Group Project: Developing An intelligent Application that applies Deep Learning to solve real world problems

Due Date of all file submissions: Last day of week 13

Purpose:

The purpose of this Project is to:

- To apply all the accumulated knowledge learned throughout the term and get some hands-on-experience in applying the deep learning techniques and concepts to a real-world problem.

General Instructions:

Be sure to read the following general instructions carefully:

1. This project must be completed in groups of 4 by all students.
2. Make sure to have a complete screenshot/figures/tables, partial screenshots/figures/tables will not earn any marks.
3. You will have to provide a **group presentation** during the scheduled class session on the week 14 (See course outline/schedule for specific date). All group members must actively participate in the presentation. See Presentation section for more details.
4. Any submission without an accompanying presentation will lose 70% of the grade.
5. In your analysis report make sure you provide an introduction and clearly state the facts and findings. Any submission missing Analysis report will lose 70%. See Submission section for more detail regarding analysis report.

Submission:

There are three elements to be submitted for this project in one zipped folder:

1. All Source Code, please create a project folder and include all project python scripts/modules and screenshot of output, as needed. Separate all generated output in a separate subfolder named 'results'.
   The complexity of the program is considered in the final mark. Note that the program must also run successfully. Programs that are complex but do not run will not receive full marks. (40 Marks)

2. ReadMe File. The ReadMe file should include the information of all group members and detailed instructions on running the code. Include information on instructions to retrieve/process dataset and dependent external libraries. (10 Marks)
3. 2-page limit analysis report. The report should follow IEEE manuscript guidelines for conferences proceedings [See link for details [IEEE Manuscript Templates]. Your report must include the following Sections: (20 Marks)
    a. Introduction = Describing your problem and dataset
    b. Methodology = Describing your Algorithms/Models
    c. Results = Describe the evaluation metrics and performance achieved
    d. Contribution = Outline the contribution of each group member

Presentation: (30 Marks, 10 Marks each)

1. The duration of the presentation should be no more than 10-minutes. The presentation will be followed up with a 5-minute Q/A session. The group should prepare 1 or 2 questions to stimulate class discussion, if there are no questions from other students.
2. All students should complete a survey report for all group presentation (excluding their own presentation).
3. Students should explain the following in their presentation:
    a. Description and scope of the problem
    b. The dataset
    c. Methodology
    d. Results/Analysis
    e. Some of the technical challenges they experience and how they addressed them

Description:

In groups of 4 students, each group will search and identify real world problems on the highly popular machine learning platform Kaggle. Kaggle is an online community of data scientists and machine learning researchers that houses a large collection of publicly available datasets and host of many competitions to address real world problems.

Groups should select one dataset with labels from Kaggle. The dataset should contain images or text for computer vision or natural language processing tasks. The dataset should be non-trivial (i.e. toy datasets like mnist will not be accepted).

Groups will then run the following 3 experiments:

1) Supervised Learning: Design and implement your own CNN/RNN model to address the specific task (i.e. image classification/segmentation/anomaly detection/sentiment analysis) of your problem. Groups should experiment and analyze the performance of the architecture using varying hyperparameters (i.e. #layers, #neurons, #kernals).

Groups should also experiment with different regularization and normalization optimization methods.

2) Unsupervised Learning: Design and implement a model for feature extraction or data augmentation task using unsupervised learning approach. Groups that address the feature extraction should use transfer learning to experiment and analyze the performance of pre-training a model for the supervised learning task. Groups that address data augmentation should generate new training samples from their generative model and analyze the performance of the supervised learning task with the inclusion of the new generate training samples.
3) State-of-the-art Models: Many state-of-the-art architectures have emerged, which achieves many state-of-the-art performances for many real-world problems. As such, many frameworks offer support for these models. Groups should research and implement one state-of-the-art model for their problem task. A pre-trained version of many of these state-of-the-art models are available in TensorFlow, groups should experiment and analyze training a new model from scratch and transfer learning from the pre-trained model.

Groups should analyze and compare the performance of all 3 experiments.

Suggested Datasets/State-of-the-art Models:

The following is a list of suggested datasets and state-of-the-art models that groups can selected. This is just a short list of some examples that groups may decide to select, but groups are not limited to the options on this list and can decide to use something not on the list.

Datasets:

- Liberals Vs Conservative on Reddits - https://www.kaggle.com/neelgajare/liberals-vs-conservatives-on-reddit-13000-posts
- IP102-Dataset - https://www.kaggle.com/rtlmhjbn/ip02-dataset
- NLP with Disaster Tweets - https://www.kaggle.com/c/nlp-getting-started/data
- Face Mask Detection - https://www.kaggle.com/andrewmvd/face-mask-detection
- Brain MRI Images for Brain Tumor Detection - https://www.kaggle.com/datasets?tags=13207-Computer+Vision
- Birds 400 – Species Image Classification - https://www.kaggle.com/gpiosenka/100-bird-species
- CelebFaces Attributes (CelebA) Dataset - https://www.kaggle.com/jessicali9530/celeba-dataset

- Top 1000 Github Repositories for Popular Keywords - https://www.kaggle.com/anshulmehtakaggl/top-1000-github-repositories-for-multiple-domains
- TMDB Box Office Prediction - https://www.kaggle.com/c/tmdb-box-office-prediction/data?select=test.csv
- iWildCam 2019 - https://www.kaggle.com/c/iwildcam-2019-fgvc6/data
- Histopathologic Cancer Detection - https://www.kaggle.com/c/histopathologic-cancer-detection/data

Models:

- ResNet
- ResNet-RS
- EfficientNet
- Vision Transformer
- Transformer
- InceptionV3
- InceptionResNetV2
- MobileNet
- Xception
- DenseNet
- NASNetMobile
- VGG

Evaluation criteria	**Not acceptable**	**Below Average**	**Average**	**Competent**	**Excellent**
	0% - 24%	**25%-49%**	**50-69%**	**70%-83%**	**84%-100%**
Functionality	Missing all functionalities required	Some requirements are implemented.	Majority of requirements are implemented but some are malfunctioning.	Majority of requirements implemented.	All requirements are implemented Correctly.
Classes	Classes have been created incorrectly or completely missing.	Classes have been defined but have errors. Instances are incorrectly used.	Classes have been defined correctly but instances are used incorrectly or not created at all.	Classes have been defined correctly but some instances are used incorrectly.	Classes are correctly defined and makes use of its own functions which are called somewhere else in the code. Instances have been created and used correctly.
Documentation	No comments explaining code changes.	Minor comments are implemented.	Some code changes are correctly commented.	Majority of code changes are correctly commented.	All code changes are correctly commented.
Design	No adherence to object design principles.	Minor adherence to object design principles.	Some object oriented and modulus design principles are adhered to.	Majority of Object oriented and modulus design principles are adhered to.	Object oriented and modulus design principles are adhered to.
Testing & Evaluation	No evidence of testing and evaluation of the requirements.	Minor evaluation and testing efforts.	Some of the requirements have been tested & evaluated.	Majority of requirements are tested & evaluated.	Realistic evaluation and testing, comparing the solution to the requirements.
Demonstration Video	Very weak no mention of	Some parts of the code	All code changes	All code changes	A comprehensive

	the code changes. Execution of code not demonstrated.	changes presented. Execution of code partially demonstrated.	presented but without explanation why. Code demonstrated.	presented with explanation, exceeding time limit. Code demonstrated.	view of all code changes presented with explanation, within time limit. Code demonstrated.