Statistical Natural Language Processing

Course syllabus SS 2019

Course Description

This course is an undergraduate introduction to (statistical) natural language processing (NLP), aiming to expose students to a large variety of topics in NLP. In the first part of the course, we will go through a number of established and 'traditional' machine learning methods, as well as some popular and 'new' ones. The second part of the course introduces common tasks, methods and applications of NLP.

This is a practical, fast-paced, broad introduction to the field. Fluency in programming and ability to learn new programming languages and/or environments will be assumed.

The course language is English.

Tentative schedule

W1	Introduction
W2–3	Preliminaries: linear algebra, probability theory
W3	ML: intro, regression
W4–5	ML: classification, evaluation
W6	ML: sequence learning
W7–8	ML: Unsupervised learning
W9	Language models
W10	Tokenization, POS tagging, morphology
W11	Dense vector representations
W12	Text classification
W12–13	Parsing
W14	Wrap up & exam

Visit the course web page for a more detailed and up-to-date version of the schedule.

Prerequisites

The students should be fluent in programming, either able to program in Python, or capable of learning by themselves in a short time. Some familiarity with (computational) linguistics is also assumed.

For the ISCL students, the above requirements are covered in courses ISCL-BA-06 and ISCL-BA-07.

Recommended literature

Daniel Jurafsky and James H. Martin (2009) *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Pearson Prentice Hall, second edition.[1]

Trevor Hastie, Robert Tibshirani, and Jerome Friedman (2009), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer-Verlag, second edition.[2]

[1] Chapters from 3rd edition draft are available at `http://web.stanford.edu/~jurafsky/slp3/`.

[2] An updated version of the complete book is available at `http://web.stanford.edu/~hastie/ElemStatLearn/`.

Course work and evaluation

In total, the coursework is worth 9 ECTS. Your grade will be determined based on 7 graded programming assignments (60 %) and a written final exam (40 %) at the end of the course. Full attendance to the course will also be rewarded with a 5 % bonus.[3]

If you are a master's student, you can take the course as a 6ECTS 'Haubptseminar' for the regular coursework, or for 9ECTS with an additional project and associated term paper.

Each assignment constitute the 10 % of the overall course grade. Only 6 best assignment scores will contribute to your final score. Late assignments up to one week are graded with a maximum of 5 % (of the total grade). Assignments later than one week will not be accepted.

[3] The following grade scale will be used to determine your final grade.

Percent	Local	ECTS
> 96	1.0	A
93–96	1.3	A
89–92	1.7	B
85–88	2.0	B
81–84	2.3	C
77–80	2.7	C
73–76	3.0	C
69–72	3.3	D
65–68	3.7	E
60–64	4.0	E
< 60	5.0	F

You are encouraged work on the assignments in pairs, but you are *not allowed* to pair with the same participant twice.

A retake of the final exam is possible, only if you failed the course, but it is still possible to get a passing overall grade by obtaining a higher grade on the exam.

If you are not able to attend the exam due to sickness, you must inform instructors by e-mail, at least 60 minutes before the beginning of the exam. A doctor's note should be presented before enrolling to the retake.

Assignment 0

We will use git version management system through GitHub classroom for distribution and submission of the assignments. Please make sure to obtain a GitHub account, and the complete a warm-up assignment which will introduce you to the environment we will be using for the assignments.

The warm-up assignment, is not graded. However, you will be able receive and work on the other assignments only after completing this assignment. Please follow the steps described in the Assignment sheet carefully.

Academic conduct

You are encouraged to discuss your assignments and other class work with others, do research on the Internet and use other sources for knowledge and inspiration. However, unless stated/cited explicitly, all the coursework you submit should be your own work. You are required to cite any source you have used. If you 'borrowed' code that is crucial for the solution of an assignment, you will lose points. Not indicating the source of external code is plagiarism.

Plagiarism or any other form of academic misconduct will not be treated lightly.

Practical information

Instructor	Çağrı Çöltekin ⟨`ccoltekin@sfs.uni-tuebingen.de`⟩
Office hours	Mon 14:00–15:00 (room 1.09)
Tutors	Maxim Korniyenko ⟨`maxim.korniyenko@student.uni-tuebingen.de`⟩
	Marko Lozajic ⟨`marko.lozajic@student.uni-tuebingen.de`⟩
Time	Mon 12:00–14:00 & Wed 10:00–12:00 & Fri 12:00–14:00
Location	Wilhelmstr. 19, room 0.02
Course web page	`http://sfs.uni-tuebingen.de/~ccoltekin/courses/snlp/`