CS205 Project Proposal

Connor Buchheit, Peter Chen, Wesley Osogo, Tianfan Xu

20 February 2024

1 Topic: Parallelizing Support Vector Machines (SVM)

Our group hopes to exploit parallelization in a machine learning technique: SVM. This machine learning technique has been around for a long time and there are many areas in this technique that could be improved through employing parallelization.

2 Areas of Parallelization/Bottlenecks

1. Batch Gradient Descent: One significant improvement we could make to the technique is divide the input set into batches, run gradient descent on each batch, and synchronize the results. This is essentially SIMD, since the same instructions (gradient descent) is used on each batch (data) of input. The key things to consider when parallelizing batch gradient descent is the optimal size of each batch compared to the total input size in order to maximize efficiency. We could use a distributed memory model because each batch only uses its own data.

2. Optimization: The optimization step to find the optimal set of weights is a computational bottleneck. To optimize, we have to solve the dual Lagrangian problem, with many parameters, which can potentially be solved by chunking our data and distributing these chunks across various processors, reducing the size of the problem. This follows MIMD, as each thread has its own set of data and delivers its own set of instructions.

3 Runtime/Memory

Since it's a machine learning problem, the accuracy and runtime is heavily correlated with the input size. Batching requires to store the entire dataset, so for this application, one constraint would be the size of the dataset we use such that we can properly utilize the memory allotted to us. Similar to runtime, the memory usage is heavily correlated with input size.