

# ライフログ解析アプリケーション実行と データ品質評価フレームワークの確立

山下 暁香<sup>†</sup> 岩木紗恵子<sup>†</sup> 小口 正人<sup>†</sup>

<sup>†</sup> お茶の水女子大学理学部情報科学科 〒 112-8610 東京都文京区大塚 2-1-1

E-mail: †{akika,saeko}@ogl.is.ocha.ac.jp, ††oguchi@computer.org

あらまし 近年のデータ収集技術やストレージの発達により、ライフログの実現が以前と比べ容易になったため、様々なライフログ解析アプリケーションが開発されてきた。本研究では、これらのアプリケーションの入力データの品質の差に注目し、実際に人の行動を言語化するライフログ解析アプリケーションにおいて、画像ストリームと加速度センサデータの2つの入力データの品質の差がアプリケーションの実行結果に及ぼす影響を定量的な指標で評価するためのフレームワークを確立し、このフレームワークを利用して入力データ品質評価実験を行った。

キーワード ライフログ、センサネットワーク、品質評価

## Execution of Lifelog Analysis Application and Realization of Framework for Evaluating Data Quality

Akika YAMASHITA<sup>†</sup>, Saeko IWAKI<sup>†</sup>, and Masato OGUCHI<sup>†</sup>

<sup>†</sup> Ochanomizu University Otsuka 2-1-1, Bunkyo-Ku, Tokyo 112-8610 Japan

E-mail: †{akika,saeko}@ogl.is.ocha.ac.jp, ††oguchi@computer.org

### 1. はじめに

近年、カメラ、携帯電話、GPS といった画像や位置、加速度等の情報を記録できる端末、PC 等のネット上の行動履歴を記録できる端末などの情報記録端末の発達により、技術的にデータ収集が容易になった。また、ストレージの発達により、無料、もしくは低廉で大量の情報を蓄積、検索、共有することが可能になった。

このデータ収集技術とストレージの発達により、ライフログの実現は以前と比べ格段に敷居が低くなったと言え、記録された一つ一つの情報は各種データベース間の連携を通じてマッシュアップする事で様々な価値が生まれる。

ライフログの実現の簡易化を受け、収集されたデータを用いて様々なライフログ解析アプリケーションが開発されてきたが、これらのライフログ解析アプリケーションに対して、収集された入力データの品質についての考慮はあまりされてこなかった。

### 2. 研究目的

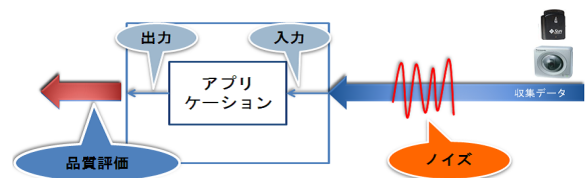


図1 研究目的

本研究では、ライフログ解析アプリケーションの開発を最終目標とするのではなく、アプリケーションに対しての入力データの品質の差に注目し、入力データの品質の差がアプリケーションの性質に及ぼす影響を定量的な指標で評価するフレームワークの確立を目指す。

図1のように入力データの品質がアプリケーションに与える影響を評価するために、ライフログ解析アプリケーションの一例として、お茶の水女子大学小林研究室で開発された「人の行動を言語化するシステム」[1]を拡張して利用し、評価実験を行った。

### 3. 人の行動を言語化するアプリケーション

#### 3.1 アプリケーションの動作

人の行動を言語化するアプリケーションとは、2つの異なる角度に設置された2台のネットワークカメラによって撮影された動画データの毎フレームに対して画像処理を施すことで、動画の中で人が行った行動を言語化するものである。例えば、図2のように「人がドアを開ける」という行動をしている動画を入力データとしてシステムに与えると、情報解析の結果として「人がドアを開ける」という言語化が行われる。

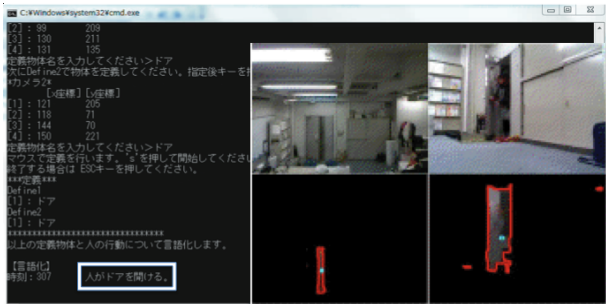


図2 人の行動を言語化するアプリケーションの動作  
赤枠：初期フレームと現フレームの差分の輪郭  
青点：差分輪郭の重心（人の重心）  
白線：言語化の結果の出力

