CS 6364. 001

Question-Answering System

Radhika Kulkarni

(rxk180002)

# Description of the System

The project aims at developing knowledge base to answer a series of questions based on a preliminary statement. This project has been adapted from Problems 12.5 and 12.6 of Artificial Intelligence – A Modern Approach textbook by Russell ad Norvig.

Initially, the content of the preliminary sentence is represented as a series of assertions. To answer the questions, background knowledge has been added to the knowledge base. The formalized domain is enough to answer a series of questions about the information in the sentence:

*‘Yesterday John went to the North Berkeley Safeway supermarket and bought two  
pounds of tomatoes and a pound of ground beef.’*

The knowledge base contains a set of assertions which are queried sequentially by Prolog. Each query of the statement logically follows the previous one, and on successfully satisfying all the necessary conditions. Prolog is a programming language particularly well suited to logic and artificial intelligence programming. In Prolog, the program logic is expressed in terms of relations, represented as facts and rules.

The logical reasoning system demonstrates the sufficiency of the knowledge base. The knowledge base is especially designed to answer questions related to the ‘meat’ in the supermarket. The knowledge base considers the hypernyms, hyponyms and holonyms of ‘meat’, ‘fish’, ‘chicken’, ‘beef’. A python script file is written to extract these features using WordNet.

# Knowledge Base

* John shopped beef at North Berkeley Safeway yesterday.

shop(john,beef,northBerkeleySafeway,yesterday).

* John shopped tomatoes at North Berkeley Safeway yesterday.

shop(john,tomatoes,northBerkeleySafeway,yesterday).

* John lives alone

livesAlone(john).

* Anyone who lives alone shops alone

shopsAlone(X):-livesAlone(X).

* If someone shops alone and if he shops then he is adult

isa(X,adult):-shop(X,\_,\_,\_),shopsAlone(X).

* If someone is not an adult, then he is a child

isa(X,child):- !,\+ isa(X,adult).

* If someone is not an adult, then he is a child

isa(X,child):- isa(X,adult), !, fail.

* Tomatoes have a unit weight of 0.5 pounds

unitWeight(tomatoes,0.5).

* If John shopped for beef, he bought 1 pound of beef

pounds(john,1,beef):-shop(john,beef,northBerkeleySafeway,yesterday).

* If John shopped for tomatoes, he bought 2 pounds of tomatoes

pounds(john,2,tomatoes):-shop(john,tomatoes,northBerkeleySafeway,yesterday).

* Mary shopped for tomatoes, she bought 1 pound of tomatoes

pounds(mary,1,tomatoes):-shop(mary,tomatoes,northBerkeleySafeway,yesterday).

* The number of items that someone has, is obtained by the no. of pounds/unit weight.

number(P,X,I):-unitWeight(I,Y),pounds(P,Z,I),X is Z/Y.

* If N>=M then N is larger or equal to M.

largerorequal(N,M):-N>=M.

* If someone has some number of items and if it is larger or equal to some number then he has at least that many number of tomatoes.

hasatleast(P,Y,I):- number(P,X,I),largerorequal(X,Y).

* If someone shopped for something somewhere sometime, then they bought that thing.

bought(X,Y):-shop(X,Y,\_,\_).

* If John bought something and if that thing is meat then John bought meat

buys(X,Y):-bought(X,Z), typeOf(Z,Y).

* Mary was buying tomatoes at North Berkeley Safeway at the same time as john.

shop(mary,tomatoes,northBerkeleySafeway,T):-shop(john,tomatoes,northBerkeleySafeway,T).

* If someone buys something at some store at sometime then that person is at that store at that time.

isAtStore(X,Y,Z):-shop(X,\_,Y,Z).

* If someone buys something at some stall at sometime then that person is at that stall at that time.

isAtStall(X,Y,Z):-shop(X,Y,\_,Z).

* If someone is at some stall at some store, and if someone else is also at the same stall and same store, then that person sees the other person

sees(X,V):-isAtStore(X,Y,Z),isAtStall(X,U,Z),isAtStore(V,Y,Z),isAtStall(V,U,Z).

* Or else they will not see each other.

sees(X,V):-!, \+isAtStore(X,Y,Z); \+isAtStall(X,U,Z); \+isAtStore(V,Y,Z); \+isAtStall(V,U,Z), fail.

* If X sees Y then Y also sees X.

sees(Y,X):-sees(X,Y).

* If something is a vegetable then it originates from nature.

originates(X,nature):-typeOf(X,vegetable).

* If something is a skincare then it does not originate in nature.

originates(X,nature):-typeOf(X,skincare), !, fail.

* If something is of form something which orginates from man, then it is made in the supermarket.

made(X,supermarket):- !,\+originates(X,nature).

