

③



般

ビッグデータ処理基盤

—クラウド環境においてビッグデータを扱うシステム—

鬼塚 真（大阪大学）

1 ビッグデータへの期待

コンピュータ将棋，しゃべってコンシェルや Siri などの音声認識，あるいは質問応答システム Watson などに代表されるように，ビッグデータを活用して人間の能力に近い情報処理をする技術が注目されている。また，Data is the new oil（データは新しい石油である）といわれるように，ビッグデータを分析することで隠れた知識やルールを発見して，社会的あるいは経済的なインパクトを生み出すことが期待されている。市場調査会社の IDC Japan の 2015 年 5 月の報告によれば，ビッグデータ分析に使われるインフラの国内市場が 2019 年までに 1,469 億円に達する（年平均成長率 27% に相当）としている。一方で，企業に対する現状のアンケート調査結果では，ビッグデータに関する技術を利用あるいは利用を検討している企業の数はこの 1 年で増加が収束傾向にあり，全体の 30% 強という割合にとどまっている。つまり，ビッグデータに取り組んでいる一部の企業だけが，今後もビッグデータに投資を拡大するという状況であると考えられる。

IT 系の人材という点について見ると，データを分析する専門家であるデータサイエンティストは 21 世紀で最もセクシー（魅力的）な仕事と言われており，またデータを分析する基盤システムの技術者も IT 系の企業において求められている。このようにビッグデータへの期待を背景とし，情報処理を専門とする学生や技術者には活躍するチャンスが大きく広がっていると考えられる。

✦ 代表的なデータ分析技術

一般的な企業においてビッグデータを活用するということは，ビッグデータを分析して知識やルールを発見してビジネスに役立てるということになる。たとえば，会員カードを活用して消費者の購買履歴を取得・分析することで，商品の在庫管理を最適化してコストダウンを図り，また消費者が求める新たな商品を開発することで利益を上げる例が挙げられる。

データ分析技術は，統計学などの数学的な基礎に基づく領域から，1990 年代から始まった大規模データを対象としたデータマイニングの領域など幅広い。特にデータマイニングについては，代表的な技術として多次元データ分析，相関ルール分析，協調フィルタなどの推薦技術，クラスタ解析・分類器などの機械学習技術などが挙げられる。多次元データ分析は，地域・期間・商品属性など多様な視点から履歴データを分析する技術であり，新規顧客開拓や商品の販売データ分析など広く利用されている。相関ルール分析は，「ビールとおむつが同時に売れる」という例にあるように，データ間の共起関係を分析する技術であり，店舗における商品配置に利用されている。協調フィルタなどの推薦技術は，利用者の購買履歴に基づいてまだ購入していない商品の購買可能性を推定することで商品を推薦する技術であり，Amazon や楽天などの Web 企業において使用されている。また，協調フィルタは Netflix が主催したビデオ推薦に関するコンテストにおいて推薦精度の壁を越えたベースの技術となっている。クラスタ解析・分類器などの機械学習技術は，データに内

在する隠れた構造を発見したり、与えられた正解データから分類方法を学習する技術であり、利用者や商品の分類などで利用されている。別の領域として、映像認識・音声認識・自然言語処理などの領域では、隠れマルコフモデルや、最近で言えば Deep Learning（深層学習）の技術が注目されている。

ビッグデータ処理基盤

ビッグデータを対象として上述したデータ分析を行うためには、ビッグデータを格納して分析プログラムを実行する処理基盤が必要である。近年は、Amazon Web Services などのクラウド環境が急速に普及して、誰でも簡単にクラウド環境が利用できるようになっている。2014 年における売上規模に関しては、Amazon Web Services は年間 50 億ドルで、年間の成長率が 50% であると予想されている。この売上規模は Microsoft、IBM、セールスフォースも同レベルの水準にある。

