IIT JODHPUR

Minor-1 Examination (OPEN-BOOK): EEL71020 Hardware Design for AI

Guidelines (Total time: 60 minutes, Maximum Marks: 15):

- Please read the question paper very carefully (both sides of this paper). **NO clarification is required in any question.** In case of any doubt, assume whatever you wish to and clearly state that in your answer.
- This is an OPEN-BOOK examination. However, usage of internet/any AI tool is **NOT** allowed.
- Note1: Meaning of hardware architecture is to show the connections between memory, computational units and so on. Further, the performance in terms of clock cycles need to be analyzed. You can also describe your design in any HDL etc. If unaware of HDL, refer to Note 2 below.
- *Note2: If you wish to not draw/describe the hardware architecture in a HDL because you do not have a proper background, you can simply write the C/C++ code of this hardware architecture taking proper assumptions for the overall setup.*

---

1. Develop a detailed hardware architecture for the implementation of the internal level block as illustrated in the below scheme for realizing the random forest algorithm to accomplish a classification task? You can assume that you have a distributed memory architecture involving many

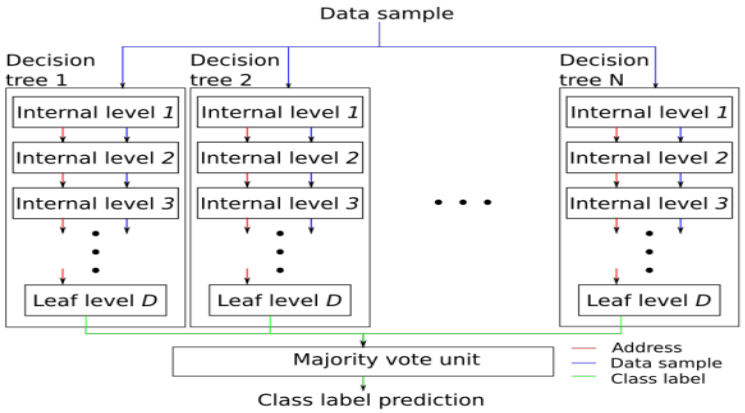

Figure 1: Random Forest Algorithm Realization

block RAMs that can store the parameters of the random forest. As can be understood from the above, each tree predicts an output class label, and the predictions of all trees are processed by a majority vote unit. Recall that each internal level shown above can be visualized as a node in each decision tree of the random forest. [4 Marks]

2. One of the important steps in K-means clustering methodology is the calculation of the distance between the data points (P) and the centroid clusters (C). There are 2 ways to compute this distance (P_i denotes the x,y co-ordinates of the data point and C_i denotes the x,y co-ordinates of the centroid):

   (a) Manhattan distance = $\sum(|P_i - C_i|)$

   (b) Euclidean distance = $\sqrt{(\sum(P_i - C_i)^2)}$

As per the methodology of K-means clustering, after the distance is computed and sorted, the centroids need to be updated in an iterative manner till all the data points are segregated in the different clusters.

Develop a hardware architecture that can enable K-means clustering inference (i.e., model testing) in a reconfigurable manner for the design allowing computations in both the above distance computation ways? Note that your complete design for doing K-means model inference allows the operation in 2 modes: first for Manhattan distance, second for Euclidean distance? You are also instructed to keep the number of output classes in your design as an input from the user. Find out an expression of the throughput of your design if the performance is to be kept very high? [3 + 1.5 Marks]

3. Feature engineering is an important task in building models for machine learning applications. This means that the input data is pre-processed in different ways and certain features are to be calculated as summary of the features present in the original dataset. Let's say that we are interested in the on-chip implementation of feature engineering. In other words, the new summarized features (p[i], q[i], r[i]) are to be computed in the hardware in runtime manner as the input data (meaning original features- a[i], b[i], c[i], d[i], e[i]) gets fed to the on-chip circuitry. The summarization process happens in an iterative manner as shown in the below diagram (elucidating that the number of input features decreases from 5 to 3 in this example). Design the hardware so that the feature engineering can be done in the best possible manner from the performance viewpoint considering sufficient resources? *Hint: To meet the specified targets, you may think to optimize the below code before implementing it in hardware.* [5 marks]

```
while (i < 500) {
p[i] = a[i] + b[i] + c[i];
q[i] = a[i]*b[i] - 5;
r[i] = c[i] + 2*d[i] + 3*e[i];
i = i + 1;
}
```

Figure 2: Example of feature engineering

4. Mention any differences that you have observed in the design for ML applications in comparison to the design of conventional computational applications? [1.5 Marks]
