EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

Handbook

Computational Linguistics
Bachelor of Arts
Minor

Valid from summer semester 2023

Updated: May 2023

Adopted: June 6th, 2023

Faculty of Humanities
Institute of Linguistics

Contents

1 Program goals and content

Computational Linguistics is an interdisciplinary combination of the subjects General Linguistics and Computer Science and is concerned with the simulation of human language production in computer-supported models. It includes the implementation of language-processing systems in various application scenarios.

Graduates of our B.A. in Computational Linguistics program obtain a long-term academic qualification aimed at the systematic, critical gaining of knowledge and development of knowledge; this qualification forms a first general academically-based professional qualification in the field of Computational Linguistics. Our students learn to deal with computer linguistic issues in an appropriate, scientific manner.

The professional goals are as follows. Graduates obtain a good overview of the central topics in Computational Linguistics - symbolic/rule-based, statistical methods and corpus linguistics. They can apply mathematical methods from logic and statistics. They are familiar with the central topics in general linguistics: Morphology, syntax and semantics. They can design computer programs and implement general data structures and algorithms from informatics and computational linguistics in an object-oriented language; equally, they have sufficient grasp of theory and the ability to implement parsers. Graduates have specialist knowledge in two areas of Computational Linguistics and are familiar with the current literature, have the ability to compare relevant works and to assess them critically, even to the point where they are able to suggest minor improvements.

Beyond the traditional confines of the discipline, graduates are qualified to program solutions generally, so that they are potentially useful to any kind of IT company They are also capable of collecting, processing, and evaluating general statistical data.

As a formal requirement for enrollment, a knowledge of English at least of level B2 of the Common European Framework of Reference for Languages must be documented. This must be in the form of an internationally-accepted test (TOEFL, IELTS, etc.). Applicants with a German school leaving certificate (Abitur) and applicants whose native tongue is English or who have a leaving certificate from an English-language secondary school do not need additional proof, as the B2 level is thereby shown.

2 General information

2.1. Structure

Studying the B.A. minor in Computational Linguistics requires regular participation in certain courses with a total of 60 credit points. The courses are listed below.

2.2. Credit points

Credit points are intended to quantify and make internationally comparable the average workload a student has to manage to pass a module. One credit point is the equivalent of approximately 30 hours. One credit point is defined as one point under the European Credit Transfer

System (ECTS). Sixty credits per year, i.e. 30 per semester, are required to complete a degree in the minimum prescribed time.

Credit points include both actual teaching time in class - contact hours - as well as private study (usually preparation and reviewing of the material taught, preparation for presentations in class, semester papers, and the Bachelor's thesis).

2.3. Assessed and non-assessed coursework

Various forms of coursework must be completed for credit points to be awarded. This includes assessment and ungraded coursework.

Assessed work is generally given a grade and counts as, or counts towards, the grade for the module. The assessed work required for each module in the Computational Linguistics study program is set out in the module handbook. Assessment may be made in the form of a grade for performance either at the end of the learning process in a module (summatively - e.g. as an exam or assignment = result = the module grade) or it may be in several stages within a module (formatively - e.g. as a programming project, exercise, or analysis task. Assessed work from each phase of assessment is incorporated into the module grade, whereby the phases may be weighted differently). A graded module has been passed if the module is graded "sufficient" (4.00) or better; it is possible to fail. Any repeat exams permitted (when and how often) are regulated in the General Provisions of the exam regulations.

Non-graded coursework is noted as "completed" / "not completed". However, non-graded coursework has only been completed when the responsible teacher attests sufficient quality, i.e. well-founded academic feedback by the teacher is possible. Non-graded coursework has no influence on the module grade. Non-graded coursework may be, e.g.: Presentation, abstract, programming project, analysis, experiment.

In the obligatory courses offered once a year, a retake exam is generally offered in the week before the next semester starts so that students who did not pass an exam have a second opportunity to pass the exam in a timely manner. Where successful, they thus can take courses building on this prerequisite without delay. Alternatively, students can also choose to repeat a course in the following year.