クラウド環境では、スケールメリットを活かすため、つまり計算機資源を低価格で大量に調達することによって利用者に対して低価格でクラウドサービスを提供するため、エクサバイト（テラバイトの 100 万倍）スケールのデータ規模あるいは 100 万台を超えるサーバ群を運用するための、ソフトウェアおよびハードウェアからなるビッグデータ処理基盤を構築することで、大規模なデータ処理を実現している。このようなビッグデータ処理基盤に関する技術は、処理の対象となるデータが蓄積されたデータ（蓄積型データ処理）であるか／時々刻々と生成されるデータであるか（ストリーム型データ処理）の観点から、2つの技術領域に分類することができる。分かりやすく言えば、蓄積型データ処理は過去のデータに対する処理であり、ストリーム型データ処理は現在のデータに対する処理である（図-1）。

❖ 蓄積型データ処理

蓄積型のデータ処理技術をさらに細分化すると、100 台程度の規模のハイエンドな計算機を用いる分

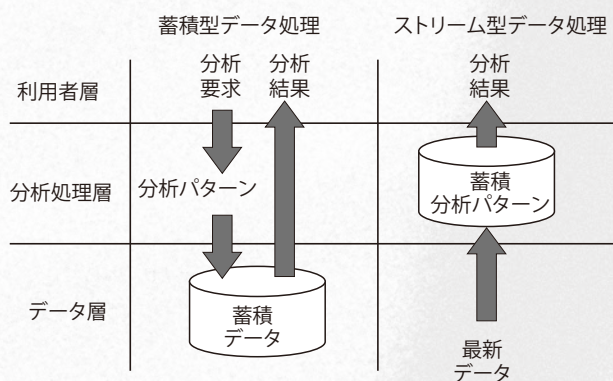


図-1 蓄積型とストリーム型のデータ処理

散データベース管理システム（分散 DBMS）の技術と、分散 DBMS の機能を簡略化して 1,000 台を超える大量のコモディティ計算機を用いる NoSQL および MapReduce に代表される分散処理基盤がある。前者の分散 DBMS については Oracle、IBM、Microsoft などのデータベース製品や SAP の HANA に代表される主記憶型の DBMS がある。最先端の技術として、大量の主記憶・メニーコア・SSD・高速ネットワークなどの最新ハードウェアを活用した分散 DBMS の研究がなされている。たとえば東大と日立が共同開発・商用化しているデータベースとして、検索命令を細分化しハードディスクレベルで実行順序を入れ替えることでハードディスクの性能を最大限に引き出す（非順序型実行原理に基づく）超高性能データベースエンジンが挙げられる。

後者の分散処理基盤技術に関しては、米国の Google、Amazon、Facebook、Twitter などの Web 系の企業が先導しており、各社の専用の用途に応じて開発が進められている。データベースの間合せ言語である SQL を高速に処理する基盤技術としては、Google がクラウドサービスとして提供している BigQuery や Amazon の Redshift がある。Google の BigQuery では、数億レコードを数秒で検索処理することが可能であり、数千台のディスクを同時に利用していると推測される。Amazon の Redshift については、ディスクアクセスを並列化すると同時に、分析対象のデータのみをディスクアクセスするためのカラム指向の技術とその軽量圧縮技術を使う

ことで高速性を実現している。一方、大学においては、機械学習処理やグラフ処理を高速化する基盤技術に関する研究が進められており、MapReduceの後継となる技術が多く提案されている。たとえば、機械学習や行列計算の中でも特に繰り返し型の分析処理を対象として、分析処理の中間結果を主記憶に保持することでディスクアクセスコストを削減した技術として Spark が注目されており、分析に関する多様なライブラリが提供されコミュニティとしても大きくなりつつある。また人・モノ・場所に関する関係情報をノードとエッジから構成されるグラフデータとして表現し、グラフデータを分析する技術も注目されており、近年の VLDB や ACM SIGMOD などデータベース系の難関国際会議において多くの論文が採択されている。これらの技術では、グラフデータを分割して複数のマシンに格納することで通信コストを削減し、グラフ構造上において計算処理を伝搬することで分析処理を実現しており、代表的な技術として Pregel や GraphLab/PowerGraph などが挙げられる。

