IEMS5726 Data Science in Practice (Fall 2022)
Project Specification

Instructions:

1. Do your own work. You are welcome to discuss the problems with your fellow classmates. Sharing ideas is great, and do write your own explanations.
2. All work should be submitted onto the blackboard before the due date.
3. You are advised to submit a single zip/rar file containing the following items.
   a. A IEMS5726_report_1155xxxxxx.pdf file containing the report for your course project.
   b. A IEMS5726_ProjectPart1_1155xxxxxx.py file storing all your programs for part 1, reproduction of the course content. (1155xxxxxx refers to your student ID)
   c. A IEMS5726_ProjectPart2ItemX_1155xxxxxx.py file storing all your programs for part 2, the work on the extension on the course project. (1155xxxxxx refers to your student ID, ItemX refers to the item number in part 2)
   d. A data.txt file storing the description and link to all your data sources.
4. Do follow the template for the course project report. It will save us time on grading your report.
5. If you do not put down your name, student ID in your submission, you will receive a 10% mark penalty out of the course project.
6. No submissions for the course project will result in failing the course immediately.
7. Due date: 19th December, 2022 (Monday) 23:59

Grading Criteria:
The course project is graded based on impression marking, (i) whether your elaboration is clear or not, (ii) whether you can reproduce the course content or not and (iii) whether you have addressed the items in part 2 well or not.

- **Project report for part 1 (40%):**
  - **Problem definition:** descriptions for your project, importance of your project;
  - **Data source:** descriptions for your data, links to your data source;
  - **Data preprocessing:** steps for preprocessing your data, visualizations for your data;
  - **Data analysis:** descriptions and explanations on the machine learning algorithm(s) used, metrics used for assessing the model(s), visualizations for the analysis;
  - **Conclusions and discussions:** findings based on the reproduction, your judgement on the performance of the reproduction.
- **Program in part 1 (20% ~ 40%):**
  - The code for your course project part 1 in a single Python program.
- **Part 2 (20% ~ 40%):**
  - In the report, you need to include the **items / critiques** you have picked at the end. For each item / critique, (i) describe what it is, (ii) explain how you accomplish it, and (iii) describe the corresponding results.
  - The code for your course project part 2 in one or multiple Python programs.

Attention: In the report, do use your own words. Do rephrase the original content from the blog / lecture notes using your own words. Besides, we accept both paragraphs and point form for the report format.

Impression grading (scale of 10):

- 10: Too amazing, beyond our expectations
- 9: Excellent job, slightly better than our expectations
- 8: Great job, meeting our expectations
- 5-7: Good job, some errors spot
- ≤ 4: far below satisfaction

Among the list of all projects, you can pick **any one** of the project topics based on your own preference. Based on the difficulties of the reproduction, the mark allocation and the number of items in part 2 are different.

#	Topics	Difficulty for part 1	Part 1 scores	Part 2 scores	#items in Part 2
1	Retail Data Analysis	Easy	60	40	4
2	Stock Market Index Prediction	Medium	70	30	2
3	Fake News Detection	Easy	60	40	4
4	Semantic Analysis on Movie Review using RNN	Medium	70	30	2
5	Semantic Analysis on Movie Review using BERT	Difficult	80	20	1
6	ImageNet Large Scale Visual Recognition Challenge (ILSVRC)	Medium	70	30	2
7	Speech Synthesis	Medium	70	30	2
8	Deep Image Processing	Difficult	80	20	1
9	Friend Suggestion System from the Facebook Network	Easy	60	40	4
10	Genetic Mutation Classification	Difficult	80	20	1
11	Protein 3D Structure Prediction	Difficult	80	20	1
12	Deep Q Learning (DQN) Agent on the CartPole-v0 task	Medium	70	30	2

In general, if you wish to work on some good features in part 2, and they are not listed in the project description below, please write an email to Danny and seek **prior approval (no later than the final exam)**. All approved new features will be shown on the Blackboard course project submission box.

