머신 러닝 모델 기반 근사 질의 처리 방법에 관한 연구

박춘서, 김성수, 남택용, 이태휘
한국전자통신연구원 스마트데이터연구실
parkcs@etri.re.kr, sungsoo@etri.re.kr, tynam@etri.re.kr, taewhi@etri.re.kr

A Study on Approximation Query Processing Method Based on Machine Learning Models

Choon Seo Park, Sung-Soo Kim, Taek Yong Nam, Taewhi Lee
Smart Data Research Section, Electronics and Telecommunications Research Institute

요 약

최근 데이터의 양이 급격히 증가함에 따라 빅데이터 환경에서 데이터 질의 처리 수행 시 연산 시간이 많이 소요되는 문제점이 발생한다. 이러한 처리 시간을 줄이기 위한 방법으로 근사 질의 처리에 대한 연구의 필요성이 대두되고 있다. 근사 질의 처리 방법은 정확도가 다소 떨어지더라도 빠른 결과를 요구하는 응용 분야에서 매우 유용하게 쓰일 수 있다. 본 논문에서는 사용자가 원하는 결과 정확도와 적시성 등을 지원하기 위한 근사 질의 처리 언어 확장, 실행 계획 생성 및 질의 최적화 기술을 제안하고, 설계 방향 및 특징 등에 대해서 설명한다.

1. 서론

최근 데이터의 양이 급격히 증가하고 복잡해짐에 따라 데이터 질의 처리 수행 과정에서 많은 연산 비용이 소요되어 사용자가 원하는 빠른 시간에 결과를 얻어 오는 데 어려움이 발생한다. 다소 정확성이 떨어지더라도 질의 처리 수행 시간을 줄여서 빠른 결과를 얻을 수 있게 하여 준 실시간성 응용 분야에서 유용하게 사용할 수 있는 근사 질의 처리 방법에 대한 연구의 필요성이 점점 높아지고 있다. 근사 질의 처리 방법은 정확 질의를 수행하기 위한 리소스 중 일부만을 사용하여 근사 질의의 결과를 빠른 시간에 제공할 수 있는 유용한 기술 중의 하나이다. 정확한 질의 결과가 아닌 근사 질의 처리 결과가 허용되고 빠른 결과가 요구되는 탐색이나 시각화와 같은 응용 분야에서 활용도가 점점 높아지고 있다.

정확 질의 처리는 전체 데이터에 대해서 질의 처리를 수행하여 질의 처리 비용이 많이 발생하는 문제점이 발생하는데, 이를 해결 하기 위하여 샘플링, 히스토그램, 웨이블릿 등을 이용한 요약 기법 기반으로 근사 질의 처리를 수행 하는 연구가 진행되었다[1, 2]. 요약 기법 기반의 근사 질의는 전체 데이터를 대상으로 질의를 수행하는 것이 아니라 전체 데이터에서 일부 데이터를 샘플링 하는 등의 축소 과정을 수행하여 요약한 정보를 대상으로 질의 처리를 수행하여 질의 처리에 대한 데이터 크기를 줄임으로써 적은 연산 비용으로 보다 빠른 결과를 얻는데 목적이 있다.

또한 데이터를 기반으로 생성한 ML 모델을 이용해 데이터에 직접 접근하지 않고 근사 질의를 처리하려는 연구가 진행되었다. 사전에 정확 질의로 데이터를 학습하여 ML 모델을 생성하는 질의 중심 모델(query-driven model)[3]과 데이터로부터 ML 모델을 학습하는 데이터 중심 모델(data-driven model)[4]의 근사 질의 방식 등의 다양한 연구가 최근 진행되었다.

본 논문에서는 근사 질의 처리시 사용자가 원하는 질의 결과의 예상 수행 시간과 오차 허용도 등을 표현하기 위하여, 질의 언어를 확장하고 실행 계획 생성 및 최적화 하는 과정을 설명한다.

2. 근사 질의 변환 및 최적화 기술의 설계 이슈

근사 질의 처리를 수행함에 있어서 사용자가 원하는 정확도와 적시성을 질의 처리 엔진에 잘 전달해야 한다. 기존 질의 언어문에는 이러한 요구사항을 표현하는 수단이 부족하므로 기존 질의 언어를 확장 제공해야 한다. 이때 정확도와 적시성의 표현하는 방식이 기존 SQL 표현과 비슷하게 확장하여 SQL 문법에 익숙한 사용자가 쉽게 이해하고 활용할 수 있도록 한다.

