Hadoopを活用した大規模データ解析の動向と今後の展望

古関　聰，佐藤　直人

高速ネットワーク環境の普及にともない，現在，社会の様々な場面においてデータの大規模な収集・蓄積が進んでおり，これを解析することで有用な情報を抽出したいという機運が高まっている．このための計算基盤として，並列分散計算の仕組みであるHadoopが有望視されており，実際にHadoopを活用した事例もいくつか報告されている．しかし，Hadoopの提供するMapReduceフレームワークは比較的低レベルな仕組みであり，データ解析利用にはプログラムの特別な設計が必要であることから，Hadoopをデータ解析に活用するためのプログラミング・モデルやツールが求められている．本稿では，このようなHadoop上でのデータ解析をとりまく状況を概観し，Hadoopとデータ解析アプリケーションとの間のギャップがどのように埋められようとしているかについて解説を試みる．

キーワード：分散処理，MapReduce，Hadoop，データ解析，ビジネス・インテリジェンス

1. はじめに

1.1 大規模データ解析をとりまく状況

現在，社会の様々な場面でデータの蓄積が急速に進んでいる．自動車のECU（Electronic Control Unit）やPOSシステム，コールセンター，Webサーバのログ・データ，あるいは監視カメラからのセンサー・データなどがその代表である．この背景には，ストレージの大容量化，カメラやRFIDに代表されるセンシング・デバイスの普及，そしてネットワークの高速化・広域化がある．

蓄積されたデータは様々な形で活用されており，例としては，自動車のECUログをもとにした車両検査[1]等が挙げられる．そこでは，車載ECU（OBD-II）のデータをサーバに集約し，故障履歴・傾向を分析することで整備の効率化を支援している．同様の履歴を使った分析は，携帯の通信ログや流通システムにおけるPOSデータのログに対しても行われている．

また，ネットワーク・サービスのクラウド化に伴い，蓄積したデータをクラウド上に配置し，Hadoopに代表される分散処理技術を用いて，データの解析をサービスとして提供する仕組みも構築されはじめている[2]．

このような社会的・技術的変化にともない，Hadoopを用いた様々なデータ解析のための基盤技術の確立が強く要請されている．

1.2 技術的課題

クラウド上での大規模データ解析をビジネスで広く活用する際に直面する主な課題は，蓄積されたデータのもつ特性それ自身，すなわち「多様であること」および「大量であること」にある．

● データの多様性および非定型性の考慮

様々なデバイス（RFID，ECU等）およびネットワークの普及により，多様なデータの蓄積が進んでいる．多くの場合，データには（いわゆるDBにおけるスキーマのような）固定的・定型的な構造はなく，XMLに代表される半構造化データとして表現されることが多い．また，ログに顕著なように構造定義の変更も頻繁に生じ得る．したがって，これら様々な形態のデータに適用する解析技術のコア（共通部分）と，個別の種類のデータに特化した部分とをどう分離し，かつ全体を一つのアーキテクチャとしてまとめるかが問題になる．すなわち，データの分散への対応のみならず，半構造化データへのアクセス・操作，スキーマ定義の変更への対応を容易にする柔軟な構造定義の仕組みが必要になる．

● データの量の考慮

解析の対象となるデータのサイズは一般に非常に大きなものになり，例えばサーバ・アクセス・ログが数百GBになることは珍しくない．クラスタへの分散が解析の前提になるが，その際，分散データ全体を一体として扱うためのシステムが必要になる．さらに，金融証券情報の解析のように，データの解析を短時間で

こせき　あきら，さとう　なおと
日本アイ・ビー・エム㈱ 東京基礎研究所
〒242-8502 大和市下鶴間1623-14

行うケースもあり，数日かかる分析処理を数時間で行うといった高速処理のための並列分散処理が必須になる．

1.3 Hadoop—急速に普及しつつある分散処理技術

クラウド上における高速処理技術として，近年Hadoop[3]が急速に普及しつつある．Hadoopは，もともとは，Web上でのテキスト検索技術として開発され，現在ではテキスト検索以外の幅広いデータ解析に応用されている．Hadoopはスケール・アウト型の処理を基本にしており，クラウド上に分散したデータに対して，同一の内容のタスクを並列に適用し，得られた個別の中間結果を最後に集約することで最終的な結果を得る．これがクラウド環境における計算モデルとして有効であることが広く認識され，データ解析技術をHadoopモデルの上に構築する試みも盛んに行われている．しかし，現時点ではまだ要素技術の開発の段階に留まっており，「共通化されたサービスが利用可能な大規模データ解析基盤の確立」といったレベルにまでは至っていない．

