PROJECT DESCRIPTION

The project is an essential part of this class. It will allow you to demonstrate your Big Data Analysis skills and create something that you are proud of. It can also be a valuable addition to your projects portfolio that you can demonstrate to prospective employers.

Project Requirements

- The project has two key components to it:
  - Understanding a recent machine learning or analytics technique and associated algorithm(s)
  - Implement and apply it to a standard dataset of sufficient complexity

- A list of acceptable techniques is mentioned later in this document. You are also free to choose any other technique provided it is of similar complexity.

- Your project deliverables should consist of two major parts:

  1. A report in IEEE conference format https://www.ieee.org/conferences/publishing/templates.html

     It should have the following sections at the minimum:
     - Abstract
     - Introduction and background work
     - Theoretical and conceptual study of the technique/algorithm you would like to implement
     - Results and analysis. Please include results in tabular or graphical formats. Be sure to analyze your results well.
     - Conclusion and future work
     - References

     The report excluding the references should be 4-6 pages long. The final file should be converted to PDF format before submission.

  2. Your code, link to dataset, results, and instructions for compiling and running

Below are the requirements:

- You should build your project such that it can run on Apache Spark. You can use either Scala or PySpark for coding. In case of Scala, you should build a jar file that can run on AWS cluster. In case of PySpark, you need to provide a notebook or other code that can run on AWS or Databricks.

- You are allowed to use public paths such as AWS S3 or UTD web account.

- Be sure to include instructions on how to compile and run your code.

- You are allowed to use Big Data libraries, such as
  1. Databricks deep learning:
  https://docs.databricks.com/en/machine-learning/train-model/deep-learning.html
  2.Azure Spark Deep Learning:
  https://learn.microsoft.com/en-us/azure/synapse-analytics/machine-learning/concept-deep-learning
  3. Spark NLP
  https://sparknlp.org/docs/en/quickstart

- Your code should run in a distributed fashion i.e. using a cluster. Just running your code on a single processor will not be sufficient.

- You are free to choose a dataset of your choice from sources like Kaggle, or any other source. Do not include the dataset as part of the deliverables, instead host it on your UTD web account or AWS S3. This will allow the TA to run your code without having to search the dataset or download huge files. If you do not know how to host data on UTD account, contact the TA.

- A log file of your experiments and parameters should be maintained and submitted. Example of a log file is shown below

Experiment Number	Parameters Chosen	Results
1	Neural Net: Number of layers = 4 Regularization Parameter =	Train/Test Split = 80:20 Training Accuracy = 95% Test Accuracy = 88%

	0.6	Training RMSE = 1.67 Test RMSE = 3.08
2	.....	.....

- Below are some further administrative requirements:

  - All contents of your report must be original. You have to write the report in your own words. It is acceptable to include figures from the references, provided you state the source clearly in the caption.

  - Your report will be checked for plagiarism. Any violation will carry strong penalties, including reporting the incident to university authorities.

  - Team size requirements: Project can be done in teams of 1 to 4 students. More than 4 students cannot be in a team under any circumstances. You can only form team within the same class and section.

  - You are highly encouraged to start early on your projects. You do not need to wait for the instructor's approval of your choice of project.

Project Topics

Below is the list of topics that you can choose from:

- Spark streaming for novel real time applications such as novel class detection, IoT, web traffic analysis, stock market prediction, real-time anomaly detection, political actor detection, etc
  Note: you cannot do anything similar to what you did in homework.

- Novel graph mining applications using GraphX. See this presentation for possible topics:
  Note: Your project should involve significant work and you cannot do anything similar to what you did in homework.
- Document summarization using NLP techniques
- Transformers for machine translation
- Transformers for question answering systems/chatbots

- Recurrent Neural Networks (RNN) for time series prediction (e.g. stock market, weather, hurricane intensity data)
- Image and video captioning with deep neural networks
- Autoencoders for bioinformatics or image processing
- Scene recognition with deep neural networks
- Deep Reinforcement Learning
- Genetic sequence analysis using deep neural networks
- Reinforcement learning for game playing
- Meta-Learning
- Adversarial Machine Learning
- Statistical Relational Learning
- Human assisted Machine Learning

Deliverables and Deadlines

Deadline	Project Phase	Deliverable
Friday July 19 Midnight (No late days can be used for this)	Project Status Report	Submit a report containing following on eLearning: • Project Topic • Team Members • Technique/Algorithm you plan to implement • Dataset details, such as number of features, instances, data distribution • Coding language / technique to be used • Preliminary Results (if available)
Sunday July 30 Midnight Friday August 2 Midnight (You can use at most 2 late days for this, if available)	Final Report	• Complete project deliverables as described in the requirements above to be submitted via eLearning ** Your report and code will be checked for plagiarism **