#### 3.2 動画データの処理方法と閾値1

既存のシステムにおける、動画データから言語化を実現する際の処理方法について説明する。動画データの処理では、1フレーム毎に、現在のフレームと初期フレームの差分の輪郭を抽出し、その輪郭で囲まれた部分の重心を求めている。輪郭は動く物体（人）、重心は人の重心と解釈される。そして、この重心が予め定義した物体（ドアなど）と重なる回数を数え、この回数がある閾値を超えた時に言語化が行われる。

ユーザはアプリケーションを実行する際に、「閾値」を設定することで言語化の基準を調節できる。尚、この2台のカメラにより撮影された動画データを処理する閾値を後に出てくる加速度センサデータ処理用の閾値と区別して「閾値1」と呼ぶ。閾値1を大きくするほど、言語化の基準が厳しくなり、言語化が消極的になり、言語化回数は減少する。また、動画データは2台のカメラによって撮影されているが、2つの動画データに対して、共通の閾値1を用いる。

#### 3.3 先行研究における評価結果

先行研究で、この既存のシステムについて評価実験が行われた[2]。評価実験の内容は、2台のカメラから収集される動画データについて、カメラを、有線の状態(LANケーブル接続)と無線の状態と撮影する事により品質の差をつけ、閾値1を変更し、言語化のタイミングと回数を評価するというものであった。「無線の状態に比べ、高品質な動画データが得られる有線の状態の方が、アプリケーションの振舞がよい」という妥当な結果が得られた。

しかし、既存のシステムでは、2台のカメラからのデータの

みを入力データとしていたため、言語化の判断が誤りやすかった。閾値1が小さい時は、積極的に言語化されるので、行動が起きていなくても言語化されてしまい、反対に、閾値1が大きい時は、言語化が消極的なので、行動が起きているにもかかわらず言語化されないというケースが多く、動画データの品質というよりは、既存のシステムの実装が不十分であるために、言語化のタイミングと回数の両方についてなかなか正解の結果が得られなかった。

### 4. 研究概要

#### 4.1 アプリケーションの実行環境の拡張

先行研究の結果を受けて、本研究では、まず、言語化の正答率を上げ、データ評価実験の内容を充実させるために、既存の「人の行動を言語化するシステム」に対して、図3に示す3つの階層を強化し、ライフログ解析アプリケーションのデータ品質評価フレームワークを実現した。

- (1) 情報解析：言語化対象となる行動数の増加
- (2) データ処理：ベイズ判定の適用
- (3) データ収集：加速度センサデータを同期

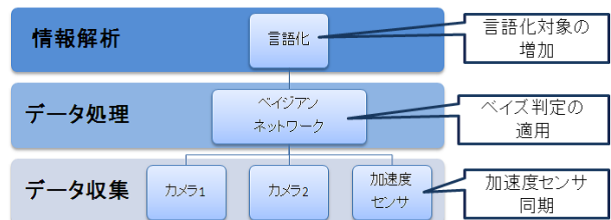


図3 アプリケーションのフレームワーク

#### 4.2 拡張後のアプリケーションの実行環境

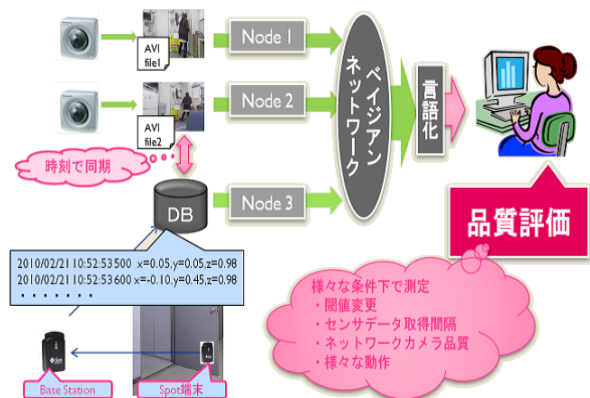


図4 アプリケーションの実行環境

拡張後のアプリケーションは、図4に示すように異なる角度に設置された2台のカメラと定義物体に取り付けた加速度センサ端末 SunSPOT [3] から得られる動画データと加速度データを時刻によって同期し、それぞれをノード1、ノード2、ノード3として、ベイズネットワークにより処理し、条件を満たした場合のみ人の行動を言語化をして、ユーザに言語化結果を提供する。