1.4 データ解析技術の基盤

Hadoopを用いたクラウド上でのデータ解析のための汎用的な技術基盤を構築するには，以下の四つの階層を考慮することが重要になると我々は考えている．

- Hadoop/Cloudインフラストラクチャー層
- プログラミングプラットフォーム層
- データ解析フレームワーク層
- アプリケーション層

まず，Hadoopで構成された最も低層のインフラストラクチャー層の上位層となるプログラミングプラットフォーム層（第2層）を導入する．Hadoop自身が提供する仕組みはMapReduceフレームワークのみであり，これらを使って様々なデータ解析を実装することは大きな負担をともなう．Pig[4]，Hive[5]，Jaql[6]に代表される，より高位の言語が必要になる．

次に，このようなプログラミング言語を用いて実装されたデータ解析フレームワーク層（第3層）を導入する．ここでは，テキスト解析，統計解析，シミュレーションなど，データ解析のための共通技術が実装される．CognosやRなどのツールとHadoopとの統合もこの層で実現される．

そして最上位に，個別のアプリケーション分野・対象に特化したアプリケーション層（第4層）を導入する．ここでは，例えば金融・証券，自動車，流通といった各々の分野ごとに必要なデータ解析を行うことになる．

1.5 本稿の構成

本稿の構成は以下の通り．まず次節で，Hadoopおよび高位レベルの言語について概説し，第1，2層で用いられる基礎技術について説明する．次に，3節では，Hadoop上でのデータ解析の試みと関連技術（第3層）について紹介する．4節において，Hadoop上での分析アプリケーション（第4層）について概観した後，5節で，今後の展望を示し，本論文のまとめを行う．

2. クラウド環境における分散処理基盤—HadoopとJaql

2.1 Hadoop

Hadoopに関してはすでに多くの解説が出ているので詳述は避けるが，HadoopはGoogle社が考案したMapReduceフレームワーク[7]のオープンソース実装である．MapReduceフレームワークは，以下のような型を持つMap処理とReduce処理から成るプログラミングモデルを提供する．

Map : (k1, v1)->list(k2, v2)

Reduce : (k2, list(v2))->(k3, v3)

ここで，Map処理により生成されたk2をキーとする値（v2）の集合は，一つにまとめられてReduce処理に渡される．Reduce処理はk2をキーとするすべての値（list(v2)）に対し，集約的な処理を行う．ユーザは，このスキームをインプリメントするようなMapやReduceのロジックを実装（値の処理やキーの生成を記述）することでプログラミングが行われる．

Hadoopでは，このようなフレームワークを実現する主要コンポーネントとして，データの格納先となる分散ファイルシステムHDFS（Hadoop Distributed File System）と，分散実行エンジンHadoop MapReduceが用意されている．

Hadoop MapReduceでは，JobTracker，TaskTrackerといったプロセスによりジョブが進行され，それぞれ，指定されたJobTrackerノード，TaskTrackerノードで実行が行われる．JobTrackerはユーザからのJob（Hadoop処理単位）を受け付け，TaskTrackerを管理して全体の分散処理をコントロールするプロセスである．JobTrackerは，まず，投入されたジョブに対しMap処理の入力となるスプリットをHDFSと通信することで計算する．次に，JobTrackerは各スプリットに対するMapタスクと設

定された数のReduceタスクを生成する．生成されたタスクは，JobTrackerにより順次TaskTrackerへの割り当てが行われる．ここで，Mapタスクの割り当ては，データのローカリティーが高くなるように行われることになっている．

