Introductory AI in the Informatics Curricula in Italy

Luigia Carlucci Aiello and Fiorella De Rosis
DIS-Università' di Roma La Sapienza
via Salaria 113
00194 Roma, Italy
aiello@assi.dis.uniroma1.it

Introduction

It is notorious that the Italian University is constrained by a heavy bureaucracy and very slow in introducing innovations. Hence, even though subjects related to AI have been taught in various courses within curricula of Informatics (Faculty of Science) or Electrical Engineering (Faculty of Engineering), the official introduction of a course named AI is very recent: it is contextual to the introduction of a curriculum in 'Informatics Engineering' in the Faculty of Engineering (1990).

AI courses are being taught in various Italian Universities, essentially in the Faculties of Engineering and Sciences, with few cases in the Faculties of Psychology. Recently the Italian Association for Artificial Intelligence AI*IA has collected some data on these courses and a workshop is planned, in the style of IIIA. Actually, the announcement of IIIA has been a reason to postpone the Italian workshop, in order to report on the results of IIIA to the Italian community.

Our contribution to IIIA is twofold:

(a) A short report on the analysis of the Italian situation at the Faculties of Engineering and Sciences performed by Fiorella De Rosis;

(b) A short description of the course being taught at the Faculty of Engineering of Rome La Sapienza (one of the three public universities of Rome) by Luigia Carlucci Aiello.

The goal is to present the selection of topics in the course and to discuss the motivations behind this choice. Should we have the courage to present the choice made as 'a successful strategy', how could we measure success in this case? The course is now being taught for the third year. Students seem to enjoy it and perform well in the seminars and at the final exam. Is this a success story?

Survey of the Italian situation

The considerations which follow result from a survey and analysis of the Introductory AI courses in the Faculties of Engineering and Sciences in Italy. Our analysis, presumably complete, counted 23 of them. The survey was promoted by the Italian Association of AI (AI*IA) and was made with the cooperation of teachers, who provided us with the relevant information (syllabus, tutorial documentation and description of laboratory works).

Informatics curricula offered at the Faculty of Engineering include two courses in the area of AI: two semesters of Artificial Intelligence and one or two semesters of Knowledge Engineering and Expert Systems (KEES). Although reasonably the second course should follow the first one, for several reasons this is not the case, in the few cases where KEES is offered students can follow KEES independently of the AI course.

Both courses are usually optional and they are offered at the end of the curriculum in Informatics; in many cases they are an introduction to subjects that the student will develop in a Master-level dissertation (our 'Tesi di Laurea'). Their content is therefore a function not only of the teacher's research interests, but also of concepts and methods that were introduced in previous courses of the curriculum. This dependence is clearly proved by the way the two main themes of AI - problem solving methods and representation formalisms - are dealt with. Although these subjects have the role of opening the course and unifying the subjects covered further on, the weight given to each of them depends on what the student is supposed to already know in the domain. If - like in some Engineering curricula - basic problem solving algorithms are part of the previous courses, the focus is mainly on heuristic methods (A*, AO*, Minmax, Alphabeta, ...). A more complete coverage of the subject is given otherwise. Similarly, when the AI course is not preceded by a Logic course - like in the Engineering curricula - AI has first to fill this gap. First order logic is therefore introduced as a language with its inference methods (the resolution principle being the pivot of this section). In several cases, the basic logic training is extended to non standard and non monotonic logics, default reasoning and truth maintenance methods.

As we will see when examining in more detail a specific experience (Section ?? of this note), this emphasis on logics has a consequence on programming languages taught and employed in laboratory works. Pro
log is the most common of them, immediately followed by LISP. Object oriented languages are not common.

Laboratory works tend to privilege programming of algorithms in these languages rather than the implementation of expert systems by existing shells. In well-structured experiences, students' programs are organised in the framework of an incrementally increasing set of tools which become useful instructional aids.

Subjects of laboratory work are related to more advanced or specialist subjects included in the course. The most common of them are: Uncertain and Metalevel reasoning, Planning, Learning, Natural Languages and Intelligent Interfaces, Distributed AI and, more recently, Neural Networks (when they are not dealt with in separate courses).