3 Program structure

3.1 Overview of modules

Semester	Module no.	Name of module	Type	ECTS
1	ISCL-BA-01	Introductory module: Computational Linguistics	Compulsory	12
3	ASW-BA-01	Linguistic Fundamentals	Compulsory	6
3	ASW-BA-02	Methods 1	Compulsory	6
2	ISCL-BA-04	Introductory module: Programming	Compulsory	12
4	ISCL-BA-05	Introductory module: Symbolic Computational Linguistics: Text Technology	Compulsory	6
5-6	ISCL-BANF-01	Introductory module: Symbolic Computational Linguistics: Parsing and Grammar formalisms	Compulsory	9
5-6	ISCL-BANF-02	Specialization module	Compulsory	9

3.2 Program structure

Module no.	Name of module	Semester						Σ
		1	2	3	4	5	6	
ISCL-BA-01	**Introductory module: Computational Linguistics**	12						12
ASW-BA-01	**Linguistic Fundamentals**			6				6
ASW-BA-02	**Methods 1**			6				6
ISCL-BA-04	**Introductory module: Programming**		12					12
ISCL-BA-05	**Introductory module: Symbolic Computational Linguistics: Text Technology**				6			6
ISCL-BANF-01	**Introductory module: Symbolic Computational Linguistics: Parsing and Grammar formalisms**					6	3	9
ISCL-BANF-02	**Specialization module**						9	9
Summe		12	12	12	6	6	12	60

Key	
Form of evaluation:	b = graded; ub = not graded (pass/fail); kP = no assessment
Assessment type:	K = exam; MP = oral exam; PA = project; H= assignment; R = presentation If several types of assessment are possible, this is shown by the use of “or” or a slash mark “/”. For instance, K/MP stands for “exam or oral exam”.
Duration:	Duration of module exam in minutes
Weighting	In courses = weighting of the exam grade as part of module grade In modules = weighting of the module grade for the final grade
Credit hours during semster	Credit hours
Status:	o = obligatory; f = facultative (not compulsory)
Class type:	VL = lecture; S = seminar; T = tutorial; Ü = exercise
LP:	Credit points under European credit transfer system (ECTS)

4 Description of modules

4.1 Compulsory modules

Module no.: ISCL-BA-01	**Module name:** Introductory module: Computational Linguistics		**Module type** Compulsory
ECTS credits	12		
Workload **- contact hours** **- private study**	Workload: 360 hours	Contact hours: 120 hours / 8 credit hours per week	Private study: 240 hours
Duration of module	1 semester		
Frequency	each winter semester		
Language of instruc-tion	English		
Module structure	a) Introduction to Computational Linguistics (2 credit hours/ week, 3 ETCS) b) Data structures and algorithms for Computational Linguistics (4 credit hours/ week, 6 ETCS) c) Data structures and algorithms tutorial (2 credit hours/ week, 3 ETCS)		
Teaching formats	a) Lecture: regular and active participation b) Seminar: regular and active participation, exercises c) Tutorial: programming exercises, homework		

■ *B.A. Computational Linguistics*

<table>
<tr><td>Content of module</td><td>This module provides an initial overview of the material, issues, and methods in Computational Linguistics.
The Introduction to Computational Linguistics lecture provides students with the necessary knowledge for advanced use of network-based computer systems. Additionally, it gives an idea of the most important areas of Computational Linguistics.
“Data and Algorithms for Language Processing I” teaches basic knowledge in a higher programming language as well as central concepts and data structures in modern programming and the complexity of algorithms.</td></tr>
<tr><td>Goals</td><td>Students gain an overview of the issues dealt with and methods used in Computational linguistics and learn mathematical concepts needed for successful studies. As this is a module providing an overview, an exam is considered appropriate to ensure the goals have been met.
Students gain basic knowledge in a higher programming language as well as an understanding of central concepts and data structures in modern programming and the complexity of algorithms. They obtain the basic knowledge needed for software development in modern programming languages.</td></tr>
<tr><td>Requirements for the awarding of credit points/ grading (and weighting, where appropriate)</td><td>
<table>
<tr><th>Name of course</th><th>Class type</th><th>Status</th><th>Credit hours in semester</th><th>LP</th><th>Assessment type</th><th>Assessment duration</th><th>Grading system</th><th>Calculation of module grade</th></tr>
<tr><td>Introduction to Computational linguistics</td><td>VL</td><td>0</td><td>2</td><td>3</td><td rowspan="3">K</td><td rowspan="3">210</td><td rowspan="3">b</td><td rowspan="3">100</td></tr>
<tr><td>Data Structures and Algorithms</td><td>S</td><td>0</td><td>4</td><td>6</td></tr>
<tr><td>Data Structures and Algorithms</td><td>T</td><td>0</td><td>2</td><td>3</td></tr>
</table>
</td></tr>
<tr><td></td><td>At the end of the module, there is to be an exam covering all areas dealt with in the module. If a student fails this exam, he/she is strongly advised to take the repeat exam in the same semester. In the tutorial, coursework in the form of programming tasks is required for a pass in the module overall.</td></tr>
<tr><td>Requirements for admission</td><td>none</td></tr>
</table>

