難しく考えずに始めようビッグデータ

データが少なくても構わない

黙っていても増えてくる。少し数理論理は必要だが、基本は、相関と分類

やってみると何かが分かる

これが新しい何かの始まり

株式会社シンク・アンド・ゴー
長岡　均

自己紹介

- 1978年　富士通入社。東大宇宙航空研究所・文部省宇宙科学研究所（現JAXA）担当
  計算センターにメインフレーム/スーパーコン、ロケット/衛星追跡・制御用システムを導入
- 1988年　米国ミニスーパーコンピュータの国内販売会社を共同設立
  計算流体力学研究所にて流体解析コンサルティング営業、可視化システムの開発・販売
- 1992年　米国カリフォルニア州サンタクララでC言語用CASEツールの開発・販売
  シリコンバレーでのソフトウェアビジネスを経験し学ぶ
- 1995年　ウッドランド株式会社にて業務システム（主に販売管理システム）の作成・納品
- 2000年　フェアウェイソリューションズにてサプライチェーンマネジメントシステムの開発・導入
  需要予測が必要なため、確率・統計を学びシステム化する
- 2008年　日本ラッドにて組み込み系システム（主にカーナビ）に関わる。クラウドデータセンターを保有していたため、センサーとサーバを繋げたシステムを想定し、ビッグデータを指向する
- 学歴　京都大学理学部　主に数学を学ぶ

なぜビッグデータなのか

これまでの製品やサービスを構成する機能の高度化、製造技術の高度化による生産効率の向上、量産によるコスト削減はこれからも必要なことですが、そこからは大きな『うま味』を期待することが難しくなってきています。その解決のひとつとして注目されているのが、『ビッグデータ』です。これまで処理できないと考えられていた大規模なデータ、これを逆に活用することで新しいビジネスを創り出そうという考えが世界中で広がってきています。

ビッグデータはまだまだ未開の「宝の山」

ビッグデータとは

きっかけ

- 2000年以前は、データは会社や組織の中、または個人のパソコンの中といった閉じた環境にあった
- 2000年以降インターネットのWebページやSNSなど開かれた環境に膨大なデータが蓄積されるようになった
- 米国GoogleやAmazonドットコムがそれに着目し、データを利用することで新しいビジネスを創り出した

米国ガートナグループ　ダグ・レイニー アナリストによる定義　–「３つのV」

- Volume　データのボリューム。膨大なデータが対象
- Variety　定型化されていない様々なフォーマットのデータが対象
- Velocity　データの入出力の速度

私が考えるビッグデータ……スモールデータでもかまわない

- ＩＴを使ってデータを分析・解析し、新しいビジネスや新しいサービス、新しい何かを創り出し、ＰＤＣＡとしてそれを回していくことによって組織体を改革していく活動
- まずは、分析・解析、それとそれらを継続することが重要。データは黙っていても増える

今後重要と考えられる３つのキーワード

「日経ビッグデータ」より

我々を取り巻く社会やビジネスのいたるところで目にするようになるでしょう。
そして、その背後にはビッグデータ技術が不可欠のものになってきている。

IoT Internet of Things

「モノのインターネット」

M2M

- 産業機械や自動車、家電など、様々な対象がインターネットを通じてデータの取得、制御ができる概念
- センサーの高性能化・小型化は、低廉化を引き起こし様々な機器に内蔵され、それらがインターネットを介してサーバ（クラウドセンター）につながってきている。こういった膨大な数のセンサーが、自動的に秒単位・分単位でデータがサーバに集約されると、そこから様々なアプリケーションの誕生が見込まれる
- 誰もが持っているスマートフォンにもGPSや動態センサーなどのセンサーが内蔵されており、人間そのものが「動くデータ収集機」になる

ディープラーニング

深層学習、機械学習の一種

- 「潜在的にはすべてのホワイトカラーの労働を代行する汎用技術」とも言われ、自動運転や他社理解、自動翻訳などが完璧にできるようになるなど、社会的に大きなインパクトをもたらすものと期待されている
- 米Googleとスタンフォード大学共同で、YouTubeからの大量の画像をコンピュータ上の仮想ニューロンを層状に配置したニューラルネットに入力。事前に何も教えなくても、「これは、猫である」と認識
- 米Google、2014年英ディープマインドテクノロジーズを買収するなど、米国企業を中心にこうした研究に大きな投資が始まっている

