EE 769: Project Proposal

Group Members:

[1] Shubham Namdev Kamble - 180020105
[2] Bhagyasree M - 203010013
[3] Aman Sunil Butoliya - 203070001

Project title:

Implement ML frameworks and their training algorithms efficiently from scratch such as:

- ❏ Neural networks
- ❏ Random forests
- ❏ SVM's
- ❏ Bonus marks for using native CUDA to speed up non-neural-network ML frameworks

Project objective, beneficiary (e.g. a client, industry, ML beginners, or only you etc.), benefit (e.g. new app, analysis, insights, technique etc.), and/or novelty (what doesn't exist that you might provide).

Ans: Project will be beneficial for ML beginners who will learn how to convert their theoretical understanding of any model into code. They will learn to implement basic models like Random Forest, SVM, and Neural Networks from scratch.

Project type: Understanding models and implementing them from scratch

What will you spend the most time on? : Most time will be spent on Building models. Mainly Neural networks.

Explain your approach in detail (including how it will be different from solutions available on the internet (papers, kaggle kernels etc. Also, include links to existing solutions).

Ans: We will select a dataset to implement these models from scratch and find accuracies for each of these models to find which one gives the best results.

Our model mainly includes:

[1] Importing packages and reading a dataset
[2] Feature Engineering/ Preprocessing
[3] Splitting data
[4] Build and implement training Algorithm on the training dataset
[5] Test model accuracy on the given dataset

If time permits we will also build and implement training algorithms of Naive Bayes and KNN. And use native CUDA to speed up non-neural-network ML frameworks.

What challenges you might face, and what will be your plan B, if some of those can't be completely solved?

Ans: As we are building and implementing our theoretical understanding into code, problems that mainly encountered will be related to coding. In that case, as most of the models are already build and implemented on the internet we will take the help of blogs and Youtube Videos that explain these models and code in an easier way. And revising the algorithm, again and again, is always a solution.

Give an estimate of your roles, responsibilities, and time estimates. E.g. "Gajodhar will work on coding the Android app. He will spend 10 hours reading about how to make apps, 5 hours on how to use ML models in apps, and 15 hours coding the app. Shaktimaan will compare different ML models in a PC to decide which model will go in the app. He will spend 10 hours understanding the models, and 20 hours coding and running experiments to compare models." We are expecting about 30 hours per team member.

Ans: Since building ML algorithms from scratch would help us in understanding algorithms in a better way, all the work must be done together. The building steps are as follows:

[1] Get a basic understanding of an algorithm - Initial time will be spent on understanding the algorithm, from different learning sources like Course Material, Books, Blogs**.** Around **6 hrs**. will be spent on this by each member.

[2] Break the algorithm into chunks - Rather than reading the whole concept, we will break it into chunks and go through each chunk in detail. Around **3 hrs**. will be spent on this by each member.

[3] Start with a simple example - As we have broken the algorithm into chunks and understood it, It's time to start coding and to keep things simple, we will start with a small dataset. Around **20 hrs.** will be spent on this by each member.

[4] Validate with trusted implementation - After understanding, breaking it down into chunks and implementing it with a simple dataset. We will validate our implementation by comparing it with the standard **scikit-learn** library. Around **6 hrs.** will be spent on this by each member**.**

[5] Writing up our process - We will finish our work by writing the report showcasing our work. Around **6 hrs**. will be spent on this by each member.

So each group member is dedicated to spend around 42 hours on this project.