СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ
SIBERIAN FEDERAL UNIVERSITY

Theoretical foundations of computer science

Course Guide

Siberian Federal University

Theoretical foundations of computer science

Course Guide

This course contributes to the requirements for the Degree of Candidate of Science in Computer Science.

Published by Siberian Federal University, Krasnoyarsk,2020

Contents

1. Course Description

This course contributes to the requirements for the Degree of Candidate of Science in Computer Science.

Course period	2 semesters Third semester: from October, the 1st to February, the 1st (18 weeks) Fourth semester: from February, the 1st to June, the 1st (18 weeks)
Study credits	6 ECTS credits
Duration	216 hours
Language of instruction	English
Academic requirements	– MSc degree in Computer Science or equivalent (transcript of records), – good command of English (certificate or other official document) **Prerequisites:** – base knowledge of computer science, programming skills.

1.1 Course overview

The course introduces the principles and the critical analysis of the main paradigms for learning from data and their applications. The course provides the Machine Learning basis for both the aims of building new adaptive Intelligent Systems and powerful predictive models for intelligent data analysis. The course introduces students to the design of A.I. based solutions to complex pattern recognition problems and discusses how to realize applications exploiting computational intelligence techniques. The course also presents fundamentals of signal and image processing. Particular focus will be given to pattern recognition problems and models dealing with sequential and time-series data.

1.2 Special features

The course "Theoretical Foundations of Informatics" is aimed at developing students' holistic understanding of the current state of the theory and practice of building intelligent systems for various purposes.

Students will learn how to apply these techniques in a wide range of applications using modern programming libraries.

1.3 Course aims and objectives

Course Aims

The course aim is to study the methods and technologies for the development of smart connected applications, i.e. applications which exhibit intelligent behavior - through the use of artificial intelligence techniques.

Course Objectives

- develop skills in representing tasks in the state space and optimizing the search for solutions;

- Study models of knowledge representation in intelligent systems;
- familiarize students with machine learning algorithms and pattern recognition;
- teach students to write efficient code in Python and R;
- to teach students to use computer tools and methods to develop artificial intelligence systems.

1.4 Learning outcomes

By the end of the course, students will know:

  - algorithms and data structures;
  - machine learning methods;
  - fundamentals of mathematical analysis and optimization methods;
  - general principles and approaches to solving problems of pattern recognition;
  - methods of computational intelligence.

By the end of the course, students will be able to:

  - formulate a task for a data science project. Put forward ideas and hypotheses and draw up a plan for solving the problem;
  - select algorithms and metrics for the task for different models;
  - build machine learning models using specialized libraries;
  - evaluate the quality of models;
  - interpret the results and draw up a research report, compare algorithms on ready-made datasets, determine quality improvement methods.

By the end of the course, students will possess:

- the necessary skills to develop applications using artificial intelligence methods;
- the necessary skills for solving machine learning and pattern recognition problems.

2. Course Lecturer, Contact Information

Sopov Evgenii,
Ph.D. in Engineering, Docent, Associate Professor of the Department of Information Systems,
School of Space and Information Technologies
Siberian Federal University
Siberian Federal University
e-mail: ESopov@sfu-kras.ru
Google Scholar page:
https://scholar.google.com/citations?user=JW4K_eQAAAAJ&hl=ru
Additional information is available at:
http://ikit.sfu-kras.ru/e/128

3. Prerequisites

A background in basic of information, data science and programming will help in faster and better understanding of every topic. Nevertheless, each part of the course includes a short introduction of methods that are required for its study. Therefore, a student without the denoted experience must be encouraged to make some additional efforts in education.