クォンティファイド・セルフ

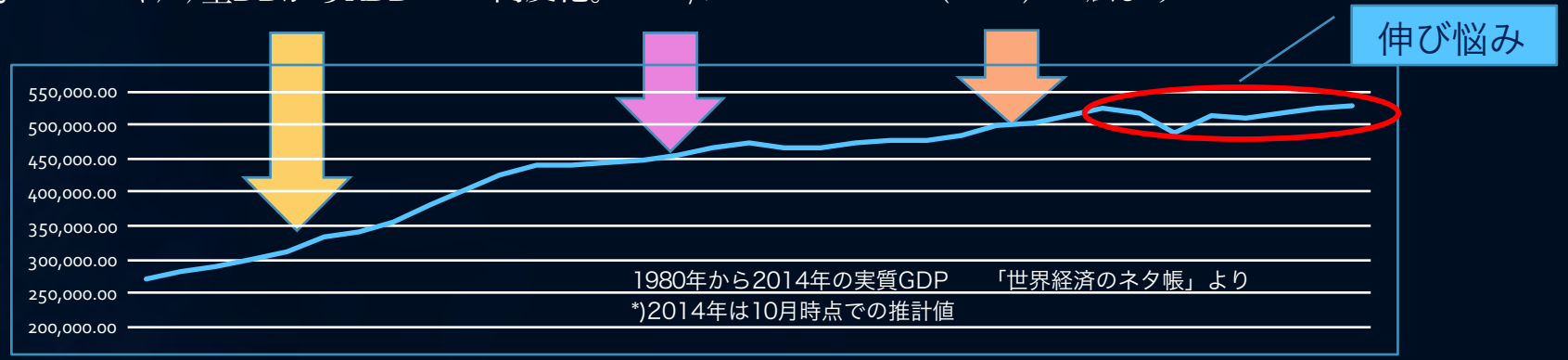
「数値化された自分」

生き方の見直し運動

- 各種センサーを組み込んだツールを使って個人行動や健康状態、睡眠などを数値化して「見える化」。より良い生き方につなげる
- 自分の行動や健康状態が見られるデジタルツールが次々に登場。スマホ用アプリも多数登場。米アップルやGoogle（「Google Glass])など参入が確実視されている

マイナーですが・・背景　これまでのコンピュータシステムの導入

	1970年代	1980年代	1990年代	2000年代	2010年代
ＩＴ	汎用ｺﾝﾋﾟｭｰﾀの登場 国産メーカー参入 バッチシステム	メインフレーム全盛期 TSS、分散処理ｼｽﾃﾑツリー型ネットワーク	UNIXオープンシステム PCの個人への普及 インターネット、RDB	オープン系サーバ定着 ｲﾝﾀｰﾈｯﾄの普及･高速化 携帯電話のi-mode接続	スマホの普及・高速化 クラウドデータセンター Ｍ２Ｍ、ビッグデータ
目的	会計システムが中心 手書きﾃﾞｰﾀの電子化 伝票・帳票出力	業務システムの拡大各業務システムは個別ﾈｯﾄﾜｰｸ型DBからRDBへ	ERPの登場と中堅・中小企業の業務システムの高度化。Web/メール	ERPによる業務ｼｽﾃﾑの統合。EDI企業連携 EC（BtoC）の広まり	



2000年代までは、経済が右肩上がりであったため、極端に言うと「やれば売れた」という時代。そのためＩＴは、業務コスト削減と業績の早期把握を主目的として導入されてきた。これまでのやり方を変えることなく、「量的」効果を生み出すものとしてＩＴは導入されてきた

これもマイナーですが、日本的視線でいくと　現在の状況

- 単純な意味での技術的優位性確保が難しくなってきている
- グローバル・オペレーションが可能になり、生産は中国・東南アジアに移転
- 「良ければ、高くても売れる」時代から「良いものが安い」時代に。内容・仕様の明確なものはとりわけ顕著
- 消費者・買い手がマーケットを主導。ＥＣサイトからの購入が急増
- 「物的豊かさ」から「精神的豊かさ」への意識の変化。「安全・安心な暮らし」と「環境」への配慮を重視