スプリットデータの取得はHDFSを介して行われるが，HDFSはファイルシステムの名前空間等のメタデータを管理するネームノードと，分散データへの実際のアクセスを行うデータノードから構成される．上記MapReduceのプロセスは，ネームノードに指示されたデータノードと直接通信を行い，分散データへのアクセスが実現される．

2.2 Jaql

Hadoopアプリケーションを開発するには，Hadoopフレームワークを利用したJavaのプログラミングが必須となる．このとき，たとえアプリケーションロジックが簡単であったとしても，MapReduceフレームワークの各オブジェクトの設定やファイル入出力に関する記述など，実際に必要な作業量は少なくない．このような開発容易性に関する問題を解決するため，いくつかの対話的簡易言語が開発されている．リレーショナルなデータモデルに基づき，スクリプト言語のような文法でデータフローを記述できるPig Latinや，SQLによく似た文法でデータ操作が可能なHiveといったものが開発されている．

ここで紹介するJaqlは，IBMアルマデン基礎研究所で開発された，JSON[8]形式のデータを処理するための言語で，HiveやPigと同様にHDFS上の大規模データ操作を容易に記述することを目的としている．

Jaqlには，HDFSやローカルファイルに対するIO機能の他，JSONデータに関する，プロジェクション，フィルタリング，ジョインといったオペレータが用意されており，これらのオペレータをUNIXのシェルのパイプのようにつなげていくことでプログラミングを行う．代表的なスクリプトは以下のような記述になる．

read(<hdfsファイル>)->オペレータ->…->オペレータ->write(<hdfsファイル>)

ここでオペレータへの入力は$という変数にバインドされJSONデータの各フィールドに演算子"."を使ってアクセスすることができる．例えばfilterといったオペレータを例にとると，例えば入力データが"xx"といったフィールドを持つとして，入力データは$にバインドされるので，"filter $. xx>0"と記述

オペレータ	記述例	意味
filter	filter $.id == xxx	入力データのidフィールドがxxxであるものを抜き出す
transform	transform {$.id, $.name}	入力データの中のidとnameフィールドを取り出す
group by	group by $g = ($.key) into {$g, count ($)}	入力データをkeyフィールドでグループ化し、key毎のカウントを取る
join	join $x, $y into $x.id == $y.id into {$x, $y}	idフィールドが等しい$x,$yのデータを結合する

図1 Jaqlの主要オペレータ

することで，「入力のxxフィールドが0以上のものを抜き出す」といったオペレータが作成できる．

図1は，Jaqlの代表的オペレータである．

また，Jaqlには拡張性があり，関数の定義を行えるほか，Javaのユーザ定義関数を呼び出すことも可能である．

以上，Jaqlについて簡単に触れたが，Jaqlの言語仕様やスクリプトサンプルに関してはJaqlのサイト[6]に豊富に情報があるので参照していただきたい．また，Jaqlを題材とした日本語の解説[9]やレポート[15]等も充実してきており，今後の関心の広まりが期待できる．

3. Hadoopを用いた，クラウド上でのデータ解析のための技術

Hadoopの登場と成功は，いわゆるWeb系の企業以外に関しても，大規模データ解析の関心を高めるといった効果をもたらしている．さまざまな業界において，企業内に大量に蓄積されたデータをうまく分析し活用することで企業のパフォーマンスを向上させる試みが行われていると考えられる．このとき重要となる問題は，データ解析をどのようにスケールさせるか，すなわち，データ解析に使われるアルゴリズムをどのように効果的にHadoop上で実装するかである．

さまざまな機械学習アルゴリズムをMapReduceフレームワーク上で実装してスケールさせることができることは2006年に発表された論文[10]で実証されている．一つの代表的な考え方としては，二乗誤差の最小化問題が行列形式で表されるような問題設定を考えたとき，最小二乗誤差を与えるパラメータの計算が分散処理に適した個別の行列演算に分解できるといったものが挙げられる．論文[10]では，回帰分析，独立成分分析，主成分分析，サポートベクターマシンといった多くのアルゴリズムの分散化方法が示されており，大規模なデータを対象にした解析がスケーラビリティ

