Schedule

1.	20.02	**Introduction**
2.	27.02	Project I plan
3.	05.03	Tutorial hours
4.	12.03	Tutorial hours
5.	19.03	Tutorial hours
6.	26.03	**Project I deadline**
7.	09.04	Project II plan
8.	16.04	Tutorial hours
9.	23.04	Tutorial hours
10.	07.05	**Project II deadline**
11.	14.05	Project III plan
12.	21.05	Tutorial hours
13.	28.05	Tutorial hours
14.	04.06	Tutorial hours
15.	11.06	**Project III deadline**

General comments for each project

- any programming language is allowed
- you can utilize code from external sources (books, articles, blogs) provided that: reference is cited in the report and some modifications to the original solution are applied (not just copy-paste)
- plan: short (1-2 pages) document which briefly describes planned architectures, experiments etc. (can be sent by e-mail)
- source code: with short readme how to run the model, set parameters etc.
- report: the most important part; all results (also failures) should be described in details; run experiments multiple times (at least 5) and present min, max, avg, std dev; justify used parameters (present tuning process); provide conclusions; high accuracy is only small part of final grade - more important is well presented experimental process

Reports

The report should include:

- description of the research problem, understandable to the person who did not see the content of the task
- instruction of the application (containing information on how to reproduce results)
- theoretical introduction
- description of the conducted experiments
- statistically processed results (presented clearly)
- conclusions, presumed reasons for successes/failures and further research proposals

Reports

Some additional remarks:

- if the experiment is not described in the report it is regarded as not conducted
- take care of reproducibility by initializing a random number generator with a constant seed
- the report is an official document, so please keep it formal (table of contents, bibliography, captions under figures, tables, etc.)
- results should be commented
- to obtain statistically significant results, each experiment ought to be repeated multiple times (when possible)
- in addition to the mean, standard deviation should also be calculated (in some scenarios worse mean with low variance may be a more desirable result than a better mean with high variance)

Project II - recurrent neural networks

Topic: **Speech commands classification with recurrent neural networks**

Dataset: Speech Commands Dataset
`https://www.kaggle.com/c/tensorflow-speech-recognition-challenge/data`

Project II - recurrent neural networks

- test and compare different network architectures (one of them should be Transformer, another one Long short-term memory (LSTM))
- investigate influence of parameters change on the obtained results
- present confusion matrix (with appropriate discussion)
- in case of accuracy or efficiency problem a subset of classes can be selected and tested (e.g. only “yes” and “no” commands)
- please pay special attention on “silence” and “unknown” classes - test different approaches (e.g. separate network for their recognition)

Project II - recurrent neural networks

Useful resources:

- https://www.kaggle.com/davids1992/speech-representation-and-data-exploration
- https://towardsdatascience.com/recognizing-speech-commands-using-recurrent-neural-networks-with-attention-c2b2ba17c837
- https://www.coursera.org/lecture/nlp-sequence-models/recurrent-neural-network-model- ftkzt
- https://pathmind.com/wiki/lstm
- https://towardsdatascience.com/transformers-141e32e69591