- これまでにはなかった新しいビジネスや新しいサービス、新しい仕組みが必要

「そのネタは、データの中にいくらでも潜んでいる」
と考えられるようになってきている

- いきなり全く新しいビジネスを創出（あえて言うと米国型）
- 身近なところで課題を発見し改善（あえて言うと日本型）

米国「そこにはチャンスがいくらでもある！」と考える傾向

Google、Amazonドットコムなど多くの事例がある（割愛）

ビッグデータは新しいビジネス・サービス・仕組みの開発活動

製造業の技術開発・製品開発

製品企画

設計・シミュレーション・評価

製造ライン設計・構築

調達

製造

物理方程式
力学系
電気系
化学系

休むことなく何度も繰り返される

製品開発では、当たり前のように実施されていた仮説・設計・シミュレーションがビジネス分野ではなされていない

ビジネス開発

販売企画

シミュレーション・評価

販売ライン設計・構築

調達

販売

これまで

価格・広告・イベントなどマスマーケティング

サマリー化された過去の販売実績

取引先との関係重視

週または月単位。場合によっては日単位

お客さんが選択したものを販売

すべきことはいくらでもある

様々なケースを評価し、因果関係の推定

全データを対象に何度でもシミュレーション

販売状況のリアル・タイムモニタリング

１日複数回も可能

お客さん毎に販売施策を実施

ビッグデータ
確率・多変量解析
相関・パターン認識

主な解析・分析の内容　アプリケーションとしては以下のものが高度に融合している

0．カウント	：	まずは数え上げる
1．平均値と分散	：	データの傾向を知る。そこから具体的なデータ値に対する確率を知る
2．相関	：	２つのデータ群についての関連性を知る。その度合いに応じてグルーピング。同質性を知ることによって、例えば、「人」であれば、その人のグループ内でとられている行動から、その人の行動を推定・予測をする
3．回帰分析	：	知りたい結果（目的変量）について、それに影響を与える要因（説明変量）を最小２乗法を使って一次式を予測式として算出。知りたい結果が２値の状態（「合格」か「不合格」）の場合は、確率として算出（ロジスティック回帰）
4．ベイズ統計	：	原因によって引き起こされる結果の確率が分かっている場合、逆に結果から原因の確率を求めることができる。検索エンジン、スパムメール検出、障害対応で使われている。原因結果とその確率のセットを多段に組むこと（ベイジアンネットワーク）によって複雑なケースにおいてその発生確率を求めることも可能
5．パターン認識	：	ある事象を特定するデータ波形の抽出と他のデータ群でのその波形の同定。 株価の分析・予想。医療分野への適用。

例：回帰分析による予測

例えば、新たにレストランを出店するが、どのくらいの売上が見込まれるか

1. レストランの売上（目的変量）を左右する要因はいろいろあるが、その町の人口、フロアサイズ、駐車場の駐車台数、駅からの距離をもとに（説明変量）売上を推定する
2. 他店舗での実績は、表の通り
3. 他店舗での実績から、目的変量である売上を上記説明変量の１次式として求める
4. この場合、各説明変量に掛かる係数を求める必要があるが、最小２乗法（求める１次式と実際のデータとの誤差の総和が最小）により求める
5. 求まった１次式に、今回出店しようとしてるレストランのデータを入力し、売り上げ予想とする

予測式を　$\acute{Y}=a^*X+b^*U+c^*V+d^*W$　とおく。これに右表のデータをあてはめたものをそれぞれ $\acute{Y}i$ とする。

$\acute{Y}i=a^*Xi+b^*Ui+c^*Vi+d^*Wi$　この１次式から見たﾚｽﾄﾗﾝiの売上予測これとYi（実績値）との差が予測誤差となる。

$S=\sum(\acute{Y}i-Yi)^2$ が最小となるa,b,c,dを求める。

	売上	人口	ﾌﾛｱｻｲｽﾞ	駐車台数	距離
レストラン１	Y1	X1	U1	V1	W1
レストラン２	Y2	X2	U2	V2	W2
レストラン３	Y3	X3	U3	V3	W3
レストラン４	Y4	X４	U4	V4	W4
レストラン５	y5	X5	u5	v5	W5

例：レコメンドエンジンで使われている相関

CDショップ　1．A君、Bさん、C君、D君が同じようなCDを購入していた
2．A君、Bさん、C君、D君の好みは似ていると判断
3．A君の除く他のメンバーは、XというCDを買っているにもかかわらず、A君は買っていなかった
4．A君もXを買うのではないか

