Semantic Type : Logical Form Template	Syntactic Category	Dependency Parse Pattern	Example
$c : \lambda x.\text{M}(x, w, c)$	N	$w =$ name for entity e	London := N : CITY : $\lambda x.\text{M}(x, \text{“london”}, \text{CITY})$
$c : \lambda x.c(x)$	N	$e \Rightarrow$ [form of to be] $\Leftarrow$ w [NN] $e \xleftarrow{appos}$ w [NN]	town := N : CITY : $\lambda x.\text{CITY}(x)$
$r : \lambda g.\lambda f.\exists x, y.f(x) \wedge g(y) \wedge r(x, y)$	$(S[Z]\backslash N_1)/N_2$	$e_1 \xrightarrow{nsubj}$ w $\xleftarrow{dobj} e_2$	bought := $(S[\text{dcl}]\backslash NP_1)/NP_2$: ACQUIRED : $\lambda g.\lambda f.\exists x, y.f(x) \wedge g(y) \wedge \text{ACQUIRED}(x, y)$
	$(S[Z]\backslash N_1)/PP_2$	$e_1 \xrightarrow{nsubj}$ w $\xleftarrow{prep}$ [IN, TO] $\Leftarrow e_2$	established := $(S[\text{pss}]\backslash NP_1)/PP_2$: HEADQUARTEREDIN : $\lambda g.\lambda f.\exists x, y.f(x) \wedge g(y) \wedge \text{HEADQUARTEREDIN}(x, y)$
$r : \lambda g.\lambda f.\lambda x.\exists y.f(x) \wedge g(y) \wedge r(x, y)$	$(NP_1\backslash NP_1)/PP_2$, $(N_1\backslash N_1)/PP_2$	$e_1 \xleftarrow{partmod}$ w [VBN] $\Leftarrow$ [IN, TO] $\Leftarrow e_2$	born := $(NP_1\backslash NP_1)/PP_2$) : PERSONBORNINLOCATION : $\lambda g.\lambda f.\lambda x.\exists y.f(x) \wedge g(y) \wedge \text{PERSONBORNINLOCATION}(x, y)$
$c : \lambda f.f$	PP/NP_1	(prepositions in the above patterns)	in := PP/NP_1 : LOCATION : $\lambda f.f$
$r : \lambda g.\lambda f.\lambda x.\exists y.f(x) \wedge g(y) \wedge r(x, y)$	$(NP_1\backslash NP_1)/NP_2$, $(N_1\backslash N_1)/N_2$	$e_1 \Leftarrow$ w [IN, TO] $\Leftarrow e_2$ $e_1 \Leftrightarrow$ [be] $\Leftrightarrow$ [NN] $\Leftrightarrow$ w [IN, TO] $\Leftrightarrow e_2$ $e_1 \xleftarrow{appos}$ [NN] $\Leftrightarrow$ w [IN, TO] $\Leftrightarrow e_2$	in := $(N_1\backslash N_1)/N_2$: MUSICIANINMUSICARTIST : $\lambda g.\lambda f.\lambda x.\exists y.f(x) \wedge g(y) \wedge \text{MUSICIANINMUSICARTIST}(x, y)$
$r : \lambda g.\lambda f.\lambda x.\exists y.f(x) \wedge g(y) \wedge r(x, y)$	$(NP_1/N_1)\backslash NP_2$	$e_1 \xleftarrow{poss}$ w $\Rightarrow e_2$ $e_1 \Rightarrow$ [be] $\Leftarrow$ [NN] $\xleftarrow{poss}$ w $\Rightarrow e_2$	’s := $(NP_1/N_1)\backslash NP_2$: DIRECTORDIRECTEDMOVIE^{-1} : $\lambda g.\lambda f.\lambda x.\exists y.f(x) \wedge g(y) \wedge \text{DIRECTORDIRECTEDMOVIE}(y, x)$
EQUALS : $\lambda g.\lambda f.\exists x, y.f(x) \wedge g(y) \wedge x = y$	$(S[\text{dcl}]\backslash N_1)/N_2$	[forms of “to be”]	is := $(S[\text{dcl}]\backslash N_1)/N_2$: EQUALS : $\lambda g.\lambda f.\exists x, y.f(x) \wedge g(y) \wedge x = y$
Create binary rule , $N \rightarrow N\backslash N$, where the returned logical form is: $\lambda f.\lambda x.\exists y.f(x) \wedge g(y) \wedge r(x, y)$ (where g is the logical form for the initial N)		$e_1 \xleftarrow{appos} e_2$	, $N \rightarrow N\backslash N$: $\lambda f.\lambda x.\exists y.f(x) \wedge g(y) \wedge \text{CITYLOCATEDINSTATE}(x, y)$ (e.g., to parse “Sacramento, California”)

Table 1: Dependency parse heuristics for CCG lexicon induction. Each heuristic is applied to a sentence containing mentions of a pair of entities e_1, e_2. The logical form templates on the left contain categories c and relations r that are filled using predicates from the knowledge base such that $c(e)$ and $r(e_1, e_2)$. Dependency parse patterns are specified using directed parse tree edges, where bidirectional edges match either edge orientation. Bracketed expressions contain part-of-speech tag restrictions for the matched token. Example lexicon entries produced by applying the patterns are shown on the right. Noun lexicon entries generated using the second rule must be generated at least twice in the corpus to be included in the lexicon.