A LogicWeb Crawler for Python

Ian Buitenkant, *Stony Brook University*

In this project, I partially implemented a tool that will allow a user to download and query LogicWeb programs from the internet. LogicWeb is an abstraction of the internet as a logic program proposed by Seng Wai Loke and Andrew Davison in 1997. In LogicWeb, web pages are rephrased as logic programs and hyperlinks between pages are treated as relationships. Using this abstraction, we can determine many more complex relationships between pages on the internet.

One of the defining characteristics of LogicWeb programs is the concept of "context". For example, if a webpage defines a LogicWeb rule of:

```
interested_in(X):-
      related(X,Y),
      interested(Y).
```

then a reader can only apply this rule to the "context" of the page on which it is defined. If the rules of multiple pages are all loaded into a LogicWeb interpreter, it must make sure that this rule only applies to the pages that define it.

While I was not able to fully implement the concept of context switching in PyLW, I was able to accomplish something similar by allowing the user to specify which programs to load from the internet. Using the `load` command will store a program in the program store and it will be queried in all future queries

Another important aspect of LogicWeb is the ability to make composite programs and queries. For example, the LW-union operator (+) can be used to query the union of two LW programs:

```
?- lw(''URLI'')+lw(''URL2''))#>p(X)
```

The above query will evaluate the goal `p(X)` in the context of the union of the two pages' programs. I have implemented by union and intersection operators in PyLW to emulate this behavior. When a user runs the `query_u` command, they will see the union of the results of running the associated query on all loaded programs. Likewise, the `query_i` command allows the user to find the intersection of the solutions found by the interpreter.

Limitations

This implementation of LogicWeb has some limitations that distinguish it from the full LogicWeb language and architecture. First, I will assume that the user is only interested in using the tool on static web pages and not on postable forms or dynamically loaded content. The original LogicWeb implementation includes functionality for interact with server side databases through the `lw(post, ``URL``)` predicate. For the sake of time management, this project will focus on the ability to query static pages.

Another limitation of this project is that the lack of a fully-fledged resolver for LogicWeb. Instead, have only developed the ability to enumerate results of a LogicWeb query over a composition of LW programs. For future work, it would be interesting to see if more dynamic programs could be interpreted, such as those that direct to other linked websites. Since no current implementation of LogicWeb exists in a usable form, it would be helpful to see how some of Loke and Davison's designs hold up in the internet today, since many things have changed since the publication of their paper.