The Case for Learned Index Structures

Tim Kraska, et al.

This paper introduces Learned Index Structures which use neural networks and justifies performance improvements with experiments over traditional index structures like binary trees, hash index, and bloom filters. The paper introduces these index structures and explains how they are actually models that are not fully taking advantage of patterns in data to maximize their performance, motivating the incorporation of learned index structures. The authors propose using a regression model to replace binary trees and learn the cumulative distribution function of the data in order to make smarter, more efficient mappings. They also propose a more robust method in the Recursive Model Index which aims to learn a hierarchy of models where the top-level makes the initial prediction of a key’s position and the lower-level models refine that prediction. The authors then ran experiments to justify this approach, concluding that learned index structures outperform binary trees by three times, and require at least one order of magnitude less memory. However, these improvements are not seen in write-heavy workloads. Additionally, there is a significant training overhead which would need to be researched further as an extension. The authors make the assumption that data follows an underlying distribution that can be effectively learned by an ML model, however, in some cases data does not follow a predictable pattern, potentially leading to worse performance over traditional index structures. In workloads where new data is being inserted often, for example, in a database for banking transactions, the model might need to be retrained too often to be practical.

Algorithms with Predictions

Michael Mitzenmacher, Sergei Vassilvitskii

This paper introduces a framework for algorithms which use machine learning predictions based on the following two principles: algorithms should perform optimally if predictions are accurate and they must not perform worse than traditional algorithms if predictions are inaccurate. The paper develops this with case studies including binary search, CM sketches, bloom filters, and caches in order to showcase how ML can be incorporated for these existing algorithms to improve their runtime when predictions are accurate but not degrading to worse performance than the existing algorithm when predictions are inaccurate. The authors make the assumption that the error in predictions is bounded, allowing them to guarantee worst-case performance. The key difference between this paper and the experimental one is that this paper is theoretically focused. Additionally, it targets ensuring worst-case performance is never worse than the traditional index structure the learned index structure is attempting to enhance. As a result, it can be interpreted as a theoretical framework for future research in the space of algorithms with embedded machine learning for enhanced performance as well as an extension of the results found in the experiments conducted in the experimental paper by addressing one of the key limitations in that work which was ensuring worst-case performance is never worse than the traditional algorithm. Therefore, there is a strong case for an extension to this theoretical paper in the form of an experiment to ensure that these bounds in performance and error do not mitigate the performance of these learned index structures in practice.