❖ ストリーム型データ処理

時々刻々と生成されるデータを対象とする技術として、一定量の限られた主記憶を利用して高速にデータ処理要求を処理するストリーム型のデータ処理技術がある。ストリーム型のデータ処理は DBMS とは対称的な技術であり、DBMS が事前に登録された過去のデータに対して入力される処理要求を処理するのに対して、ストリーム型のデータ処理では事前に登録された処理要求に対して入力されるデータストリームを処理する。製品としては、各データベースベンダが提供している。Web 系の企業においては、Twitter は Storm や Storm を発展させた Heron を開発しており、Twitter サービスのシステムの負荷の監視などの目的で、集計あるいは機械学習によってテラバイト規模のデータ分析を日々行っている。またストリームデータを対象とした機械学習エンジンとして NTT と PFI が共同で開発した Jubatus があり、オンライン型の分類器や異常検知

などの機能に加えて、映像認識の応用向けに Deep Learning の機能開発がされている。

❖ Hadoop 開発コミュニティ

ビッグデータ処理基盤については、Yahoo! を中心に Hadoop の OSS（オープンソースソフトウェア）の開発コミュニティが 2006 年に立ち上がり、Google が開発したシステムである分散ファイルシステム GFS、分散処理基盤 MapReduce および BigTable、分散ロック Chubby の OSS 版の開発がなされてきた。現在では、Hortonworks や Cloudera などの Hadoop を利用してビジネスを展開する企業を中心となってコミュニティ開発が継続されている。日本でも Hadoop Conference Japan が 2009 年から始まり、2014 年では参加者が 1,300 名に達するイベントに成長していて、OSS 開発が全盛の時代にあると言える。参加者は IT 系企業の人々が中心であり、Hadoop の利用方法や実装の詳細に関する情報共有、あるいは Hadoop の OSS コミュニティに貢献するという目的で開発者たちが集まっている。実際、2014 年に日本から Hadoop コミッタが 3 名輩出されており、Hadoop コミュニティに対する日本の貢献が大きくなりつつある。企業側の立場から見ると、コストダウンを図るという目的で OSS を活用するという機運が高く、OSS を活用するためには OSS に詳しい技術者を育成する必要がある状況にある。

❖ 基盤技術開発の難しさ

ビッグデータ処理基盤を開発する際の技術的な難しさは何であろうか？ 大きくは、次の 3 つの特徴に起因していると考えられる。1) 大量のマシンが同時に動作すること、2) システムがネットワーク分散していること、3) データ規模が大きいことである。

まず大量のマシンが同時に動作することに起因する難しさについて 3 点説明する。1 点目は、マシンが大量であると定期的に一定数のディスクやマシンが壊れるため、ハードウェア故障が起こる前提でソ

ソフトウェアを設計する必要があることである。具体的には高い可用性を実現するための機能が必要であり、プロセスおよびデータを多重化しておくことで、故障発生時にはシステムの正常系の動作に極力影響を与えないように、プロセスおよびデータを復旧させる機能を有する必要がある。実際、Googleの分散ファイルシステムではデータおよびプロセスの多重化が実現されている。2点目は、マシン台数の増加に伴ってシステムの内部状態が組合せ的に増加するため、故障発生時に原因究明が難しくなることである。たとえば、マシン台数が10台から100台に増えるだけでシステムの内部状態数が組合せ的に増加し、より難しい障害が発生する。特に永続データを管理する分散ファイルシステムはプロセスの状態に加えて永続データの状態が加わるために、障害がより複雑になりやすい。Amazonでも、ネットワーク容量をアップグレードしようとした際の操作ミスによってシステムの負荷が上がり、データのミラーリング機能が連鎖的に働いてシステム全体が停止した例がある。3点目は長期に渡って運用していると負荷が偏ることがあるため、負荷分散するようシステムを設計する必要があることである。たとえば、NoSQLではデータのキーを用いてデータの分散を決定しているが、一般に負荷が均等になるようにキーを設計することは難しいため、システム側で負荷の不均衡を検出してデータの再配置を行うことが望ましい。実際、GoogleのNoSQLであるBigTableでは負荷分散の機能を提供しているが、マシン台数に線形の性能が出せない最大の原因は負荷分散が難しいことであると報告されている。

