Computational Linguistics

LIN 567/467 – Spring 2018

Project

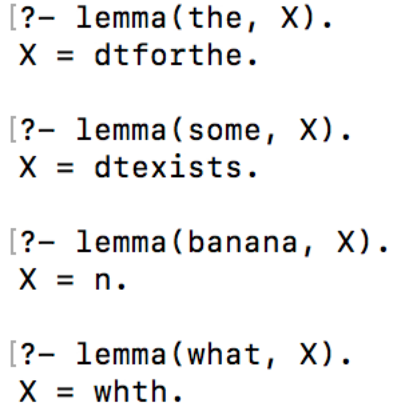
Shefali Sharma: 50247677
Priya Murthy: 50248887

We have developed a ‘smart refrigerator system’ that can interact with the user and allows the user to pass queries(declarative or a question) to the system.

2.1 Grammar

Lemmas

We use lemmas to define the words that are present in the smart refrigerator’s vocabulary. We use it to identify the Parts-of-speech (POS) tagging - to mark words against their respective POS tags.

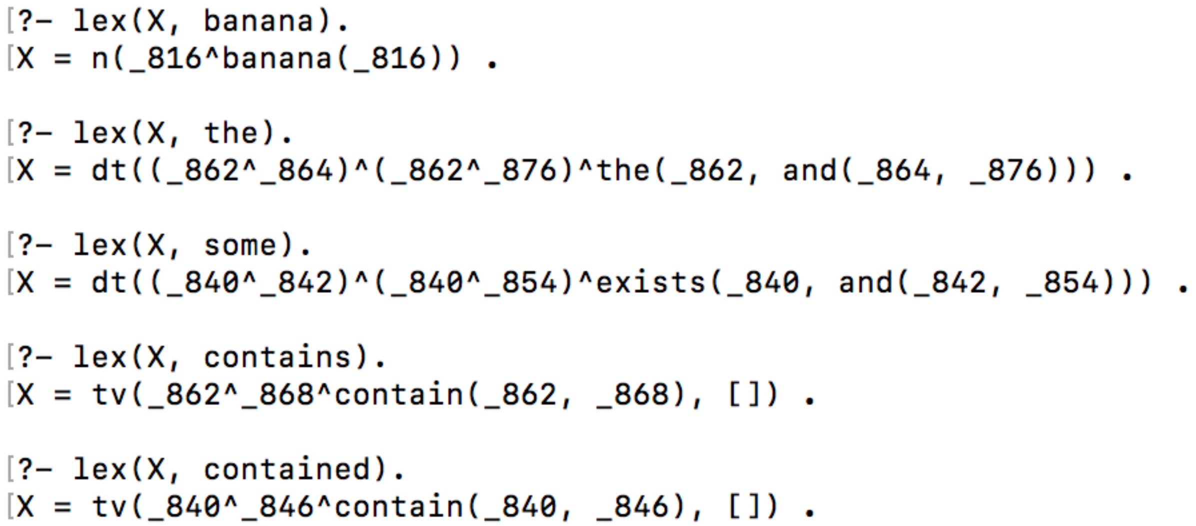


```
[?- lemma(the, X).
 X = dtforthe.

[?- lemma(some, X).
 X = dtexists.

[?- lemma(banana, X).
 X = n.

[?- lemma(what, X).
 X = whth.
```

Lex

Generating lex to represent semantic rules for different words against each tag.

We have defined lex for transitive verbs, intransitive verbs, nouns, adjectives, etc.



```
[?- lex(X, banana).
[X = n(_816^banana(_816)) .

[?- lex(X, the).
[X = dt((_862^_864)^(_862^_876)^the(_862, and(_864, _876))) .

[?- lex(X, some).
[X = dt((_840^_842)^(_840^_854)^exists(_840, and(_842, _854))) .

[?- lex(X, contains).
[X = tv(_862^_868^contain(_862, _868), []) .

[?- lex(X, contained).
[X = tv(_840^_846^contain(_840, _846), []) .
```

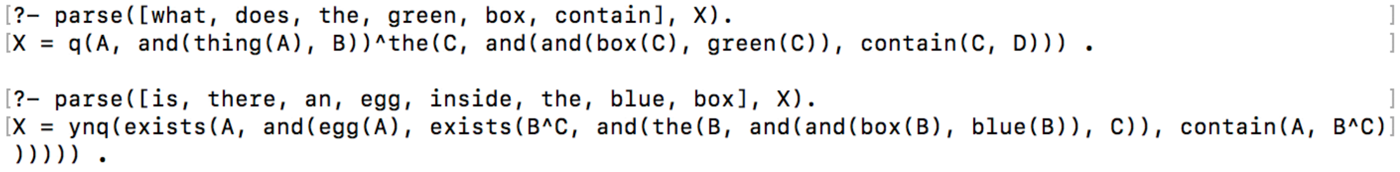
2.2 Semantic Analysis

Rules

We define rules to develop a way for the system to combine the lex and make meaningful sentences.

Parser

We have made use of the Shift Reduce parser to define our parser. It parses the recent parse trees and joins them to form a new root symbol. It creates the First Order Logic.

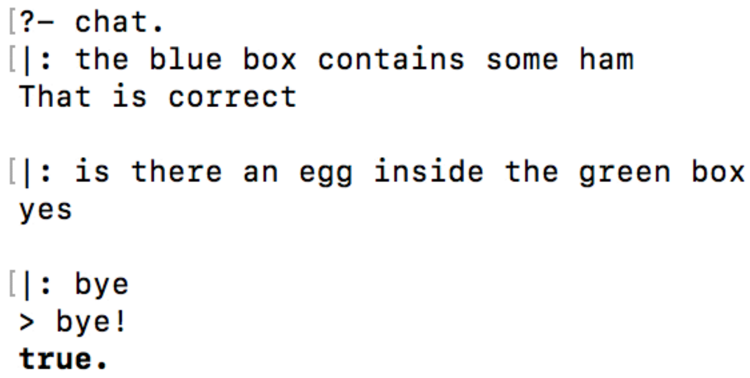


```
[?- parse([what, does, the, green, box, contain], X).                                                ]
[X = q(A, and(thing(A), B))^the(C, and(and(box(C), green(C)), contain(C, D))) .                      ]

[?- parse([is, there, an, egg, inside, the, blue, box], X).                                          ]
[X = ynq(exists(A, and(egg(A), exists(B^C, and(the(B, and(and(box(B), blue(B)), C)), contain(A, B^C)]
 ))))) .
```

2.3 Model checker

Model, ModelChecker and Response

We first create a model for our refrigerator. It is the data that our refrigerator uses to interact with the user. It validates and queries the model to get appropriate answers for the users.
For each query, our system, answers in ‘Yes’ or ‘No’, or gives the output as required.



```
[?- chat.
[|: the blue box contains some ham
 That is correct

[|: is there an egg inside the green box
 yes

[|: bye
 > bye!
 true.
```