### 4.3 データ品質評価実験

前節のデータ品質評価フレームワークを用いて、入力データに品質の差を付けた時に、アプリケーションに与える影響を評価する。具体的には、言語化のタイミング、回数について主観的に評価する。

## 5. 開発環境

### 5.1 人の行動を言語化するライフログ解析アプリケーション

プログラムの開発は、Microsoft Visual C++ 2008 Express Edition を用いて、C 言語で行い、画像ストリーム処理は OpenCV を用いた。画像ストリーム撮影用のネットワークカメラは Panasonic BB-HCM715 (最大 130 万画素、有線・無線 LAN 対応)[4] を使用した。

### 5.2 加速度センサ端末

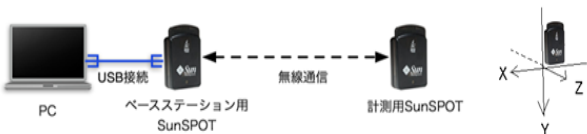


図 5 加速度センサ端末 SunSPOT

また、図 5 で示すサン・ラボで開発された無線センサネットワークデバイス SunSPOT を加速度センサ端末として用いた。加速度、温度、照度センサと 8 個の LED が搭載されており、Java プログラミングにより制御可能であり、Java の開発環境は Net Beans IDE 6.5.1 である。SunSPOT 端末にはベースステーション用端末と計測用端末があり、ベースステーション用端末は PC と USB 接続して用い、計測用端末は物体に取り付けてベースステーション用端末と無線通信を行う。また、1 台のベースステーション用端末に対し、複数台の計測用端末を同時に用いるマルチ通信が可能である。

加速度センサ端末の情報を保持するためのデータベースは PostgreSQL を用いた。

## 6. データ品質評価フレームワークの実現

### 6.1 データ収集

既存のシステムで強化した 3 つの階層のうち、データ収集の部分における加速度センサデータの追加について説明する。本研究では、誤判定が多かった既存のシステムに対して、2 台のカメラのノードに加え、加速度センサデータのノードを追加した。加速度センサノードにおけるデータ処理方法について説明する。まず、時刻順に並んだ加速度データを  $x$  軸、 $y$  軸、 $z$  軸のそれぞれの値について 2 つの時刻毎に差をとった。この差を前述の動画データ処理用の閾値 1 と区別して、閾値 2 とする。次に、この閾値 2 がある回数だけ連続した時に「人の行動が起こった」と判断し、加速度センサデータのノードのビットを立てる。この連続数を閾値 3 とする。

### 6.2 データ処理と情報解析

#### 6.2.1 ベイジアンネットワーク

収集されたデータの処理は、理論的に裏付けるために、従来

のアプリケーションでは実装されていなかったベイジアンネットワークモデルを基にモデル化した。

ベイジアンネットワークとは、因果関係を条件付き確率で記述し、結果から原因を予測するグラフィカルモデルである。結果ノードである  $T_1, T_2, T_3$  と原因ノードである  $D_i$  に対し、与えられた CPT (Conditional Probability Table) に従って、以下の公式を最大にする  $D_i$  が尤もらしい原因として判断される (図 6)。

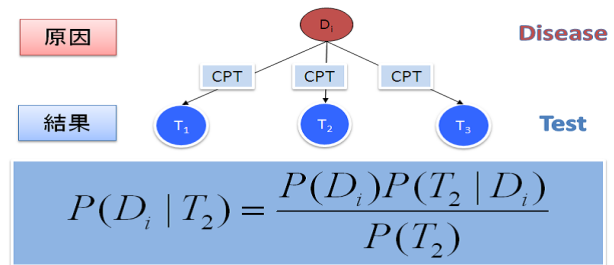


図 6 ベイジアンネットワーク

#### 6.2.2 アプリケーションのモデル化

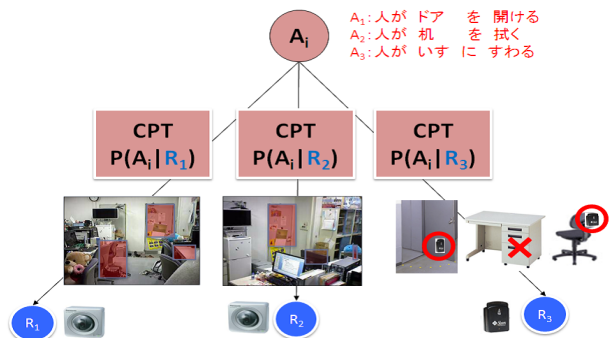


図 7 アプリケーションのモデル化

前節のベイジアンネットワークに基づき、本アプリケーションをモデル化した結果を図 7、モデル化の際に与えた CPT を表 1 に示す。

$R_1, R_2, R_3$  は結果ノードでそれぞれ、2 台のカメラと SunSPOT の反応を表す。 $A_i$  は原因ノードで、人の行動を表す。なお、言語化対象となる人の行動については、データ品質評価フレームワークの情報解析の部分で、既存のシステムでは 1 つのみであった言語化対象を複数個に増加させた。評価実験では、以下の 3 つの行動を扱う。

