

Degree programme	Type	Course
Bachelor in Artificial Intelligence	OB	2

Contact lecturer

Name : Marco Schorlemmer

Email : wernhermarco.schorlemmer@uab.cat

Group languages

You can consult this information at the [end](#) of the document.

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

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.

Learning outcomes

- CM04 (Integrate knowledge representation systems in artificial intelligence applications in different fields, in a way that is appropriate to the requirements of the application.) Integrate knowledge representation systems in artificial intelligence applications in different fields, in a way that is appropriate to the requirements of the application.
- KM13 (Describe the algorithmic, logical, and mathematical underpinnings of the different ways of representing knowledge in artificial intelligence systems.) Describe the algorithmic, logical, and mathematical underpinnings of the different ways of representing knowledge in artificial intelligence systems.
- SM13 (Select the appropriate knowledge representation techniques for a given problem of artificial intelligence application.) Select the appropriate knowledge representation techniques for a given problem of artificial intelligence application.
- SM14 (Apply knowledge representation systems in the resolution of artificial intelligence problems) Apply knowledge representation systems in the resolution of artificial intelligence problems

Contents

1. Introduction: Knowledge, ontology, representation. Knowledge-based systems. 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, default values. Description logics. Knowledge graphs.
4. Knowledge representation with feature spaces: Feature spaces. Knowledge graph embedding and neuro-symbolic systems.
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 situation calculus.

Learning activities and methodology

Title	Hours	ECTS	Learning outcomes
Problem solving seminars	14	0.56	SM13, SM14
Theory lectures	24	0.96	CM04, KM13, SM13
Personal study and readings	36	1.44	KM13, SM13
Development of knowledge engineering project	36	1.44	CM04, SM13, SM14
Problem solving	21	0.84	SM13, SM14
Practical sessions on knowledge engineering	10	0.4	CM04, SM13, SM14

The course follows a tech-free teaching methodology in theory lectures and problem-solving seminars, while encouraging the use of computing technology and generative AI in practical, knowledge engineering sessions.

The content of theory lectures is presented through problem-driven group assignments in class, with occasional use of the blackboard. This is complemented by problem-solving seminars in which students are expected to complete exercises and solve problems in teams under supervision. Problems and exercises are either fully solved during problem-solving seminars or accompanied by hints, allowing students to complete them autonomously. In theory lectures and problem-solving seminars, students are expected to take handwritten course notes because laptops and other computing devices are not allowed during class. Finally, students are to carry out a knowledge engineering project in small teams, during which most of the course content is applied. In supervised practical sessions, the engineering methodology and development steps are introduced, along with the necessary skills for using programming and system development environments. The bulk of the engineering project, however, is carried out autonomously by the student teams. Unlike theory lectures and problem-solving seminars, these sessions require students to use their laptops or other computing devices.

In this course, the use of Artificial Intelligence (AI) technology is allowed as an integral part of the development of the work, provided that the final result reflects the student's significant contribution to the analysis and personal reflection. The student must clearly identify which parts have been generated with this technology, specify the tools used and include a critical reflection on how these have influenced the process and the final result of the activity. The lack of transparency in the use of AI will be considered a lack of academic honesty and may result in a penalty to the activity grade or, in serious cases, greater sanctions.

The Virtual Campus platform (<http://cv.uab.cat/>) will be used to share complementary teaching materials, submit the engineering project, check subject marks, and communicate with the teaching staff, among other uses.

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.

Assessment

Continuous assessment activities

Title	Weight	Hours	ECTS	Learning outcomes
Problem-solving exercises	20%	3	0.12	SM13, SM14
Knowledge engineering project	20%	2	0.08	CM04, SM13, SM14
Partial exam on theory and problems	30%	2	0.08	KM13, SM13, SM14
Final exam on theory and problems	30%	2	0.08	KM13, SM13, SM14

The evaluation of the course's level of achievement takes into account teamwork in problem seminars and practical sessions, as well as the students' scientific and technical knowledge of the subject. 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 defence of the knowledge engineering project to which the student teams will be summoned
- the exercises of student teams submitted according to the assignment given in problem-solving seminars
- the individual, in-person 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:

(a) 20% of the final grade corresponds to the arithmetic mean of the marks for three written exercises completed and submitted in small teams; after each submission, a subset of teams will be randomly selected and required to answer questions about the exercise in person.