ポイントは、「同じようなCDを購入」を適切に数値化すること

購入履歴より、A君、Bさんのどちらかが買ったCDを抽出

	CD1	CD2	CD3	CD4	・・	CDn-1	CD n
A君	0	1	1	1	・・	0	1
Bさん	1	1	0	1	・・	1	1

1:そのCDを買った
0:そのCDを買っていない

以降、次のように表記
Ai：A君のCDiの購入状態
Bi：BさんのCDiの購入状態

$$\frac{\sum A_i * B_i}{\sqrt{\sum A_i^2} * \sqrt{\sum B_i^2}}$$

を求める。A君、Bさんが全く同じCDを買っている場合、1
すべて別のCDの場合、0
この関数は0から1の値をとる。ちなみにこの関数は「コサイン関数」と呼ばれる

- コサイン関数値に対して、「同じさの判定」をパラメータで与えることで、A君、Bさんは似ているかどうか判断パラメータを「1」に近くすると人数が減るが、「同じさ」が強いメンバを抽出
- この方式では、「意外性」の小さなレコメンドになるという問題がある

相関を計算する場合ケース数がnC2と膨大な計算量になる

人間が、グループ分けする場合、「大・小」や「大、中、小」などせいぜい5グループ程度だろう。
システムであれば、100や1,000グループに分けることができ、「同じ」度合いの緊密性を上げることができる。

簡単にベイズ統計に触れておきます

1. 「ベイズ」という名前は、高校の教科書で例えば、「くじ引きで、Aさんが当たりを引いた後（くじ、当たりくじそれぞれ１つ減っている）、Bさんがくじを引いた時の当たる確率」といった条件付確率（「ベイズの条件付確率」という）ということでてた。
2. ベイズは300年前の人で、ベイズの確率論がそれ以降の確率論の主流だった
3. 1900年前半に推計統計学を確立したネイマン、ピアソンにベイズ確率は否定された。以降ネイマン、ピアソンの推計統計学が主流になり、検定・推定実績をあげてきました
4. 近年、ベイズ確率は∞の試行ではネイマン、ピアソンの確率と矛盾しないことが証明され、現実の中での経験に合致していることから見直され、採用されてきています
5. 両者の確率論の大きな違いは、ネイマン、ピアソンの確率では、「サイコロの１の目が出る確率は1/6」、「コインの表の出る確率は1/2」と決まっているということが前提ですが、ベイズの確率では、それは不明という点です。そのため、最初に「まずは、1/4でも1/3でも決めてしまおう（事前確率：これが数学的に曖昧）。サイコロを振った結果で確率を見直そうとなる
6. 例えば、当たりくじがどれだけ入っているか分からないくじを引くとき、「５回も引けば１回は当たるだろう」といったことを想定するだろう。それで１回目に当たりを引くと、「結構当たりが入っているな」と感じ、２回目も当たりが出ると「これは当たりが多いぞ」となるだろう。ベイズの確率はこのように確率が事前確率・事後確率共にを変遷する
7. ちなにベイズの定理は次の式で、P(B|A)がP(A|B)で表されることから、Bを原因、Aを結果とすれば、原因に対する結果の確率が分かっていれば、逆に結果から原因の確率が求まる点がポイント

$$P(B|A) = \frac{P(A|B) * P(B)}{P(A)}$$

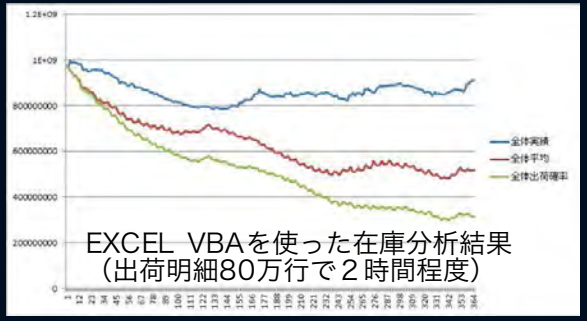
スパムメールの３割に「A」という単語が使われている。メールに単語「A」が含まれているがスパムメールの確率は？

分析の目的と道具立て

知見を得る　分析者がプログラミングをすることなく、様々な分析がツールとの対話形式でできる。

EXCEL
ほとんどすべてのPCに入っており、基本的な操作は皆知っている。100万行入力でき、DBアクセス・データ抽出、VBAプログラミングができる。また数学関数や統計関数も網羅されているので、必要な分析は十分できる

SPSS
データマイニングツール。データを入力するだけで必要と思われる分析計算は自動的にしてくれ、グラフィカルに可視化してくれる。何を分析したいか、それに応じた関数は何かを一切気にする必要はない
辞書が感情辞書など用意されており、テキストマイニングも一通りのしてくれる



EXCEL VBAを使った在庫分析結果
（出荷明細80万行で２時間程度）

業務で使う　分析して知見を得ることも重要だが、それを日常業務の中で稼働させる必要がある。ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ要