Some final considerations about textbooks and other tutorial documentation. Although the recent years have seen a flourishing of interests and research activities in the area of AI in Italy there are no Italian textbooks of AI. In addition, no book seems to fully satisfy expectations of instructors. Therefore, documentation suggested to students is a smorgasbord (to use this Workshop's terminology) of selected chapters from about 50 different books, integrated with articles from the main journals (Artificial Intelligence, IEEE on AI-related subjects and so on). The most frequently adopted books are the following: the Italian translation of Rich and Knight, the Genesereth and Nilsson's Logical Foundations of AI (the two most popular ones), Nilsson's Principles of AI and Jackson, Winston, Rich, Charniak and McDermott's textbooks. The consequence of this heterogeneity of tutorial documentation is that lecturers and their notes have to create a methodologic continuity among subjects, by unifying notational differences, filling up gaps, highlighting repetitions of the same subjects in different terms and so on.

Some remarks on the course of AI at the Faculty Engineering of the University of Rome 'La Sapienza'.

The course is now being offered for the third year. It is placed at the end (5th year) of a curriculum in Informatics Engineering, a mixture of a curriculum in Electrical Engineering and in Computer Science, when students have been exposed to several (mandatory) courses in programming, data structures, compiler construction, theoretical foundations, operation research, systems theory, control theory, electronics, etc. Students have usually been exposed to PASCAL, LISP, PROLOG and C in previous courses in addition, to set the context better, it is worth noting that students are also offered a (optional) course of robotics.

The syllabus of the AI course is:

1. Introduction: definition, some history.
2. Knowledge Representation

Formalisms and methods
Methods for dealing with non monotonicity,
Truth maintenance, uncertainty, time
3. Automated Deduction Resolution, natural deduction, sequent calculus, semantic tableaux
4. Problem solving; planning
5. Learning
6. Natural Language Understanding
7. Knowledge Engineering and Expert Systems

The emphasis is as follows 1/2 of the time is devoted to topics 1 to 3 (about forty hours), 1/4 of the time is devoted to topics 4 and 5 (about twenty hours), 1/4 of the time is devoted to topics 6 and 7 (about twenty hours).

The main emphasis of the course is Logics as a representation formalism: the programming language chosen for the laboratory work is PROLOG.

Each year a topic is selected and a "specilistic module" is taught on that topic. This module is usually offered also to post-graduate students. Last year the selected topic was 'Natural Language Processing', this year the topic is 'Representation of structured knowledge and concept description languages'. In addition, students deepen a particular topic of their choice and refer on it in a seminar to their colleagues, some of them participate in a programming project.

The suggested readings are:
2. Console, Lamma, Mello, Programmazione Logica e Prolog, UTET, 1991

These books are complemented with chapters from other books and with articles on specific topics. Lectures are mandatorily in Italian, readings in English are considered no problem.
Curricula / study plan. Exams and thesis. Graduation. Welcome. This new course is a great opportunity to study what is wanted in the real world of jobs. Internet companies, consulting companies, startups and high tech industries, public administrations, research centers. Take a look! Contact: datascience@math.unipd.it. Available places: European candidates or non-European candidates residing in Italy. Non-European candidates residing outside Italy: 20 (2 for candidates joining the Marco Polo Project). Duration 2 years. Language English. Machine-readable TU Informatics Curricula. Contribute to fsinf/curricula development by creating an account on GitHub. The Fachschaft Informatik maintains machine-readable versions of the TU Informatics Curricula because the faculty of informatics fails to do so. The data is maintained in XML on GitHub and converted to JSON for easier consumption. The data is provided "As is", without warranty of any kind. For the authoritative sources please consult the official curricula PDFs, the Übergangsbestimmungen and the TISS newsletter. If something seems off with an XML or JSON version, please open a GitHub issue. The index of all curriculas is available as HTML, XML and JSON. Additionally a JavaScript API is provided. Claudia d'amato curriculum vitae. updated at September, 7th 2016. Contents. 1. definition of similarity and dissimilarity measures for expressive knowledge representation formalisms adopted in the SW and formalization of their theoretical foundation. 2. analysis and formalization of supervised and unsupervised learning methods for the SW 3. analysis and formalization of methods for managing uncertain knowledge representation and.