* If something is a Vegetable or meat, it can be eaten.

can(X,eaten):-typeOf(X,vegetable);typeOf(X,meat).

* Precedence of days

after(yesterday,daybeforeyesterday).

after(today, daybeforeyesterday).

after(tomorrow, daybeforeyesterday).

after(today,yesterday).

after(tomorrow,yesterday).

after(tomorrow,today).

* Someone has something at sometime if he has shopped for it.

has(X,Y,T):-shop(X,Y,\_,T).

* Someone eats something at sometime if someone has something at that time and if that thing can be eaten.

whatWill(X,Y,T,eatThem):- has(X,Y,K), can(Y,eaten),after(T,K).

* Safeway has a branch called North Berkeley Safeway supermarket

hasbranch(safeway,northBerkeleySafeway).

* Skin care products, vegetables, meat are available in supermarkets.

available(skincare,supermarket).

available(meat,supermarket).

available(vegetable,supermarket).

* Something sells something if it has a branch which is a supermarket which has those class of products

sells(X,Y):- hasbranch(X,U),typeOf(U,supermarket),typeOf(Y,Z),available(Z,supermarket);

hasbranch(X,U),typeOf(U,supermarket),available(Y,supermarket).

* If anyone shops at someplace for anything at some times then they pay money for that thing at that time at that place.

pay(X,Z,Y,T):-shop(X,\_,Y,T).

* If someone has to pay money at sometime at a store which is of some type then they must bring that money to that place at that time.

bring(X,Z,Y,T):-pay(X,Z,K,T),typeOf(K,Y),typeOf(Z,money).

* If anyone shops at someplace for anything at some times then they pay money for that thing at that time at that place.

If someone who pays money at sometime at someplace of some type then he spent money that time in that type of place

spentMoney(X,Z,T) :-pay(X,W,Y,T),typeOf(Y,Z),typeOf(W,money).

* Define after and before yesterday

after(after,yesterday).

after(yesterday,before).

* If someone spend money at someplace at some time then he had less money after that time at the same place.

moneyLeft(W,less,Y,T):- spentMoney(W,Y,U),after(T,U).

* If someone shops meat then he is non vegeterian

isNonVeg(X):- shop(X,Y,\_,\_),typeOf(Y,meat).

* If someone is not a non vegeterian then he is vegeterian

isVeg(X):-!,\+isNonVeg(X).

* If someone shopped 1 pound of something, then he bought 16 ounces of that thing

ounces(X,Y,Z):-pounds(X,W,Z), Y is W\*16.

* If John has some beef and if it is larger or equal to some number then he has atleast that many ounces of beef.

hasAtleastOunces(X,U,Y):- ounces(X,V,Y),largerorequal(V,U).

* Capacity of John's car trunk is 4

carTrunkCapacity(john,4).

* Volume occupied by each pound of tomato is 1

volume(tomatoes,1).

* Volume occupied by each pound of beef is 1

volume(beef,5).

* Space occupied by the item bought by a person is number of pounds of item bought times the volume occupied by the item

spaceOccupied(P,I,X):- pounds(P,W,I),volume(I,V), X is W\*V.

* If the space occupied by the number of pounds of item bought is than or equal to car trunk capacity of the person then the item fits in car.

fitInCar(P,W):- spaceOccupied(P,W,Y),carTrunkCapacity(P,X),largerorequal(X,Y).

* If a person shops at supermarket then the supermarket is open at that time.

If the supermarket is open then the staff is present

Hence when a person is at the supermarket and shops any item then there are other people in safeway

isOpen(P,L, T):- isAtStore(P,L,T).

staffPresent(P,L,T):-isOpen(P,L,T).

peopleAtStore(P,L,T):-staffPresent(P,L,T).

owns(X,Y):-sells(X,Y).

%ownsP(P,I,L):-shop(P,I,L,\_).

%person(john).

%person(mary).

%owns(P,I,L):-person(P)->ownsP(P,I,L);ownsS(L,I).

% North Berkely Supermarket is a type of Supermarket.

typeOf(northBerkeleySafeway,supermarket).

% Soap is a type of skin care product.

typeOf(soap,skincare).

%Tomatoes are type of vegetables

typeOf(tomatoes,vegetable).

%Deodrant is type of skincare

typeOf(deodorant,skincare).

typeOf(creditcard,money).

typeOf(money,money).

typeOf(cash,money).

%Yesterday John went to the North Berkeley Safeway supermarket and bought two

%pounds of tomatoes and a pound of ground beef

# Questions

# Section 12.5

1. **Is John a child or an adult?**

Ans:

Explanation:

John lives alone.

Anyone who lives alone shop alone.

So John shops alone.

John shops tomatoes and beef at North Berkeley Safeway.

If someone has to shop alone and shops then its an adult.

Hence John is an adult.

Prolog Demonstration:

