

# 集合知および Web2.0 の国際開発への応用

山下 幸彦

東京工業大学・国際開発工学専攻

平成 26 年 5 月 14 日

# 発表概要

- 集合知 (Wisdom of crowds, Collective intelligence) の一般的な内容
  - Diversity (多様性)
  - Independency (独立性)
  - Decentralization (分離性)
  - Aggregation (集約性)
- インターネット
- Web
- Web 2.0
- 研究室での取り組み
  - 集合知による説明システム
  - 集合知による半機械翻訳システム

# 群集の狂気 (Madness of crowds)

## 集合狂気 (Collective madness)

### Charles Mackay

- Extraordinary Popular Delusions and the Madness of Crowds, 1841
- 群集は「ばか (Dumb)」か「狂っている (Crazy)」かその両方

### Henry David Thoreau

- 大衆 (mass) は優秀なグループの標準に追いつくことはない。
- その最低レベルまで悪化する。

### Gustave Le Bon

- The Crowd: A Study of the Popular Mind, 1885
- 群集の知性は、その中の個々の個人に劣る。

# 集合知における問題

- 集合知：The wisdom of crowds, Collective intelligence
- 集団 (社会) が知的に振る舞うか？
- 問題の分類
  - Coordination problem (協調)
  - Cooperation problem (協力)
  - Cognition problem (認知)**
- ここでは，Cognition problem に関して詳しく述べる。  
(Wikipedia のようなものは，Cognition problem)
  - 集団は個人より優れているか？
  - 大きな集団は小さな集団より優れているか？
- とりあえず，Coordination problem, Cooperation problem についても簡単に説明する。  
(社会学がいらっしゃるとかなり僭越なのですが。)

# Coordination problem

- 例：交差点で人とぶつからない。
- 全体として知があるように振る舞う。
- 個人は他の人の判断も予測して行動する。
- **The great equilibrium (経済学)**
- レストラン：店が混むと不快  
相手の行動を予想して、最適解で均衡する。
- 個人が**自分に対して最適化**しても，**社会的に最適**なアウトカムが出力される。
- レストランの例も，全体や他人のために予想して行動しているのではない。
- (これでうまく行けば，世の中楽なんですけれど)

# Cooperation problem

- Coordination に比べては，社会的な問題
- 人が歩道を歩く。税金を支払う。家の前の道の雪かき。
- **Collectively rational**(集合的合理)にはならないことが多い。
- Ultimate Game:
  - 10ドルを2人で分ける。1人が金額を決定できる。
  - もう1人は，拒否(両方とも1ドルももらえない)できるだけ。
  - 基本的には，5ドルずつになる。
  - 学業成績を提示する，結果が変わる。
- Public goods experiments (E. Fehr and S. Gächter)
  - 4名。1 token 支出すると全員に 0.4 token 戻る。
  - 20回実行する。
  - 全員が1 token 支出すると 32 token 得る。(Collectively rational)
  - 全員が全く支出しないと 20 token 得る。
  - 最初は多くの学生が 0.5 token ぐらいを支出する。

# Cooperation problem

- Public goods experiments 続き
  - 支出しない学生が存在する影響で、最後は 70~80%の学生が支出しなくなる。
  - 0.3token 支払うことで他人の 1token を減らすことができるようにする。  
(自分も損をするが、罰を与えることができる)  
⇒ ほとんどの学生が支出するようになる。
- 法律では cooperation を生み出せないが、cooperation をより成功するようにすることができる。
- ビジネスに広い信頼を得ることは重要  
(狭い強い関係はビジネスに発展に不利)
- 市場がある地域の方がより人々が社会的になる。

# Cognition problem

- 個人でも認知できる問題
  - 状態の推定
  - 原因の推定
  - 未来の予想
  - 複雑なものを作る
  - 知識をまとめる
- これらの問題の解決にあたる主体：
  - 個人
  - 少人数のグループ
  - 集団
- 集合知 ⇒ 集団による結果のほうが良い。  
(Collective madness と反対の主張)

## 集合知の例 (その 1)

### Hazel Knight

- 1920 年代
- クラスの学生による温度推定
- 学生の意見による平均値
- 推定値 72.4 度 (実際 72 度)

### Kate H. Gordon

- クラスの学生による瓶の中のゼリー粒の個数推定
- 学生の意見による平均値
- 推定値 871 度 (実際 850 個)
- 平均より良かった学生は, 56 人中 1 名
- 平均値より推定精度が良い学生は, 実験ごとに異なる。

## 集合知の例 (その 2)

### Normal L. Johnson

- Collective solution
- 迷路における方向選択 (コンピュータシミュレーション)
- 個々の平均 12.8 回でゴールに到達
- 決定を平均した場合 9 でゴールに到達

### 機械学習

上の例に関連して (私の専門の一つ) :

- アンサンブル学習 (文字認識)  
異なる種類の学習器の結果の多数決
- AdaBoost (デジタルカメラの顔検出)  
数多くの弱学習機 (重み付き平均)

## 集合知の例 (その3)

### Google

- 1998年
- Page Rank (単語で検索したページを表示する順番)
- そのページを参照先とするリンクをもつページ数

### Iowa Election Markets (IEM)

- 選挙の勝者を予測する。
  - 当たれば1ドルが得られる。
  - 外れても損失なし。
- 米国の国の選挙で得票率を1.37%の誤差で予測した。
- どの世論調査よりも正確

### Wikipedia

- みんなで作る百科事典 (詳しくは後で)

## 集合知の例 (その 4)

### 多数の部品メーカーを使った大量生産

- 部品も自社製が多かった。
- Olds Motor Works (1903 年):  
様々な部品を様々な会社から調達した最初の自動車会社

(⇔ 最近の日本の半導体メーカー)

### スペースシャトルチャレンジャー事故時の株価

- Rockwell International: メインエンジン
- Thiokol : 個体ロケット
- Martin Marietta : 外部燃料タンク
- 1 時間で 3 社とも 6% 下落した。
- その日の終わりに Thiokol 社だけ 12% 他は 3

# 集合知

- Douglas Richard Hofstadter (1979 年) 認知物理学
- Peter Russell (1983 年)

## 群衆の知恵 (James Surowiecki, 2005 年)

少数の権威者による意思決定・結論や情報よりも多数の意見の集合によるものの方が役に立つ, あるいは正しい結論や予測につながる。

関連する単語 : Bees algorithm, Cellular automaton, Civic intelligence, Collaborative filtering, **Collaborative human interpreter**, Collaborative innovation network, Collaborative intelligence, Collaborative software, Collective action, Collective consciousness, Collective decision-making, Collective effervescence, Collective memory, Connectivity (graph theory), Crowd psychology, Crowdsourcing, Customer engagement, Cybernetics,

# 集合知

Dispersed knowledge, Distributed cognition, Enterprise bookmarking, Global brain, Global Consciousness Project, Group behaviour, Group mind (science fiction), Facilitation (business), Facilitator, Human-based computation, Hundredth monkey effect, Information Routing Group, Keeping up with the Joneses, Knowledge ecosystem, Library, Library of Alexandria, Meme, **MIT Center for Collective Intelligence**, Noosphere, Open-space meeting, Open source intelligence, **Prediction Markets**, Preference elicitation, Recommendation system, Smart mob, Social commerce, Social information processing, Stigmergy, Superorganism, Swarm Intelligence, Think tank, **Wiki**, **Wikipedia**

# 集合知の成功条件

- Cognition problem において，集合知が成功する条件
- 多様性 (Diversity) :  
様々な参加者の意見が表明できる。
- 独立性 (Independency) :  
他の参加者に影響されない。
- 分散性 (Decentralization) :  
分散して意思決定が行われる。
- 集約性 (Aggregation) :  
最終的には，意見をまとめることができる。

これらが満たされていないと，

- Collective madness に陥る可能性がある。

## 多様性 (Diversity)

- コンピュータシミュレーション (Scott Page)
  - smart な人々によるグループは, smart な人々 + not-so-smart な人々によるグループにより全体としては smart でなくなる。
  - smart なグループは意見を似せてしまう。
  - 少ない知識でも different skill を持っている人を加えることは, グループのパフォーマンスを向上させる。
  - どんなに少なくとも新しい知識は, グループをより smart にする。
- 専門家は過信していることが多い。
- ただ, 専門家に限らず大体の人が過信している。  
過信が重要な場合もある。(後述)
- (予測で当たるのは天気予報ぐらい。)

# 多様性 (Diversity)

- Smart な人はいるかもしれない:  
Warren Buffett は 1960 年代より S&P500 より良かった。  
そのような人をあらかじめ見つけることは極めて難しい。
- Diversity は不確実な未来の予想と行動の決定に重要
- 小さなグループが知性で大きなグループにかなうわけではない。
- 小さいグループは一般に Diversity におとる。  
→ 小さいグループの方が Diversity がより重要
- ただし、何もわからない人 (例えば幼児) を加えて Diversity は生まれない。  
様々なバックグラウンドを持つ人々が重要である。

(国際開発工学専攻は Diversity の意味では最高)

## Information cascade (情報連鎖)

- 情報連鎖は独立性を失わせる。
- 誰が成功するとその情報を元に皆がまねする。
- ニホンザル (Imo)
  - さつまいもを洗う。
  - 麦と砂を水で分ける。
  - 両方とも 2~3 年後にはほとんどのさるが行う。
- スクリューの標準化：アメリカ海軍が採用  
Early adopters の影響が大きい
- まねは良いアイデアを広げる強力な方法
- 発明は個人の才能である。
- しかし、数多くの発明物からの選択は **collective** なこと  
(適正な技術が選ばれるとは限らない。)

# Information cascade の問題

- 働き蟻の 1,200feets 程度の輪
  - 働き蟻が前の蟻のあとについていくために生じる。
  - 大量の働き蟻を目的地に移動させるためには有効
- 1990 年代の IT バブル
  - Internet の通信の伸びは年間 10 倍を越えていた。(1996 年)
  - 経営破綻した
- Plank-road (1940 年代 New York)
  - 民間会社がカナダ産の木を敷いた道を作った。
  - 土の道より良かった。
  - 通行料を徴収し、初めは成功した。
  - 維持費が高く長く続かなかった。
  - NY 州だけで 352 社が同様の有料道路を作った。

# Groupthink の犠牲

- **Groupthink** も独立性を損ねる。
- キューバ革命
  - 1,200 人による革命
  - ケネディのグループは成功すると思わなかった。
  - CIA にもキューバデスクにも相談しなかった。
- カードの大きさ (Solomon Asch)
  - 同じ大きさでもクラスの他の人が大きいといえ、70%の学生が大きいと答える。
- 5 人の人が空を見ると 45%の人が空を見る。  
人を増やすと 70%の人が空を見る。
- グループに賛成する方が、グループに反対するよりも楽
- グループになるほどリスクな選択をしなくなる。  
(No one ever got fired for buying IBM.)

# 独立性 (Independency)

- 完全な独立は存在しない。
- 人間は自律的なものであると同様に社会的なもの
- 他者の影響から相対的な自由
  - 意見が相関することによる間違いが生じなくなる。
  - 独立な人々は新しい意見を持っている (多様性)。
- グループ内の影響が強いと、グループの決定は賢くなる。
- 独立性を高めるためには：
  - 最初に意見と情報を広く持つ。
  - グループの判断の前に自らの判断を持つ。
  - 意見は同時に集める。
  - 独立性のためには過信 (勇気) も重要  
(リーダーは過信すると独立性・分離性に問題が生じるが)
- (国際開発工学専攻は Independency の意味でも最高)
  - 教員間の独立性は高い、若い教員が元気

## 分散性 (Decentralization)

- 9.11 の兆候を示すデータが，FBI, CIA, NSA は持っていた。
- 予測できなかった理由は，それらをまとめる組織がないから？  
(大統領だけ)
- というよりは，情報共有ができていなかった。  
(それぞれの組織の近い分野の人々が，直接情報を共有する方が良い)
- 問題に近い人がいればいるほど，良い判断ができる。
- Linux (成功例)
  - 1991 年 Linus Torvalds 氏が公開
  - 10 人がダウンロードして，バグを直した。
  - 中心がなくても，うまく行っている。
- (国際開発工学専攻は Decentralization の意味でも最高)
  - 直接問題に対応している。

# 予測に集合知を利用

- FutureMAP
  - 集合知ベースで安全保障の判断を改善する方法論
- Prediction markets
  - IEM のような仕組みで未来の生じる確率を予想する。
- Internal market
  - 各組織から集まった：12～13 人
  - 外部：数人
- Policy analysis market (2003 年)
  - United States' Defense Advanced Research Projects Agency (DARPA)
  - 外部の人々，無報酬
  - 予測内容
    - ヨルダン政府はどのくらい安定化される。
    - パレスチナのトップは誰になる。

# 集約性 (Aggregation)

- 多様性, 独立性の環境で様々な情報や決定を生成  
⇒ 集約
- 具体例：投票，市場，  
Collective significance of information
- 集約は，具体的な方法やコストなどの問題で難しい。
- 現在集合知が具現化してきている大きな理由：  
インターネットや Web2.0 の発展により，集約が以前よりも容易になってきたから。
- ネットワークの発展は，これからも世界中に広まる。  
⇒ 集合知がさらに重要性になっていく。
- (国際開発工学専攻は **Aggregation** の意味でも最高)
  - 会議，メール，Wiki，まとめる力

## 学問での集約性の問題

	Country	Documents	Citable docs	Citations
1	United States	537,308	493,337	341,608
2	China	392,164	383,117	105,523
3	United Kingdom	152,877	137,413	106,306
4	Germany	143,284	132,505	95,320
5	Japan	118,768	111,893	50,816
6	France	102,474	95,534	61,977
7	India	98,081	91,366	25,665
23	Malaysia	20,838	20,134	5,135
42	Thailand	10,824	10,213	3,752
59	Viet Nam	2,836	2,720	1,097
70	Philippines	1,405	1,327	688

<http://www.scimagojr.com>

- 論文数はまだまだ増える ⇒ 集約性に問題

# 学問での集約性の問題

- 論文数が増えて，集約できるのか。
- Citation 数は，**Collective significance**  
独立性の問題  
⇒ **分散性，独立性が重要**  
Citation 数が増えると，それが原因で Citation が増える。
- 1 人 or 少人数のグループでサーベイや本を書く。  
⇒ **多様性の問題**
- 論文を書くだけならば，それほど ICT 知識はいらないが，より高度な集約のためには，ICT 知識が重要
- 文系，理系に関わらず，**情報集約のための研究者の ICT リテラシーの充実が必要** (個人的意見)
- (前フリが終了。Web 2.0 の話へ。)

## 集合知のまとめ

- 多様性 (Diversity) :  
様々な参加者の意見が表明できる。
- 独立性 (Independency) :  
他の参加者に影響されない。
- 分散性 (Decentralization) :  
分散して意思決定が行われる。
- 集約性 (Aggregation) :  
最終的には、意見をまとめることができる。
- インターネットは、多様性、独立性、分散性、集約性
- Web 2.0 と集合知が可能にした：即時性、低コスト

## 国際開発と集合知

- 情報は重要：最新，高品質の情報が安価で手に入る。  
過去の情報の集積，サーベイ
- 途上国側からの情報発信

# インターネット

- **ARPANET** (1969 年) が原型
  - 「ネットワークつなく」という意味
  - 学内のネットワークをつないだ。
- UCLA, UC サンタバーバラ, スタンフォード, ユタ大学
- 負荷共有, メッセージサービス, 情報の提供, プログラムの共有, 遠隔ログイン
- 東北大学のハワイの ALOHAnet(ARPANET) への参加 (1981 年)
- 日本国内: **JUNET** (1984 年): 東大, 慶応大, 東工大
- WWW (World Wide Web) により一般に普及
- 通信量の飛躍的増大を光通信の進歩 (Wavelength Division Multiplexing) が支える。
- ユビキタスネットワーク ← 無線通信の進歩

# World Wide Web (WWW)

- Tim Berners-Lee が開発
  - 欧州原子力研究機関，1990 年初頭
  - 論文を任意のプラットフォームで閲覧可能にする。
  - 日本最初のホームページ：  
森田洋平，高エネルギー加速器研究機構 1992/9/30
- **HTTP** (Hypertext Transfer Protocol)
  - Hypertext：ハイパーリンクが埋め込まれたテキスト
  - サーバクライアント型システム
  - Hypertext を相互にやりとりするための規約
- **HTML** (Hyper Text Markup Language)
  - Hypertext を記述するための言語 **タグ付き言語**  
SGML (Standard Generalized Markup Language) に準拠

## タグ付き言語

```
<table border="1">  
  <tr><td> 項目 1 </td><td> 項目 2 </td></tr>  
</table>
```

# Web ブラウザ

- **WorldWideWeb** 世界最初の Web ブラウザ (1991 年)
  - HTTP で通信した NTTL 文書を表示
  - Nexus と改名された
- Mosaic (1993 年) :画像表示  
NCSA (National Center for Supercomputing Applications)
- Mosaic から Netscape Navigator (1994 年) に分岐
  - ブラウザ上で動作するプログラミング言語 Javascript (2005 年)
- Mozilla (1996 年) 設立
  - Netscape 社がオープンソース化
  - Firefox (2002 年)
- Internet explorer (Microsoft)
- Chrome (Google), Safari (Apple), Opera
- プラグイン:ブラウザの動作を拡張  
機能豊富な HTML5 により, 安全性のために使わない傾向

# URL

- **URL** : Uniform Resource Locator
- 同じ形でネットワーク上のすべてのリソースの場所を示そうとするもの
- Web の文書などを指定するために使う。
- 例 : `http://www.ide.titech.ac.jp:80/ja/index.php`
  - `http` : スキーム (プロトコルを表す)
  - `www.ide.titech.ac.jp` (ホスト・ドメイン)  
ネットワーク上のマシンの名前
  - `80` : ポート番号 (UDP, TCP)  
スキーム標準のポート番号の場合は省略できる。
  - `/ja/index.php` : パス・データ名
- Web のほかでも , `ftp`, `rscs`, `svn` など使われる。

# HTTP のデータ

- HTTP におけるデータの受け渡し形式 :
- ヘッダ部
  - 通常はユーザーには見えない。
  - html 文書の<head> </head>のことではない。
  - Cookie の情報  
Web サーバごとに，ブラウザが保持している情報
- 空行
- データ部
  - HTML ファイルに記述されるもの  
タイトル，文字コード，見出し，フォントの種類と大きさ，  
表，ハイパーリンク，画像の参照
  - 画像などがテキスト化されたもの  
MIME (Multipurpose Internet Mail Extensions)
  - POST の送信データ ユーザがブラウザに入力した情報

# HTTP の手順

- HTTP サーバ

- Apache
- IIS (Microsoft)

- HTTP のデータの受け渡し手順：

- 1 ブラウザの URL を入力
- 2 DNS を使ってサーバの IP アドレスを調べる
- 3 ブラウザからサーバにデータを要求を送る。(GET)  
ブラウザ側に Cookie がある場合サーバに送る。
- 4 サーバがブラウザに HTTP のデータを送る。
- 5 データ部の HTML のファイルの表示必要な情報を要求する。  
(GET)
- 6 サーバがブラウザに HTTP のデータを送る。  
画像, css ファイル, Javascript ファイルなど  
(繰り返すことがある)

## Web 2.0

- Tim O'Reilly が提唱 (2005 年)
- **Web** : サーバを管理する情報提供者から , 情報受領者への一方通行
- **Web 2.0** : **参加型のアーキテクチャ**  
一般参加者が情報を提供し , 一般参加者が情報を受領

### Web 2.0 が現れた背景

- 高速通信の低コスト化 :  
マルチメディア : 音声 , 動画 (youtube)
- 計算機の高速度化 :  
サーバーでは検索 , クライアントでは Javascript
- ストレージの低コスト化 :  
**ロングテール**に属する情報の保存と提供が可能



## Web 2.0 のキーワード

□ コミサイト, Amazon Mechanical Turk, Enterprise 2.0, EyeOS, FreedomBox, Gears, Gmail, Google リーダー, Lightbox, Mturk, Napster, Netvibes, OpenDNS, Pandora Radio, Pinterest, Rotten Tomatoes, Skype, WebSocket, Windows Live, Wuala, **YouTube**, Zend Framework, Zimbra, Ipernity, Employee Generated Media, **Wiki**, Wikimapia, Wikileaks, **Wikipedia**, ウェブ 2.0 会議 (2004 年), 開心網, Google ニュース, GUBA, Crunchyroll, **経済産業省アイディアボックス**, Confluence, コーチ・ユナイテッド, サウンドペディア, 新浪微博, Stage6, SnapDish, ソーシャル・セマンティック・ウェブ, ソーシャルブックマーク, So-net Photo, Tudou, Digg, Trippiece, BitTorrent, Furpeace, **Facebook**, フォークソノミー, フォト蔵, フリーミアム, Flickr, Fring, Myspace, マッシュアップ, MaYoMo, Meebo, Midomi, **みんな**, Yahoo!□□, YouOS, Youku, Last.fm, レビューサイト, 人人網

# Web 2.0 を支える技術

- **LAMP** (WAMP, MAMP)
  - Linux (Windows, Macintosh)
  - Apache : HTTP サーバー (IIS, Zope)
  - MySQL : データベース (MariaDB, PostgreSQL, Oracle, DB2)
  - PHP : 処理言語 (Perl, Ruby, Python, Java, C)
- **CSS** (Cascading Style Sheets)
- **Javascript** (ブラウザ上で処理を実行)
- SSL (Secure Sockets Layer) : トランスポート層の暗号通信
- 応用されているもの
  - Wiki, Wikipedai (後述)
  - CMS (Contents Management System)  
多くの CMS が SAapche+MySQL+PHP)
  - e-Commerce, e-Education, e-Government, ...

# MySQL

- 関係 (relational) データベース (E. F. Cood, IBM, 1960 年)
- 2次元の表でデータを格納する。
- 当初は階層化データベースに比べて重要視されななかった。
- 関係論理：制限，射影，結合
- 表を組み合わせて，データ構造を作る。  
(表自体の構造を複雑にする試みはうまく行かない)

grouptbl	
gid	name
1	東アジア
2	東南アジア
3	アフリカ

groupuser	
gid	pid
1	2
1	3
1	4
1	5
2	1
2	2

usertbl	
pid	name
1	伊藤博文
2	黒田清隆
3	三條実美
4	山縣有朋
5	松方正義
6	大隈重信

# SQL

- **SQL** (非公式 Structured Query Language)
- データ操作言語
  - 検索 (select)
  - データの書き換え (update)
  - 行の追加 (insert)
  - 行の削除 (delete)
  - データベースの管理 (create, drop)
- 例: 「東アジア」グループの所属する人の名前  
**select** usertbl.name from grouptbl,groupuser,usertbl  
where grouptbl.gid=grouptuser.gid  
and grouptuser.pid=usertbl.PID  
and grouptbl.name='東アジア'
- 検索時間が短くなるように, コンピュータが自動的に検索の方法を工夫する。
- ハッシュテーブル, ソートなど

# PHP

- **Hypertext Preprocessor**
- サーバ上で動作するオブジェクト指向のプログラミング言語

## PHP の処理手順

- 1 ブラウザからサーバにデータを要求を送る。(GET, POST)
- 2 要求されたデータがHTML ファイルの場合はそれをブラウザに送る。

PHP のファイルの場合，PHP プログラムを実行し，プログラムが出力するテキスト (print 文など) をブラウザに送る。

- form などによって，ブラウザからデータが送られてきた場合は，それを PHP プログラムに渡す。  
\$\_GET, \$\_POST 変数に格納される。
- MySQL のコマンドが入っている場合は，データベースの検索，操作を行う。

## デモ

## ● Cascading Style Sheets

- データ (Data Object Model) と表現を分ける。
- フォントの大きさ，色，配置，表のスパン，背景色
- ページをきれいに見せるために，細かく指定されている。
- 優先順番が決まっている (Cascading)
  - 作成者 > ユーザ > ブラウザ
  - インライン > 埋め込みスタイルシート > 外部
  - ID，属性 > クラス > タイプ (html タグ)
- セレクタ：タイプ (子孫，子，隣接)，ID，クラス，属性
- ID：1 つの html タグを指定することができる。
- クラス：複数の html タグを指定することができる。
- <div> (division)，<span>：  
スタイル，ID，クラスを記述するタグ

# JavaScript

- Netscape 社が 2005 年に開発した。
- ブラウザ上でインタープリターで動作する。
- PHP , Java , Perl などはずべてサーバ上で動作し , 生成した HTML テキストをブラウザに送ることが役割。
- 動的なページを作ることができる。
- スクリプト系 (PHP, Perl, Ruby, Python) の言語
- オブジェクト指向のプログラミング言語
- プログラミング言語の Java とは無関係
- html データやスタイルの参照 , 書き換え , 追加が可能
- DOM (Document Object Model) により HTML 文書を扱う。
- ファイルなどにはアクセスできない (安全性のため)
- デモ

# Ajax

- **Asynchronous JavaScript and XML**
- 1999 年 Internet Explorer 5 に XMLHttpRequest が実装された。
- Javascript を使って，非同期にサーバからデータを取得し，表示を書き換えることが可能になった。
- Google Map (2005 年) で有名になった。
- **XML (eXtensible Markup Language)**
  - html のようにタグによって記述される構造化されたテキスト。
  - DTD (Document Type Definition) より厳密に文法を定義できる。
  - xhtml, docx, xlsx などは XML で記述されている  
(zip 化されている場合がある)
- ただし，Ajax は XML を使わなくてもはいけないものではなく，非同期通信することを意味している場合が多い。
- JQuery：標準的に使われる Ajax のためのライブラリー
- デモ

# Wiki

- Ward Cunningham が Wiki Wiki Web を公開 (1995 年)
- ウェブブラウザからページの作成・編集を行うことができるコンテンツサーバ
- 特徴
  - Wiki の文法に従い、テキストをブラウザ上で入力
  - 整形された文書を表示
  - 一般的には、入力は **WYSIWYG (What You See Is What You Get)** ではない。
  - 多人数が 1 つのコンテンツを記述・管理
  - PHP, Java, perl, ruby, C++ で実装された各種 Wiki システムが存在
- **Wikipedia** がスタート (2001 年)
  - Wiki による辞書, 百科事典
  - 途上国では無料で得られる情報源
  - 先進国の高等教育・研究開発においても重要な情報源

# Wikipedia (Wikimedia)

- 百科事典に劣らない正確さ。(Jim Giles, 2005 年)
- 内容の多様さ (百科事典の数十倍の項目数,  
ブリタニカ百科事典：項目数 65,000 以上)。
- オンライン版ブリタニカ百科事典は、購読者が修正案を提案できるようになった。

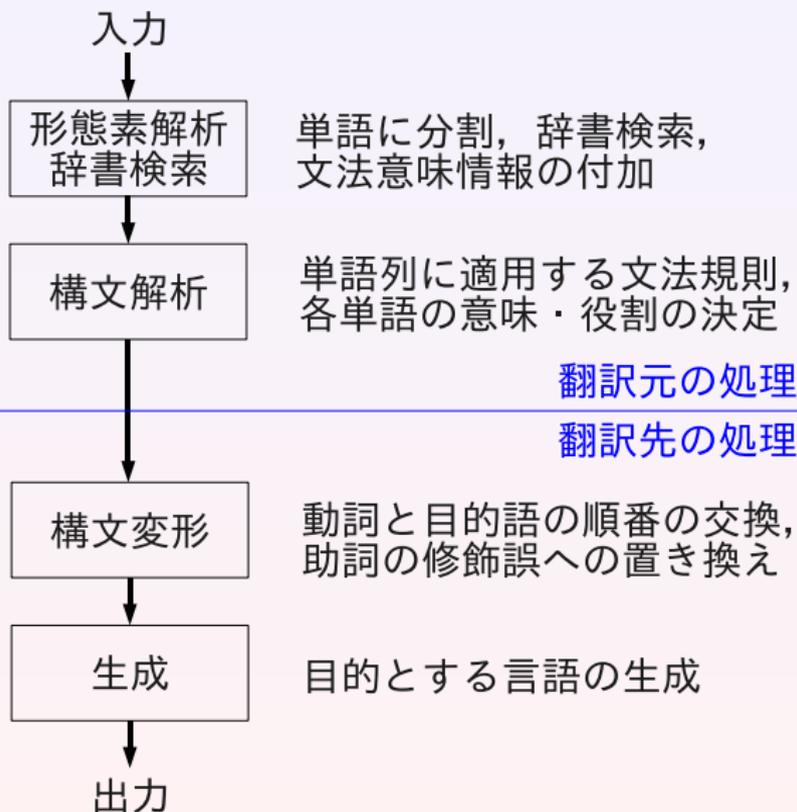
順位	言語	記事数	順位	言語	記事数
1	英語	4,483,887	10	ワライ語	959,527
2	オランダ語	1,769,013	11	日本語	902,972
3	ドイツ語	1,703,015	15	中国語	760,363
4	スウェーデン語	1,615,174	53	タイ語	87,160
5	フランス語	1,490,623	286	カヌリ語	1