JAVA
多くのエンジニアが使用している言語。JVMによりプラットフォームに依存しない。Web環境などのフレームワークも充実
PCの高速化、大規模メモリー化から、10億件程度のデータのハンドリングであれば、処理時間は10分程度か
実測例）乱数を使ったPOSｼﾞｬｰﾅﾙﾃﾞｰﾀ生成　総明細数：328百万、サイズ：14.4GB、処理時間：6分31秒

Hadoop（後述）
専用のハードは不要で通常のパソコンで分散処理クラスターが構築できる。Apacheで公開されているので誰でも入手可能
基本的にはバッチシステムのため、フロントシステムが必要でバックエンドで大規模計算エンジンとして使用
Javaがメインだが、種々の言語に対応。しかし、いずれもMap/Reduceという記述スタイルに従わなければならない

Erlang
米国Ericsonがフォールトトレラントシステムの構築を目的に開発した関数型言語。Hadoopのバッチ処理に対して、リアルタイム処理向け。例えばプロの将棋士に対して700台のサーバで2億手/秒を計算し勝っている。関数単位にプロセスを生成するため、実行中多数のプロセスが発生し、クラスター内の任意のサーバで非同期に動き、メッセージ通信により同期をとる

Hadoopについて

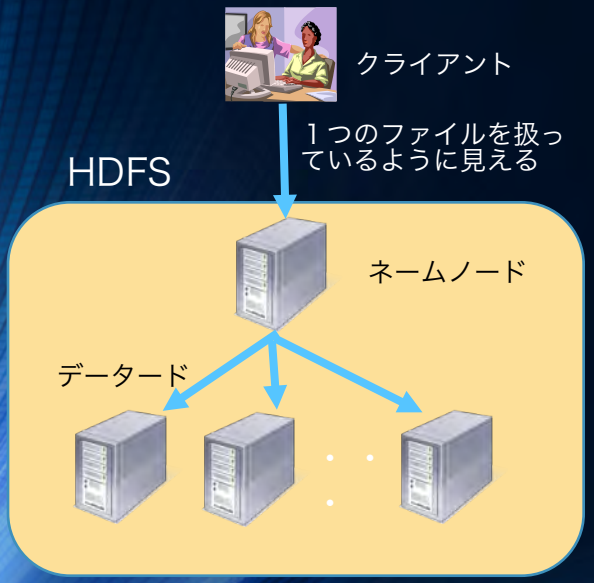
米国Googleが自ら開発した検索エンジンの技術について、プログラム自体は公開しなかったが、その仕様を論文で公開（2003年、2004年）。その公開された仕様をもとに、Doug Cutting氏らが中心になって開発。現在、Aapcheのトップレベルのプロジェクトとして公開されており、誰でもが入手可能なため、ビッグデータの基盤として定着

1. 大規模並列分散システム
2. 高価な専用ハードを必要とせず、廉価で市販のPCで分散処理クラスターが構成でき、オープンソースなため、費用をかけずに環境が作ることができる（ベースOSはLinux）
3. スケールアウト思想のため、必要に応じ処理ノードを増やすことが可能。2008年2月 Yahoo!が稼働させている検索インデックスが10,000コアのHadoopクラスタで生成されていることを公表
4. データのレプリケーションや稼働ノードの監視など、障害の検知および対策が充実しており、全体として安定稼働が期待できる
5. Hadoop自体は、分散ファイルシステムHDFSとフレームワークMapReduceで構成されており、データベースを直接サポートしていないが、周辺にHiveといったSQLライクなプログラミング環境、MapReduceというHadoop用プログラミングを隠ぺいするPigなど多くの周辺ソフトが出てきている（多いため、一つひとつ調べるのは大変）
6. 分散ファイルシステムHDFSのファイル思想としては、大量のデータをハンドリングすることを前提としているため、読み込みと書き込みのみの処理しかなく、データの一部更新はできない。
7. 処理は、MapReduceという考え方に従わなければならない。Map処理は、ノードに分散されたファイルからデータを読み込み、それぞれのノードがキーとバリューのペアを作り、その後、Reduce処理が、キー単位に集約されたデータを処理する
8. バッチ処理のため、基本的にはバックエンドの計算エンジンとして稼働させ、その結果をフロントエンドでDB化してリアルタイム処理に対応するという構成になる

Hadoop技術 その１ 分散ファイルシステム（HDFS)

