1 Introduction

This document gives basic information about the graduate module LG519.

2 Overview, Aims, Assessment, etc.

This course provides an introduction to the programming language Prolog and its use for Natural Language Processing (NLP), and so to fundamental issues in NLP.

Teaching Method: The course involves both lectures/seminars, where the ideas and techniques are introduced and discussed, and supervised laboratory sessions where students can see what the ideas and techniques mean in practice and develop a thorough practical appreciation.

Students are expected to attend the weekly lecture/seminar and at least one of the lab sessions every week. They are also expected to spend several hours a week in the labs at other times.

The aims are:

- To introduce basic ideas of Prolog, and to get used to writing, debugging, and documenting simple Prolog programs;
- To introduce the basic idea of the Definite Clause Grammar (DCG) formalism, which is closely related to Prolog, and to learn about and gain some facility in the standard techniques;
- To appreciate the strengths and limitations of DCGs;
- To develop an appreciation of the basic ideas and techniques of Phrase Structure description, on which DCGs depend;
- To develop a basic understanding of key ideas in parsing at both theoretical and practical levels.

Learning outcomes By the end of the course, you will have a sound understanding of some key issues in NLP from both theoretical and practical viewpoints. You will have a good basic understanding of the core parts of Prolog, and be able to write programs for a variety of tasks. You will have written programs which can parse (and produce surface syntactic representations of) fragments of English. As a by-product you will have extensive experience of working in a Unix and X-based environment, and using a variety of computer tools (text processors and formatters, mail systems, web browsers).

Assessment Assessment is by two pieces of course work (counting 50% each).

The course work assignments are given below.

Students are reminded of the existence of penalties for late submission of course work (see the Departmental postgraduate handbooks).
3 Topics

1 Basics: syntax and semantics of Prolog. Matthews (1998), Rogers (1986, 1,2)
2 Facts, rules, recursion, processing, debugging. Matthews (1998), Rogers (1986, 3,4)
3 List Processing Matthews (1998), Rogers (1986, 5)
4 Linguistic Knowledge: Grammars, Representation, Constituency. Borsley (1991),
5 Definite Clause Grammars (DCGs) (1) Matthews (1998), Gazdar and Mellish (1989, 4,5),
Basics: CFGs, recognition, generation parsing, Agreement. Matthews (1998), Pereira and Shieber (1987, 1,2),
7 DCGs (3): Subcategorization "
8 DCGs (4): Long distance dependencies, etc. "
9 Evaluating Grammar Formalisms, Limitations of DCGs. "
10 Towards full NLP "


4 Reading and Course Material

There is a web page for this course:

http://courses.essex.ac.uk/lg/LG519/

which contains course notes materials, exercises, and assorted demonstration programs.

Especially recommended is Matthews (1998), which is very accessible, and straightforward, and covers many topics that will be dealt with on the course.

Supplementary material for the first part of the course can be found in the early chapters of almost any introduction to Prolog (see below). Rogers (1986), which is sadly long out of print, is especially accessible. Once the most important features and techniques of Prolog programming have been introduced, the course concentrates on the use of use of Definite Clause Grammars (DCGs) for analyzing various constructions of English (DCGs are a simple extension of Prolog). Background material for this part of the course can be found in Covington (1994), Gazdar and Mellish (1989), Pereira and Shieber (1987), Smith (1991), and Gal et al. (1991) (as well as Matthews (1998), of course).
The bibliographic search tools at the following site may be useful:

http://www.essex.ac.uk/linguistics/clmt/bibsearch

A ‘Reader’ containing a collection of useful articles and excerpts from relevant books will be made available in week 2. It contains the following:


A much longer list of useful references can be found at the end of this handout.

5 Assignments

The following are the course work assignments for both the graduate module LG519, and the undergraduate half-course LG419. They are weighted equally. For undergraduates taking LG419 there is also an in-course test which accounts for half the mark (so, for undergraduates, the final mark is made up of as follows Assignment 1: 25%; Assignment 2: 25%; in course-test: 50%).

5.1 Assignment 1

This is the first of two assignments for LG419/LG519. It counts for 50% of the coursework marks on LG419, and 50% of the total marks on the graduate module (LG519). There are 10 parts to the assignment – do all of them. The deadline is 4:00pm on the Monday of week 6.

1. Give brief definitions, with examples, of the following Prolog concepts:
   - atom, variable, compound term, clause, comment, procedure;
   - matching, unification;
   - database, query, goal.
2. There are three syntax errors in this definition. Write out a correct form, indicating them.
bugs([]).
bugs([Head|Tail]) :-
    write(Head),
    bugs(Tail)

3. Suggest English paraphrases for the various clauses in the following database:

stupid(X) :- man(X).
stupid(X) :- dog(X).

man(X) :- male(X), human(X).
dog(X) :- male(X), canine(X).

male(socrates).
male(rover).

human(socrates).
canine(rover).

