

論文要旨

学位論文題目： リアルタイムデータアクセス処理機構の最適化

氏名： 榎 美紀

近年、様々なデバイスから位置情報・気象・ソーシャル・センサーデータなど、膨大な量のデータが日々発信され続けており、地球上で生成されるデータは 2013 年から 2020 年の間で 4 兆 4,000 億ギガバイトから 44 兆ギガバイトへと 10 倍の規模に拡大すると予想されている。このような膨大なデータを利用して関連性を見つけたり、変化を検知したりと、様々な観点から分析する要求は年々高まっており、それらを実現するための処理基盤は大規模なデータをリアルタイムに、また高速に扱えることが重要となる。このようなストリームデータをリアルタイムに分析するため、2000 年以降ストリームデータ処理を主にした DSMS (Data Stream Management System) が登場した。従来の DBMS (Data Base Management System) と異なり、連続的に発信されるデータを対象にリアルタイムにデータ加工や分析結果を返し続けるシステムである。

従来の DBMS は、データがあらかじめデータベースに格納されている状態から、そのデータを対象に SQL によるクエリが発行され、所望のデータが返されるアーキテクチャである。発行されてくるクエリのパターンは様々である。一方、DSMS は流れてくるストリームデータに対して、連続的にクエリが実行される。データへの問合せ部分に着目した時、あらかじめ指定したウィンドウと呼ばれる単位にデータを区切り、その範囲内に到着したデータに対してクエリの演算を施すという特徴がある。また、ストリームデータはリアルタイムにデータが発信されてきており、その流量が時間や何かのタイミングで変化する場合は、常に一定の性能を保って処理し続けるシステムを保証することは難しい。そこで、インプットデータは、ランダムにサンプリングする等のフィルタリングを実施する。

DSMS で演算処理された後のデータは、一般的に破棄されていた。それは現在の状態における何らかの演算結果が重要であり、ストリーム処理システムはそのような要件を満たすためのシステムであるからである。また、ストリームデータはリアルタイムに到着し続けるため、全データをストレージに格納することは現実的ではなかった。ところが、近年のストレージの格納量の大規模化、SSD 等によるデータ入出力処理の高速化にも伴い、ストリームデータもそのままストレージに格納され、それをオフラインで分析対象にすることも増えてきた。しかしながら、ストリームデータのリアルタイムの分析とストレージに格納したオフラインでの分析を同時に考えたシステムは検討されていない。

そこで本論文では、リアルタイムなストリームデータを逐次的にストリーム処理する機構

と、蓄積されたデータを処理する機構を兼ね備えたリアルタイム分析システムを提案する。ストリームデータのリアルタイム性の収束のタイミングを議論し、リアルタイム分析とオフライン分析との使い分けの手法を検討する。

また、近年、あらゆるものがインターネットに繋がり、ネットワークを介して様々な情報を交換するようになる IoT (Internet of Things) の拡大や、スマートフォンの普及も影響して、世界中の各個人がアカウントを持ちテキストや写真、動画を送受信するソーシャルネットワークサービス、車等のデバイスからの GPS や機器情報のセンサーの発信など、世界中のデータは今も増加し続けて大規模になっており、一般にリアルタイムデータといっても、多岐にわたる。これらの情報を利用することにより、ストリーム処理を分析する際に、各データの特性を考慮した最適化が行えるのではないかと考えられる。そこで本論文では、大規模リアルタイムデータの一つであるソーシャルメディアデータを対象として、ソーシャルメディアデータがもつ特徴を利用したデータアクセスの最適化手法を、逐次的にストリーム処理する機構と、蓄積されたデータを処理する機構それぞれに提案する。

リアルタイム分析を実施するための、逐次的にストリーム処理をする部分については、出来るだけフレッシュなデータをストリーム処理内に維持するために、各メッセージの流行が収束するタイミングを拡散モデルを用いて早期に推定し、時間ウィンドウ幅をカスタマイズしてメンテナンスする手法を提案し、Twitter の実データを用いてその効果を示す。次に、代表的なリアルタイムの問合せパターンを実際に測定し、性能を評価する。インメモリなデータアクセスは高速にクエリが実行できるが、複雑な問合せパターンは演算のオーバーヘッドがあり、単純なクエリより応答時間が長くなる。そのようなクエリに対しては、計算の中間結果を保持したビューを導入して問合せの高速化が実現できることを示す。

また、ソーシャルメディアには、ある出来事が突然大きく話題になると瞬間的に多くのユーザーが一斉にメッセージをを発信し出し、バースト状態を起こす特徴がある。このようなデータバースト時のクエリの応答時間の劣化を回避するため、各メッセージに付随する情報に着目して重要度を計算し、重要度が低いと判断されたメッセージをフィルタリングして処理するデータ量をコントロールする手法を提案する。これにより、従来のランダムなフィルタリングと比較して、同程度のサンプリング量において、問合せ結果の精度高く維持しながら、問合せ時間をより短く処理できていることを示す。

提案システム内の蓄積されたデータを処理する部分については、データアクセスの処理性能の評価し、提案したデータアクセス最適化手法の効果を検証する。データアクセスには、POJO ベースの O/R マッピングアーキテクチャである JPA を利用するため、まずその基本性能を評価した。データアクセス処理の高速化には、キャッシュの利用が効果的であるが、更新クエリの割合が増えてくると、従来のキャッシュメンテナンス手法では性能が低下することを確認し、我々の提案手法である列単位、値単位で更新の影響が及ぶキャッシュのみを無効化する、より細粒度のキャッシュメンテナンス手法により、性能が改善することを示す。また、無効化すべきデータを特定するための無効化用インデックスが効果を維持できる

よう、メモリ領域の容量にあわせてサイズ変更可能な無効化用ハッシュインデックス導入し、キャッシュヒット率を大幅に向上できることを示す。クエリアクセスに偏りがある場合は、無効化用重み付ハッシュインデックスを導入することにより、さらに性能を高く維持できることを示す。

また、ソーシャルメディアの情報拡散ネットワークのようなグラフデータアクセスの高速化のため、ビットマップのインデックスを導入する。このインデックスは大規模な粗行列になることが想定されるため、実際に **Twitter** のデータを用いて、行列を圧縮してサイズ削減の効果を評価する。ソーシャルメディアデータの特性である、情報拡散と拡散されやすい経路等を考慮して、あらかじめ行列を並び替えることにより、さらに圧縮率を高めることが出来ることを示す。