Descriptions for course projects:

1. Retail Data Analysis

   Part 1: Reproduction for the content below: (60%)
   You can choose any one of the retail data analysis discussed in class.
   - https://towardsdatascience.com/retail-data-analytics-1391284ec7b8
   - https://towardsdatascience.com/machine-learning-for-retail-price-suggestion-with-python-64531e64186d

   Part 2: Pick any 4 items below: (40%)
   - Improve the visualizations for your analysis
   - Perform the analysis using the consideration of the data split of time series data
   - Perform comparison for multiple machine learning algorithms
   - Explain and tune the parameters for the original model used
   - Look for another retail dataset, and perform a better analysis
   - Lodge some applications for the analysis
   - Other features you wish to work on (prior approval from Danny via email)

2. Stock Market Index Prediction

   Part 1: Reproduction for the content below: (70%)
   Follow the blog below, which is discussed in class.
   - https://towardsdatascience.com/predicting-future-stock-market-trends-with-python-machine-learning-2bf3f1633b3c

   Part 2: Must implement the feature below (15%)
   - Propose and verify a money making model

   Pick any 1 item below: (15%)
   - Convert the problem to a regression problem
   - Improve the visualizations for your analysis
   - Perform comparison for multiple machine learning algorithms
   - Explain and tune the parameters for the original model used
   - Lodge some applications for the analysis
   - Other features you wish to work on (prior approval from Danny via email)

3. Fake News Detection

Part 1: Reproduction for the content below: (60%)
Follow the blog below, which is discussed in class.

- https://data-flair.training/blogs/advanced-python-project-detecting-fake-news/

Part 2: Pick any 4 items below: (40%)

- Perform some visualizations for the analysis
- Apply more NLP skills on data preprocessing
- Include extra features as the input for the analysis
- Perform comparison for multiple machine learning algorithms
- Use some deep learning approaches to improve the fake news detection accuracy
- Lodge some applications for the analysis
- Other features you wish to work on (prior approval from Danny via email)

4. Semantic Analysis on Movie Review using RNN

Part 1: Reproduction for the content below: (70%)
Follow the blog below, which is discussed in class.

- https://www.kaggle.com/code/arunmohan003/sentiment-analysis-using-lstm-pytorch/notebook

Part 2: Pick any 2 items below: (30%)

- Run a hyperparameter search to optimize the configurations
- Use pre-trained word embeddings like GloVe word embeddings
- Increase the model complexity by adding more layers
- Increase the model complexity by using bidirectional LSTMs
- Lodge some applications for the analysis
- Other features you wish to work on (prior approval from Danny via email)

5. Semantic Analysis on Movie Review using BERT

Part 1: Reproduction for the content below: (80%)
Follow the steps, which are discussed in class.

- Define the problems using your own words
- Use the dataset: https://jmcauley.ucsd.edu/data/amazon/
- Preprocess the data using at least 4 different NLP skills
- Split the data into training and testing sets
- Train the BERT model
- Perform some visualizations for your data and results

Part 2: Pick any 1 item below: (20%)

- Train a RoBERTa model, and report model performance
- Train a XLNet model, and report model performance
- Append non-text features to the input, and re-train the BERT model
- Use multimodal transformer to improve the model performance
- Lodge some applications for the analysis
- Other features you wish to work on (prior approval from Danny via email)

6. ImageNet Large Scale Visual Recognition Challenge (ILSVRC)

Part 1: Reproduction for the content below: (70%)
Follow the steps, which are discussed in class.

- Define the problems using your own words
- Use the ImageNet dataset 2017: https://image-net.org/challenges/LSVRC/
- **[EDIT] You can pick a subset of data, and do not pick the too small subset of data**
- Train an AlexNet based on the training set
- Report the top 5 error rate for the AlexNet

Part 2: Pick any 2 items below: (30%)

- Train a VGG-Net, and report model performance
- Train a ResNet, and report model performance
- Lodge some applications for the analysis
- Other features you wish to work on (prior approval from Danny via email)

7. Speech Synthesis

Part 1: Reproduction for the content below: (70%)
Follow the blog below, which is discussed in class.