4. Course Outline

<table>
<tr><th>Week</th><th>Lectures</th><th>Seminars/ Assignments</th><th>Hours
Lec/Lab/HA</th></tr>
<tr><td colspan="4">Semester 1</td></tr>
<tr><td>1-8</td><td>Module 1. Artificial intelligence fundamentals
Lecture 1 “Advanced search”
Lecture 2 “Constraint satisfaction problems”
Lecture 3 “Knowledge representation and reasoning”
Lecture 4 “Rules systems: use and efficient implementation.”
Lecture 5 “Planning systems”</td><td>Lab 1 “Python Basic”</td><td>8/10/28</td></tr>
<tr><td>9-18</td><td>Module 2. Machine learning
Lecture 1 “Computational learning tasks for predictions, learning as function approximation, generalization concept”
Lecture 2 Linear models and Nearest-Neighbors
Lecture 3 Principles of learning processes: elements of statistical learning theory, model validation.
Lecture 4 “Neural Network”</td><td>Lab 1 “R programming”
Lab 2 “Learning algorithms and properties, regularization”
Lab 3 “Support Vector Machines and kernel-based models”</td><td>10/8/80</td></tr>
</table>

Semester 2			
1-12	**Module 3. Formal and statistical approaches to NLP** **Lecture 1** "The basic methods Language technology" **Lecture 2** "Statistical methods" **Lecture 3** "Applications"	**Lab 1** "NLP libraries: NLTK, Theano, Tensorflow."	12/5/5
13-18	**Module 4. Computational mathematics for learning and data analysis** **Lecture 1** "Introduction of Computational mathematics for learning and data analysis" **Lecture 2** "Matrix factorization, decomposition and approximation - Eigenvalue computation" **Lecture 3** "Nonlinear optimization: theory and algorithms" **Lecture 4** "Least-squares problems and data fitting "Lab 1 "MATLAB for data analysis"	**Lab 1** "MATLAB for data analysis"	6/4/4

	36	27	180
36	Final Exam		36

4.1 Course requirements

4.1.1 Web-page of the course

Course materials and required reading materials are available on the course webpage “Theoretical foundations of computer science”. The webpage is available through the SibFU E-learning portal www.e.sfu-kras.ru. You must be logged in to access this course. https://e.sfu-kras.ru/enrol/index.php?id=9071.

4.1.2 Required reading

The book “Artificial Intelligence: A Modern Approach 3rd Edition» by Stuart Russe discusses issues of classification and regression using supervised learning, predictive analytics with ensemble learning, detecting patterns with unsupervised learning and also binding recommender systems.

The book «Computer Age Statistical Inference: Algorithms, Evidence, and Data Science» contains all information that is required for study beginning with classical inferential theories – Bayesian, frequentist, Fisherian – individual chapters take up a series of influential topics: survival analysis, logistic regression, empirical Bayes, the jackknife and bootstrap, random forests, neural networks, Markov chain Monte Carlo, inference after model selection. It will help students to reach a deeper understanding of methods and applications of data analysis together with optimization.

The book “Deep learning for NLP and speech recognition” explains recent deep learning methods applicable to NLP and speech, provides state-of-the-art approaches, and offers real-world case studies with code to provide hands-on experience.

Book “Functional Data Analysis with R and MATLAB” is designed to show readers how to perform functional data analysis using MATLAB. The text provides MATLAB code for a set of data analyses that showcase functional data analysis techniques. Topics include an introduction to functional data analysis and how to specify basis systems for building functions.

The book “Measurement and Data Analysis for Engineering and Science” is recommended for studying up-to-date coverage of experimentation methods in science and engineering.

The book “Artificial Intelligence with Python” by Joshi Prateek is a comprehensive Guide to Building Intelligent Apps for Python Beginners and Developers.

Some of the course topics include practical tasks in R, a book «R-Software for Data Analysis» by can provide students with additional information.

4.1.3 Course materials

Student’s Home Assignment reports must be attached as a separate pdf file. Student’s name and group number should be written on the first page of the file. It is recommended to insert code with short comments for key elements of the code. Students send this report in electronic form only before the deadline.

4.1.4 Required feedbacks

Students are free to contact the lecturer by email. The name of department and a number of a group should be written in the subject or in the beginning of the letter for convenience. More information on how to contact the lecturer can be found in «Lecturer information» section of this Guide.

Student's Home or Lab Assignment reports must be attached as a separate pdf file. Student's name and group number should be written on the first page of the file. Students send this report in electronic form only before the deadline.