- $A_1$ : 人がドアを開ける
- $A_2$ : 人が机を拭く
- $A_3$ : 人がいすにすわる。

予め与えられた CPT を参考に、公式

$$P(A_i | R_1, R_2, R_3) = \frac{P(A_i)P(R_1, R_2, R_3 | A_i)}{P(R_1, R_2, R_3)}$$

を最大にする  $A_i$  が尤もらしい原因と判断し、言語化する。

ただし、 $P(A_i | R_1, R_2, R_3)$  の詳しい値は必要なく、公式が最大となるときの  $i$  がわかればよいので、データ処理の計算をする時には、以下のように変形した後の公式を用いる。

$$\begin{aligned} & \underset{i}{\operatorname{argmax}} P(A_i | R_1, R_2, R_3) \\ &= \underset{i}{\operatorname{argmax}} \frac{P(A_i)P(R_1, R_2, R_3 | A_i)}{P(R_1, R_2, R_3)} \dots \text{分母は共通} \\ &= \underset{i}{\operatorname{argmax}} P(A_i)P(R_1, R_2, R_3 | A_i) \dots P(A_i) \text{は同じ値と想定} \\ &= \underset{i}{\operatorname{argmax}} P(R_1, R_2, R_3 | A_i) \end{aligned}$$

### 6.2.3 CPT

表1 モデルに与えた CPT

| R \ A | A <sub>1</sub> | A <sub>2</sub> | A <sub>3</sub> | R \ A | A <sub>1</sub> | A <sub>2</sub> | A <sub>3</sub> | R \ A | A <sub>1</sub> | A <sub>2</sub> | A <sub>3</sub> |
|-------|----------------|----------------|----------------|-------|----------------|----------------|----------------|-------|----------------|----------------|----------------|
| 0     | 0.1            | 0.4            | 0.5            | 0     | 0.1            | 0.9            | 0.8            | 0     | 0.01           | 0.9            | 0.9            |
| 1     | 0.9            | 0.6            | 0.5            | 1     | 0.9            | 0.1            | 0.2            | 1     | 0.99           | 0.1            | 0.1            |

| R \ A | A <sub>1</sub> | A <sub>2</sub> | A <sub>3</sub> | R \ A | A <sub>1</sub> | A <sub>2</sub> | A <sub>3</sub> | R \ A | A <sub>1</sub> | A <sub>2</sub> | A <sub>3</sub> |
|-------|----------------|----------------|----------------|-------|----------------|----------------|----------------|-------|----------------|----------------|----------------|
| 0     | 0.9            | 0.1            | 0.9            | 0     | 0.9            | 0.9            | 0.9            | 0     | 1.0            | 1.0            | 1.0            |
| 1     | 0.1            | 0.4            | 0.1            | 1     | 0.1            | 0.1            | 0.1            | 1     | 0.0            | 0.0            | 0.0            |

| R \ A | A <sub>1</sub> | A <sub>2</sub> | A <sub>3</sub> | R \ A | A <sub>1</sub> | A <sub>2</sub> | A <sub>3</sub> | R \ A | A <sub>1</sub> | A <sub>2</sub> | A <sub>3</sub> |
|-------|----------------|----------------|----------------|-------|----------------|----------------|----------------|-------|----------------|----------------|----------------|
| 0     | 0.9            | 0.9            | 0.1            | 0     | 0.5            | 0.9            | 0.6            | 0     | 0.9            | 0.9            | 0.01           |
| 1     | 0.1            | 0.1            | 0.9            | 1     | 0.5            | 0.1            | 0.4            | 1     | 0.1            | 0.1            | 0.99           |

CPT とは、条件付確率表の事である。

図7のように、本研究のデータ処理のモデルでは、ノード  $R_1, R_2, R_3$  に対して、それぞれ物体の定義の数（評価実験では3つ）だけ CPT を割り当てた。表1の中の小数は条件付確率  $P(R_j | A_i)^{(\text{注}1)}$  である。

