

Homework 6: Logic Programming II

Introduction	1
Code	1
Testing.....	2

Introduction

This report introduces a logic program designed to translate sentences from Spanish to English. The program operates on predefined rules and facts, ensuring accurate translations while maintaining word order and handling stop words. By leveraging Prolog, a declarative programming language, the report presents the implementation and some tests proposed in the statement of the provided task of the translation algorithm. The lesson learned from this work is, mainly, the use of cut operator and manipulation of lists.

Code

First of all, the facts are defined in the way they are provided, as a predicate *wordtrans(original, translation, grammatic category)*. The other predicates are stop words that, when they appear, the translation is automatically stopped and returned.

A base case for the recursion is defined in order to avoid infinite loops. After that, if the analyzed word is a stopword, then it returns an empty list and translation until this point is returned.

In English, adjectives go before nouns. This is solved by checking bigrams and their POS Tags so that, in case the original language has a noun preceding an adjective, they are translated in reverse order.

Finally, in the las lines of code the translation is performed so that, if the word is in our knowledge base, then its corresponding translation is returned. Otherwise, it is replaced by “(?)” symbol.

The cut operators are used as green cuts, i.e., they are not necessary for the program to run (red cuts) but they are included to make it more efficient and avoiding backtracking in our rule-based cascade of conditions.

```


wordtrans(casa,house,noun).
wordtrans(grande,big,adjective).
wordtrans(roja,red,adjective).
wordtrans(marron,brown,adjective).
wordtrans(una,a,article).
wordtrans(la,the,article).
stop(vaca).
stop(foca).

% When both lists are empty, then it is finished
senttrans([],[]).
% If it's a stop word, immediately stop translation
senttrans([S|_], []) :- stop(S), !.
% In case of having a noun followed by an adjective, whitch them and go to the
% next rule.
senttrans([S1, S2 | TS], [E2, E1|TE]) :- wordtrans(S1, _, noun),
                                         wordtrans(S2, _, adjective),
                                         senttrans([S2, S1 | TS], [E2, E1|TE]),
                                         !.
% Check if there is a translation for each of the word and move forward.
senttrans([S|TS], [E|TE]) :- wordtrans(S, E, _), senttrans(TS, TE), !.
% Finally, if it is not a stopword and it is not in KB, then translate it as (?).
senttrans([_|TS], ["(?)"|TE]) :- senttrans(TS, TE).

```


Testing

The tests proposed in the problem statement have been successfully passed by our code. In the image below, in the left we have the expected results and in the right the ones obtained by us. As we can see, they are equal.

 <code>senttrans([una,casa,roja,grande],P).</code>	 <code>senttrans([una, casa, roja, grande], P)</code>
<code>P = [a, red, big, house]</code>	<code>P = [a, red, big, house]</code>
 <code>senttrans([una,casa,roja],P).</code>	 <code>senttrans([una, casa, roja], P)</code>
<code>P = [a, red, house]</code>	<code>P = [a, red, house]</code>
 <code>senttrans([una,vaca,roja],P).</code>	 <code>senttrans([una, vaca, roja], P)</code>
<code>P = [a]</code>	<code>P = [a]</code>
 <code>senttrans([],P).</code>	 <code>senttrans([], P)</code>
<code>P = []</code>	<code>P = []</code>
 <code>senttrans([una,casa,amarilla],P).</code>	 <code>senttrans([una, casa, amarilla], P)</code>
<code>P = [a, house, (?)]</code>	<code>P = [a, house, "(?)"]</code>
 <code>senttrans([esta,pelota,azul],P).</code>	 <code>senttrans([esta, pelota, azul], P)</code>
<code>P = [(?), (?), (?)]</code>	<code>P = ["(?)", "(?)", "(?)"]</code>

Some extra tests proposed by us

When there are 2 stop words, the first one is the one which stops the process

 `senttrans([una, foca, y, una, vaca], P)`

P = [a]

Stop word works even when it's in the dictionary, indicating the priority of these:


`wordtrans(vaca, cow, noun).`
`stop(vaca).`
`stop(foca).`

 `senttrans([una, vaca], P)`

P = [a]

A grammatic rule has been introduced so that, when a word can be either noun or adjective and is followed by a noun, the adjective is chosen.

`wordtrans(roja, red, noun).`
`wordtrans(roja, red, adjective).`

 `senttrans([una, casa, roja], P)`

P = [a, red, house]