4.2 Course Structure

Learning Activities	**Hours**
Lectures	36
Practice sessions / Seminars,	27
Self-study Assignments	117
Final Exam (including preparation)	36
Total study hours	216

4.3 Time schedule of the course and course outline

№	*Theme*	*Week*	*Learning Activities*	*Hours*	*Home Assignment and Reading*
	Theme 1. Artificial intelligence fundamentals		**Lecture 1** “Advanced search”	**2**	Artificial Intelligence: A Modern Approach 3rd Edition pp. 64-108
			Lecture 2 “Constraint satisfaction problems”	**2**	Artificial Intelligence: A Modern Approach 3rd Edition pp. 202-227
			Lecture 3 “Knowledge representation and reasoning”	**2**	Artificial Intelligence: A Modern Approach 3rd Edition pp. 234-244
			Lecture 4 “Rules systems: use and efficient implementation.”	**2**	Artificial Intelligence: A Modern Approach 3rd Edition pp. 149-178
			Lecture 5 **“**Planning systems”	**2**	Artificial Intelligence: A Modern Approach 3rd Edition pp. 401-430
			Lab 1 “Python Basic”	**10**	Artificial intelligence with python
			Home Assignment №1		
2	**Theme 2. Machine**	**10-18**	**Lecture 1** “Computational learning tasks for	**2**	Artificial Intelligence: A Modern Approach 3rd

	learning		predictions, learning as function approximation, generalization concept”		Edition pp. 523-649
			Lecture 2 Linear models and Nearest-Neighbors		Artificial Intelligence: A Modern Approach 3rd Edition p. 719
			Lecture 3 Principles of learning processes: elements of statistical learning theory, model validation.	**2**	Artificial Intelligence: A Modern Approach 3rd Edition pp. 709-711
			Lecture 4 “Neural Network”	**2**	Artificial Intelligence: A Modern Approach 3rd Edition pp. 563-598
			Lab 1 “R programming”	**4**	R-Software for Data Analysis
			Lab 2 “Learning algorithms and properties, regularization**”**	**2**	R-Software for Data Analysis
			Lab 3 “Support Vector Machines and kernel-based models**”**	**2**	Artificial Intelligence: A Modern Approach 3rd Edition pp. 744-748 Artificial intelligence with

					python
			Home Assignment №2		
3	**Theme 3. Human language technologies**	**19-28**	**Lecture 1** “The basic methods Language technology”	**4**	Artificial Intelligence: A Modern Approach 3rd Edition pp. 691-724
			Lecture 2 “Statistical methods”	**4**	Artificial Intelligence: A Modern Approach 3rd Edition pp. 651-690
			Lecture 3 “Applications”	**2**	Measurement and Data Analysis for Engineering and Science pp 34-56
			Lab 1 “NLP libraries: NLTK, Theano, Tensorflow.”	**4**	Deep learning for NLP and speech recognition
			Home Assignment №3		
	Theme 4. Computational mathematics for learning and data analysis	**29-36**	**Lecture 1** “Introduction of Computational mathematics for learning and data analysis”	**2**	Computer Age Statistical Inference: Algorithms, Evidence, and Data Science pp 12-69

			Lecture 2 "Matrix factorization, decomposition and approximation - Eigenvalue computation"	**2**	Computer Age Statistical Inference: Algorithms, Evidence, and Data Science pp 112-145
			Lecture 3 "Nonlinear optimization: theory and algorithms**"**	**2**	Computer Age Statistical Inference: Algorithms, Evidence, and Data Science pp 120-129
			Lecture 4 "Least-squares problems and data fitting **"**	**2**	Computer Age Statistical Inference: Algorithms, Evidence, and Data Science pp 156-164
			Lab 1 "MATLAB for data analysis"	**5**	Functional Data Analysis with R and MATLAB
			Home Assignment №4		
5	**Final exam**			36	Prepare to final exam. Preparation for answering exam questions (available at e-courses and course book).

