Neural Networks for Machine Learning Project
- Part I: Convolutional neural networks for image analysis
  - Preliminary work
  - Exploratory data analysis
  - Reproduce research results
- Part II: Transformers for text analysis
  - Preliminary work
  - Exploratory data analysis
  - Classify text
- Part III: Variational autoencoder for image analysis
  - Preliminary work
  - Migrate from one framework to another framework
  - Experiment with different model configurations

Neural Networks for Machine Learning Project

The project comprises three parts that survey different learning paradigms and different modalities. Conceptually, the project will exercise your understanding of important machine learning concepts such as input representation, error functions, optimization methods, regularization, and validation. Practically, the project will give you some experience with popular frameworks (e.g., TensorFlow and PyTorch).

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in RFC 2119.

You shall submit two files:

- A LaTeX report (.pdf) in ACM article format with your results and discussions: PROJECT.LAST_NAME.FIRST_NAME.pdf
- A notebook (.ipynb) with all of your code for each part of the project: PROJECT.LAST_NAME.FIRST_NAME.ipynb

Part I: Convolutional neural networks for image analysis

Context:

- **Paper.** Deep Residual Learning for Image Recognition
- **Data.** TensorFlow Datasets is a collection of datasets ready to use, including ImageNet and CIFAR-10
- **Models.** Keras Applications are canned architectures with pre-trained weights, including ResNet models

Preliminary work

Optional but highly recommended:

1. You should go to Getting started with TensorFlow 2 on Coursera[1] and watch every video[2] seriatim.
2. You should go to Customising your models with TensorFlow 2 on Coursera and watch each video in Weeks 1-2.

Exploratory data analysis

Requirements:

- You shall conduct a comprehensive statistical analysis of any of the datasets used in the paper and discuss your results.
- You shall develop data visualizations of examples/instances in any of the datasets or your statistical analyses and discuss your results. You may use dimensionality reduction libraries to plot high-dimensional data.

Reproduce research results

Requirements:

- You shall use TensorFlow APIs to imitate Fig. 4 in the paper for ResNet-50 and ResNet-101 and reproduce the results in Tab. 3 and Tab. 4 for both ResNet-50 and ResNet-101 and discuss your experiment.

[1] Click *Enroll for Free* and then click *Audit*.
[2] Many videos can be played (harmlessly) using a playback speed up to 1.5.

Part II: Transformers for text analysis

Context:

- **Paper.** Attention Is All You Need
- **Data.** Datasets
- **Source.** Text classification with Transformer

Preliminary work

1. You should go to Customising your models with TensorFlow 2 on Coursera and watch each video in Weeks 3-4.

Exploratory data analysis

Requirements:

- You shall conduct a comprehensive statistical analysis of any of the natural language datasets in `tf.keras.datasets` and discuss your results.
- You shall develop data visualizations of examples/instances in any of the natural language datasets or your statistical analyses and discuss your results. You may use dimensionality reduction libraries to plot high-dimensional data.

Classify text

Requirements:

- You shall use the source implementation to train and evaluate a text classifier with a Transformer block on any of the natural language datasets in `tf.keras.datasets` and discuss your results.

Part III: Variational autoencoder for image analysis

Context:

- **Paper.** beta-VAE
- **Source.** 11-VAE.ipynb

Preliminary work

- You should go to Probabilistic Deep Learning with TensorFlow 2 on Coursera and watch each video in Weeks 1, 2, and 4.

Migrate from one framework to another framework

Requirements:

- You shall reproduce the results obtained with the source implementation, a VAE in PyTorch, which simply entails running the notebook without any errors to serve as a baseline.
- You shall migrate the source implementation to TensorFlow and reproduce the baseline results using your TensorFlow implementation.

Experiment with different model configurations

Requirements:

- You shall modify the `optimizer` (in your TensorFlow implementation) to use a different optimization method than Adam and discuss the results as compared to the baseline.
- You shall parameterize the coefficient `0.5` in the expression for the KL divergence in the loss function (in your TensorFlow implementation) and run experiments using different values for this parameter, which varies the amount of regularization.[3] Report and discuss your results.

[3] Recall \beta=1 corresponds to the original VAE formulation [Higgins et al., 2017].