Technical Perspective on 'OmniSketch: Streaming Data Analytics with Arbitrary Predicates'

Graham Cormode
University of Warwick, UK
G.Cormode@warwick.ac.uk

Many applications in data management can benefit from a "data sketch": a compact data structure that captures the important features of a much larger object. Over the years, many different sketches have been proposed for finding a variety of foundational statistics about an evolving dataset: the number of distinct elements, the median element, the frequently occurring elements (heavy hitters), and several other metrics and norms. These have been used by the data management community for a number of purposes: for efficient measurements when the full data is too large to retain; to guide query planning; and to support approximate query answering. For a longer discussion of the history of sketches and their changing uses, see [1].

One key limitation in the use of sketches is that they tend to be designed for a single purpose. A certain sketch may give accurate answers for one specific query (possibly with a query parameter), but does not yield any information about other types of query. For instance, a sketch that approximates the number of distinct elements does not provide any insight into what is the median of that input distribution. This may limit the applicability of sketching technology, since we need to design and build out a collection of sketches in readiness to answer a fixed set of pre-determined queries. It would be much preferable to provide a single summary that can handle a wider range of future queries, much as a traditional DBMS supports answering any relational query over stored data.

The following "OmniSketch" paper, first appearing in PVLDB in 2023, makes a substantial step in this direction. It provides a single sketch that can accurately answer any counting query over multidimensional data. Such counting queries are at the foundation of any kind of data exploration and analysis, and are needed by modern applications such as populating dashboards and instantiating baseline machine learning models. A naive approach to this task would be to pre-emptively compute all such queries in advance: the number of such possible queries grows exponentially with the dimension, and so is unfeasible for even moderate cardinality attributes in modest dimensionalities.

The OmniSketch approach is to build on existing sketching techniques by combining them in novel ways. Specifically, it uses (frequency) hashing and (min-wise) sampling. In traditional hash tables, hashing is used to record all elements exactly, with fast lookups and collision handling. In sketching applications, hash collisions are treated as the norm, not the exception: all elements mapping to a particular cell are combined there in some way. In traditional statistical applications, samples pick elements from the input at random according to a chosen probability distribution. With sketching, it is more common to perform sampling via "permanent random numbers": ensuring that sampling decisions remain consistent across time by basing them on randomly chosen but fixed functions.

OmniSketch combines these two methods, by using hashing on each dimension in turn to map each element to one of a number of cells. Within each cell, the min-wise hashing retains a fixed-sized sample of the elements mapped there, which can be updated as more data points are observed. Range queries are answered by inspecting all the cells across the sketch structure where matching data points would have been mapped, and extracting information about the prevalence of such data items from what has been retained by the sampling. The analysis of this randomized data structure provides bounds on the tightness of the approximation, as a function of the parameters that determine the size of the sketch. The experimental study demonstrates that this gives answers that are accurate with a small error bound, and much more scalable than prior approaches that scale exponentially with the dimension.

The natural extensions for the data management community to consider are to design sketches that tackle different classes of query. The class of arbitrary complex SQL aggregation queries is likely too broad to yield useful solutions, since the lower bounds on sketch sizes inherited from communication complexity theory mean that there is no prospect for strong approximation guarantees on queries that involve selective joins, for example [2]. Driven by current applications, it may be more productive to work on query classes that unlock important workloads, such as more general data analytics (with aggregations such as sum, mean and median in addition to count), and materializing the statistics for building predictive models (such as conditional probabilities and Bayesian models).

1. REFERENCES

[1] G. Cormode. Gems of pods: Applications of sketching and pathways to impact. In *ACM Principles of Database Systems (PODS)*. ACM, 2023.

[2] A. Rao and A. Yehudayoff. *Communication Complexity and Applications*. Cambridge University Press, 2020.