4.4. Theme 1: Artificial intelligence fundamentals

The theme consists of:

- Lecture 1 “Advanced search”
- Lecture 2 “Constraint satisfaction problems”
- Lecture 3 “Knowledge representation and reasoning”
- Lecture 4 “Rules systems: use and efficient implementation.”
- Lecture 5 “Planning systems”
- In-lab activity «Python Basic»
- Home assessment work 1

The Theme 1: Artificial intelligence fundamentals aims to offer a view of the classical/symbolic approach to Artificial Intelligence and serves as a basis for more in depth treatment of specific theories and technologies for building complete A.I. systems integrating different approaches and methods.

In-lab activity «Python Basic». This is a seminar about Python programming. These skills will be required for the rest of the course. Students will learn Python types and statements, names of a special method, inbuilt functions, exceptions, and other frequently used standard library modules. The In-lab activity « Python Basic» also covers Built-in object types, syntax, statements for creating as well as processing objects, functions, modules for structuring and reusing code. It also includes special operator overloading methods, standard library modules, and extensions important Python idioms and hints.

The sample of the exercises for Theme 1

1. Explain why problem formulation must follow goal formulation.

2. Your goal is to navigate a robot out of a maze. The robot starts in the center of the maze Exercises 113 facing north. You can turn the robot to face north, east, south, or west. You can direct the robot to move forward a certain distance, although it will stop before hitting a wall.

a. Formulate this problem. How large is the state space?

b. In navigating a maze, the only place we need to turn is at the intersection of two or more corridors. Reformulate this problem using this observation. How large is the state space now?

c. From each point in the maze, we can move in any of the four directions until we reach a turning point, and this is the only action we need to do. Reformulate the problem using these actions. Do we need to keep track of the robot's orientation now?

d. In our initial description of the problem we already abstracted from the real world, restricting actions and removing details. List three such simplifications we made.

3. Give a complete problem formulation for each of the following. Choose a formulation that is precise enough to be implemented.

a. Using only four colors, you have to color a planar map in such a way that no two adjacent regions have the same color.

b. A 3-foot-tall monkey is in a room where some bananas are suspended from the 8-foot ceiling. He would like to get the bananas. The room contains two stackable, movable, climbable 3-foot-high crates.

c. You have a program that outputs the message "illegal input record" when fed a certain file of input records. You know that processing of each record is independent of the other records. You want to discover what record is illegal.

d. You have three jugs, measuring 12 gallons, 8 gallons, and 3 gallons, and a water faucet. You can fill the jugs up or empty them out from one to another or onto the ground. You need to measure out exactly one gallon.

4. Consider the problem of finding the shortest path between two points on a plane that has convex polygonal obstacles as shown in Figure 1. This is an idealization of the problem that a robot has to solve to navigate in a crowded environment.

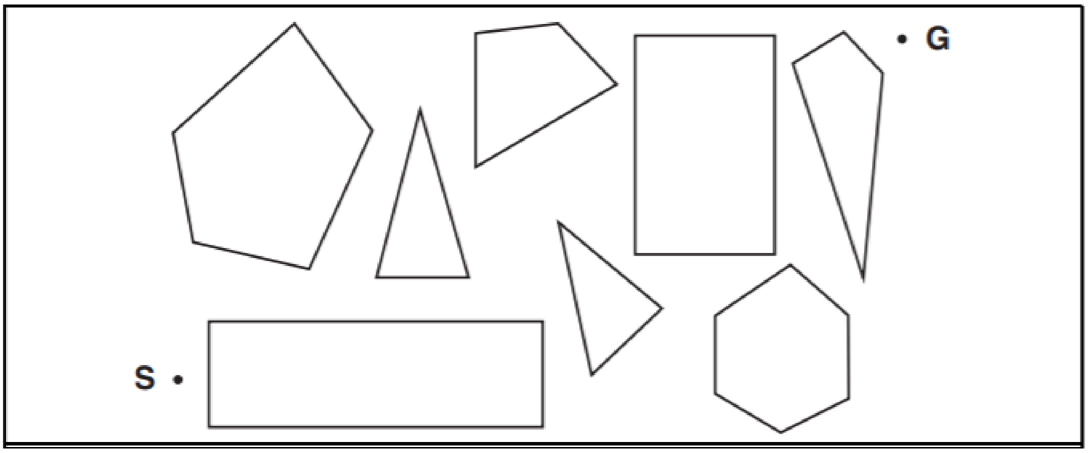

Figure 1 A scene with polygonal obstacles. S and G are the start and goal states.