- https://github.com/vincentherrmann/pytorch-wavenet/blob/master/WaveNet_demo.ipynb

Part 2: Pick any 2 items below: (30%)

- Compare your performance with the Google text-to-speech
- Apply WaveNet to another dataset (can be other problems related to sound data)
- Apply WaveNet to other sound problems (e.g. data compression, sound description, … etc)
- Train both parametric model and concatenative model for the same dataset, and compare the models performance
- Other features you wish to work on (prior approval from Danny via email)

8. Deep Image Processing

Part 1: Reproduction for the content below: (80%)
Follow the Colab below, which is discussed in class.

- https://github.com/pytorch/examples/blob/main/dcgan/main.py
- **[EDIT] You can use any image datasets to train your GAN, and then your GAN will be an expert on that types of objects**

Part 2: Pick any 1 item below: (20%)

- Compare your GAN with any variations of GAN, e.g. DCGAN, GenForce, BigGAN, StyleGAN, SeFa, … etc
- Apply your GAN for some image processing techniques, e.g. GAN inversion, colorization, super-resolution, image reconstruction, image inpainting, semantic image manipulation, … etc
- Other features you wish to work on (prior approval from Danny via email)

9. Friend Suggestion System from the Facebook Network

Part 1: Reproduction for the content below: (60%)
Follow the specification below, which is discussed in class.

- https://courses.cs.washington.edu/courses/cse140/14wi/homework/hw4/homework4.html
- https://github.com/vineetamonkar/Friend-Recommendation-System-using-Python

Part 2: Pick any 4 items below: (40%)

- Perform some graph visualizations for the recommender system
- Apply content-based filtering or collaborative filtering for some other datasets
- Use matrix factorization to construct a Facebook recommender system
- Lodge some applications for the analysis
- Other features you wish to work on (prior approval from Danny via email)

Or pick any 2 items below:

- Learn the latent variables using deep learning
- Learn the bag-of-words representation using deep learning
- Learn the hidden features from user's and item's behaviour using two parallel neural network (DeepCoNN)
- Other features you wish to work on (prior approval from Danny via email)

Or pick 2 items from the former part 2, and pick 1 item from later part 2.

10. Genetic Mutation Classification

Part 1: Reproduction for the content below: (80%)
Follow the publication below, which is discussed in class.

- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7563092/

Note: You need to write to the author to request data. Sample code is also unavailable.

Part 2: Pick any 1 item below: (20%)

- Train other BERT models (e.g. RoBERTa, XLNet, … etc), and report model performance
- Lodge some applications for the analysis
- Other features you wish to work on (prior approval from Danny via email)

11. Protein 3D Structure Prediction

Part 1: Reproduction for the content below: (80%)
Follow the publication below, which is discussed in class.

- https://www.nature.com/articles/s41467-021-23303-9

Note: You need to write to the author to request data.

Part 2: Pick any 1 item below: (20%)

- Perform some visualizations for the protein complexes
- Compare your work with DeepFRI web service
- Lodge some applications (except web service) for the analysis
- Other features you wish to work on (prior approval from Danny via email)

12. Deep Q Learning (DQN) Agent on the CartPole-v0 task

Part 1: Reproduction for the content below: (70%)
Follow the tutorial below, which is discussed in class.

- https://pytorch.org/tutorials/intermediate/reinforcement_q_learning.html

Part 2: Pick any 2 items below: (30%)

- Use both policy-based and value function-based for the CartPole-v0 task
- Apply reinforcement learning for some other games with action states (You can pick two different game applications as 2 items)
- Perform visualizations for your game applications
- Other features you wish to work on (prior approval from Danny via email)