

FAST: A New Sampling-Based Algorithm for Discovering Association Rules

Bin Chen
Exelixis, Inc.
bchen@ece.northwestern.edu

Peter J. Haas
IBM Almaden Research Ctr.
peterh@almaden.ibm.com

Peter Scheuermann
Dept. of ECE, Northwestern U.
peters@ece.northwestern.edu

An obvious way to speed up the process of discovering association rules is to generate them from a small sample of the database rather than the database itself. The primary challenge in developing sampling-based algorithms stems from the fact that the support of an itemset in a sample almost always deviates from the support in the entire database.

We present FAST (Finding Associations from Sampled Transactions), a refined sampling-based mining algorithm that is distinguished from prior algorithms by its novel two-phase approach to sample collection. In Phase I a large sample is collected to quickly and accurately estimate the support of each item in the database. In Phase II, a small final sample is obtained by excluding “outlier” transactions in such a manner that the support of each item in the final sample is as close as possible to the estimated support of the item in the entire database. We propose two approaches to obtaining the final sample in Phase II: “trimming” and “growing.” The “trimming” procedure starts from the large initial sample and removes “outlier” transactions until a specified stopping criterion is satisfied. In contrast, the growing procedure selects “representative” transactions from the initial sample and adds them to an initially empty data set. The simplest versions of FAST use a fixed-size stopping criterion: they terminate when the sample size reaches a user-specified value n . The basic FAST-TRIM algorithm is as follows:

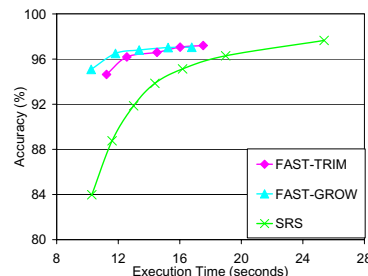
```

obtain a simple random sample  $S$  from  $D$ ;
compute  $f(A; S)$  for each item  $A$  in  $S$ ;
set  $S_0 := S$ ;
while ( $|S_0| > n$ ) {
  divide  $S_0$  into disjoint groups of size  $\min(k, |S_0|)$ ;
  for each group  $G$  {
    compute  $f(A; S_0)$  for each item  $A$  in  $S_0$ ;
    if  $|S_0| > n$ , set  $S_0 = S_0 - \{t^*\}$ , where
       $\text{Dist}(S_0 - \{t^*\}, S) = \min_{t \in G} \text{Dist}(S_0 - \{t\}, S)$ ;
  }
}
run a standard association-rule algorithm against  $S_0$  to obtain
the final set of association rules;

```

Here $f(A; X)$ is the frequency of item A in data set X

and $\text{Dist}(S, T)$ measures the distance between the item supports in transaction sets S and T . One of the distance functions that turned out to be most useful is: $\text{Dist}(S_0, S) = \sum_A (f(A; S_0) - f(A; S))^2$. By choosing a value of k between 1 and $|S|$, the user can strike a balance between ineffective but cheap “oblivious” trimming and very effective but very expensive “pure greedy” trimming. FAST-GROW works in a similar manner.



The above figure shows some experimental results for FAST (with $k = 10$) and SRS (Simple Random Sampling). *Apriori* is called to process the final sample of each algorithm. Observe that, if required to finish the mining task within ten seconds, FAST-TRIM and FAST-GROW achieve an accuracy of approximately 95%, compared to an accuracy of about 84% for SRS. The final sample produced is 5% of the size of the original database. Looked at in another way, FAST achieves an accuracy comparable to SRS in about 35% less time. We also compared the performance of FAST with Toivonen’s algorithm (*Proc. VLDB*, 1996). It takes Toivonen’s algorithm about 100 seconds to finish the mining task using the same 5% sample size as FAST.

Based on our experience, FAST is a promising algorithm in situations where good approximate answers suffice and interactive, real-time response is essential. In addition, users can explicitly trade off accuracy and speed. We emphasize that the sample created by FAST can be subsequently processed by *any* existing (non-sampling-based) association-rule mining algorithm, so that FAST complements, rather than replaces, current algorithms such as *DepthMiner*, *Max-Miner*, *DIC*, or *FP-tree*. Details of FAST and its extensions are in a Technical Report available from Northwestern University.