(b) 60% of the final grade corresponds to the arithmetic mean of the marks of the written partial and final exams on theory and problems carried out individually on the days of examination.

(c) 20% of the final grade corresponds to the knowledge engineering project completed in small teams, based on a submission and an in-person defence in the lab, for which the attendance of all team members is required.

To pass the course in the first call, it is necessary to obtain at least a mark of 5 in each one of the assessment items (a), (b), and (c). In addition, for item (b), it is necessary to obtain at least a mark of 4 in each exam.

In the second call, it is possible to improve the mark for assessment item (b), i.e., 60% of the course's final grade, by retaking either the partial or the final exam, or both.

To pass the course in the second call, it is necessary to obtain, as for the first call, at least a mark of 5 in each one of the assessment items (a), (b), and (c); and for item (b), it is necessary to obtain at least a mark of 4 in each exam.

This course does not foresee a single evaluation system.

Planning of Assessment Activities

Dates for continuous assessment, such as problem-solving exercises and project presentations, will be announced on the Virtual Campus and in the lecture slides. Dates may change when necessary. Any such modification will always be communicated to students in class and through the Virtual Campus, in accordance with the schedule set by the centre or degree programme.

Non-Assessment

If students have not been assessed for more than 30% of the subject, there will be insufficient evidence of assessment, and the course will be graded as non-assessable.

Honours

Awarding an honours degree is the decision of the teaching staff responsible for the subject. UAB regulations dictate that Honours can be granted only to students who have obtained a final grade of 9 or higher, and that no more than 5% of enrolled students can be awarded an Honours degree.

Plagiarism

Without prejudice to others deemed appropriate and in accordance with current academic legislation, irregularities committed during an assessment activity can result in a mark of 0. Assessment activities marked in this way and by this procedure will not be recoverable. If any of these assessment activities are required 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 teamwork that has not been entirely done by the members of the team;
- present as their 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, individually or in teams.

If a student has committed irregularities in any assessment activity (and therefore will not be able 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, letting 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.

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
- van Harmelen, F. et al. (2008) *Handbook of Knowledge Representation*. 1st edition. Amsterdam: Elsevier. https://bibcercador.uab.cat/permalink/34CSUC_UAB/1eqfv2p/alma991010507167406709

- Baader, F. (2003). *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press.
https://bibcercador.uab.cat/permalink/34CSUC_UAB/rokuu2/cdi_nii_cinii_1971993809743743412
- Hogan, A., Blomqvist, E., Cochez, M., & al. (2021). Knowledge Graphs. *ACM Computing Surveys*, 54 (4), 1-37.
https://bibcercador.uab.cat/permalink/34CSUC_UAB/rokuu2/cdi_proquest_journals_2671351403
- Wang, Q., Mao, Z., Wang, B., & Guo, L. (2017). Knowledge Graph Embedding: A Survey of Approaches and Applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12), 2724-2743.
https://bibcercador.uab.cat/permalink/34CSUC_UAB/rokuu2/cdi_crossref_primary_10_1109_TKDE_2017_
- Badreddine, S., d'Avila Garcez, A., Serafini, L., & Spranger, M. (2022). Logic Tensor Networks. *Artificial Intelligence*, 303, Article 103649.
https://bibcercador.uab.cat/permalink/34CSUC_UAB/rokuu2/cdi_crossref_primary_10_1016_j_artint_2021_
- Noy, N. F., McGuinness, D. (2004). Ontology Development 101: A Guide to Creating Your First Ontology. https://protege.stanford.edu/publications/ontology_development/ontology101.pdf
- Lamy, J.-B. Ontologies with Python (2020): Programming OWL 2.0 Ontologies with Python and Owlready2. Apress. https://bibcercador.uab.cat/permalink/34CSUC_UAB/1c3utr0/cdi_askewsholts_vlebooks_9781484265529

Software

The knowledge engineering project will use freely available programming and development tools 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.

Course groups and languages

The information provided is provisional until November 30. After this date, you will be able to consult the language of each group through this [link](#). To access the information, you will need to enter the course CODE

Type of teaching	Group	Language	Semester	Shift
(TE) Theory	71	English	first semester	afternoon
(PAUL) Classroom practices	711	English	first semester	afternoon
(PLAB) Practical laboratories	711	English	first semester	afternoon