Short Paper: ScaleGANN — Scaling ANN Index Build using Cost-effective and Cloud-native GPUs

Table 1: Index Construction Time (s).
Ovr.=Overall Time, B.O.=Build-Only Time

Dataset		Nv. CAGRA	GGNN	DiskANN	ScaleGANN
Sift100M	Ovr.	1790	1287	9597	4727
	B.O.	1693	1287	7848	2570
Deep100M	Ovr.	1775	2004	21159	5222
	B.O.	1673	1840	19155	2797
MSTuring 100M	Ovr.	2063	10977	20221	6672
	B.O.	1871	10744	18089	3480
Laion100M	Ovr.	5483	44735	62109	11259
	B.O.	3850	43131	57163	6504
Sift1B	Ovr.	18155	13899	119039	70617
	B.O.	17362	13025	83732	27676

Table 2: Index Build-Only Time (s) over GPU Parallelism.

	Sift100M	Deep100M	MSTuring100M	Laion100M
1 GPU	2570	2797	3480	6504
2 GPU	1495	1557	1992	3394
4 GPU	877	882	1158	2003

1 Spot Instance Cost Analysis

In this supplemental material, we provide an indexing cost analysis of DiskANN and ScaleGANN. For index construction, DiskANN uses a single CPU machine and disk, while ScaleGANN uses one/multiple GPU spot instance(s) for shard index construction and a regular CPU machine with disk for partitioning and merging. Generally, the cost can be directly evaluated by the overall index construction time multiplied by the regular machine price. However, in ScaleGANN we also have to consider the time taken for data transfers between GPU spot instances and CPU machine. Therefore, the total cost for ScaleGANN using GPU spot instance(s) is computed as: (overall construction time + data transfer time) $\times$ CPU price + (index build-only time + data transfer time) $\times$ GPU spot instance price. Meanwhile we estimate data transfer time by simulating shard data sent to GPU and the index returned to CPU. Each task involves at most 16GB of data bounded by GPU memory, so we estimate total transfer time as: number of shards $\times$ 16GB / network bandwidth.

Assuming the use of 4 GPUs for ScaleGANN build parallelism. According to AWS ECS [1], a regular Linux CPU machine with around 80 threads, 200G RAM and 2T disk (e.g., c5d.24xlarge, c5n.18xlarge) is at \$3.9-4.6/h, while a Linux GPU machine with 4 16G V100 (e.g., p3.8xlarge) has regular price \$13.7/h and spot instance price changing normally between \$1.22-3.67/h. For a case study, we select c5d.24xlarge for CPU and p3.8xlarge for GPU spot instance which both offer at least 10Gbps network bandwidth. Using the GPU-compatible Laion100M as an example, 100 shards is enough for index build in Table 1, indicating a 1600G upper bound for data transfer amount and correspondingly at most 160s, namely 0.045h, data transfer time. According to Table 1, DiskANN's overall build time is 17.25h, while ScaleGANN's partition and merge takes 1.32h calculated by overall time minus the index build-only time. As shown in Table 2, Laion100M takes 0.56h to build the index using 4 V100, thus obtaining an overall build time of 1.88h after adding the partition and merge time. Therefore, the cost estimate for DiskANN with regular CPU is at least \$67.3 (17.25×3.9), while ScaleGANN costs at most \$11.1 ($(1.88+0.045) \times 4.6 + (0.56+0.045) \times 3.67$) which is even cheaper than DiskANN.

Notably, in this scenario, even on-demand cloud GPUs can be more cost-effective than CPUs. This is primarily due to the efficiency of GPU acceleration on high-dimensional datasets, which substantially reduces runtime and, consequently, the overall cost. For low- and mid-dimensional datasets, using GPU spot instances still offers indexing speedups while greatly lowering the cost. Moreover, although older regular GPUs are already relatively affordable, newer GPUs with larger memory and more threads enable faster and more stable indexing by reducing the number of shards and speeding up distance computations. In such cases, spot instances of the latest GPUs can achieve better performance at the same or even lower cost than older regular GPUs.

References

[1] AWS. Amazon ec2 spot instances. https://aws.amazon.com/cn/ec2/spot/, 2024.