

【WWW2012勉強会】

Session 27: Data and Content Management 1

担当：渡辺陽介（東京工業大学）

Declarative Platform for Data Sourcing Games

▶ 著者

- ▶ Daniel Deutch (Ben-Gurion Univ.), Ohad Greenshpan, Boris Kostenko, Tova Milo (Tel-Aviv Univ)

▶ 概要

- ▶ クラウド(Crowdの方)データソーシングによって得られた不確実なデータのクリーニング処理を, SQLベースの宣言的言語で記述・実行可能とした

▶ 解いている問題

クイズ形式のクラウドソーシングシステム

(例) 国名と首都のペアのデータベースを作りたい

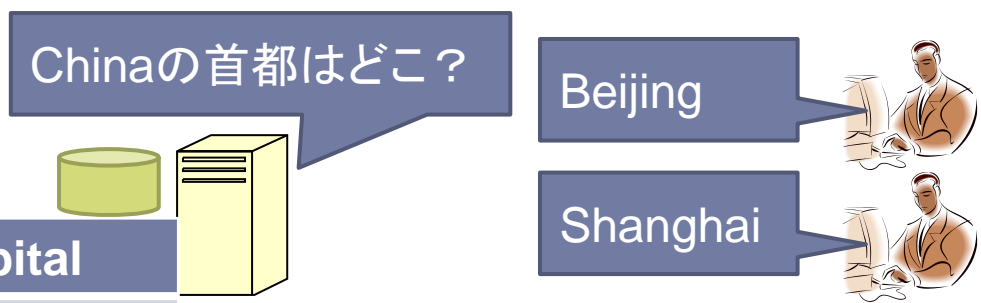
正解と不正解が混ざっている場合どうするか？

Chinaの首都はどこ？

Beijing

Shanghai

ユーザの回答の信頼度をどうやって見積もるか？



Country	Capital
China	Beijing
China	Shanghai

Repair-Key演算とPossible Databases

CapitalsByUsers

Country	Capital	User
China	Beijing	Alice
Netherlands	Amsterdam	Alice
Netherlands	Amsterdam	Bob
Netherlands	Hague	Carol
China	Shanghai	Dan

Users

User	Authority
Alice	2
Bob	5
Carol	3
Dan	4

Table2, Table3より引用

結合(と射影)



Country	Capital	Authority
China	Beijing	2
Netherlands	Amsterdam	2
Netherlands	Amsterdam	5
Netherlands	Hague	3
China	Shanghai	4

Repair-key
演算



Possible Databases

Country	Capital	Country	Capital
China	Beijing	China	Beijing
Netherlands	Amsterdam	Netherlands	Hague
Country	Capital	Country	Capital
China	Shanghai	China	Shanghai
Netherlands	Amsterdam	Netherlands	Hague

各Databaseの確率はauthorityから計算

User-based PageRank

- ▶ Authorityの高いユーザが入れた回答は正解(の可能性が高い)
- ▶ 正解(の可能性が高い答え)を多く答えるユーザはAuthorityが高い

Users

User	Authority	Correctness
Alice	2 → 2	1
Alice	2 → 2	0
Dan	4 → 1	1
Dan	4 → 1	0

④ユーザ毎Authorityの更新
Count(distinct country)

①Repair-key演算

User	Auth.	Corr.	User	Auth.	Corr.
Alice	2	1	Alice	2	0
Dan	4	1	Dan	4	1

User	Auth.	Corr.	User	Auth.	Corr.
Alice	2	1	Alice	2	0
Dan	4	0	Dan	4	0

②選択演算(correctness=1)

User	Auth.	Corr.	User	Auth.	Corr.
Alice	2	1	Dan	4	1

User	Auth.	Corr.	User	Auth.	Corr.
Alice	2	1			

③結合

CapitalsByUsers

Country	Capital	User
China	Beijing	Alice
Netherlands	Amsterdam	Alice
China	Shanghai	Dan

Country	Capital	User	Auth.
China	Beijing	Alice	2
Netherlands	Amsterdam	Alice	2
China	Shanghai	Dan	4

Country	Capital	User	Auth.
China	Beijing	Alice	2
Netherlands	Amsterdam	Alice	2

Country	Capital	User	Auth.
China	Shanghai	Dan	4

Country	Capital	User	Auth.