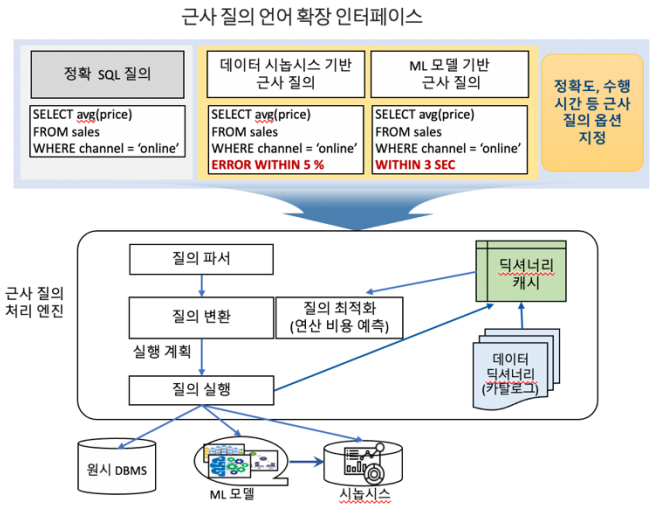
이렇게 확장 제공하는 근사 질의 언어문에 대해서 질의 파서 기능을 수행하고, 근사 질의에 대한 실행 계획을 생성해야 한다. 생성된 여러 개의 실행 계획 중 가장 최적의 실행 계획을 선정하여 해당 실행 계획을 수행하고, 근사 질의 결과를 사용자에게 전달하는 기능을 제공해야 한다. 이때 여러 실행 계획 중

사용자가 요구하는 정확도와 적시성을 만족하면서 가장 좋은 성능을 나타낼 수 있는 최적의 실행 계획을 선정하는 것이 무엇보다 중요한 과제이다.

근사 질의 처리 수행시 ML 모델 활용하는 방법으로는 결과 추론형 모델과 시놉시스 생성형 모델 방식이 고려되며 각 모델의 장단점을 파악하고, 질의 유형과 실행 환경 등에 따라 가장 적절한 모델을 선정하여 근사 질의를 수행하는 방법을 고민해야 한다.

3. 근사 질의 변환 및 최적화 기술

본 절에서는 근사 질의 처리를 지원하기 위한 근사 질의 언어 확장, 실행 계획 생성 및 최적화 방법에 대해서 설명한다.



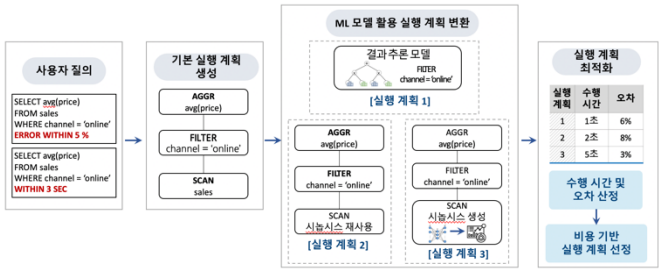
(그림 1) 근사 질의 언어 확장 구성도.

그림 1 은 근사 질의 언어 확장 구성도의 전체 모습을 간략하게 나타낸 것이다. 입력된 사용자 질의문을 파싱하여 정확한 결과를 요구하는 정확 질의문과 사용자가 원하는 정확도와 적시성을 표현하는 근사 질의문 등으로 구분하는 과정을 수행한다. 근사 질의 처리 언어인 경우에는 ML 모델 기반 및 요약 기반 근사 질의 수행 등으로 구분한다. 질의 파서를 수행하고 질의를 변환하여 다수의 실행 계획을 생성한다. 다수의 실행 계획에 대해서 최적화 과정을 수행하여, 최종 실행 계획을 선정하고 해당 실행 계획에 대한 질의 처리를 수행한다.

(A) : SELECT avg(price) FROM sales
WHERE channel = 'online' **ERROR WITHIN 5 %**
(B) : SELECT avg(price) FROM sales
WHERE channel = 'online' **WITHIN 3 SEC**
(C) : SELECT avg(price) FROM sales
WHERE channel = 'online' **ERROR WITHIN 5 % and WITHIN 3 SEC**

(그림 2) 근사 질의 언어 확장 사용 예.