を持つことが確認できる．

3.1 R と Jaql の統合

ここでは，前述の Jaql と R を統合した大規模機械学習に関する研究結果を紹介する．IBM アルマデン研究所では，前述の論文[10]の結果を援用し，分散処理の計算を Jaql で記述しその結果を R 上で操作する実証が行われている[11]．論文[11]では，推薦システムに応用される非負行列の因子分解について，R で最適化問題を記述し，最適化関数に渡される二乗誤差を計算する関数，二乗誤差の勾配を計算する関数を Jaql で定義することで，うまく問題の記述性を高めながらも適切なスケーラビリティが得られることが示されている．

3.2 シミュレーション

シミュレーションは，データ解析において最も幅広く用いられる計算手法のひとつであり，また並列分散化によって性能の大幅な向上が期待できる分野でもある．その中でもモンテカルロ・シミュレーションは適用分野の広さから特に重要であるが，逆にどの分野に適用するかによって実際の処理の内容に大きな違いが生じ，それが効率的な並列分散化の方法にも影響を与える結果になる．そのため，Hadoop を用いた「モンテカルロ・シミュレーションのための高性能で汎用的なツール」と呼べるものはまだ確立していない．このような状況において，以下で紹介する MCDB[12]では，リレーショナル DB における SQL クエリをデータ解析のアルゴリズム記述に用いることで，Hadoop/Jaql 上での効率的なモンテカルロ・シミュレーションを実現している．

Monte-Carlo DB（MCDB）は，IBM アルマデン研究所において開発された拡張リレーショナル DB の一種（確率 DB と総称される）であり，確率的に分布するデータを，SQL の FLWOR（For/Let/Where/OrderBy/Return）表現を用いて定義することを可能にしている．MCDB では，正規分布等の確率分布関数にしたがってデータが分布する仮想的な DB を定義できる．このとき，分布の特性（正規分布における平均・分散など）を陽に与える必要はなく，FLWOR 表現において宣言的に定められたデータ間の関係から間接的に定めることができる．データが定義され，それに対する計算が SQL クエリとして与えられると，MCDB は実際のモンテカルロ・シミュレーションのためのデータ生成を行う．

Hadoop 環境での実行モジュールは MC3 と呼ばれ，SQL 処理におけるデータの生成・読み出しを Map に，ジョイン演算を Reduce に変換する．MC3 は，多段 MapReduce 処理の効率的なスケジューリングを実現している．

3.3 テキスト分析

インターネットに蓄積された大規模なテキストを解析することも企業の大きな関心事項の一つである．大規模テキストを単独で分析する試みはすでに始められており，今後は，さらにテキストのデータに企業が持っているこれまでのデータを組み合わせて分析するといった試みが行われるようになると考えられる．

このようなテキスト分析の基礎となる技術がテキストの構造化である．テキストの構造化は，フリーソフトでも提供されている形態素解析を含め，構文解析，意味解析を利用した高度な構造化，さらに，一度構造化したデータを，人物，商品，企業といった概念でまとめあげるエンティティ・アナリシスといったものまでを含めた概念である．

このように構造化されたデータは，既存のデータマイニングの新しい入力となり，回帰，クラスタリングといった応用範囲の広い分析の効果を高めていくことが期待できる．

3.4 ビジネス・インテリジェンス

多くの企業では，データを可視化することで企業活動の状態把握や企業の意思決定を行っており，このような可視化を行うための数多くのパッケージが利用可能になっている．企業が抱える大規模なデータに対しても可視化を中心としたビジネス・インテリジェンスを活用するモチベーションは高まってきている．IBM では，HDFS 上のデータを表計算パッケージのように操作できる機能や，IBM Cognos BI[13]から Jaql を呼び出し，Jaql によるデータ操作結果をレポートに取り込む機能が開発されており，ビジネス・インテリジェンスのニーズに答える試みを行っている．

