히스토그램을 이용한 근사적 집단 연산과 효과적인 오차 추정

안성준 °, 배진욱, 심마로, 이석호
{junny, oblody, maro}@db.snu.ac.kr, shlee@comp.snu.ac.kr
서울대학교 컴퓨터공학과

Approximate Aggregation and Effective Error Estimation using Histogram

Seongjoon Ahn °, Sukho Lee
Dept. of Computer Engineering, Seoul National University

요 약

히스토그램은 데이타베이스 질의 최적기가 사용하는 통계정보 중의 하나이다. 최근에는 데이타베이스의 크기가 기하급수적으로 커짐에 따라, 데이타의 전체적인 성향을 빠르게 파악할 수 있는 방법의 하나로 히스토그램을 활용하는 방안이 고려되고 있다.

그를 위해서, 히스토그램에서 얻어진 근사값의 오차를 추정할 수 있는 방법이 요구되었다. 기존의 기법에서는 히스토그램의 각 버켓에 실제 빈도와 평균 빈도의 최대차를 추가하고, 이 값을 이용하여 오차 추정을 하였다. 그러나, 이 값이 히스토그램 버켓의 전체적인 데이타 분포를 잘 반영하지 못하기 때문에 실제 오차에 근접한 오차 추정을 할 수가 없는 단점이 있었다.

본 논문에서는 이를 극복하기 위해, 히스토그램에 데이타의 분포를 잘 반영하는 정보 즉, 평균값, COUNT/SUM 연산에 대한 최대 오차를 추가하였다. 이 정보들을 이용하여 실제 오차에 보다 근접한 오차 추정을 할 수 있었으며, 부가적으로 SUM/AVG 연산에 대한 보다 정확한 근사값을 얻을 수 있었다.

1. 서론

히스토그램은 데이타베이스 시스템의 질의 최적기에서 특정 애트리뷰트 값의 분포를 얻기 위한 통계 정보이다. 최근에는, 방대한 데이타의 처리가 요구되는 데이타웨어하우스/OLAP 시스템에서 데이타의 전체적인 성향을 빠른 시간에 알아보기 위해, 히스토그램을 이용하여 집단 함수 질의에 대한 근사적인 결과를 얻는 기법이 고려되고 있다.

히스토그램은 다른 자료 구조와 비교해 볼 때, 적은 저장 공간을 사용하며 시스템에 주는 부하가 거의 없다는 장점이 있다. 반면, 많은 양의 데이타를 요약한 형태로 되어 있기 때문에 히스토그램에서 얻어지는 결과는 오차를 가지게 되는데, 이 오차의 크기와 오차에 대한 정확한 추정이 결과값의 유용성에 큰 영향을 미치게 된다. 따라서, 오차를 최소로 하면서 동시에 가능한 한 실제 오차에 근접한 오차 추정을 할 수 있도록 히스토그램을 구성하는 것이 중요하다.

히스토그램에 대한 연구는 질의 최적화를 위한 통계 정보에 관한 연구에서 시작되었다. Equi-width 히스토그램을 이용한 질의 최적화 기법[1]이 가장 먼저 시도되었으며, 히스토그램의 구성 요소(버켓 분할 방식, 버켓 정렬 방식 등)에 대해서 히스토그램의 정확도를 향상 높일 수 있는 기법[2]이 제안되었다.

기존 히스토그램이 여러 애트리뷰트가 연관된 데이타 값의 분포(Joint Distribution)를 정확히 예측하기 힘든 단점을 극복하기 위해 다차원 히스토그램[3]이 제안되었다. 여기서는 문제를 간단히 하기 위해서 1차원 히스토그램만을 사용하였다.

그리고, 히스토그램을 근사적 집단 연산에 응용하기 위하여 근사값의 오차를 추정하는 기법[4]이 제안되었다. 여기서는 각 히스토그램 버켓의 실제 빈도와 평균 빈도의 최대차를 이용하여, 주어진 질의에 대한 오차 상한을 추정하는 기법을 제안하였다. 그러나, 이 기법은 데이타의 전체적인 분포를 잘 반영하지 못하기 때문에, 실제 오차에 근접한 오차 추정이 어렵다는 단점이 있으며, COUNT이외의 SUM과 AVG 연산에 대한 정확한 결과 예측과 오차 추정이 어렵다.

이 논문에서는 히스토그램에 데이타의 전체적인 분포를 반영하는 정보를 추가함으로써 다양한 집단 연산 질의 결과를 정확하게 예측하고, 효과적으로 그 오차를 추정할 수 있는 기법을 제안하고자 한다.

2절에서는 기존 기법과 그 문제점에 대해 정리하고, 3절에서는 히스토그램의 오차를 실제 오차에 근접하게 추정할 수 있는 기법을 제안하였다. 4절에서는 실험을 통한 성능 평가 내용과 결과 분석을 정리하였고, 5절에서 결론을 지었다.

