kNN-STUFF: Performance Models (v0.1)

João Vieira

January 12, 2020

Single Accelerator

$$\#\text{cycles per classification} = (M + M \times N) \times \varepsilon_{\text{DMA}} + 17 + k$$

Parallel Configuration 0

$$\#\text{cycles per classification} = \frac{(M \times C \times A + M \times N) \times \varepsilon_{DMA} + 17 + k \times C \times A}{C \times A}$$

Parallel Configuration 1

$$\#\text{cycles per classification} = \frac{(M \times A + \frac{M \times N}{C}) \times \varepsilon_{DMA} + 17 + k \times C \times A}{A}$$

Notes

1. Note that when using parallel configuration 1, there is an additional software phase that merges the results calculated in hardware. This additional software phase incurs in an overhead with the complexity per classification $f(M, k, C) = k \times C \times (M + k)$.
2. All these models assume a perfect CPU that is capable of explore 100% of kNN-STUFF capabilities. For small datasets, this may not be true.
3. When using parallel configurations, one can only achieve the maximum performance if:
   - when using parallel configuration 0, the number of testing samples is a multiple of $A \times C$;
   - when using parallel configuration 1, the number of testing samples is a multiple of A.
4. The bigger the dataset, the more reliable are these models.

Symbols

- M: number of features per sample;
- N: number of training samples;
- ε_{DMA}: Xilinx DMA efficiency (≈ 1);
- k: k nearest neighbors;
- C: number of clusters;
- A: number of accelerators per cluster.