Information Integration Over Time in Unreliable and Uncertain Environments

▶ 著者

- ▶ Aditya Pal (Univ. Minnesota), Vibhor Rastogi, Ashwin Machanavajjhala, Philip Bohannon (Yahoo! Research)

▶ 概要

- ▶ 本来同じ情報を提供するはずの複数の情報源から、更新遅れやミス等で異なる値が届いてしまった場合に、どれを信じるかという問題の解決

▶ 例：レストラン情報サイト（あるお店の電話番号）

サイト	時刻t1	時刻t2	時刻t3(現在)
S1	555-4444	555-8566	555-1234
S2	555-8566	555-1234	555-1234
S3	555-4444	555-4444	555-4444

Table 1より引用

本来の更新系列Zの推測 「555-4444 → 555-8566 → 555-1234」
サイトごとの更新遅延や更新ミスの特性(マッピング O)の推測

Semi-Markovian

- ▶ 本来の更新系列 Z が, 途中の操作 O により変更されて, 各サイトでの観測系列 S となった, と考える
- ▶ 想定する操作
 - ▶ OP1: S は Z の更新のいくつかを報告しない
 - ▶ OP2: S は Z の本来の更新とは異なる値を報告する
 - ▶ OP3: S はある Z の更新を重複して報告する
 - ▶ OP4: S は Z の値を遅れて報告する

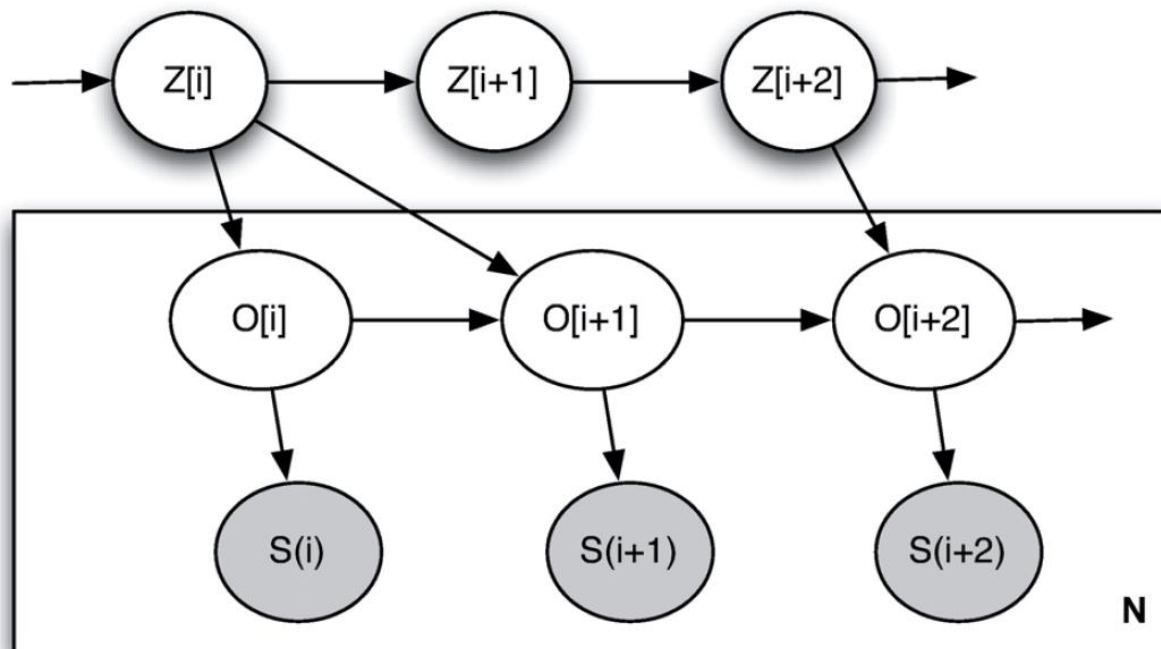


Figure 1より引用

Z と O の推論アルゴリズム

- EMアルゴリズム
- Gibbs sampling
- ハイブリッド?

SAFE Extensibility of Data-Driven Web Applications

▶ 著者

- ▶ Raphael Reischuk, Michael Backes (Saarland Univ.),
Johannes Gehrke (Cornell Univ.)

