CAGRA: Highly Parallel Graph Construction and Approximate Nearest Neighbor Search for GPUs

Hiroyuki Ootomo · Akira Naruse · Corey Nolet · Ray Wang · Tamas Feher · Yong Wang
Nvidia Corporation
ICDE 2024

Presenter: Lim Yuan Jee
National Cheng Kung University

國立成功大學
National Cheng Kung University

Discussion topics

- Introduction
- Why Graph Based ANNS
    - Challenges
- CAGRA
    - Key feature
    - Search Algorithm
    - Search Algorithm Optimization
- Evaluation
    - Build Performance
    - Search Performance
    - Scalability
- Benchmark
    - Results
    - Further Benchmarks
- Conclusion

Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

Introduction

- The paper introduces CAGRA, a novel algorithm for graph construction and ANNS specifically optimized for NVIDIA GPUs.
- Limitations of Existing ANNS Approaches for Large Datasets
  - On large datasets, computational costs become prohibitive
  - ANNS offers a viable solution by striking a balance between throughput and accuracy
- At the end of the paper, we will be able to see:
  - How does it work?
  - How much performance improvement? (compared to CPU)

Discussion topics

- Introduction
- Why Graph Based ANNS
    - Challenges
- CAGRA
    - Key feature
    - Search Algorithm
    - Search Algorithm Optimization
- Evaluation
    - Build Performance
    - Search Performance
    - Scalability
- Benchmark
    - Results
    - Further Benchmarks
- Conclusion

Why Graph Based ANNS

- Approach: Constructing a proximity graph
  - represents similarity relationships between data points
- Graph Transversal
  - find the k closest nodes to the input query

Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

Challenges

- Existing graph-based methods are not optimised for modern hardware like GPUs
- Existing graph-based methods are not optimised for GPUs

Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

CAGRA

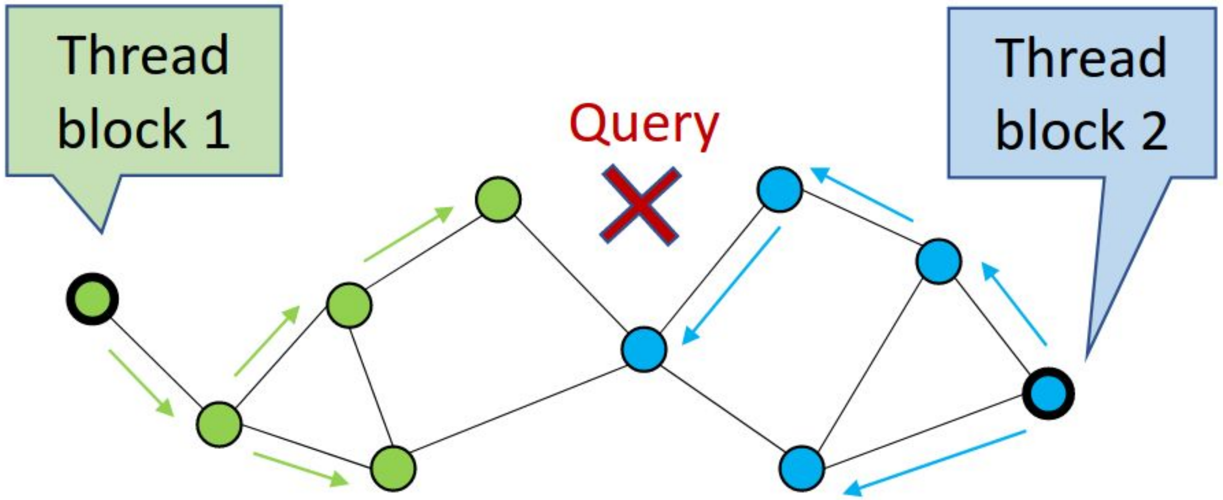
- CAGRA (Cuda Anns GRAph-based)
  - a novel parallel computing hardware-based proximity graph and search algorithm designed for GPUs.
- Addresses the limitations of existing methods by:
  - offering a proximity graph and search implementation optimized for NVIDIA GPUs.

Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

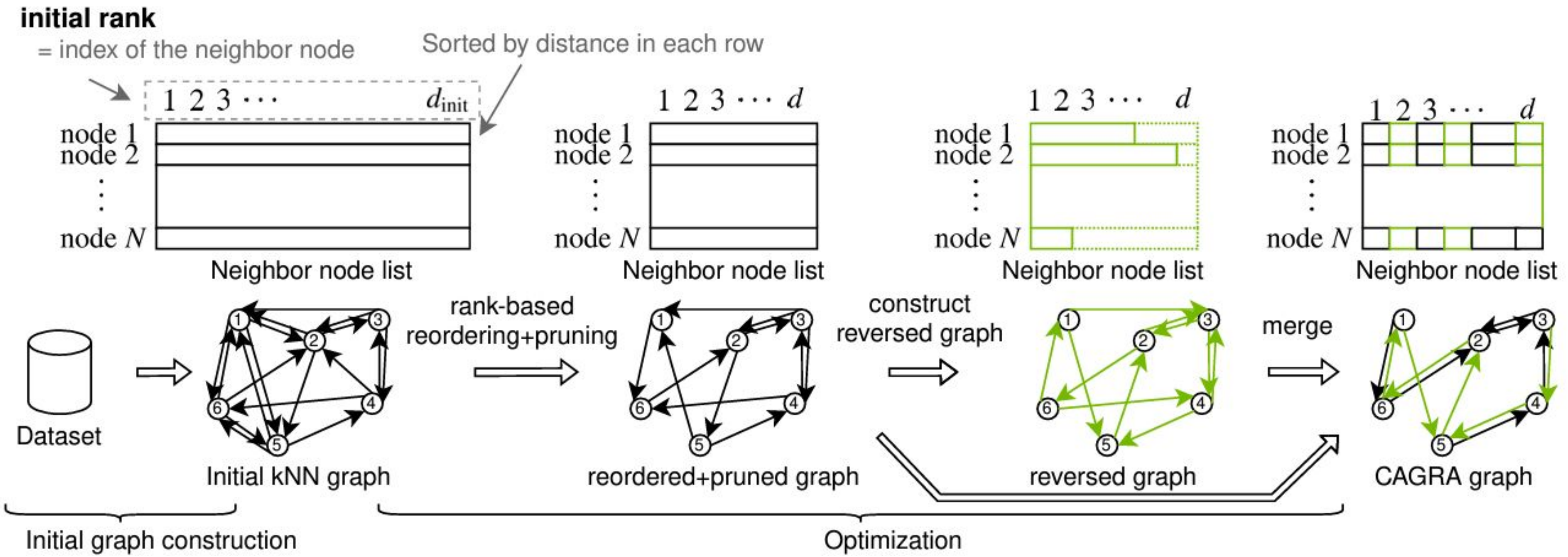
CAGRA: Key Features

- Fixed out-degree (d)
  - Parallelism of GPU is utilized effectively
  - Allows expansion of search space
- Directional
- No hierarchy



CAGRA: Key Features(cont.)

- Two-Stage Construction
  - Building an initial k-NN graph with NN-descent
  - Optimizing the graph through edge reordering and reverse edge