사용자가 근사 질의를 요청할 때 사용자의 요구 사항을 표현할 수 있도록 질의 문법 확장 기능을 제공해야 한다. 이때 기존의 SQL 사용자가 쉽게 이해할 수 있도록 SQL 문법과 비슷하게 확장한다. 그림 2 는 사용자가 원하는 근사 질의의 정확도(오차 허용 범위)와 적시성(질의 처리 시간) 등을 지정할 수 있도록 질의 언어 확장 기능을 제공하는 모습을 나타낸 예이다. 사용자가 원하는 질의 결과의 오차 허용 범위(A)를 지원하기 위하여 근사 질의 언어를 확장하거나, 질의 처리 시간(B)을 각각 요청할 수 있다. 또한 사용자의 정확도와 적시성(C)을 한번에 모두 요청할 수 있도록 질의 언어 확장 지원한다.



(그림 3) 근사 질의 실행 계획 최적화 방법.

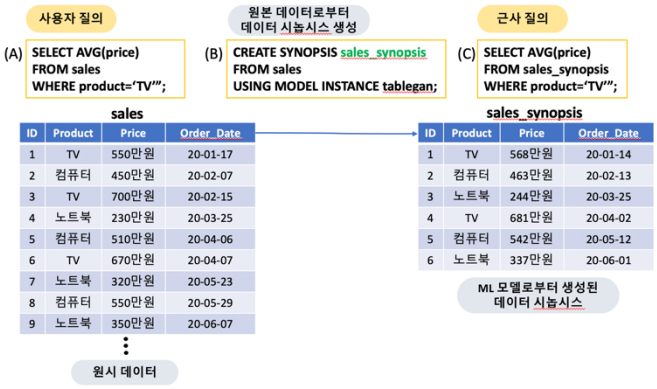
그림 3 은 실행 계획에 대한 최적화 과정을 도식적으로 간략하게 나타낸 것이다. 입력된 사용자 질의에 대해서 질의 파서 기능, 질의 변환 및 최적화 과정을 수행한 후 선정된 실행 계획에 대해서 최종 질의 처리를 수행해야 한다. 첫번째 단계로 입력 질의문에 대한 질의 파서 과정을 수행하여 기본 실행 계획을 생성한다. 생성된 기본 실행 계획을 기반으로 ML 모델을 활용한 다수의 실행 계획을 생성하게 되는데, 이때 활용하는 모델은 크게 결과 추론형 모델과 시놉시스 생성형 모델로 나눌 수 있다. 각 모델에 맞는 다수의 실행 계획을 생성하고 그중 사용자의 요구사항(오차 범위, 처리 시간)에 만족하면서 질의 처리 연산 비용을 최소화 할 수 있는 최적의 실행 계획을 최종 선정한다.

최적의 실행 계획을 선정하는 최적화 방법은 사용자가 요구하는 정확도와 적시성 등을 고려하여 각 실행 계획에 대한 질의 수행 시간과 결과 오차 등을 예측하여, 사용자가 원하는 요구 조건을 모두 충족할 가능성이 가장 높고, 그중 연산 비용이 상대적으로 적어 비용 효율성이 가장 높은 실행 계획을 우선적으로 선정하게 된다. 또한 근사 질의 처리 과정에서 필요한 다양한 메타 데이터 정보는 별도의 장소인 카탈로그에 저장 관리하게 된다.

본 논문에서 기술하고 있는 ML 모델 활용 방법으로는 결과 추론형 모델 방식과 시놉시스 생성형 모델

방식으로 구분할 수 있다. 결과 추론형 모델 방식은 사용자의 특정 형태의 분석 질의의 예측 결과를 추론하는 ML 모델을 생성하고, 해당 질의에 대한 질의 수행을 실행하기 위한 실행 계획을 구성하는 것이다. 학습된 질의 형태에는 최적화되어 있으나 질의 형태가 변경되면 ML 모델을 새로 생성해야 하는 단점이 존재한다. 시놉시스 생성형 모델 방식은 전체 데이터에 대해서 질의 수행하는 것이 아니라, 질의 처리에 사용할 수 있는 합성 데이터인 시놉시스를 생성하는 것이다. 시놉시스는 원시 데이터와 형태는 같으나 생성된 값을 갖도록 생성할 수도 있고, 연산자 처리를 지원하는 다른 형태를 갖도록 생성할 수도 있다.