表1にある9つ<sup>(注2)</sup>の CPT については、一番上の青枠で囲んだ行の3つの CPT は  $A_1$  を言語化する時に用いるもの。中央の行の3つの CPT は  $A_2$  の言語化、一番下の行の3つの CPT は  $A_3$  の言語化のためのものである。さらに、一番左の列の3つの CPT は  $R_1$ 、つまり、行動が起きた時にカメラ1で撮影された動画データ中の定義物体範囲内が反応する確率、中央の列の3つの CPT は  $R_2$ 、つまり、行動が起きた時にカメラ2で撮影された動画データ中の定義物体範囲内が反応する確率、一番右の列の3つの CPT は  $R_3$ 、つまり、行動が起きた時に SunSPOT が反応する確率を表している。

なお、3列目の CPT に注目すると、「ドア」と「いす」の言語化の際に用いる CPT の中に赤字で示した数字がある。これは、行動が起きた時に、SunSPOT が反応する確率をカメラ1とカメラ2が反応する確率より大きくし、重みを付ける事で、SunSPOT が反応した時に言語化が行われやすいようにしたものである。更に、中央の行の右端の CPT では、行動が起きたときに、SunSPOT が反応する確率が0になっているが、これは、机には SunSPOT 端末を取り付けていないためである。

例えば、青枠で囲んだ3つの CPT のうち、左端の CPT は、 $P(R_1 | A_i)$  を示しており、「行動が起きた時にカメラ1の「ドア」の定義範囲内が反応する確率と反応しない確率」を表している。

- $P(R_1 = 0 | A_1 = 1) = 0.1$   
「人がドアを開けた」時、反応しない確率は0.1

- $P(R_1 = 1 | A_1 = 1) = 0.9$   
「人がドアを開けた」時、反応する確率は0.9
- $P(R_1 = 0 | A_2 = 1) = 0.4$   
「人が机を拭いた」時、反応しない確率は0.4
- $P(R_1 = 1 | A_2 = 1) = 0.6$   
「人が机を拭いた」時、反応する確率は0.6
- $P(R_1 = 0 | A_3 = 1) = 0.5$   
「人がいすにすわった」時、反応しない確率は0.5
- $P(R_1 = 1 | A_3 = 1) = 0.5$   
「人がいすにすわった」時、反応する確率は0.5

### 6.2.4 データ処理手順

「人がドアを開ける」と言語化する場合を例として、データ処理の手順を説明する。

$A_1$  の言語化判定なので、表1の青枠で囲んだ3つの CPT を参考にする。 $R_1, R_2, R_3$  のビットが000から111まで、全部で8通りの場合分けを行い、表1の青い枠で囲んだ CPT を参考に、 $A_1$  「人がドアを開ける」が公式を最大にする時のみ、言語化する。この処理は動画データの1フレーム毎に行われる。

