**Project Description**

1. The project will be about a small-scale deep learning application.
2. Tentative project titles:
   1. Optical character recognition
   2. Handwriting recognition
   3. Speech to text generation
   4. Text to speech generation
   5. Machine translation
   6. Sentiment analysis
   7. Image captioning
   8. Multivariate stock price forecasting
   9. Game playing
   10. Object detection and classification
   11. Question answering chatbot
   12. Document summarization
   13. Language modeling
3. You can propose other projects
4. You will work in a group of 5 students
5. Each of you will have a dedicated responsibility

|  |  |  |
| --- | --- | --- |
| **Role** | **Deliverable files** | **Code** |
| Collector | Data.py & Data.zip | Suitable data collection, generation, preparation, and visualization |
| Coder | Train.py & Test.py | Designing architecture, building training and testing module |
| Trainer | Tune.py & Results.docx | Platform preparation, model training and hyperparameter tuning |
| Writer | Report.docx | Report, specification and documentation writing |
| Leader | Report.pptx | Searching, supervising, coordination and presentation |

1. Deliverables are given below\
   1. Data.py: All data processing code
   2. Train.py: Takes hyperparameter as command-line-argument, learns and saves model
   3. Tune.py: Does a grid search over hyperparameters, calls train.py, output performances over grid
   4. Test.py: Loads saved model and predict for new data
   5. Data.zip: All of data used for train and test
   6. Results.docx: All performance results during hyperparameter tuning
   7. Report.docx: Introduction, problem description, architecture, workflow, results, discussions
   8. Report.pptx: 10-minute presentation of the project (12-15 slides)
2. You will not be graded solely over performance. Process is very important.
3. You should prepare sufficiently small dataset so that it fits memory but you have to show that more data will improve performance. You need progressively improving 3 datasets of increasing size.
4. During training you need to show progressive improvement for 3 increasing epochs and 3 increasingly complex architecture
5. During tuning, identify 3 most important hyperparameters besides network architecture and use at least 3 values of each.
6. You can use any high-level library. Comment each line of your code
7. You can use suitable datasets and platforms. But we strongly recommend Python environment using Keras+Tensorflow.
8. Allowed to use any code or data from any source with proper acknowledgement
9. Don’t fall into analysis paralysis
10. Don’t get paralyzed either
11. Try to strike right balance among performance, memory and time
12. Your code must be working and at least it has to run on my laptop within reasonable time
13. Following datasets should be very useful.
    1. IAM Handwriting Database
    2. LibriSpeech ASR Corpus
    3. Europarl Parallel Corpora
    4. IMDB Movie Review
    5. Flickr 8K
    6. Yahoo Finance
    7. Arcade Learning Environment
    8. ImageNet
    9. Stanford Question Answering Dataset
    10. Legal Case Reports Data Set
    11. Brown Corpus
14. Grading
    1. Individual deliverables: CSE471 class test score (individual)
    2. Viva about whole project: CSE472 quiz score (individual)
    3. Overall project quality: CSE472 assignment 4 score (group + individual sum)