2. 히스토그램을 이용한 근사값에 대한 오차 추정

이 절에서는 기존의 히스토그램 오차 추정 기법[4]과 그 문제점에 대해 설명하겠다.

2.1. 히스토그램의 구성과 근사적 집단 연산

히스토그램의 각 버켓은, 애트리뷰트 값 중 최소값($v_{i,\min}$), 최대값($v_{i,\max}$), 평균 빈도($f_{i,avg}$), 서로 다른 값의 개수(n_i)로 구성된다[2]. ($i=1,\cdots,N$, N은 버켓의 개수) 히스토그램을 이용하여 아래 질의에 대한 근사값을 얻는 기법에 대해서 생각해보자.

SELECT COUNT(ATTR) FROM R WHERE ATTR <= m

그림 1을 보면, i번째 버켓 왼쪽의 세 버켓은 질의에서 주어진 범위에 완전히 포함되며, i번째 버켓은 일부만이 범위에

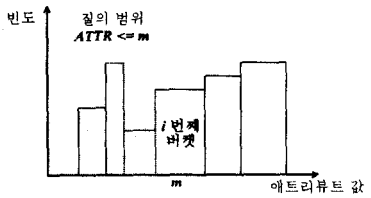

그림 1. 히스토그램을 이용한 범위 질의의 처리

포함됨을 알 수 있다.

범위에 완전히 포함된 버켓(k번째라 하자)의 빈도합은 $\sum_{j=1}^{n_k} f_j$ (f_j는 각 아이템의 빈도)이 된다. $f_{k,avg} \times n_k = \sum_{j=1}^{n_k} f_j$ 이므로, 이 값은 히스토그램 내의 정보로 정확하게 계산할 수 있다. 따라서 이러한 버켓에서는 오차가 발생하지 않는다.

그림 1의 i번째 버켓과 같이, 일부분이 범위에 포함된 버켓에서는, $v_{i,\min}$과 m사이에 있는 아이템의 수와 평균 빈도를 이용하여 COUNT 연산의 결과를 추정하게 되는데, 여기서 오차가 발생하게 된다.

2.2. 기존의 오차 추정 기법과 문제점

[4]에서는 각각의 버켓마다 실제 빈도와 평균 빈도의 최대차 $\Delta f_{i,\max} = \max\{|f_j - f_{i,avg}| \mid j = 1, \ldots, n_i\}$를 추가하고, 이를 이용하여 히스토그램의 오차를 추정하고 있다. 그림 2에서, (a)는 실제 데이터, (b)는 (a)로 구성한 히스토그램 버켓에 해당된다. 이 예에서는 (a)의 오른쪽에서 두번째 데이타의 빈도와 평균 빈도와의 차가 $\Delta f_{i,\max}$에 해당된다. 그림 1의 i번째 버켓과 같이 일부분이 범위에 포함되는 버켓에서 발생할 수 있는 최대 오차는 아래와 같다.

$$\min(m - v_{i,\min} + 1, v_{i,\max} - m + 1) \times \Delta f_{i,\max} \quad [4]$$

이 기법의 문제점은 크게 두 가지로 볼 수 있다.

- 각 버켓에서 평균 빈도와의 최대차만을 고려하고 있으므로, 버켓 안의 빈도의 분포가 비교적 균등하지 않은 경우, $\Delta f_{i,\max}$가 커지게 되고, 효과적인 오차 추정이 어렵다.
- 히스토그램의 주된 용도가 선택도의 예측이기 때문에 SQL의 COUNT 연산은 잘 지원할 수 있지만, SUM, AVG 연산에 대한 보다 정확한 결과값과 효과적인 오차 추정에 대해서는 고려하고 있지 않다. SUM 연산의 예를 보면, v_i, f_i를 버켓 안의 실제 애트리뷰트 값, 빈도라 할 때, 각 버켓의 실제 SUM 연산 결과는 $\sum_{j=1}^{n_i} v_j \times f_j$ 가 되는데, [4]의 히스토그램에서는 $\sum_{j=1}^{n_i} (v_j \times \sum_{k=1}^{n_i} f_k / n_i)$로 추정된다. 따라서 COUNT 연산과는 달리 질의 범위에 포함되는 모든 버켓에서 오차가 발생하게 된다.

3. 효과적인 오차 추정을 위한 히스토그램

여기서는 보다 정확한 오차 추정을 위해서, 각 히스토그램 버켓에 $\Delta f_{i,\max}$ 대신 3개의 새로운 값을 추가하였다.