Module no.: ASW-BA-01	**Module name:** Linguistic Fundamentals		**Module type** Compulsory
ECTS credits	6		
Workload **- contact hours** **- private study**	Workload: 180 hours	Contact hours: 60 hours / 4 credit hours per week	Private study: 120 hours
Duration of module	1 semester		
Frequency	each winter semester		
Language of instruction	English		

■ *B.A. Computational Linguistics*

Module structure	Introduction to Linguistics (4 credit hours per week, 6 ECTS)								
Teaching formats	Proseminar (lecture-style teaching, exercises, group work)								
Content of module	"Introduction to Linguistics" provides a foundation in linguistics. The goal is to give the student an overview of the research fields within linguistics: their methodology, historical development, and the ways in which they are linked as well as their significance beyond the boundaries of linguistics as a discipline.								
Goals	Students will be able to analyze linguistic data on all levels of description (phonology, syntax, semantics, pragmatics) using the conceptual means of linguistics. They will also gain insight into the specific approaches of different linguistic schools to linguistic phenomena. In addition, they will become aware of how current quantitative methods from machine learning and data science enrich and change linguistics.								
Requirements for the awarding of credit points/ grading (and weighting, where appropriate)	*Name of course*	*Class type*	*Status*	*Credit hours in semester*	*ECTS*	*Assessment type*	*Assessment duration*	*Grading system*	*Calculation of module grade*
	Introduction to Linguistics	S	O	4	6	K	90	b	100
Requirements for admission	none								

Module no.: ASW-BA-02	**Module name:** Methods 1		**Module type** Compulsory
ECTS credits	6		
Workload **- contact hours** **- private study**	Workload: 180 hours	Contact hours: 60 hours / 4 credit hours per week	Private study: 120 hours
Duration of module	1 semester		
Frequency	each winter semester		
Language of instruction	English		
Module structure	a) Mathematical Methods: Logic (2 credit hours per week, 3 ECTS) b) Programming and Data Analysis (2 credit hours per week, 3 ECTS)		
Teaching formats	a) Introductory seminar (lecture-style teaching, exercises, group work) b) Introductory seminar (lecture-style teaching, presentation)		

Content of module	The introductory seminar "Mathematical Methods: Logic" teaches basic methodological knowledge for the formal-structural analysis of expressions, their structure and meaning. "Programming and Data Analysis" takes students into the programming of simple algorithms with the primary goal of familiarizing students with the processing and sifting of empirical data and several of the usual relevant analyses which build upon them and which are relevant from the linguistics point of view.								
Goals	Students learn the terminology basics of mathematical and logical concepts needed in linguistics and are able to follow linguistic argumentation and analysis.								
Requirements for the awarding of credit points/ grading (and weighting, where appropriate)	*Name of course*	*Class type*	*Status*	*Credit hours in semester*	*LP*	*Assessment type*	*Assessment duration*	*Grading system*	*Calculation of module grade*
	Mathematical Methods: Logic	S	O	2	3	K	90	b	100
	Programming and Data Analysis	S	O	2	3	kP	-	ub	-
Requirements for admission	none								

Module no.: ISCL-BA-04	**Module name:** Introductory module: Programming	**Module type** Compulsory
ECTS credits	12	
Workload **- contact hours** **- private study**	Workload: 360 hours Contact hours: 120 hours / 8 credit hours per week Private study: 240 hours	
Duration of module	1 semester	
Frequency	each summer semester	
Language of instruction	English	
Module structure	a) Data Structures and Algorithms for Computational Linguistics II (4 semester credit hours, 6 ECTS) b) Tutorial Data Structures and Algorithms for Computational Linguistics II (4 semester credit hours, 6 ECTS)	
Teaching formats	Programming tasks, programming projects, homework, regular and active participation	

