a
                     alcohol
i_c
                     curtains
i_e
                     fire extinguisher
i_n
                     scented candle
i_r
                     radiator
// unary predicates
```

```
teenager → {Erin, Orla, Clare, Michelle, James}

adult → {Fionnula, Mary, Sarah, Colm}

female → {Erin, Orla, Clare, Michelle, Fionnula, Mary, Sarah}

male → {James, Colm}

extinguisher → {fire extinguisher}
```

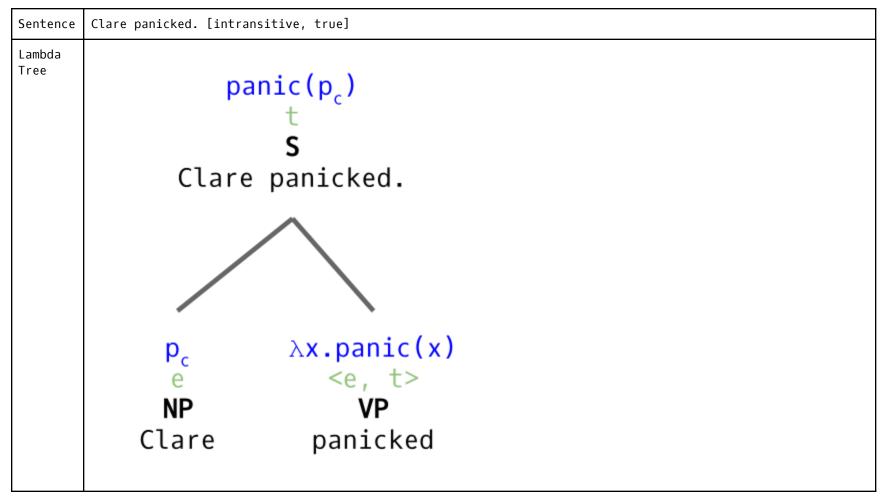
{curtains}

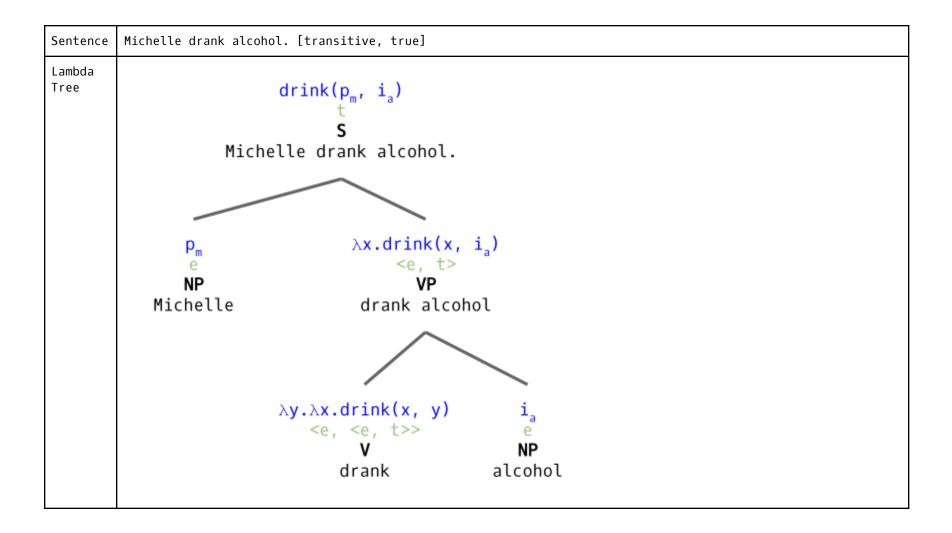
```
{scented candle}
scentedCandle→
radiator
                   {radiator}
panic
                   {Erin, Clare}
shocked
                   {Fionnula}
                   {Colm}
longWinded
// binary predicates
                   {<Michelle, alcohol>, <Fionnula, alcohol>}
drink
                   {<Michelle, alcohol>}
drop
                   {<Erin, radiator>, <Orla, radiator>, <Clare, radiator>, <Michelle, radiator>, <James, radiator>}
tie
// ternary predicates
                   {<James, fire extinguisher, curtains>}
spray
]
```