高スループットを目指した分散ファイルシステム

インテリジェントなRAIDとも言えるが、処理系のMapReduceと結びつき、ファイルを構成するブロックをMapReduceが直接処理する（データ・ローカリティ）「大量データをハンドリングする」ということが前提のため、応答性よりも全データをハンドリングするスループットを重視。ランダムアクセスはできない。また、部分更新もない。読み込みと書き込みのみ



ネームノード

- 分散ファイルの一元管理。属性情報やファイルブロックの所在管理。アクセス要求への応答性を上げるため、メモリで管理
- 使用状況管理
- データノードの死活管理
- クライアントからのアクセス要求の受付

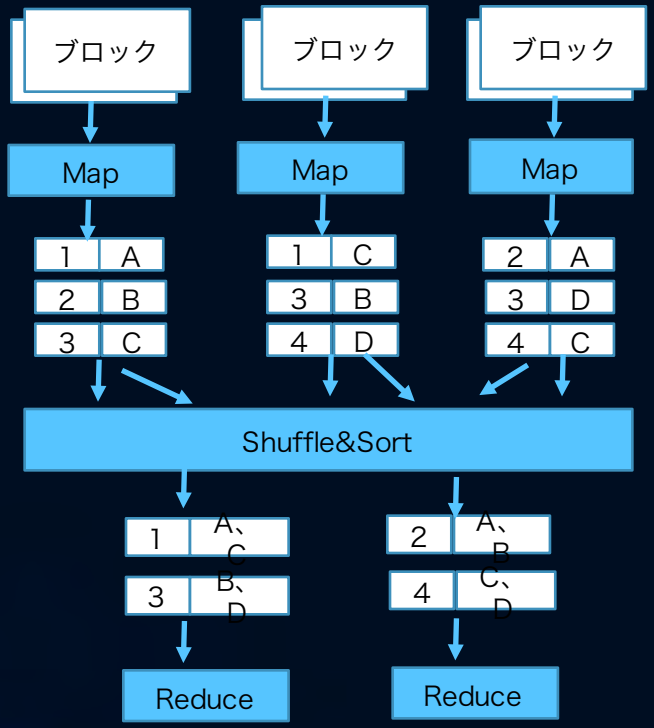
データノード

- HDFSでは、ファイルは、64MB単位に分割（ブロック）され、ブロック単位にデータノードで保管される
- ブロックレプリケーションとラックアウェアネス。データノードの障害を想定し、ブロックは複数のデータノードに保管される（デフォルトは３つ）　全レプリカがすべて同一ラック内に存在しないよう配置をコントロール

Hadoop技術 その2 フレームワーク（MapReduce)

処理の単位を相互に関連しないデータに分割（キーとバリューのセット）することで、並列処理を実現

例えば、店舗の売上を集計する場合、「山田君はA店」、「斎藤君はB店」、「吉田さんはC店」と分担することで、並行して作業を進めることができる。こういった考えをとっている。この場合、A店、B店、C店が「キー」で、それぞれの店の売上伝票が「バリュー」



Map処理

- HDFSに格納されたファイルは、データノードにブロックとして保存されている。Map処理は、ブロックを保持するタスクノードがそのブロックを直接処理する（データローカリティ）
- 処理対象は、１行単位でキーとバリューのペア作る

Shuffle&Sort処理（Hadoopが自動的に実行）

- 複数のタスクノードのMap処理で作られたキー/バリューのペアに対して、キーごとにデータを集約し、ソートをかける
- タスクノード間でのデータ転送が発生するため、ボトルネックの要因になりえる

Reduce処理

- キーごとに集約されたデータ群（バリューのリスト）について、必要な処理を実行

Hadoop構築・利用上の課題　その１

Hadoopは、廉価な市販ハードウェアで、場合によっては数万台のクラスタを構成して大規模データ処理を実現することができますが、それを構築し、利用するためには習得すべき技術が多い

1. 適した領域
   - 大規模なデータを裏方でバッチ処理で愚直に動かす、ということに向いたシステム。計算結果もDBに格納されるわけではなくCSVファイルのため、人がインタラクティブに何かしようという場合は、計算結果をDBにロードし、フロントで動かすことになる
2. システム設計
   - 計算のボリュームが事前に見積もることができないため、必要となる計算資源が分からない。
   - スケールアウト：「足らなければ追加できる」の考え方のため、同程度の能力のサーバをどういう構成で何台用意しておけばよいのか分からない。データ転送のオーバヘッドを考慮すると、矛盾するようだが、高性能なハードウェアを台数少なく用意した方が良いか・・
3. プログラミング
   - Javaのフレームワークではあるが、MapReduceの記述は必須。ﾌﾟﾛｸﾞﾗﾐﾝｸﾞ論理をMapReduceの考え方に置き換えるのには、かなりこなす必要がある。通常の処理手順ではなく、キー/バリューのペアをどのように変遷させるかがプログラム論理設計のポイント
   - Javaプログラムで、元々キー毎に処理をしている場合は、比較的Hadoopへの移行は容易だが、そうでなければ全く違うプログラムを作成する必要がある
   - EclipseでMapReduceプログラムが動くかどうか？入力ファイルがHDFSであることから、開発マシンはLinuxになる
   - Hadoopはバックグラウンドで計算エンジンとして稼働するため、その結果を使う処理については別途フロントサイドで作成する必要がある