(2014 年 4 月 1 日)

## 問題点

- 言語間での情報の量と質の格差

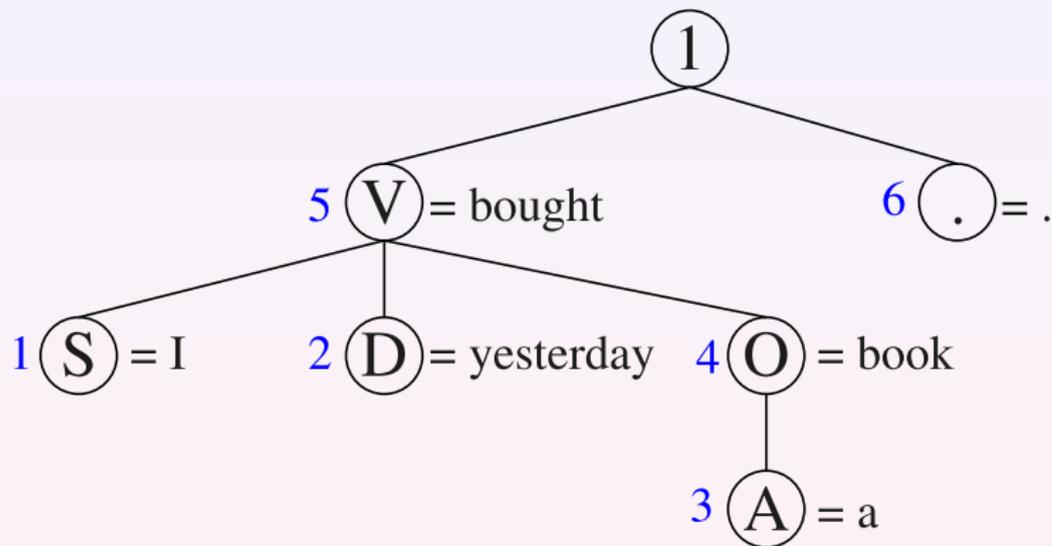
# 機械翻訳の手順



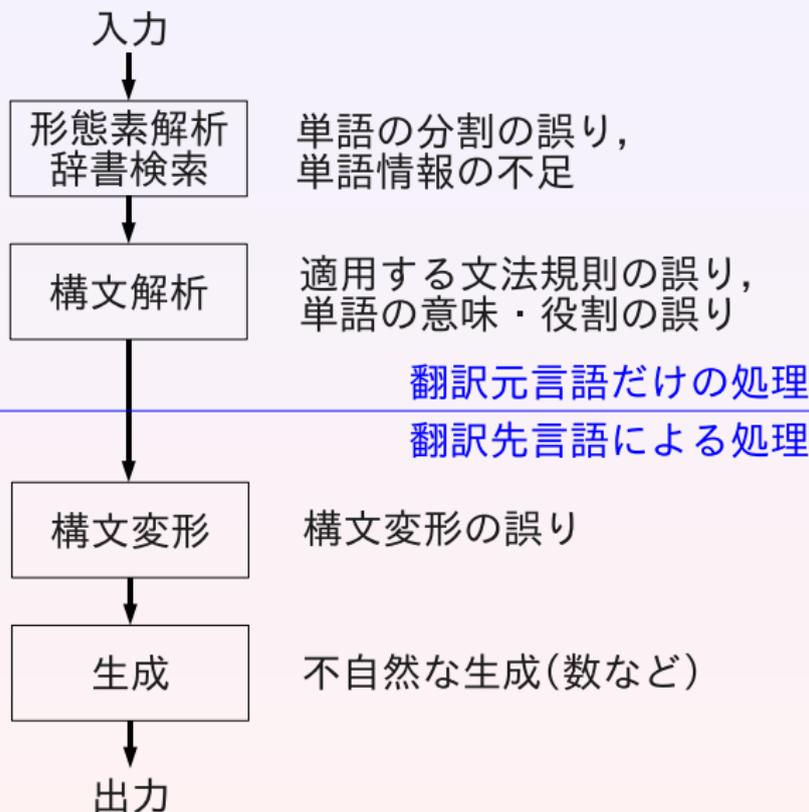
# 構文解析



# 構文変形



## 機械翻訳で生じる誤り (1/2)



## 機械翻訳で生じる誤り (2/2)

### 誤訳の例例

**原文** : RSS is a family of web feed formats. Web feeds benefit publishers by letting them syndicate content automatically.