- $\max\{|\sum_{j=1}^{k}(f_{i,avg} - f_j)| \mid k = 1, \cdots, n_i\} = \sum \Delta f_{i,\max}$
  : 이 값은 각 버켓의, 실제 빈도의 누적 합과 평균 빈도 누적합의 차(그림 2,3의 히스토그램 버켓에서 검은색으로 표시된 부분)의 최대 절대치이다. 그림1의 i번째 버켓과 같이 일부가 범위에 포함되는 버켓에서, COUNT 연산에 대한 최대 오차

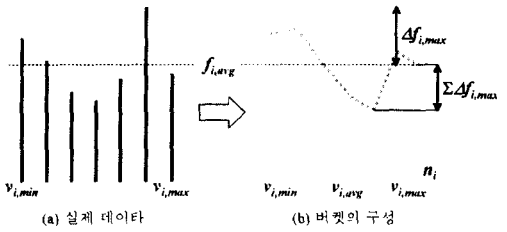

그림 2. 히스토그램 버켓의 구성

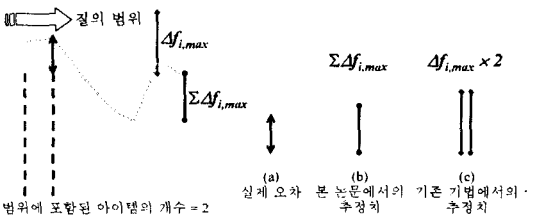

그림 3. 기존 기법과 오차 추정 효과 비교

로 사용된다. 이 값은 $\Delta f_{i,\max}$ 보다 버켓 내의 데이타 분포를 더 반영하고 있다고 볼 수 있다.

- $\sum_{j=1}^{n_i}(v_j \times f_j) / \sum_{j=1}^{n_i} f_j = v_{i,avg}$: 이 값은 각 버켓의 평균값이 되는데, 이 값을 이용하면 SUM, AVG 연산의 범위 질의에 완전히 포함되는 버켓에 대해서 오차없이 결과를 구할 수 있다. ($v_{i,avg} \times f_{i,avg} \times n_i = \sum_{j=1}^{n_i}(v_j \times f_j)$)
- $\max\{|\sum_{j=1}^{k}(v_j \times (f_j - f_{i,avg})| \mid k = 1, \cdots, n_i\} = \sum v\Delta f_{i,\max}$
  : 이 값은 각 버켓의 실제 합과, 평균 빈도와 평균 값의 누적합의 최대차가 된다. 이는 SUM 연산에서, $\sum \Delta f_{i,\max}$과 같은 역할을 한다.

이 값들을 사용한 히스토그램에서의 COUNT, SUM, AVG 연산 질의에 대한 오차 추정은 다음과 같이 할 수 있다.

- COUNT 연산의 경우, 일부분만 범위의 포함되는 버켓(그림 1의 i번째 버켓)의 $\sum \Delta f_{i,\max}$를 오차로 추정한다.
- SUM 연산의 경우, 일부분만 범위에 포함되는 버켓(그림1의 i번째 버켓)의 $\sum v\Delta f_{i,\max}$를 오차로 추정한다.
- AVG 연산의 경우, COUNT와 SUM의 결과와 추정된 오차를 이용하여 결과값 및 오차를 계산할 수 있다.

그림 3은 [4]에서 제안된 기법과 본 논문에서 제안하는 기법의 차이를 도식적으로 보여주고 있다. 그림 3(a),(b),(c)의 막대의 길이가 각각 실제 오차, 이 기법에서 추정한 오차, [4]에서 추정하는 오차에 해당된다. 이 기법에서 추정한 오차가 [4]에서 추정한 오차보다 실제 오차에 근접함을 알 수 있다.

4. 성능 평가

4.1. 실험 환경

성능 평가는 2가지 데이타 집합을 생성하여 수행하였다(그림4, 그림5). 데이타1은 전체 데이타가 비교적 균등한 분포를 가지며, 데이타2는 애트리뷰트 값 사이의 빈도수 차가 큰 특징이 있다. 두 데이타 모두 서로 다른 값의 종류는 2048개이다.

히스토그램은 1차원 히스토그램 중 MaxDiff(V,A)[2]를 사용하였으며, 데이타의 10%를 임의 추출하여 구성하였다. 이 경우에 임의 추출에 의한 오차는 무시할 수 있다[5].

