Dan Roberts

CS 497 Honors RS Project

10/31/12

Mid-semester progress report

**Project Title:** PLGraph

**Advisor:** Zhiyuan Li

**Question:** Is there a significant and useful difference between the graphs of the grammars of

different programming languages, and if so, what does this tell us about the languages in

question?

**Project Description:** When a parser is compiled using a parser-generator such as yacc or bison, one of the intermediate stages involves a transition graph, which the generator can make a copy of. By analyzing these structures from a graph-theoretical standpoint, I will examine quantitative relationships between languages. I will then compare these relationships to qualitative descriptions provided by language designers and expert users, and attempt to combine the two sources of information into a common thread.

**Project Deliverable:** A short conference/workshop-style paper detailing my methods and results.

**Work Accomplished So Far:**

1. Read: learning about current efforts in language analysis, figuring out what graph metrics would be useful to test, and remembering my basic statistics.
2. Acquire language graphs: Yacc/Bison is the most convenient method for generating such a graph, but the parsers of many languages are not written in yacc, so I've had to search for suitable languages. This is especially true of modern languages like Haskell and Erlang, which tend to be self-compiling, and of very old languages like FORTRAN, which were designed before BNF was invented, and as such aren't amenable to expression in that way.
3. Build a language graph analyzer: I used python (particularly the networkx graph library) to parse the files output by Yacc/Bison and derive basic metrics from them.
4. Acquire qualitative descriptions: This has mostly involved poking around on project websites and Wikipedia to fill in a spreadsheet.
5. Begin comparing attributes: Using the scipy and matplotlib libraries for Python, I've been working to tie the two sets of attributes together. Most of the work here has just been learning the tools, which I haven't used extensively before.

**TODO List:**

1. Finish hashing out my statistical methods (this week)
2. Build a BNF parser: though many languages are not compiled using Yacc/Bison, almost all have a full BNF or E-BNF grammar available online. If I were to build something to turn these into transition graphs, I would greatly expand my dataset. I estimate this to be at least 20~30 hours of work, so it probably won't be done before Thanksgiving.
3. Write the paper (start the week of Thanksgiving, end by presentation date)
4. Put together the presentation (parallel to or slightly after the paper). The difficult part here is going to be connecting to my audience – this is pretty abstract stuff, and I want to make sure people aren't falling asleep.
5. Submit to a conference or workshop (pending advisor approval)