a. Suppose the state space consists of all positions (x, y) in the plane. How many states are there? How many paths are there to the goal?

b. Explain briefly why the shortest path from one polygon vertex to any other in the scene must consist of straight-line segments joining some of the vertices of the polygons. Define a good state space now. How large is this state space?

c. Define the necessary functions to implement the search problem, including an ACTIONS function that takes a vertex as input and returns a set of vectors, each of which maps the current vertex to one of the vertices that can be reached in a straight line. (Do not forget the neighbors on the same polygon.) Use the straight-line distance for the heuristic function.

d. Apply one or more of the algorithms in this chapter to solve a range of problems in the domain, and comment on their performance.

4.5. Theme 2: Machine learning

The theme consists of:

- Lecture 1 “Computational learning tasks for predictions, learning as function approximation, generalization concept”
- Lecture 2 “Linear models and Nearest-Neighbors”

- Lecture 3 “Principles of learning processes: elements of statistical learning theory, model validation”
- Lecture 4 “Neural Network”
- Lab 1 “R programming”
- Lab 2 “Learning algorithms and properties, regularization”
- Lab 3 “Support Vector Machines and kernel-based models”
- Home assignment work 2

The Theme 2: Machine learning introduce the principles and the critical analysis of the main paradigms for learning from data and their applications. The lectures provide the Machine Learning basis for both the aims of building new adaptive Intelligent Systems and powerful predictive models for intelligent data analysis.

The goal of this theme is to give learners basic understanding of modern neural networks and their applications in computer vision and natural language understanding. The lecture starts with a recap of linear models and discussion of stochastic optimization methods that are crucial for training deep neural networks. Learners will study all popular building blocks of neural networks including fully connected layers, convolutional and recurrent layers. Learners will use these building blocks to define complex modern architectures in TensorFlow and Keras frameworks.

In-lab activity «R programming». This is a seminar about basics of R programming including basic structures in R, functional programming, R objects, functions, and basics of R packages from applied point of view.

In-lab activity “Learning algorithms and properties, regularization” The seminar is devoted to the study of methods for adding some additional restrictions to a condition in order to solve an incorrectly set task or prevent overfitting. Most often, this information takes the form of a penalty for the complexity of the model.

In-lab activity "Support Vector Machines and kernel-based models". The seminar is devoted to solving the classification problem using Support Vector Machines and kernel-based models on the example of a certain data set.

The sample of the exercises for Theme 2

Suppose we generate a training set from a decision tree and then apply decision-tree learning to that training set. Is it the case that the learning algorithm will eventually return the correct tree as the training-set size goes to infinity? Why or why not?

In the recursive construction of decision trees, it sometimes happens that a mixed set of positive and negative examples remains at a leaf node, even after all the attributes have been used. Suppose that we have p positive examples and n negative examples.

Consider the following data set comprised of three binary input attributes (A1, A2, and A3) and one binary output:

Example	A1	A2	A3	Output y
X1	1	0	0	0
X2	1	0	1	0
X3	0	1	0	0
X4	1	1	1	1
X5	1	1	0	1

Use the algorithm to learn a decision tree for these data. Show the computations made to determine the attribute to split at each node.