그림 4. 데이타1

그림 5. 데이타2

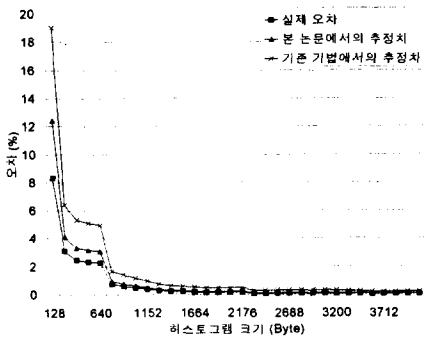

그림 6. 데이타1에 대한 COUNT 질의 결과

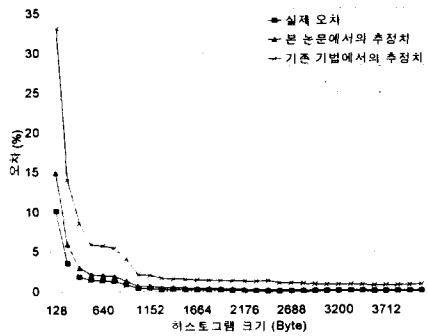

그림 7. 데이타1에 대한 SUM 질의 결과

4.2. 성능 분석

그림6에서 그림9까지는 데이타1, 데이타2에 대해 임의의 범위 질의 1000개에 대해 COUNT, SUM 연산을 적용한 결과를 보여주고 있다. [4]에서는 SUM 연산에 대해 고려하고 있지 않지만, $\Delta f_{i,\max}$를 이용하여 오차를 추정할 수 있는 방법을 사용하였다. 가로축은 히스토그램의 크기(128Byte에서 4096Byte까지), 세로축은 오차에 해당된다.

전체적으로, 이 논문에서 제안하고 있는 기법이 [4]에서 제안한 기존 기법보다 효과적으로 오차를 추정하고 있음을 알 수 있다. 특히 데이타2와 같이 한 버켓 안에 있는 애트리뷰트 값 사이의 빈도차가 큰 경우에, 성능의 차이가 두드러짐을 알 수 있다.

이 기법을 사용하면, 한 버켓의 크기가 커지므로 전체 버켓의 수가 줄어드는 단점이 있다. 하지만, 그에 따른 실제 오차의 차이는 미미한 정도였으며, 오히려 오차의 추정의 정확도를 비교해 볼 때, 본 논문에서 제시하고 있는 기법은 충분히 의미있는 기법이라고 할 수 있다.

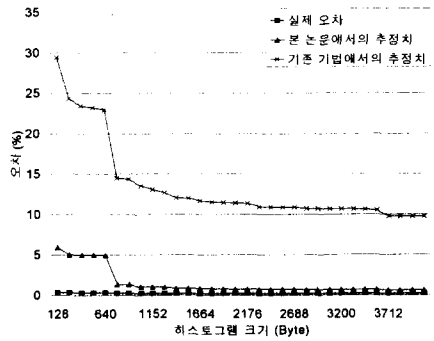

그림 8. 데이타2에 대한 COUNT 질의 결과

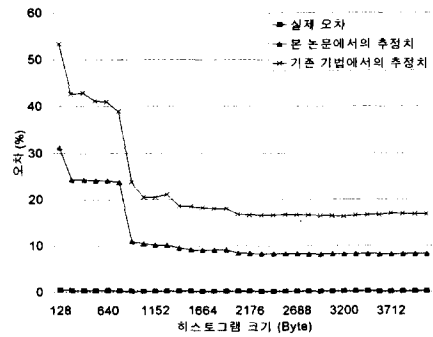

그림 9. 데이타2에 대한 SUM 질의 결과

5. 결론

급속히 늘어나는 데이타의 전체적인 성향을 빠르게 파악하기 위하여 히스토그램을 이용하는 방법이 고려되고 있다.

이 논문에서는 히스토그램에 데이타의 분포를 나타내는 정보를 추가하여, 기존 기법보다 정확한 근사적 집단 연산과 효과적인 오차 추정을 수행할 수 있는 기법을 제안하였다. 이 기법을 이용함으로써, 질의 최적기에서의 선택도 예측이나 데이타웨어하우스/OLAP에서의 근사적 질의 처리에서, 보다 효과적이고 다양하게 히스토그램을 활용할 수 있을 것이다.

참고 문헌

[1] R.P. Kooi, "*The optimization of queries in relational databases*" PhD thesis, Case Western Reserver University, 1980

[2] Viswanath Poosala, Yannis E. Ioannidis, Peter J. Haas, Eugene J. Shekita, "*Improved Histograms for Selectivity Estimation of Range Predicates*", ACM SIGMOD Conference 1996

[3] Viswanath Poosala, Yannis E. Ioannidis, "*Selectivity Estimation Without the Attribute Value Independence Assumption*", VLDB, 1997

[4] H.V. Jagadish, Nick Koudas, S. Muthukrishnan, Viswanath Poosala, Ken Sevcik, Torsten Suel, "*Optimal Histograms with Quality Guarantees*", VLDB, 1998

[5] Surajit Chaudhuri, Rajeev Motwani, Vivek Narasayya, "*Random Sampling for Histogram Construction: How much is enough?*", ACM SIGMOD Conference, 1998