4. Hadoop 上での分析アプリケーション

ここでは，Hadoop のアプリケーションエリアについて整理を行い，大規模データ解析の展望について示唆を与えることを試みる．

大規模データ解析の応用エリアについては，データの観点と分析手法の観点から二次元のマトリクスが構成できる．このマトリクスの要素を業界別に埋めることで，Hadoop の応用エリアのマップを示すことができる．

大規模なデータは，センサー，通信，電子的操作等のアクションを源とし，刻々と生成されて蓄積されている．企業における応用を考えた場合，このデータの蓄積場所ということから，プライベート-パブリックといった分類を考えることが有効であると考えられる．これは，プライベート・クラウド，パブリック・クラウドといった分類と対応するものである．

また，分析手法は，前節で紹介したものが挙げられる．シミュレーション，データマイニング，テキスト分析といったものが代表的であると考えられる．

このようなマトリクスを，業界固有のデータや分析をもとにして埋めることで，Hadoopアプリケーションの概観図を得ることが可能である．図2は，金融業界を例にとり，試みにマトリクスを構成し大規模データ解析を整理したものである．

ここで，蓄積される社内データとしては，金融機関が保有している金融資産，負債データや，金融取引によって発生するトランザクションデータの他，内部格付けを行うための各種信用データ等が挙げられる．

また，社外データとしては，公開されているマーケット時系列データの他，ニュースやレポート等のテキストが大規模に蓄積され利用できると考えられる．

このような蓄積データに対するシミュレーションの適用エリアとしては，将来価値の分布計算が挙げられる．例えば，公開されたマーケットデータからリスク・ファクターの変動モデル（時系列モデル）を構築し，将来のリスク・ファクターをシミュレーションで多数生成し，ポートフォリオ価値の分布を得ることでVaR（Value at Risk）等を求めることが考えられる．また，企業内プライベートデータである社内格付けやデフォルト率をリスク・ファクターとしたVaRをシミュレーションにより計算し，信用リスク管理を行うことも考えられる．

	大規模に蓄積されるデータ	
	社内データ ・保有ポジション ・トランザクション ・社内、格付け、信用データ	パブリック・データ ・マーケット時系列データ ・経済ニュース、レポート
シミュレーション	・モンテカルロ法VaRによる信用リスク管理 ・金融商品プライシング ・IFRS「期待損失アプローチ」による償却原価測定	・モンテカルロ法VaRによる市場リスク管理 ・シナリオ分析
データマイニング・最適化	・不正検知 ・クレジットモデル構築	・裁定機会発見
テキスト	・大量の目論見書分析	・ニュースの分析によるマーケット理解

図2 金融業界における大規模データ解析応用例

蓄積データに対するデータマイニングも様々な用途が考えられる．クレジットカード等のトランザクションデータを分析して不正を検知するためのモデルの構築をすることや，企業の信用力を推定するためのモデルを大規模に蓄積された財務データの判別分析をすることで構築することが挙げられる．また，公開されたマーケット時系列データの蓄積から統計的な裁定機会を発見し，統計アービトラージ戦略によるアルゴリズムトレーディングに活用するといったことが考えられる．具体的な例として，VISAでは，大量のログをHadoop上に蓄積し不正検知のためのモデルを構築するといった試みが行われている[14]．

金融テキストデータの解析は，テキスト構造化技術の進歩とともに今後注目されるエリアだと考えられる．企業の財務報告のテキストを大規模に分析することで信用モデルの高度化を図ることや，保有金融資産の目論見書を分析することで，ポートフォリオ全体の価値に深刻な影響を及ぼすようなシナリオの抽出等を行うことも可能になると考えられる．また，ニューステキストをマーケット時系列データと関連させ，時系列の変動パターンを説明するようなテキストを抽出させるような分析がパブリックデータを利用して可能になるといったことが考えられる．

5. まとめと今後の展望

本稿では，現在急速に普及しつつある分散処理技術Hadoopをもとにした，クラウド環境における大規模データ解析の現状について概説してきた．

2節ではHadoopの概要とその核心であるMapReduce処理について説明し，さらに，Hadoop上に構築されたより高位のプログラミング言語を紹介した．複雑なデータ解析アルゴリズムの並列実装にはHadoopのMapReduce機構だけでは不十分であり，高位言語で提供される高い記述性が重要である．

