

Knowledge Representation

Code: 106564
ECTS Credits: 6

Degree	Type	Year	Semester
2504392 Artificial Intelligence	OB	2	2

Contact

Name: Marco Schorlemmer
Email: wernhermarco.schorlemmer@uab.cat

Use of Languages

Principal working language: english (eng)
Some groups entirely in English: Yes
Some groups entirely in Catalan: No
Some groups entirely in Spanish: No

Teachers

Tommaso Flaminio

Prerequisites

To have taken the courses on Computational Logic, Fundamentals of Programming I, and Fundamentals of Programming II; and to have basic knowledge of the Python programming language.

Objectives and Contextualisation

Representing and reasoning with knowledge in computationally effective ways is an important and fundamental aspect of artificial intelligence. It determines the efficacy and efficiency of computational systems in their capacity for learning and problem solving. This course is an introduction to the main approaches and techniques for representing knowledge in computational systems in order to build knowledge-based systems.

The objective of the course is: to study logic-based and feature-based approaches to knowledge representation, which provide the formal structure and inference mechanisms; to analyse the foundations of ontologies, by which we define the entities that are we think are relevant in particular application domains; and to introduce the main computational techniques that allow the use of knowledge for learning, reasoning, and problem solving. The course also aims at examining the main difficulties of formally representing knowledge in computational systems and of deploying knowledge-based systems in real-world applications, tackling thereby the issues of uncertainty, vagueness, and change.

Competences

- Analyse and solve problems effectively, generating innovative and creative proposals to achieve objectives.
- Develop critical thinking to analyse alternatives and proposals, both one's own and those of others, in a well-founded and argued manner.
- Identify, understand and apply the fundamental concepts and techniques of knowledge representation, reasoning and computational learning for the solution of artificial intelligence problems.

- Introduce changes to methods and processes in the field of knowledge in order to provide innovative responses to society's needs and demands.

Learning Outcomes

1. Analyse and solve problems effectively, generating innovative and creative proposals to achieve objectives.
2. Conceptualise, model and solve planning problems.
3. Develop critical thinking to analyse alternatives and proposals, both one's own and those of others, in a well-founded and argued manner.
4. Model and solve problems where uncertainty is an essential component using Bayesian networks and different logic-based approaches.
5. Recognise and understand the scope and limitations of the most representative methods of knowledge representation.
6. Recognise the difference between uncertainty and imprecision.
7. Represent the knowledge of a domain through ontologies and know how to reuse this knowledge.
8. Understand the main knowledge representation schemes and their inference procedures, as well as how to select the most appropriate scheme to represent knowledge in a specific environment.
9. Weigh up the risks and opportunities of both your own and others' proposals for improvement.

Content

1. Introduction: Knowledge, Ontology, Representation. Knowledge-Based System. Knowledge Engineering.
2. Knowledge Representation with Logic: Vocabulary, Meaning. Basic, Complex, and Terminological Facts. Rules, Production Systems. Varieties of Logic.
3. Ontologies: Concepts, Properties, Individuals, Data. Taxonomies, Meronymy. Frames, Inheritance, Defaults. Description Logics. Knowledge Graphs.
4. Knowledge Representation with Feature Spaces: Feature Spaces, Conceptual Spaces. Prototype Theory of Concepts. Concept Similarity.
5. Uncertainty, Vagueness, and Degrees of Belief: Conditional Probability. Objective and Subjective Uncertainty. Belief Networks, Bayesian Inference. Vagueness, Fuzzy Sets.
6. Action and Planning: Situation Calculus. Actions, Fluents. Frame Problem. Planning in the Situation Calculus.

Methodology

The course follows a flipped-class methodology, complemented with exercises and practical engineering work.

Students are expected to autonomously complete readings or listen to video lectures on the theoretical contents of the course, so that theoretical lectures can be devoted to clarify and deepen the core theoretical concepts. This is complemented with problem solving seminars in which students are expected to do exercises and solve problems in team work, under supervision. Problems and exercises are either fully solved during problem seminars, or else some hints are given so that students can complete the problems autonomously. Finally, students are to carry out a knowledge engineering project in small teams, during which most of the content of the course is put into practice. In supervised practical sessions, the engineering methodology and development steps are introduced, as well as the necessary skills for using the programming and system development environments. The bulk of the engineering project, however, is carried out autonomously by the student teams.

The Virtual Campus platform (<http://cv.uab.cat/>) will be used to share teaching materials, deliver the engineering project, consult the marks of the subject, communicate with the teaching staff, etc.

Annotation: Within the schedule set by the centre or degree programme, 15 minutes of one class will be reserved for students to evaluate their lecturers and their courses or modules through questionnaires.

Activities

Title	Hours	ECTS	Learning Outcomes
Type: Directed			
Theoretical lectures	24	0.96	8, 5, 6
Type: Supervised			
Practical sessions on knowledge engineering	10	0.4	1, 3, 9, 7, 2, 4
Problem solving seminars	14	0.56	1, 3, 7, 2, 4
Type: Autonomous			
Development of knowledge engineering project	35	1.4	1, 3, 9, 7, 2, 4
Personal study and readings	35	1.4	8, 5, 6
Problem solving	24	0.96	1, 3, 7, 2, 4

Assessment

The evaluation of the level of achievement of the course takes into account the team work done in problem seminars and practical sessions, as well as the scientific and technical knowledge of the subject gained by the students. To carry out this evaluation, the following aspects are taken into account:

- the knowledge engineering project together with the required documentation submitted by the student teams
- the defenses of the knowledge engineering project to which the student teams will be summoned
- the exercises of student teams submitted in problem solving seminars
- the individual, face-to-face written exam to adequately and individually assess the level of achievement

Assessment Items and Their Relative Weight Towards the Final Grade

The final grade reflects the overall level of achievement for the course by each student, and it is determined, in the first call, by combining the marks of the various assessment items as follows:

60% of the grade corresponds to theory and problems, broken down as follows:

(a) 30% corresponds to the arithmetic mean of the marks of three written exercises that are carried out and delivered in small teams in the problem solving seminars (i.e., each exercise contributes 10% to the final grade).