**Yahoo 翻訳** : RSS は、ウェブ供給フォーマットの家族です。ウェブは、彼らに自動的に内容をシンジケートにさせることによって、利益出版者に食事させます。

### 機械翻訳の見通しは明るくない

- 上の翻訳は 2009 年と変わっていなかった。
- 翻訳業者は機械翻訳を文の翻訳のためには使わない。(機械翻訳を直すよりも原文から翻訳した方が楽)
- コンピュータには文書の意味は分からない。
- プロの翻訳者も、意味を取り違えた翻訳をする。

# 半機械翻訳情報提供システム

## ● ADAM

- Auxiliary Description to Attach Meaning の略
- 英文に意味と文法情報などを付加した **XML 文書**
- **集合知に基づいて** ADAM 文書を記述できるようにする。
- 意味と文法情報を付加することによって、  
翻訳元言語の処理における間違いをなくす。
- 付加情報を付加することによって、  
翻訳先言語の処理における間違い削減する。

## ● LILITH

- LInguaL Interpreter To Human
- ADAM 文書を翻訳先の言語に翻訳する。

## ● eVA

- electronic Vocabulary Assistant
- 多言語辞書管理システム

# ADAM 文書記述システム

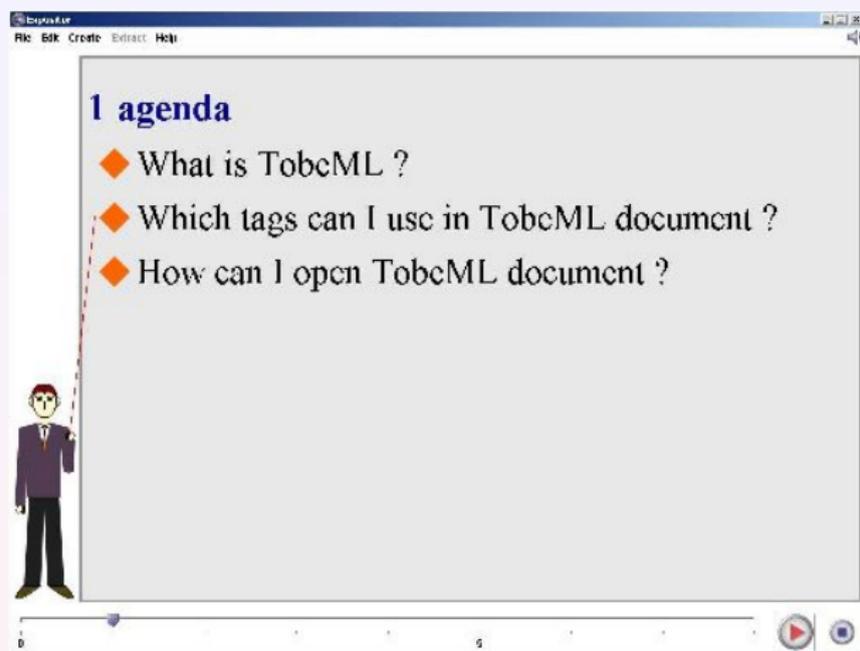
- インターネット上で，ADAM 文書を記述するためのシステム
- 英文に対して，文法情報，意味情報を表すタグを付加
- 練習により，1 単語につき 3 ~ 5 秒で必要なタグを付加可能

## 付加する情報の例：them

- 文法情報：「代名詞」「人称」「目的格」
  - 意味情報「辞書の 5 番目の意味」
- 
- クライアント側処理：javascript
  - サーバ側処理：PHP

- Text Oriented Bi-stream Explanation Markup Language
- XML ベースの説明記述言語
- board 文書と talk 文書に分離  
    <board>, <talk>
- セマンティクスを表すためのタグ  
    <definition>, <theorem>, <proof>, <question>
- 節, 箇条書き, 表, 図などのためのタグ  
    <section>, <subsection>, <itimize>, <table>, <fig>,
- 数式には, MathML を使用
- board を指し示すためのタグ  
    <target>, <point>
- その他の動作  
    <pause>, <nod>, <smile>

# Expositor



- **tobeML 講義形式プレーヤ**

- Java

- Swing (表示) , TreeTTS (音声合成) , JEuclid (数式)

# まとめと今後の課題

## まとめ

- 集合知, Web, Web 2.0 の基本技術, 機械翻訳  
人間が得意なこと vs. 計算機が得意なこと
- 研究室での取り組み
  - 半機械翻訳情報提供システム
  - tobeML

## 今後の課題

- プログラムの維持
- 開発の後継者
- FEP (Front End Processor) 形式の ADAM 記述システム
- 多数の参加者を集める。