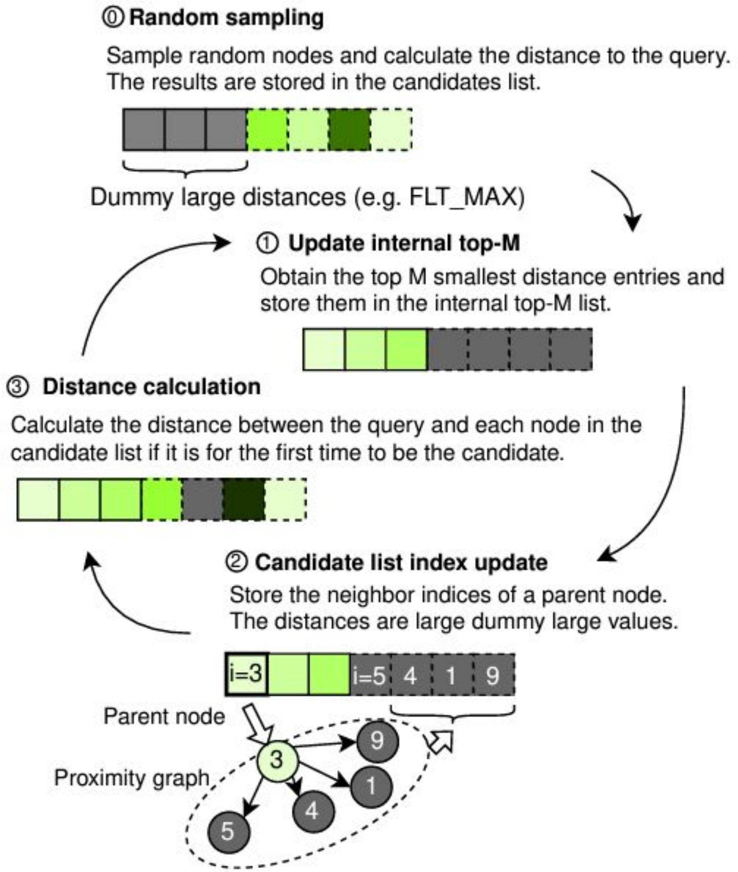


Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

CAGRA: Search Algorithm

- Key idea
  - Traverse a highly optimized proximity graph in parallel
  - Use multiple threads to explore different parts of the graph simultaneously.



Discussion topics

- Introduction
- Why Graph Based ANNS
    - Challenges
- CAGRA
    - Key feature
    - Search Algorithm
    - Search Algorithm Optimization
- Evaluation
    - Build Performance
    - Search Performance
    - Scalability
- Benchmark
    - Results
    - Further Benchmarks
- Conclusion

CAGRA: Search Algorithm Optimization

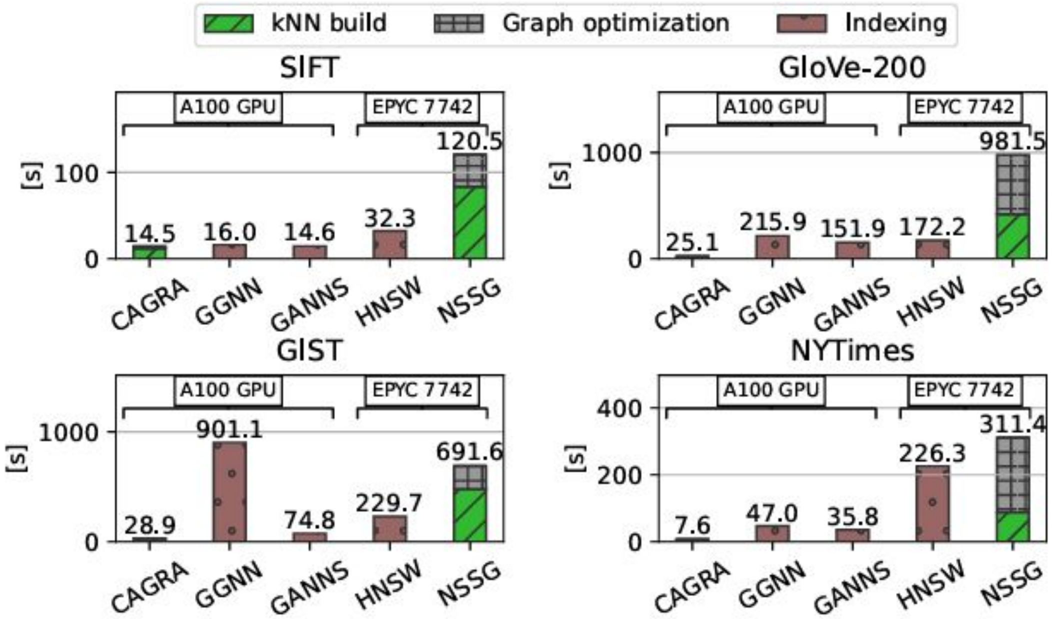
- Warp Splitting
  - Splits each warp into smaller teams of threads
- Hash Table for Visited Nodes
  - Ensuring that distances are only calculated once per node
- Forgettable Hash Table Management
  - Reduces memory usage
- Top-M Calculation Optimization
  - Small buffer size: warp-level bitonic sort
  - Large buffer size: radix sort using shared memory
- Multi-CTA Mode
  - Multiple CTAs collaboratively process a single query

Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

Evaluation: Build Performance

- Speedup Over CPU-Based Methods:
  - CAGRA is 2.2–27× faster than HNSW
- Speedup Over GPU-Based Methods:
  - CAGRA outperforms GGNN by 1.1–31× and GANNS by 1.0–6.1×
- Vs NSSG:
  - Faster knn build time and Graph Optimization



Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

Evaluation: Search Performance

- Throughput Advantage:
  - 33–77× higher throughput compared to HNSW (CPU-based) and 3.8–8.8× higher throughput compared to GPU-based methods (GGNN, GANNS)
- Recall vs Throughput:
  - Maintains a balance between high recall and high throughput

Dataset	Dimension	Batch Size	Recall (%)	Throughput (QPS) – CAGRA	Throughput (QPS) – HNSW	Speedup
SIFT-1M	128	10,000	95	1.5M	19.4K	77×
GIST-1M	960	10,000	90	148K	4.4K	33×
GloVe-200	200	10,000	95	355K	45K	8×
NYTimes	256	10,000	90	685K	61K	11×

Evaluation: Search Performance(cont.)

- Throughput Advantage at single query:
  - 3.4–53× faster performance than HNSW (CPU) at 95% recall
- Multi-CTA mode for small batches:
  - Single query is processed by multiple thread blocks (CTAs) on the GPU

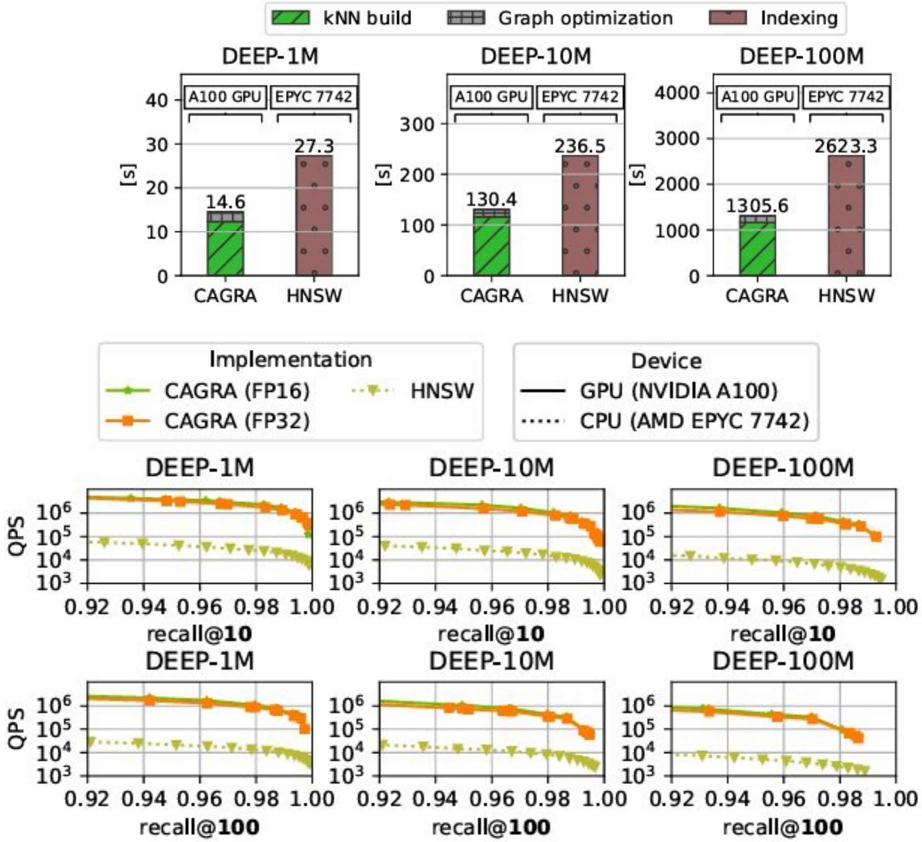
Dataset	Dimension	Recall (%)	Throughput (QPS) – CAGRA	Throughput (QPS) – HNSW	Speedup
SIFT-1M	128	95	3.5K	66	53×
GIST-1M	960	90	142	8	18×
GloVe-200	200	95	418	39	11×
NYTimes	256	90	633	83	7.6×

Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

Evaluation: Scalability

- Graph construction time and search performance scale linearly with dataset size
- CAGRA maintains high throughput and recall



Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

Benchmark

- cuVS-bench provides a tool for us to perform benchmark on different algorithms (notably CAGRA)
- A benchmark was ran on these two setups on the same system:

CPU

- Database Used: Milvus (HNSW)
- Dataset: wiki-all-1m
- Entries: 1M
- Dimension: 768-dim
- Page Size(k): 10

GPU

- Database Used: rapids-cuVS (CAGRA)
- Dataset: wiki-all-1m
- Entries: 1M
- Dimension: 768-dim
- Page Size(k): 10
- Graph degree: 64

Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

Results

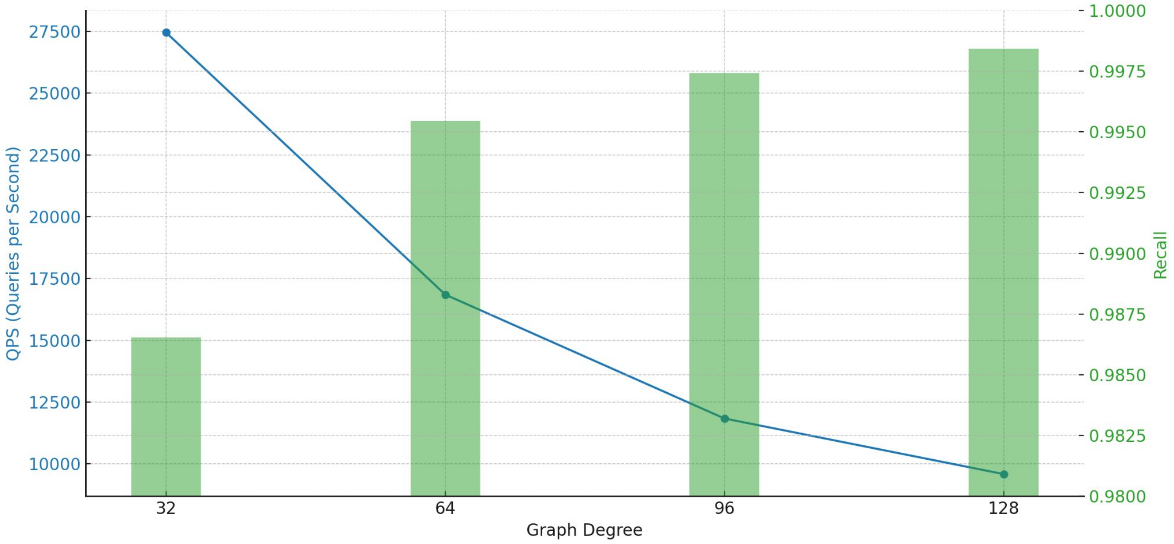
	CPU(Milvus-HNSW)	GPU(cuVS-CAGRA)
Build Time	00:18:9.50	00:00:12.37
QPS	249	16848

Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

Further Benchmarks

- Another benchmark is performed to see:
  - The effect of graph degree towards recall and QPS
- Takeaway:
  - Higher graph degrees improve recall but reduces QPS



Discussion topics

- Introduction
- Why Graph Based ANNS
  - Challenges
- CAGRA
  - Key feature
  - Search Algorithm
  - Search Algorithm Optimization
- Evaluation
  - Build Performance
  - Search Performance
  - Scalability
- Benchmark
  - Results
  - Further Benchmarks
- Conclusion

Conclusion

- This paper presents a new alternative to existing vector database algorithms. Heavily focusing on GPU usage and optimization.
- Future Work:
  - Multi-GPU Environments
  - Memory Efficiency Improvements
  - Broadening Use Cases
- Personal takeaways:
  - CAGRA sets a new benchmark for GPU applications for VDBMS.
  - Possible integration with other databases for persistent storage (as cuVS does not support persistence)