Tutorial on Approximate Nearest Neighbor Search (ANNS) – Techniques and Open Problems

Magdalen Dobson
Carnegie Mellon University
mrdobson@andrew.cmu.edu

Laxman Dhulipala
University of Maryland
laxman@umd.edu

Zheqi Shen
UC Riverside
zshen055@ucr.edu

Harsha Vardhan Simhadri
Microsoft Research
harshasi@microsoft.com

1 ANNS AND RECENT DEVELOPMENTS

Approximate nearest neighbor search (ANNS) is a classical algorithmic problem that is increasingly relevant in practice today across a variety of AI application domains. For AI-first applications, ANNS is the key index that connects neural networks for search [27], recommendation [6] and content generation/summarization [10, 18, 19, 25, 26] with relevant items in the knowledge store. This linking process helps better ground the output of the models and generates higher quality results.

While ANNS is a well-studied problem, classical algorithms like LSH [2] and its derivatives fail to meet the requirements of this new generation of applications. Some of the shortcomings include inadequate query performance for large indices of high dimensional vectors ($10^7$ to $10^{12}$ vectors with dimensions in the range of $10^2$ – $10^5$) and memory-inefficiency and difficulty scaling in the external-memory setting. Perhaps most importantly, existing ANNS approaches are difficult to adapt to new and natural feature requirements motivated by emerging applications. Examples include (a) designing streaming indices, (b) designing ANN indices that can support a combination of hard matches and nearest neighbor search [34], and (c) designing indices that can quickly answer filtered queries.

Addressing these new requirements has been a productive area, with numerous research papers [4, 15, 20, 22–24, 31, 32] and open search packages [11, 12, 22, 28] aiming to push the research and development frontier. There are also vector-search-as-a-service companies (e.g. Milvus, Pinecone, Qdrant, Vespa, Weaviate, Zilliz) that package this research for commercial use, not to mention extensively customized in-house solutions at larger companies where several products stand on ANNS.

2 SCOPE OF THE TUTORIAL

In this 1.5 hour tutorial, we will survey:

- New systems and feature requirements that the next generation of ANNS indices must support to be usable in large-scale deployments.
- Algorithmic techniques developed for ANNS in the last 10 years, and their relative merits and applicability to different use cases. We will focus on graph, and clustering based methods [24, 32] as well as quantization techniques [16, 21].
- Relevant datasets [30], software packages [22, 28], scripts and benchmarking tools [3, 30] for getting started.

An overview of open research problems in generalizing the capabilities of ANN indices and possible directions. For example:

- Complex predicates [34]
- Building updateable vector databases with important properties like crash-recovery, serializability and checkpointing
- Designing massive distributed multi-node indices
- Developing better theoretical analysis of empirical algorithms
- Adapting the index as the model parameters that generate the embeddings change, e.g., when ANNS is used inside model training
- Adapting to different query distributions arising, say, a different modality (e.g., text vs image) or from the far tail of the distribution

3 GOALS AND AUDIENCE

Many SPAA attendees have experience in areas relevant to ANNS, such as processing large graphs in parallel and concurrent settings, parallelizing existing ANNS algorithms, and in state-of-the-art data structures for vector databases. The goal of this tutorial is to help such researchers learn about this problem and understand how their expertise relates to open problems in ANNS.

Another goal of the tutorial is to make it easier for new researchers to get started and acquainted with this field. To this end, our tutorial provide a fully-reproducible demonstration of running ANNS on several interesting datasets, showcasing both the power and flexibility of current ANNS systems, as well as their limitations in certain more challenging scenarios, e.g., under dynamic updates or historical queries, in restricted-memory settings, and under filtering, among many others.

Tutorial attendees will run ANNS algorithms using the Problem Based Benchmark Suite (PBBS) [1]. These ANNS implementations have been developed using the ParlayLib toolkit [7] and have been highly optimized to yield performance competitive with the original published ANNS. Furthermore, the implementations are provided under the uniform test framework provided in PBBS with easy-to-use interfaces. We will also introduce other existing benchmarking tools for ANNS, such as the ANN-Benchmarks framework [5]. Although this tutorial will focus on relatively small-scale datasets to simplify the problem settings and let the audience focus on the key techniques used in the state-of-the-art ANNS algorithm, the ideas are extended to the billion scale in both the PBBS and the Big ANN Benchmarks framework [29].

4 WHY SPAA?

Approximate nearest neighbor search is a highly parallel problem with interesting algorithmic techniques at its core. The authors of this tutorial have used numerous ideas originating in the SPAA community in ANNS; examples include fundamental parallelism...
research [7, 8, 17], randomized incremental algorithms [9], and persistent and/or purely functional graph frameworks [13, 14, 33].

Despite its highly parallel nature, there is a lack of researchers whose primary expertise is in parallelism and concurrency working on this problem. One of our main goals in this tutorial, therefore, is to expose the SPAA community to recent developments in ANNS, with a focus on algorithmic techniques used across multiple ANNS systems and open-problems pertaining to these systems. For example, we will pose open questions surrounding finer-grained (potentially input-dependent) analyses of the work, accuracy, and parallelism in widely-deployed ANNS systems.

Furthermore, most of the work on practical ANNS has not provided a theoretical understanding explaining the empirical success of recent ANNS systems. We will also briefly survey the existing theoretical work on ANNS from SPAA and other theory conferences to highlight the current gap between theory and practice.

REFERENCES