つづいて3節ではRとJaqlの統合に代表される，Hadoop上で利用可能なデータ解析技術・ツールを紹介した．S-plusと互換性をもつオープンソース言語Rは，現在最も広く利用されている多目的のデータ解析・統計処理言語の一つであり，RとJaqlを組み合わせることにより，Hadoop上でのデータ解析プログラム開発がより容易になることが期待できる．

また，4節では，Hadoop上での分析アプリケーションについて整理し，特に金融分野における応用例を

紹介した．4節で挙げたVISAの例などを含め，海外ではHadoopを用いた分析の事例が出てきているようであるが，国内においては，いまのところ大規模な事例は知られていない．著者らの知る限り，実証実験的な事例がいくつかあるだけである．ただし，今後の積極的な活用を現在検討している例はいくつかあり，近い将来に様々な発表や報告が行われることが予想される．

最後に，クラウド上での大規模データ解析の今後の展望についての見通しを述べる．HadoopのMapReduceモデルにおけるプログラム開発環境の整備が進むが，Jaqlなどの高位の言語による開発の割合も増えていくだろう．これらの言語が幅広く普及していくには，Hadoopの高性能をそのまま享受できることが前提になるが，Java処理系の研究で蓄積された技術の適用が考えられ，今後の大きなパフォーマンス改善が予想される．汎用データ解析ツールについては，RのHadoop対応に加え，今後Cognosなど産業界で広く用いられているツールのHadoop対応が進むだろう．実際Hadoop対応がすでに告知されているものもいくつかあり，これまでに蓄積された企業固有のデータ解析アプリケーションのクラウド上での展開が進むだろう．アプリケーションについては，4節で紹介したVISAの事例に続いて国内外での活用が大きく増加していくだろう．その際に，ログなど主としてすでに蓄積した大量のデータを時間をかけて解析するデータマイニング的な活用に加え，センサーなどからの入力を準リアルタイム的に解析するアプリケーションが次第に増えていくことが予想される．実際，金融証券分野におけるマーケット情報の分析・解析などへの活用が検討されている．

参考文献

[1] 第二回国際自動車通信技術展，attt. jp

[2] 中井，“企業システムにおける大規模データの活用とHadoopの動向，”G-CLOUD maganize 2011.

[3] http://hadoop.apache.org/

[4] http://pig.apache.org/

[5] http://hive.apache.org/

[6] http://code.google.com/p/jaql/

[7] “MapReduce : Simplified Data Processing on Large Clusters,” J. Dean and S. Ghemawat, Proc of OSDI '04 (December 2004), pp. 137-150.

[8] http://www.json.org/

[9] “「Jaql」を使ってMapReduceをより簡単に，”米持幸寿，http://codezine.jp/article/detail/5646/

[10] C.-T. Chu, S.K. Kim, Y.-A. Lin, Y. Yu, G.R. Bradski, A.Y. Ng and K. Olukotun, Map-Reduce for machine learning on multicore, In NIPS, pp. 281-288, 2006.

[11] “Ricardo : Integrating R and Hadoop,” S. Das, Y. Sismanis, K.S. Beyer, R. Gemulla, P.J. Haas and J. McPherson, Proc. 2010 ACM SIGMOD Intl. Conf. Management of Data.

[12] “MCDB : A Monte Carlo Approach to Managing Uncertain Data,” R. Jampani, L. Perez, M. Wu, F. Xu, C. Jermaine and P.J. Haas, Proc. 2008 ACM SIGMOD Intl. Conf. Management of Data, 687-700.

[13] http://www-06.ibm.com/software/jp/data/cognos/products/8bi/

[14] J. Cunningham, Hadoop@Visa, Hadoop World NY, 2009. http://www.slideshare.net/cloudera/hw09-large-scale-transaction-analysis/

[15] 中井，“EucalyptusのHadoopクラスタとJaqlでBasket解析をしてHiveとの違いを味わってみました，”http://www.slideshare.net/enakai/eucalyptus-hadoop jaqlbasket-20101203/