4. Given this database, describe how Prolog would go about solving the following query:

?- stupid(rover).

5. What does this predicate do:

mystery([],).
mystery([H|T]) :-
    mystery(T).

6. Describe what the following ‘built in’ Prolog predicates do: consult, halt, listing, trace, spy, nospy.

7. Draw out a family tree consisting of at least four generations, and containing at least 12 people, and describe this tree by means of a database of prolog facts using a 2 place predicate *parent*, and predicates *male* and *female*.
(For this you should hand in a drawing of the tree, and a print out of the database, i.e. some prolog code)

8. Write prolog queries to find out the following from this database:

the names of people who are male;
the names of people who are parents of somebody or other
the names of people who are both male and parents of somebody;
if anybody is both male and female
(Hand in some prolog queries, they can be handwritten, but should be correct).

9. Write *brief* English definitions for the following English kinship terms:

- father
- mother
- grandparent
- sibling
- brother
- sister
- aunt
- uncle
- cousin
10. Write prolog rules to define these terms, and briefly note any discrepancies between the English definitions and the prolog definitions, or any other ways the definitions seem wrong. (For this you should give in a piece of commented prolog code)

5.2 Assignment 2

The deadline for this assignment is the last day of the Autumn Term. If you are a postgraduate, this counts for 50% of the marks on this module (lg519). If you are an undergraduate, it counts for 50% of the course work marks on lg419.

1. Write a phrase structure grammar for a fragment of English that includes the following:
   1. The Ns (or NPs): he, him, she, her, it, they, them, who, whom, what, and Sunday;
   2. A representative sample of Verbs (i.e. verbs with different subcategorization frames, including those that take embedded sentences);
   3. PP Adjuncts such as near them, on Sunday;
   4. Declarative Sentences;
   5. Yes/No Interrogatives;
   6. Wh-Interrogatives, including cases of ‘unbounded’ Wh-movement from embedded sentences (Who do you think Sam said he saw?).

2. Motivate the main features of the analyses which your grammar assumes, and discuss the limitations of your grammar.

3. Write a small set of test sentences with which you expect your grammar to accept/reject (“small” here means less than 30).

4. Write a Definite Clause Grammar that has the same coverage as the phrase structure grammar, and which builds a surface structure representation.

5. Show that the grammar works — produce a ‘log’ of the grammar working.


Some Notes of How to Do This.

The basic idea is to put together what you have learned this term. I am interested in:

- whether you can produce a reasonable description of a small fragment of English, some of which is familiar (VP), or easy (NPs containing just pronouns), but some of which will not be obvious to you (the structure of Interrogatives, perhaps);
- whether you can use the DCG techniques we have looked at, combining them together;
- whether you can present your grammar/DCG clearly with sufficient comments to make it/them plausible and comprehensible (NB there are marks for presentation in this);
- whether you can produce arguments to support what you have done, and see what is wrong with what you have done (even if you do not know how to put it right);
- whether you have some ideas about why DCGs are a good/bad thing – I want to make you reflect on your practical experience of writing DCGs.

If it helps, you might imagine that we are engaged in building some large scale grammar of English, and you have been charged with making a first attempt at this part of the grammar. Of course, you want to get as much of it right as possible, but it is an early stage, so you cannot be sure about all details of analyses that will be needed. You can expect someone else to have to take over your work later on, so you have to make it clear, and make it clear what is good/bad about what you suggest.

Here are some useful phrases:
“The grammar given above assumes that the structure of structure of interrogatives sentences is as in (1). Some motivation for this analysis can be seen in the examples in (2), which indicate that ....”

“Since it is not central to my analysis, I assume without discussion that expressions such as today are PPs, like on Tuesday. For the same reason, I do not consider the internal structure of PPs at all, simple listing a small sample in the lexicon.”

“The DCG grammar formalism offers the following advantages: ... There are however a number of problems, of which the following are the most obvious: ...

Don’t Waffle. Stick to the point. Do not start: “Before I present my grammar, it is necessary to define what a phrase structure grammar is, consider whether PSGs are adequate for the description of Natural Languages, and discuss the nature of grammaticality, and how one can test for constituency.” Similarly, when justifying the analyses, don’t go on about how there is a VP in English: you can take that for granted.

6 Other Readings

6.1 Books about Computational Linguistics


D. Dowty, L. Kartunnen, and A. Zwicky, editors. *Natural Language Parsing*. CUP, Cambridge, 1985


S. Shieber. *An Introduction to Unification Based Approaches to Grammar*. Number 4 in CSLI Lecture Notes. CSLI, Stanford University, Ca., 1986


6.2 Books about Prolog and other Tools


6.3 General Books about Linguistics/Grammar