Hadoop構築・利用上の課題　その２

4. デバッグ/テスト
   - プログラミングで挙げたが、EclipseでMapReduceプログラムが動くかどうか？ということで、Hadoopのメッセ-ジに頼らざるを得ない
   - ジョブの実行をトレースできない。データにより一部ノードでAbortした場合、どのデータで落ちたのか特定しづらい
   - 小さなデータで、MapReduceを一つの単位で区切りながら確認する
5. 性能評価
   - ハードウェア単体での性能に対して、台数分を考慮した性能が出なかったことがあった。環境の設定によるチューニング、Hadoopの持つ機能によるチューニング、プログラミングによるチューニングなどがあるが、それぞれ習熟する必要がある
   - 所定時間内にジョブが完了しない場合、「このくらいでも増やしておこうか」ということになり、その上で評価することになる（スケールアウトの考え方）
6. メンテナンス
   - Hadoopには障害に対する機能がいろいろ用意されているが、ハードの切り離しや復旧などの操作を知っておく必要がある
   - 多数のノードの状態を一元管理する機能があり、それらを習得する必要がある
   - 数百台、数千台のサーバを維持するためには、運用要員を用意することは別に問題ではないかもしれない
7. その他　周辺ソフトがいろいろと出てきており、逐次調査・評価をする必要がる

いろいろ課題はあるが、これまで不可能と考えられてきた処理が可能になる、新しいビジネスを作ることができることを考えれば、それだけの価値は十分あるだろう。いずれにしても経験を積み、技術を蓄積することが重要

ビッグデータ構築にあたって（従来のシステムエンジニア技術は役に立たない）

1．ビッグデータは生き物

- ◆これまでの業務システムや基幹システムのように、５年、10年変わらず使うというものではない。常に改良しながら成長させる必要がある
- ◆そのため旧来の要件定義から始まり、設計、プログラミング、テストといったウォータフォール型は適さない

2．始めることが重要

- ◆「それをやって、どれだけの効果を出すのか？」といった、固いことは考えない、言わない。結局、「やらない」ということに押し込むだけ
- ◆より大きな問題は、「やろうというテーマが出てこない」こと。トップ・上層部の「やる姿勢」がテーマを生み出し、人を作る