次に、システムがネットワーク分散していることに起因する難しさについて2点説明する。1点目は死活監視が難しいことである。一部のマシンが高負荷状態になった場合、ネットワークの外から監視した場合にマシンが障害で停止しているのか、単に高負荷で一時的に応答がないだけなのかを区別することが難しい。ネットワークを運用系と監視系の2系統用意して、負荷の少ない監視系を使うことでマシンが動作しているか否かは判断できるように

なるが、このケースでもプロセスが高負荷なのか障害で停止しているのかの区別をすることができない。このような問題に対して、分散ロックが開発されている。分散ロックはシステム全体のプロセスの死活監視を行うものであり、プロセスが一時的に高負荷で応答がない状態であっても、そのプロセスに障害が起きたものと判断して、プロセスおよびデータを強制的に復旧させる。ただし、復旧の際にシステム全体として不整合が起きないようにシステム全体を設計しておく必要がある。2点目はマシン間での時刻同期が難しいことである。データの更新に対しては、タイムスタンプやトランザクション番号などを用いて更新操作の前後関係を判定することでデータの一貫性を保証する必要がある。しかし、タイムスタンプあるいはトランザクション番号のいずれの場合もシステム全体で大域的な（複数マシン間で同期された）値を取得する必要がある。マシン台数が多い場合にはシステム全体で同期をとるコストが非常に大きくなる問題がある。この問題に対して、ベクタークロックを用いて大域的な値を利用しない代わりに競合が生じた際にはアプリケーションレベルで競合を解消する方法や、GoogleではGPSや原子時計を用いることで、異なるデータセンタ間においても個々のマシンが独立に正確なタイムスタンプを取得することを実現している。

最後に、扱うデータ規模が大きい点から生じる難しさについて説明する。大規模な人工データでは実データと異なりバリエーションが少ないため、試験工程においてバグが発見しにくい問題がある。この問題に対しては、現在稼働しているシステムと並列に新システムを小規模で導入して段階的にシステムを入れ替えるという方法が利用されている。

今後の展望

ビッグデータを分析することで社会的あるいは経済的なインパクトを生み出すためには、多くの応用において時間をかけて分析技術を改善あるいはチューニングする必要がある。実際にIBMのWatson

や Netflix の例では、分析精度を向上するために数年が費やされている。今後の研究の方向性としては、上記のような人間が試行錯誤する分析工程を自動化する技術が重要であると考えられる。たとえば、機械学習におけるハイパーパラメータの最適化や、多次元データ分析において特徴的な分析クエリを自動的に探索するなどの課題が挙げられる。一方、ビッグデータの処理基盤に関しては、最新ハードウェアを活用した高速化と低コスト化の観点で技術がこれ

からも発展するとともに、グラフデータ構造や応用ごとに専用化された基盤技術が今後も発展するものと考えられる。

(2015 年 5 月 25 日受付)

鬼塚 真 (正会員) oni@acm.org

大阪大学大学院情報科学研究科教授、博士 (工学)。1991 年東京工業大学工学部卒業。同年、NTT 入社。2000～01 年ワシントン大学客員研究員。2010～14 年電通大客員教授。2012～14 年 NTT 研究所特別研究員、2014 年より現職。