4.6. Theme 3: Human language technologies

The theme consists of:

- Lecture 1 “The basic methods Language technology”
- Lecture 2 “Statistical methods”
- Lecture 2 “Applications”
- Lab 1 “NLP libraries: NLTK, Theano, Tensorflow.”
- Home Assignment №2

The Theme 3: Human language technologies presents principles, models and the state of the art techniques for the analysis of natural language, focusing mainly on statistical machine learning approaches and Deep Learning in particular. Students will learn how to apply these techniques in a wide range of applications using modern programming libraries. The lecture discusses topics such as formal and statistical approaches to NLP; generative vs Discriminative Models; linguistic essentials (tokenization, morphology, PoS, collocations, etc.). parsing (constituency and dependency parsing); processing Pipelines; Lexical semantics: corpora, thesauri, gazetteers; Distributional Semantics: Word embeddings, Character embeddings; Deep Learning for natural language; Opinion mining, Sentiment Analysis; Question answering, Language inference, Dialogic interfaces; Statistical Machine Translation. The theme deals with Statistical methods such as Language Model, Hidden Markov Model, Viterbi Algorithm, Generative vs Discriminative Models. Entity recognition, Entity linking, classification, summarization will be presented in the lecture 3.

The seminar “NLP libraries:” is devoted to learning NLP libraries such as NLTK, Theano, Tensorflow.” This libraries are an open source machine learning software libraries for solving problems of building and training neural networks in order to automatically find and classify patterns, achieving the quality of human perception.

The sample of the exercises for Theme 3

1. This exercise explores the quality of the n-gram model of language. Find or create a monolingual corpus of 100,000 words or more. Segment it into words, and compute the frequency of each word. How many distinct words are there? Also count frequencies of bigrams (two consecutive words) and trigrams (three consecutive words). Now use those frequencies to generate language: from the unigram, bigram, and trigram models, in turn, generate a 100- word text by making random choices according to the frequency counts. Compare the three generated texts with actual language. Finally, calculate the perplexity of each model.

2. Write a program to do segmentation of words without spaces. Given a string, such as the URL "thelongestlistofthelongeststuffatthelongestdomainnameatlonglast.com," return a list of component words: ["the," "longest," "list," ...]. This task is useful for parsing URLs, for spelling correction when words runtogether, and for languages such as Chinese that do not have spaces between words. It can be solved with a unigram or bigram word model and a dynamic programming algorithm similar to the Viterbi algorithm.

3. Consider the problem of trying to evaluate the quality of an IR system that returns a ranked list of answers (like most Web search engines). The appropriate measure of quality depends on the presumed model of what the searcher is trying to achieve, and what strategy she employs. For each of the following models, propose a corresponding numeric measure.

a. The searcher will look at the first twenty answers returned, with the objective of getting as much relevant information as possible.

b. The searcher needs only one relevant document, and will go down the list until she finds the first one.

c. The searcher has a fairly narrow query and is able to examine all the answers retrieved. She wants to be sure that she has seen everything in the document collection that is Exercises 887 relevant to her query. (E.g., a lawyer wants to be sure that she has found all relevant precedents, and is willing to spend considerable resources on that.)

d. The searcher needs just one document relevant to the query, and can afford to pay a research assistant for an hour's work looking through the results. The assistant can look through 100 retrieved documents in an hour. The assistant will charge the searcher for the full hour regardless of whether he finds it immediately or at the end of the hour.

e. The searcher will look through all the answers. Examining a document has cost \$A; finding a relevant document has value \$B; failing to find a relevant document has cost \$C for each relevant document not found.

f. The searcher wants to collect as many relevant documents as possible, but needs steady encouragement. She looks through the documents in order. If the documents she has looked at so far are mostly good, she will continue; otherwise, she will stop.

4.7 Theme 4. Computational mathematics for learning and data analysis

The theme consists of:

- Lecture 1 “Introduction of Computational mathematics for learning and data analysis”
- Lecture 2 “Matrix factorization, decomposition and approximation - Eigenvalue computation”
- Lecture 3 “Nonlinear optimization: theory and algorithms”
- Lecture 4 “Least-squares problems and data fitting ”Lab 1 “MATLAB for data analysis”
- Home Assignment №3

The Theme 4. Computational mathematics for learning and data analysis introduces some of the main techniques for the solution of numerical problems that find widespread use in fields like data analysis, machine learning, and artificial intelligence. These techniques often combine concepts typical of numerical analysis with those proper of numerical optimization, since numerical analysis tools are essential to solve optimization problems, and, vice-versa, problems of numerical analysis can be solved by optimization algorithms. The course has a significant hands-on part whereby students learn how to use some of the most common tools for computational mathematics; during these sessions, specific applications will be briefly illustrated in fields like regression and parameter estimation in statistics, approximation and data fitting, machine learning, artificial intelligence, data mining, information retrieval, and others.