<table>
<tr><td>Content of module</td><td colspan="9">Data structures and algorithms are the core elements of linguistic programming. The data structures are particularly relevant for the storage and selection of data, and algorithms may be seen as recipes for the processing of that data. “Data Structures and Algorithms for Computational Linguistics II” enables students to develop their general understanding of fundamental data structures such as linked lists and trees, and how they are used in Java. The implementation of computer linguistic programs which store and select such data is also a part of this course. Among the data structures and concepts this module also goes in-depth into are fields, lists, stacks and trees, as well as specification and analysis of algorithms.</td></tr>
<tr><td>Goals</td><td colspan="9">Students’ basic knowledge from the introductory course is expanded upon, such that they obtain a broad and solid foundation in the data structures and algorithms required in computational linguistics practice. Note that this module includes work in project groups.</td></tr>
<tr><td rowspan="4">Requirements for the awarding of credit points/ grading (and weighting, where appropriate)</td><td>Name of course</td><td>Class type</td><td>Status</td><td>Credit hours in semester</td><td>LP</td><td>Assessment type</td><td>Assessment duration</td><td>Grading system</td><td>Calculation of module grade</td></tr>
<tr><td>Data Structures and Algorithms for Computational Linguistics II</td><td>S</td><td>0</td><td>4</td><td>6</td><td>K</td><td>120</td><td>b</td><td>100</td></tr>
<tr><td>Data Structures and Algorithms for Computational Linguistics II</td><td>T</td><td>0</td><td>4</td><td>6</td><td>kP</td><td>-</td><td>ub</td><td>-</td></tr>
<tr><td colspan="9">This class concludes with a written examination. In the tutorial, coursework in the form of programming tasks is required for a pass in the module overall.</td></tr>
<tr><td>Requirements for admission</td><td colspan="9">a pass in module ISCL-BA-01</td></tr>
</table>

Module no.: ISCL-BA-05	**Module name:** Introductory module: Symbolic Computational linguistics: Text Technology	**Module type** Compulsory
ECTS credits	6	
Workload **- contact hours** **- private study**	Workload: 180 hours Contact hours: 60 hours / 4 credit hours per week	Private study: 120 hours
Duration of module	1 semester	
Frequency	each summer semester	
Language of instruction	English	
Teaching formats	Regular and active participation	

<table>
<tr><td>Content of module</td><td colspan="9">The use of texts in Computational Linguistics requires them to be first made available in electronic form. Text Technology supplies approaches to solving this problem as well as for the coding and making accessible of textual data for many applications. A number of markup languages play a special role in this. Text Technology includes methods of automatically classifying words and phrases into corpora as well as general methods enabling collections of texts to be annotated linguistically, and general methods enabling collections of texts to be annotated linguistically.</td></tr>
<tr><td>Goals</td><td colspan="9">Students learn the key technical requirements for processing large volumes of text. They become familiar with basic processes in automatic annotation and can demonstrate their use in example applications. They learn about the problems which arise in the electronic processing of texts as well as in further steps such as the correction of typographical errors and linguistic annotation with morphosyntactic markers. The basis of this skill is familiarity with the classifications used in English and German and the relevant, widely used methods for automatically classifying words and phrases into corpora. Students learn to apply such methods within a defined area. Furthermore, they become familiar with the usefulness of processing text corpora in the identification and extraction of information from large collections of texts. At this point in their studies, students are able to understand and apply simple statistical analysis.</td></tr>
<tr><td rowspan="3">Requirements for the awarding of credit points/ grading (and weighting, where appropriate)</td><td>Name of course</td><td>Class type</td><td>Status</td><td>Credit hours in semester</td><td>LP</td><td>Assessment type</td><td>Assessment duration</td><td>Grading system</td><td>Calculation of module grade</td></tr>
<tr><td>Text Technology</td><td>S</td><td>0</td><td>4</td><td>6</td><td>K</td><td>90</td><td>b</td><td>100</td></tr>
<tr><td colspan="9">This class concludes with a written examination. Coursework in the form of programming tasks during semester is required for a pass in the module overall.</td></tr>
<tr><td>Requirements for admission</td><td colspan="9">none</td></tr>
</table>

<table>
<tr><td>Module no.:
ISCL-BANF-01</td><td colspan="2">Module name:
Introductory module: Symbolic Computational Linguistics: Parsing and Grammar formalisms</td><td>Module type
Compulsory</td></tr>
<tr><td>ECTS points</td><td colspan="3">9</td></tr>
<tr><td>Workload
- contact hours
- private study</td><td>Workload:
270 h</td><td>Contact hours:
120 hours / 8 credit hours per week</td><td>Private study:
150 h</td></tr>
<tr><td>Duration of module</td><td colspan="3">2 semesters</td></tr>
<tr><td>Frequency</td><td colspan="3">every winter semester (Parsing), every summer semester (Grammar formalisms)</td></tr>
<tr><td>Language of instruction</td><td colspan="3">English</td></tr>
<tr><td>Module structure</td><td colspan="3">a) Parsing (4 SWS, 6 ECTS)
b) Grammar formalisms (4 SWS, 3 ECTS)</td></tr>
</table>