▶ 概要

- ▶ カスタマイズが容易なRIA (Rich Internet Application)を設計するための階層的プログラミングに基づくフレームワーク

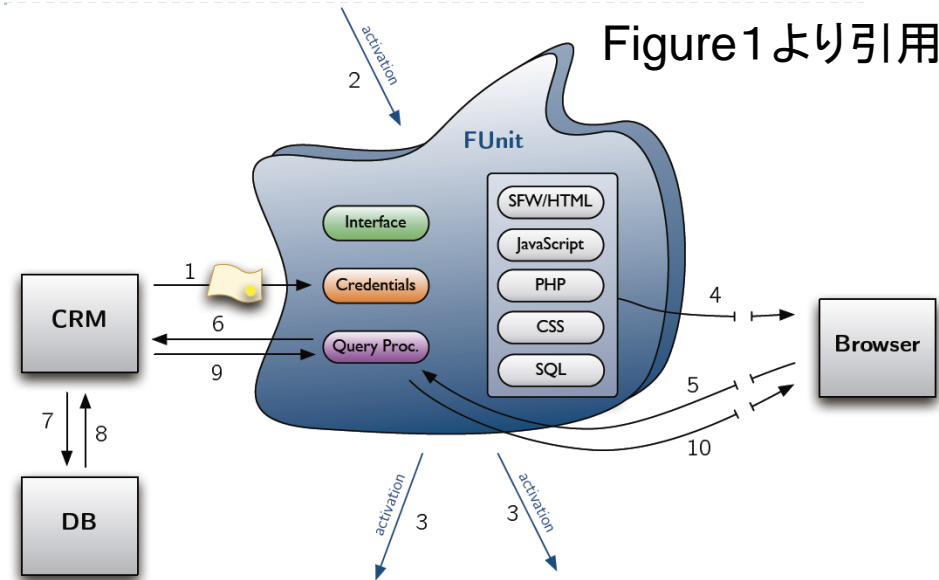
▶ 対象とする問題

- ▶ ユーザ定義(第3者定義)のコンポーネントを組み込んで、自由にカスタマイズできるWebアプリケーションを作りたい
 - ▶ 拡張可能性を実現する設計とは？
 - ▶ 第3者定義のコンポーネントを組み込んで安全か？
 - コンポーネント間の呼び出し関係
 - データベースへのアクセス許可

SAFE: Safe Activation Framework for Extensibility

▶ f-unit

- ▶ HTMLの断片についての表示やビジネスロジックをまとめた基本コンポーネント
- ▶ SFWという言語で記述し、コンパイルによりサーバ用コード、クライアント用コードを生成



▶ Activation tree

- ▶ F-unit間の呼び出し関係を表すツリー
- ▶ HTMLの構造に対応して親子関係が定義される

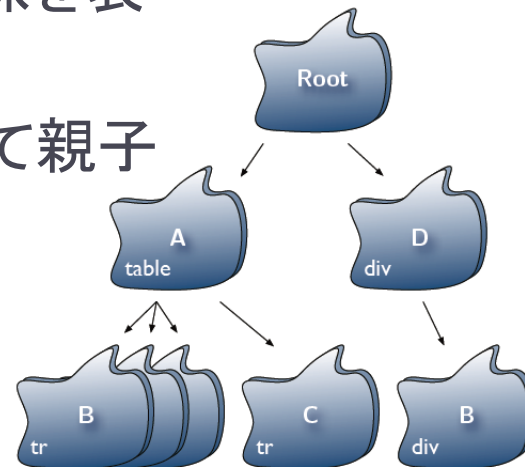


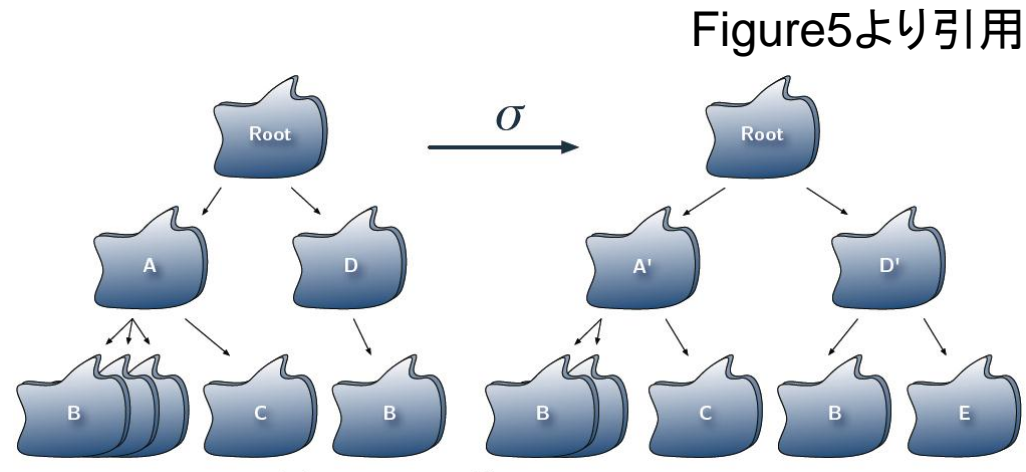
Figure 2より引用

```
<HTML>
...
<table id='Root.A'>
  <tr id='Root.A.B_1'> ... </tr>
  <tr id='Root.A.B_2'> ... </tr>
  ...
  <tr id='Root.A.C'> ... </tr>
</table>
<div id='Root.D'>
  <div id='Root.D.B'> ... </div>
</div>
...
</HTML>
```


Extensibility and Security

▶ 拡張

- ▶ Activation Tree内のf-unitを別のものに置き換える
- ▶ 他のコンポーネントの連携に支障がでないかをチェックする仕組み
 - ▶ インターフェース



▶ インターフェース

- ▶ F-unit間の呼び出し関係
- ▶ データベースの各表・カラムへのアクセス
- ▶ 他のf-unitによる更新通知を受け取るためのサブスクリプション

Figure6より引用

```
INPUT:
  init = (!$!reviewerId)
  activate = {@$@id, @$@title, @$@comments, @$@grade}

ACTIVATION:
  FooBar.init = (!$!reviewerId)
  FooBar.activate = {@$@id}

DATABASE:
  read = {reviews.id}
  write = {reviews.comments, reviews.grade}

SUBSCRIPTION:
  com: SELECT comments FROM reviews
  acc: SELECT * FROM reviews WHERE
        BEFORE grade<0 AND
        AFTER  grade>0
```