DOT-K: Distributed Online Top-K Elements Algorithm using Extreme Value Statistics

Nick Carey, Tamás Budavári, Yanif Ahmad, Alexander Szalay

Johns Hopkins Whiting School of Engineering

Department of Computer Science

ABSTRACT

DOT-K is a one pass, communications-efficient technique useful for both estimating upper order quantiles and selecting the largest 'k' elements in the context of an extremely large (peta-scale) and highly distributed data environment. Our novel approach draws its foundations from Extreme Value Statistics (EVS) to reason about the statistical relationships between the tail distributions of dataset partitions while minimizing communications costs.

EXTREME VALUE STATISTICS

- Characterizes the tail distributions, or extreme values, of random variables
- Traditionally concerned with modeling frequency of extreme environmental phenomena

PICKANDS, BALKEMA, DE HAAN THEOREM

- The distribution of threshold exceedances of a sequence of I.I.D. random variables with a common continuous underlying distribution function is approximated by the Generalized Pareto Distribution (GPD) and that the approximation converges as the tail threshold rises

$$p(x|\xi,\sigma,\mu) = \frac{1}{\sigma}\left[1+\xi\cdot\left(\frac{x-\mu}{\sigma}\right)\right]^{-\frac{1}{\xi}}$$

Equation 1. Generalized Pareto Distribution probability density function including parameters **ε** (shape) **σ** (scale) and **μ** (location, or threshold)

- For a large class of common data distributions, the 'k' largest values in a dataset may be approximated by a GPD provided the k'th order statistic is an appropriately high threshold
- Bias-variance trade-off: lower threshold results in theoretically worse GPD approximation while higher threshold limits amount of available threshold exceedances leading to greater parameter estimation uncertainty

M-OBSERVATION RETURN LEVEL

- For a given GPD, one may calculate the threshold x_m that is exceeded on average once every m observations

$$\zeta_\mu\left[1+\xi\cdot\left(\frac{x_m-\mu}{\sigma}\right)\right]^{-\frac{1}{\xi}} = \frac{1}{m}$$

Equation 2. Coles' M-Observation Return Level equation. ζ_υ is a constant estimated by the number of observations exceeding μ divided by total observations

$$\sum_{i=1}^{p} n_i\,\zeta_{\mu_i}\left[1+\xi_i\cdot\left(\frac{x_m-\mu_i}{\sigma_i}\right)\right]^{-\frac{1}{\xi_i}} = k$$

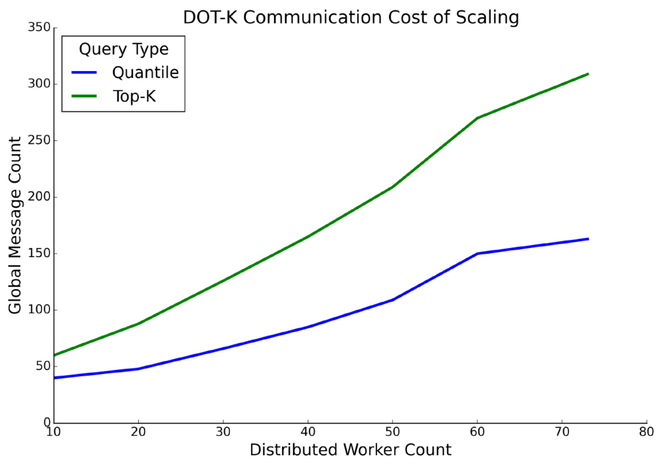
Equation 3. Our modification of Coles' M-Observation Return Level. Numerically solving for x_m, this equation estimates each distributed data partition's expected contribution to the top-k query result.

DOT-K ALGORITHM

- Assuming a dataset row-partitioned across many nodes, our goal is to estimate the k'th largest element and subsequently retrieve all elements greater than the estimate.
  1. Each node collects its largest 'k' local values and using a parameter estimator such as MLE, calculates the GPD parameters that best fit the local data
  2. By relating the GPD parameters collected from each node, the query issuer estimates the global k'th largest element by numerically solving Equation 3
  3. The k'th order statistic estimate is communicated to the distributed nodes and the exceedances are relayed back to the query issuer

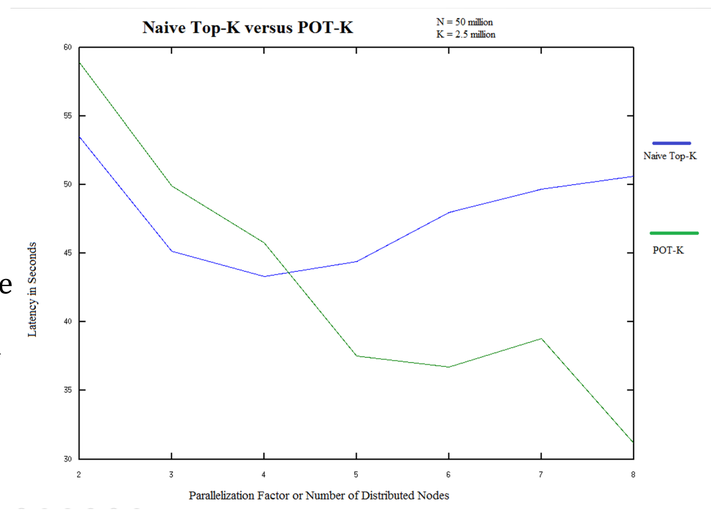
EXPERIMENTS

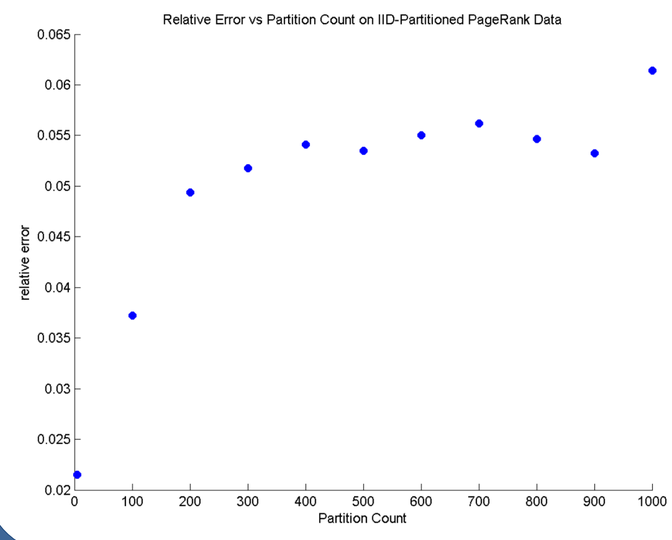
- DOT-K was executed within the Apache Storm stream processing framework on an Amazon AWS cluster to test communication costs and query latency
- A pseudo-distributed Matlab implementation was used to evaluate DOT-K query accuracy at larger scales



- With 'P' as parallelization factor, an ideal implementation communicates 4*P total messages between workers with 6*P + ~k total real values transmitted
- Slightly more than 4*P total messages were transmitted in our empirical evaluation due to Storm overhead

- Query result latency quickly overtakes a naïve top-k implementation as parallelization rises.
- DOT-K retrieves only the elements estimated to be in the final query result from the distributed workers, and at very large scale this property reduces the query result sort cost





- Relative error is given as: $\delta k = \left|\frac{k-\hat{k}}{k}\right|$
- Berkeley Big Data Benchmark PageRank dataset was used to test DOT-K accuracy on a highly partitioned real dataset
- Each data point is the result of the average of ten trials with the dataset randomly shuffled between nodes for each trial