

Summer 2021: CSEE5590 – Special Topics

Extra Credit Project

Project Overview:

In this project, we want you to use the prior knowledge and work from your assignments and use cases to build a model. But the important part is that we want to make this model available for use, so you have to build an application (interface) for your model.

Project Teams:

Teams can be in Max. of 3 students.

Teams of two or more students are expected to show more effort and work compared to others.

Project Description:

In module-2 we learned how to build a neural network, and from the assignments and use cases, you should have the skills to work on any available dataset or build your dataset for your problem. For this project, we want you to build a model on a dataset of your choice depending on the problem you would like to work on.

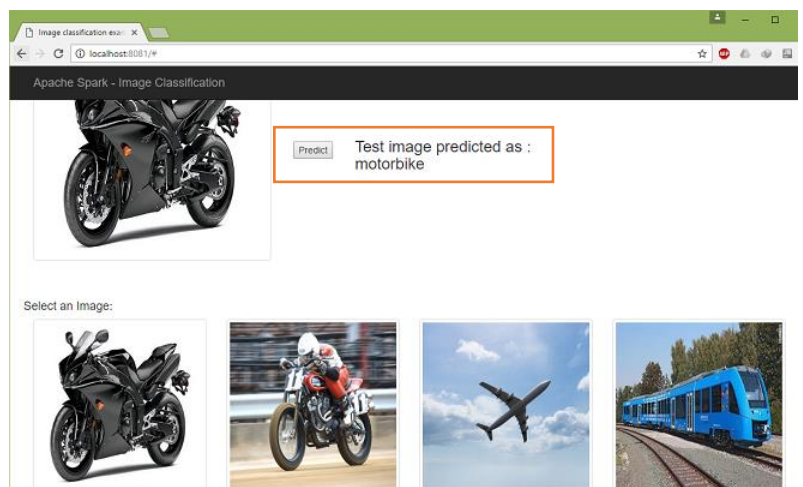
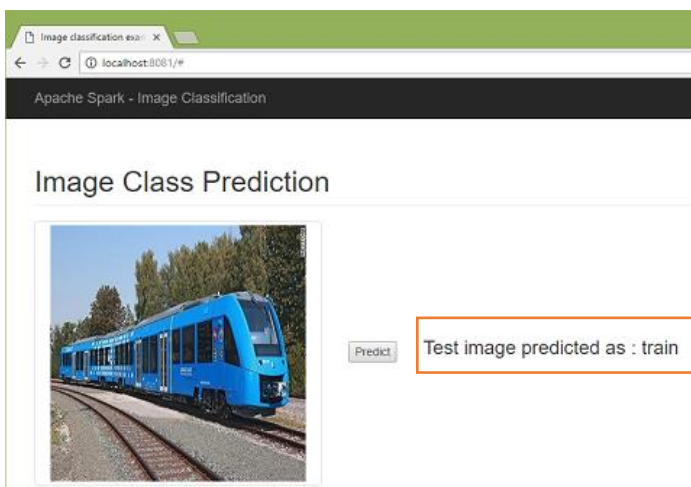
For example:

- Text generation.
- Text classification.
- Text Sentiment analysis.
- Image classification.

The goal of this project is to build an application interface for your model. A web page to be able to load images and make a prediction or write text and predict the sentiment as an example. The application can be of your choice, like a web page or a mobile app. The important part is that you have saved your trained model, and the server or mobile app is loading the model and making a prediction every time a request is initiated.

You can use Flask (python micro web framework) or JavaScript to build the mode interface. Here is an example for an image classification webpage:

- Train your model and save your model weights.
- For predictions load your model weights and do model prediction only.
- Show your model class prediction in your application.



Project Evaluation:

The maximum score is: 5 points

- 3 Points for the technical part and implementation.
 - *Based on the submitted code and execution.*
- 2 Points for documentation and technical explanation for the method used.
 - *Based on the PDF Report quality, and Turnitin score.*

Execution: an important part of the project implementation, you have the option to host your model in a cloud service ([AWS](#), [Google Cloud](#), [Heroku](#), [gradio](#)...) and have your project publicly available with a web address. Otherwise, you can use your local machine to deploy this and record a video of 1~3 minutes as proof of execution.

Documentation: This is a very important part of the project; we do not expect you to write a very detailed report of more than 10 pages. But we want to see your understanding and the reason you applied the methods on the dataset and choice of layers, activation functions. Make sure you add a references section for the sites and tutorials you used.

The quality of the report is more important than the quantity.

Resources:

- 1- Flask: <https://flask.palletsprojects.com/en/2.0.x/quickstart/>
- 2- [How to build a web application using Flask and deploy it to the cloud](#)
- 3- YouTube: [Deploy Keras Neural Network to Flask web service | Part 1 - Overview](#)
- 4- [Building a simple Keras + deep learning REST API](#)
- 5- [Keras + Flask on AWS instance.](#)
- 6- [Gradio getting-started](#) check the [sharing](#) option for free cost
- 7- [Gradio ML examples](#)
- 8- [Gradio hosted hello world](#)
- 9- [Heroku + FastAPI example](#)
- 10- [Turning Machine Learning Models into APIs in Python](#)
- 11- [Building a Plant disease classification web app in Keras and Tensorflow.js \(PART 2\)](#)