The lab "MATLAB for data analysis". Students will learn how to make automatic machine learning (AutoML) including feature selection, model selection and hyperparameter tuning using MATLAB.

The sample of the exercises for Theme 4

1. Use the distillation column data set (https://openmv.net/info/distillation-tower) and choose any two variables, one for *x* and one as *y*. Then fit the following models by least squares in any software package you prefer:

$$y_i = b_0 + b_1 x_i$$
$$y_i = b_0 + b_1 (x_i - \overline{x})$$
$$(y_i - \overline{y}) = b_0 + b_1 (x_i - \overline{x})$$

rove to yourself that centering the *x* and *y* variables gives the same model for the 3 cases in terms of the b1b1 slope coefficient, standard errors and other model outputs.

5. Assessment

Assessment strategy	Points, max	Evaluation criteria
Lab works	50	Lab report
Final exam	50	2 questions and a practical task in the form of a simulated experimental dataset

Grade policy for final assessment is:

- A (excellent work) 91–100 points
- B (above average work) 81–90 points
- C (average work) 71–80 points
- D (below average work) 50–70 points
- F (failed work) < 50 points

6. Attendance Policy

Students are expected to attend classes regularly. In case of missing an in-lab activity a student should perform additional work submitted to the instructor within a week after a class was missed.

Every topic involves an assignment. A written report on the assignment should be submitted within two weeks from the moment students received a list of problems. The final mark will rely on the same grading policy as for the final exam.

7. Required Course Participation

There are no special requirements for the course participation. The preferred type of report submission is the electronic one. Students can use the web-version of the course (link) for a better progress. All problems for solution could be found there together with text from the course book.

8. Facilities, Equipment and Software

Software:

Eclipse Platform 3.5

R-Studio

Microsoft Office®.

Annex 1 Final Oral Exam Questions

1. Multivariate and matrix calculus
2. Matrix factorization, decomposition and approximation
3. Nonlinear optimization: theory and algorithms
4. Least-squares problems and data fitting
5. Formal and statistical approaches to NLP.
6. Statistical methods: Language Model, Hidden Markov Model, Viterbi Algorithm, Generative vs Discriminative Models
7. Linguistic essentials (tokenization, morphology, PoS, collocations, etc.).
8. Parsing (constituency and dependency parsing).
9. Processing Pipelines.
10. Lexical semantics: corpora, thesauri, gazetteers.
11. Distributional Semantics: Word embeddings, Character embeddings.
12. Deep Learning for natural language.
13. Applications: Entity recognition, Entity linking, classification, summarization.
14. Opinion mining, Sentiment Analysis.
15. Question answering, Language inference, Dialogic interfaces.
16. Statistical Machine Translation.
17. NLP libraries: NLTK, Theano, Tensorflow.
18. Computational learning tasks for predictions, learning as function approximation, generalization concept.
19. Linear models and Nearest-Neighbors (learning algorithms and properties, regularization).
20. Neural Networks (MLP and deep models, SOM).
21. Probabilistic graphical models.

22. Principles of learning processes: elements of statistical learning theory, model validation.
23. Support Vector Machines and kernel-based models.
24. Introduction to applications and advanced models.
25. Advanced search
26. Constraint satisfaction problems
27. Knowledge representation and reasoning
28. Non-standard logics
29. Uncertain and probabilistic reasoning (Bayesian networks, fuzzy sets).
30. Foundations of semantic web: semantic networks and description logics.
31. Rules systems: use and efficient implementation.
32. Planning systems.