(b) 30% corresponds to the mark of the written exam on theory and problems that is carried out individually on the day of the final examination.

40% of the grade corresponds to the knowledge engineering project, broken down as follows:

(c) 30% corresponds to the mark of the knowledge engineering project done in small teams, and based on a delivery and a face-to-face defense of the project, to which the attendance of all members of each team is mandatory.

(d) 10% corresponds to the mark of the written exam on the engineering project, which is carried out individually on the day of the final examination.

To pass the course in the first call, it is mandatory to obtain at least a mark of 5 in each one of the four assessment items (a), (b), (c), and (d).

In the second call it is possible to improve, by means of a written retake exam, the mark corresponding to all theory-and-problems assessment items (items (a) and (b), i.e., 60% of the final grade of the course) and the

mark of the written exam on the knowledge engineering project (item (d), i.e., 10% of the final grade).

To pass the course in the second call, it is mandatory to obtain at least a mark of 5 in each one of these two recoverable parts, in addition to having passed, with a minimum of 5, the knowledge engineering project done in teams (which is evaluated in one of the practical sessions, and therefore cannot be recovered in the second call).

Planning of Assessment Activities

Continuous-assessment dates such as problem-solving exercises and project presentations will be made public on the Virtual Campus and on lecture slides. Dates may change when necessary. Any such modification will always be communicated to students in class and through the Virtual Campus within the schedule set by the centre or degree programme.

Non-Assessment

The final grade of the student will be 'ABSENT' as long as the student has not been assessed more than 30% of the subject.

Honors

Awarding an honors degree is the decision of the teaching staff responsible for the subject. UAB regulations dictate that Honors can only be granted to students who have obtained a final grade equal to or greater than 9, and that only up to 5% of the total number of students enrolled can be awarded an Honors degree.

Plagiarism

Without prejudice to others that are deemed appropriate and in accordance with current academic legislation, irregularities committed by a student during an assessment activity can lead to a change of mark to 0. Assessment activities marked in this way and by this procedure will not be recoverable. If it is necessary to pass any of these assessment activities to pass the course, the student will not pass the course, with no opportunity to recover it in a second call in the same academic year. These irregularities include, among others:

- the total or partial copy of a practice, report, or any other evaluation activity;
- let others copy your exercises/exam/work;
- present a team work that has not been entirely done by the members of the team;
- present as own those materials produced by a third party, even if they are translations or adaptations, and in general works with non-original and exclusive elements of the student;
- use communication devices (such as mobile phones, smartwatches, tablets, etc.) during assessment activities, individual or in teams.

In case the student has committed irregularities in any assessment activity (and therefore it will not be possible to pass the course in a second call), the final grade of the course will be the lowest of the value 3 and the weighted average of the marks. In summary: copying, let others copy your work or plagiarising in any of the assessment activities is equivalent to a failure with a grade lower than or equal to 3.

Assessment Activities

Title	Weighting	Hours	ECTS	Learning Outcomes
Exam on the knowledge engineering project, individual	10%	1	0.04	1, 8, 5, 7, 4
Exam on theory and problems, individual	30%	2	0.08	1, 8, 5, 6, 7, 2, 4
Knowledge engineering project, in teams	30%	2	0.08	1, 8, 3, 9, 5, 7, 4
Problem-solving exercise, in teams (1)	10%	1	0.04	1, 8, 3, 9, 7

Problem-solving exercise, in teams (2)	10%	1	0.04	1, 8, 3, 9, 7, 4
Problem-solving exercise, in teams (3)	10%	1	0.04	1, 8, 3, 6, 2, 4

Bibliography

- Brachman, R. J. & Levesque, H. J. (2004) *Knowledge representation and reasoning*. 1st edition. Amsterdam: Morgan Kaufmann.
https://bibcercador.uab.cat/permalink/34CSUC_UAB/1eqfv2p/alma991010367030506709
- Sowa, J. F. (2000) *Knowledge representation: logical, philosophical, and computational foundations*. Pacific Grove: Brooks/Cole.
https://bibcercador.uab.cat/permalink/34CSUC_UAB/1eqfv2p/alma991004680439706709
- Gärdenfors, P. (2000) *Conceptual spaces: the geometry of thought*. Cambridge (Mass.): MIT Press.
https://bibcercador.uab.cat/permalink/34CSUC_UAB/1eqfv2p/alma991008913449706709
- van Harmelen, F. et al. (2008) *Handbook of knowledge representation*. 1st edition. Amsterdam: Elsevier.
https://bibcercador.uab.cat/permalink/34CSUC_UAB/1eqfv2p/alma991010507167406709

Software

The knowledge engineering project will make use of freely available and downloadable programming and development tools available for the main operating systems (Windows, macOS, and Linux) such as the ontology editor Protégé (<https://protege.stanford.edu/>) and the programming language Python (<https://www.python.org/>). They will be introduced during the practical engineering sessions.