```
Lexical Entries
```

```
Nouns (NP)
             \langle Clare \rangle = p_c (e)
             \langle Michelle \rangle = p_m (e)
             \langle James \rangle = p_i (e)
             \langle Fionnula \rangle = p_f (e)
             \langle Colm \rangle = p_1 (e)
             \langle alcohol \rangle = i_a (e)
             \langle \text{fire extinguisher} \rangle = \lambda x.extinguisher(x) (<e, t>)
             \langle \text{curtains} \rangle = \lambda x. \text{curtain}(x) (<e, t>)
             \langle man \rangle = \lambda x.man(x) (\langle e, t \rangle)
             \langle someone \rangle = \lambda P. \exists z P(z) (\langle e, t \rangle, t \rangle
             \langle scented candle \rangle = \lambda x.scentedCandle(x) (<e, t>)
             \langle \text{teen} \rangle = \lambda x. \text{teen}(x) (\langle e, t \rangle)
             \langle radiator \rangle = \lambda x.radiator(x) (<e, t>)
Verbs (V)
             \langle panicked \rangle = \lambda x.panic(x) (<e, t>)
             \langle drank \rangle = \lambda y \cdot \lambda x \cdot drink(x, y) (<e, <e, t>>)
             \langle sprayed \rangle = \lambda z.\lambda y.\lambda x.spray(x, z, y) (<e, <e, <e, t>>>)
             \langle was \rangle = \lambda P.P (<<e, t>, <e, t>>)
             \langle is \rangle = \lambda P.P (<<e, t>, <e, t>>)
             \langle dropped \rangle = \lambda y \cdot \lambda x \cdot drop(x, y) (<e, <e, t>>)
            \langle \text{tied} \rangle = \lambda y.\lambda x. \text{tie}(x, y) (<e, <e, t>>)
Adjectives (AP)
             \langle shocked \rangle = \lambda x.shocked(x) (<e, t>)
             \langle long-winded \rangle = \lambda x.longWinded(x) (<e, t>)
Determiners (D)
             \langle the \rangle = \lambda P.\iota x.P(x) (<<e, t>, e>)
             \langle a \rangle = \lambda P.P (<<e, t>, <e, t>>)
             \langle \text{every} \rangle = \lambda Q.\lambda P. \forall z Q(z) \rightarrow P(z) (<<e, t>, <<e, t>, t>>)
             \langle a \rangle = \lambda P \lambda Q. \exists x P(x) \wedge Q(x) (<<e, t>, <<e, t>, t>>)
Prepositions (P)
             \langle at \rangle = \lambda x.x (<e, e>)
             \langle to \rangle = \lambda x.x (<e, e>)
```

## Translations





Sentence	James sprayed the fire extinguisher at the curtains. [ditransitive, true]					
Lambda Tree	<pre>spray(p<sub>j</sub>, ix.extinguisher(x), iz.curtain(z))  t  S  James sprayed the fire extinguisher at the curtains.</pre>					
	P <sub>j</sub> e <b>NP</b> James	e				
	<pre>λz.λy.spray(y, ιx.extinguisher(x), z)</pre>		e PP at the curtains			
	λz.λy.λx.spray(x, z, y) <e, <e,="" t="">&gt;&gt;  V  sprayed</e,>	tx.extinguisher(x)  OP  the fire extinguisher	XX.X <e, e=""> P at</e,>	e DP the curtains	e DP	
		AP.ix.P(x) <e, t="">, e&gt;  D  the  Ax.extinguisher(  <e, t="">  NP  fire extinguisher(</e,></e,>		\( \lambda P.ix.P(x) \\	>	