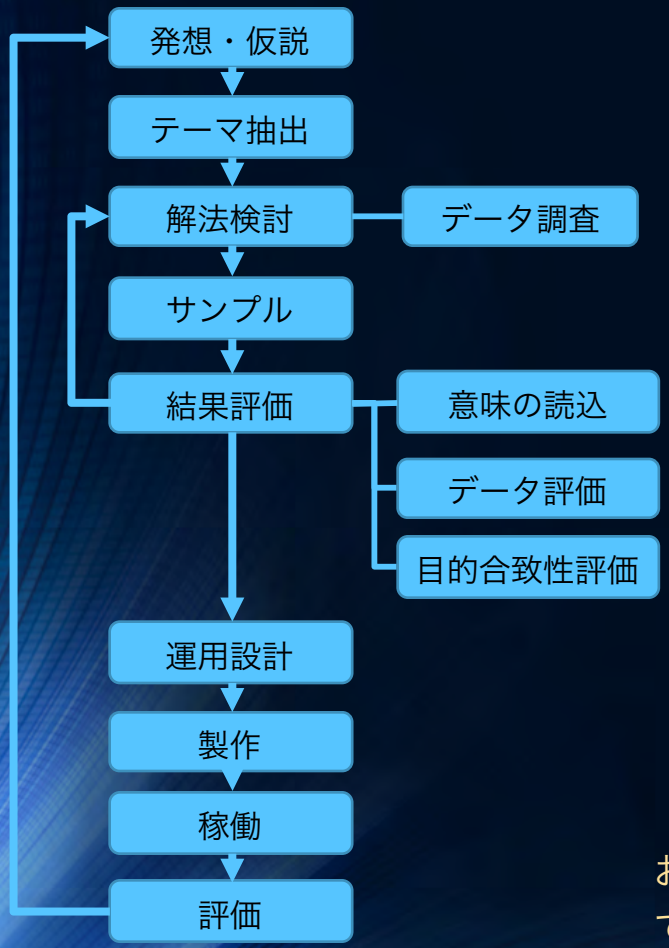
3．自らが作るという姿勢

- ◆ビッグデータ自体が会社に新しいビジネスを作るもの。外部からの助言・情報はあっても、これ自体を外部に任せることはできない。

4．人材の育成

- ◆これまでのプログラミング技術を主としたITエンジニアではなく、自らが問題を設定し、解決立案、テスト・評価ができる新しいタイプのエンジニアの育成が必要。難しいか？しかし、ビッグデータは、そういった職場環境に変えることもできる
- ◆確率・統計など数理論理に慣れた人材育成が必要。実際には教科書のようにはいかないことが多い

ビッグデータ構築の手順



現場業務は現場で起きている様々な音を知っている。問題、意見、疑問。それらを汲み取る

どのようにしたらよいと考えるか、その結果どうよくなると考えるか（定性的で構わない）からテーマを抽出

具体的にどのように解決するか検討。解決に必要なデータの入手の調査

評価のため、場合によってはプログラムを作ることも必要。BIツールを利用する場合、その準備

データを投入。結果についてその背後にあるものを考慮。どういうことがどういう形で分かるのかなど意味的評価

実際のデータは多岐にわたるため想定した結果にならない。どういうデータを意味あるデータとするか検討

「どう良くなると考えるか」ということが、数値的に表現され、以降の処理に反映されるか評価

すべてのケースが想定した結果にならない。想定外のケースの頻度・ボリュームの評価、それへの対応の検討

お客様のエンジニアがこの手順をフォローできるためには、人材教育・育成が必要であり、それも勉強会および作業を通じて指導 (OJT) も行う必要があります。

ビッグデータ構築に必要となる技術

1. 感じている問題を人に問題と理解できるよう表現できる技術。解を提示し納得を得る技術
2. システム内でそれをどのように表現できるか考える技術
3. どのような数理論理で実現するか設計する技術
4. 結果について、意味が分かり、その背後の要因を想定できる技術
5. データの実態を把握し、それに対する対応を設計できる技術
6. ベースとして確率の考えを使うが、例えば「2σ」以上の場合、どのような影響があるか、対処するか設計できる技術
7. システム実装技術
8. 運用技術

データサイエンティストの要件

2013/12/11の日経新聞「93%の経営者がビッグデータの専門家不足に悩む」という記事があった。自らが投資決定をできずにどうしたことだろうか。ビッグデータもスモールデータも自社にとっての新しいビジネス創出活動と考えれば、トップデシジョンは不可欠。トップの意思とそれを裏付ける投資は、人材育成の前提で、それが明確ならば人は育つ。

最後に

1．データは「未開の宝の山」

ビッグデータもスモールデータもデータを分析すれば、必ず何か宝を見つけることができる。「情報を持っている」、「予測できる」（確率になるが）その意味と重要性を改めて認識すべき時に来ている。都合の良い情報だけを外部からとろうとしても、それは無理だろうまた、今後確実にデータは増え続ける。その活用のバリエーションを考えれば、データはいくらでも開拓できる「未開の宝の山」と見なせる。

2 「新しい何か」にもっと重きを置くべき時代になった

アメリカ、とりわけシリコンバレーでは、「新しい」ということが最も評価され、「新しい」ことの評価もできるようになっている。一方、「実績あるもの」は、分かっているものとしてコストパフォーマンスが重視される。こういったことが、文化として定着している。一方、日本では、「実績」が最優先され、「新しいもの」は「危ない」と考えられがち。こういったことを変えていくべき時に来ている。このまま行くと、アメリカの一人勝ちになり、気質的に中国もアメリカに近いため、中国の後塵を拝することなりそうだ

3．ビッグデータに前向きな投資を

2013/12/11の日経新聞「93%の経営者がビッグデータの専門家不足に悩む」という記事があった。自らが投資決定をできずにどうしたことだろうか。ビッグデータもスモールデータも自社にとっての新しいビジネス創出活動と考えれば、トップデシジョンは不可欠。トップの意思とそれを裏付ける投資は、人材育成の前提で、それが明確ならば人は育つ。