■ *B.A. Computational Linguistics*

<table>
<tr><td>Teaching formats</td><td colspan="9">a) Proseminar (regular and active participation)
b) Proseminar (regular and active participation)</td></tr>
<tr><td>Content of module</td><td colspan="9">At first glance, a language is a means of communication. In the field of Computational linguistics, a language becomes a formal object which may be described using a (finite) system of rules (formal grammar). Parsing a sentence using grammar means to break it up in such a way that demonstrates how this sentence was created by the rules of grammar. This analysis is an initial step towards enabling a computer to “understand” a language data set.</td></tr>
<tr><td>Goals</td><td colspan="9">Students learn the main classes of formal language as well as formal grammar for the description of linguistic units such as words, phrases, and sentences with phrase-structure rules in the framework of such languages. Students become familiar with the base algorithms for the analysis of formal and natural language data sets, and with the allocation of analysis trees (parsing algorithms), and the necessary transformation processes. In this unit, students gain competence in mastering the advanced symbolic and quantitative approaches used in the following modules in the second and third years.</td></tr>
<tr><td rowspan="3">Requirements for the awarding of credit points/ grading (and weighting, where appropriate)</td><td>Name of course</td><td>Class type</td><td>Status</td><td>Credit hours in se-</td><td>LP</td><td>Assessment type</td><td>Assessment dura-tion</td><td>Grading system</td><td>Calculation of mod-ule grade</td></tr>
<tr><td>Parsing</td><td>S</td><td>0</td><td>4</td><td>6</td><td>K</td><td>120</td><td>b</td><td>100</td></tr>
<tr><td>Grammar formalisms</td><td>T</td><td>0</td><td>4</td><td>3</td><td>-</td><td>-</td><td>ub</td><td>-</td></tr>
<tr><td>Requirements for admission</td><td colspan="9">successful completion of the modules ASW-BA-01, ASW-BA-02, ISCL-BA-01</td></tr>
</table>

Module no.: ISCL-BANF-02	**Module name:** Specialization module		**Module type** Compulsory
ECTS points	9		
Workload **- contact hours** **- private study**	Workload: 270 h	Contact hours: 60 hours / 4 credit hours per week	Private study: 210 h
Duration of module	1 semester		
Frequency	every summer semester		
Language of instruction	English		
Teaching formats	Advanced seminar Computational Linguistics (9 ECTS, 6 SWS, regular and active participation)		

<table>
<tr><td>Content of module</td><td colspan="9">The specialization module pursues topics from the basic modules in greater depth. These topics are linked with the relative current core research in the discipline and are subject to regular review and updating. Relevant areas include Computer Lexicography, Information Retrieval, Logic, Computational Semantics, Machine Translation, Machine Learning, Text Technology, Quantitative Methods, etc. Students are required to obtain more comprehensive mastery using examples from research and development fields of Computational Linguistics in class in the first two years of study, with the aim of now obtaining a qualification for professional work in the field of information technology.</td></tr>
<tr><td>Goals</td><td colspan="9">Students learn to master in-depth an advanced and complex topic and are familiar with the current computational linguistic working methods and results. They are also able to work out and adequately present complex connections in proper academic fashion. Students are able to discuss and present in a proper structure both orally and in written form the knowledge they have gained in a methodologically correct way, and in a way which is appropriate to the material.</td></tr>
<tr><td rowspan="3">Requirements for the awarding of credit points/ grading (and weighting, where appropriate)</td><td>Name of course</td><td>Class type</td><td>Status</td><td>Credit hours in semester</td><td>LP</td><td>Assessment type</td><td>Assessment duration</td><td>Grading system</td><td>Calculation of module grade</td></tr>
<tr><td>Hauptseminar</td><td>S</td><td>0</td><td>4</td><td>9</td><td>R (or A / K / mP)

H</td><td>15

-</td><td>b

b</td><td>20%

80%</td></tr>
<tr><td colspan="9">In advanced seminars, grading is usually carried out on the basis of a presentation and a written assignment, whereby in this advanced module an independent project or program development often provides the basis of the written assignment. The precise form of examination is decided by the relevant member of academic staff.</td></tr>
<tr><td>Participation requirements</td><td colspan="9">successful completion of the modules ISCL-BA-01, ASW-BA-01</td></tr>
</table>