시놉시스 생성형 모델 방식은 질의 대상 데이터 크기를 줄임으로써 질의 수행 시간을 줄이는데 목적이 있다. 질의의 정확도는 다소 떨어지더라도 질의 처리 시간을 줄임으로써 빠르게 처리 결과를 얻을 수 있다. 시놉시스 기반 질의는 시놉시스를 새로 생성하여 질의 처리하는 방법과 기존에 미리 생성된 시놉시스를 재사용하여 생성 비용을 줄이는 질의 처리하는 방법 등으로 처리할 수 있다. 시놉시스 기반 근사 질의 처리 방법은 특정 질의 타입에 최적화된 것이 아니라 단지 데이터 크기를 줄이는 구조여서 질의 형태가 자주 변경되는 모델에도 효율적인 방법이 될 수 있다.



(그림 4) ML 모델 기반 근사 질의 수행 방법.

그림 4 은 ML 모델을 기반으로 근사 질의를 수행하는 방법을 간략하게 도식적으로 나타낸 것이다. 예를 들어, 사용자가 원시 데이터에 대한 분석 질의(A) 요청을 처리하기 위해 (B)와 같은 형태의 구문을 통해 데이터 시놉시스를 사전에 생성해 두거나 질의 처리 시 새로 생성할 수 있다. 이때 데이터 시놉시스 생성에 활용하는 ML 모델 인스턴스는 별도의 구문을 통해 미리 등록, 훈련되어 있어야 한다. 시놉시스를 생성하고 나면 사용자 질의 요청 시에는 원시 데이터에 접근하지 않고 그 대신 생성된 시놉시스를 이용해 질의를 처리하여 결과를 제공할 수 있다.

ML 모델을 등록하고 이를 특정 테이블의 컬럼 집합이나 연산자 등에 맞게 훈련시키기 위해 CREATE MODEL, TRAIN MODEL INSTANCE 등의 별도 구문을 지원한다. 원시 데이터에 접근하지 않고 질의를 처리하기 위해서는 ML 모델과 모델 인스턴스, 원시 데이터의 테이블과 컬럼 정보 등 메타데이터를 관리하고 있어야 하는데, 이러한 메타데이터 정보는 별도의 데이터 카탈로그 스토어에 저장 관리하게 된다.

질의 처리 엔진에서는 사용자가 원하는 정확도, 수행 시간 등의 요구사항을 충족시키기 위해서 생성하는 데이터 시놉시스의 양을 조절하거나, 같은 연산을 지원하나 구조가 다른 여러 ML 모델을 훈련해 두고 이를 선택적으로 활용하여 최종 결과를 반환하게 된다. 이렇게 얻은 최종 결과는 정확도는 다소 낮으나 질의 결과를 적시에 제공함으로써 사용자가 데이터의 경향을 빠르게 파악할 수 있도록 해준다. 이러한 ML 기반 근사 질의 처리 기술은 데이터 탐색이나 시각화 같은 응용 분야에서 유용하게 쓰일 수 있다.

4. 결론

본 논문에서는 빅데이터 환경에서 근사 질의 처리를 위한 사용자의 요구사항을 표현을 할 수 있도록 SQL 문법과 비슷하게 질의 언어를 확장하고, 사용자 질의 파싱 과정을 수행한 후 다수의 실행 계획 생성 및 최적화 과정을 수행하여 사용자의 요구사항을 만족하면서 질의 처리 연산 비용 등을 고려하여 효율이 가장 좋은 실행 계획을 선정하는 과정을 제시하였다. 또한 결과 추론 모델 방식과 시놉시스 생성형 모델 방식에 대한 특징과 시놉시스 생성 및 메타데이터 관리 등의 근사 질의 처리 방법을 제안하였다.

* 이 논문은 2021 년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2021-0-00231, 빅데이터 대상의 빠른 질의 처리가 가능한 탐사 데이터 분석 지원 근사질의 DBMS 기술 개발)

참고문헌

[1] Sudipto Guha et al., “XWAVE: Approximate Extended Wavelets for Streaming Data”, VLDB, 2004.

[2] Yongjoo Park et al., “Database Learning: Toward a Database that Becomes Smarter Every Time”, SIGMOD, 2017.

[3] Fotis Savva et al., “ML-AQP: Query-Driven Approximate Query Processing based on Machine Learning”, arXiv, 2020.

[4] Moritz Kulessa et al., “Model-based Approximate Query Processing”, EDBT, 2019.