- (1)  $(R_1, R_2, R_3) = (0, 0, 0)$   
 $P(R_1, R_2, R_3 | A_1) = 0.1 * 0.1 * 0.01 = 0.0001$   
 $P(R_1, R_2, R_3 | A_2) = 0.4 * 0.9 * 0.9 = 0.324$   
 $P(R_1, R_2, R_3 | A_3) = 0.5 * 0.8 * 0.9 = 0.36$   
 公式を最大にするのが  $A_1$  でないので、言語化しない。
- (2)  $(R_1, R_2, R_3) = (0, 0, 1)$   
 $P(R_1, R_2, R_3 | A_1) = 0.1 * 0.1 * 0.99 = 0.0099$   
 $P(R_1, R_2, R_3 | A_2) = 0.4 * 0.9 * 0.1 = 0.036$   
 $P(R_1, R_2, R_3 | A_3) = 0.5 * 0.8 * 0.1 = 0.04$   
 公式を最大にするのが  $A_1$  でないので、言語化しない。
- (3)  $(R_1, R_2, R_3) = (0, 1, 0)$   
 $P(R_1, R_2, R_3 | A_1) = 0.1 * 0.9 * 0.01 = 0.0009$   
 $P(R_1, R_2, R_3 | A_2) = 0.4 * 0.1 * 0.9 = 0.036$   
 $P(R_1, R_2, R_3 | A_3) = 0.5 * 0.2 * 0.9 = 0.09$   
 公式を最大にするのが  $A_1$  でないので、言語化しない。
- (4)  $(R_1, R_2, R_3) = (0, 1, 1)$   
 $P(R_1, R_2, R_3 | A_1) = 0.1 * 0.9 * 0.99 = 0.089$   
 $P(R_1, R_2, R_3 | A_2) = 0.4 * 0.1 * 0.1 = 0.004$   
 $P(R_1, R_2, R_3 | A_3) = 0.5 * 0.2 * 0.1 = 0.01$   
**公式を最大にするのが  $A_1$  なので、言語化する。**
- (5)  $(R_1, R_2, R_3) = (1, 0, 0)$   
 $P(R_1, R_2, R_3 | A_1) = 0.9 * 0.1 * 0.01 = 0.0009$   
 $P(R_1, R_2, R_3 | A_2) = 0.6 * 0.9 * 0.9 = 0.486$   
 $P(R_1, R_2, R_3 | A_3) = 0.5 * 0.8 * 0.9 = 0.36$   
 公式を最大にするのが  $A_1$  でないので、言語化しない。
- (6)  $(R_1, R_2, R_3) = (1, 0, 1)$   
 $P(R_1, R_2, R_3 | A_1) = 0.9 * 0.1 * 0.99 = 0.0891$   
 $P(R_1, R_2, R_3 | A_2) = 0.6 * 0.9 * 0.1 = 0.054$   
 $P(R_1, R_2, R_3 | A_3) = 0.5 * 0.8 * 0.1 = 0.04$   
**公式を最大にするのが  $A_1$  なので、言語化する。**
- (7)  $(R_1, R_2, R_3) = (1, 1, 0)$   
 $P(R_1, R_2, R_3 | A_1) = 0.9 * 0.9 * 0.01 = 0.0081$

(注1): 事象  $A_i$  が起きた時に、事象  $R_j$  が起きる確率

(注2): 結果ノード数 \* 定義物体数 = 3\*3 = 9

$$P(R_1, R_2, R_3|A_2) = 0.6 * 0.1 * 0.9 = 0.054$$

$$P(R_1, R_2, R_3|A_3) = 0.5 * 0.2 * 0.9 = 0.09$$

公式を最大にするのが  $A_1$  でないので、言語化しない。

$$(8) (R_1, R_2, R_3) = (1, 1, 1)$$

$$P(R_1, R_2, R_3|A_1) = 0.9 * 0.9 * 0.99 = 0.8019$$

$$P(R_1, R_2, R_3|A_2) = 0.6 * 0.1 * 0.1 = 0.006$$

$$P(R_1, R_2, R_3|A_3) = 0.5 * 0.2 * 0.1 = 0.01$$

公式を最大にするのが  $A_1$  なので、言語化する。

これは、 $A_1$  の言語化について示したものであるが、 $A_2$  の言語化<sup>(注3)</sup>と  $A_3$  の言語化<sup>(注4)</sup>も同様に処理する。なお、CPT では、加速度センサのノードの条件付き確率(表1の赤字)に重みをおいたため、 $R_3$  ノードのビットが立った時に、言語化が行われやすいようになっている。

## 7. データ品質評価実験

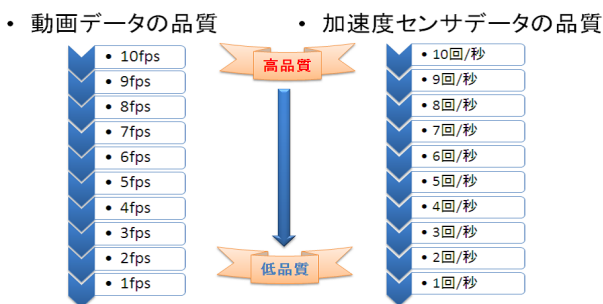


図8 動画データと加速度センサデータの10段階の品質

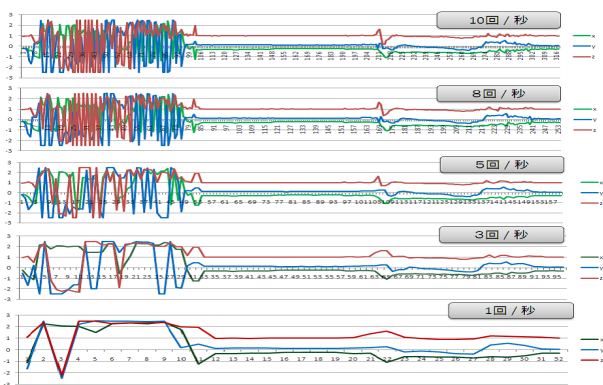


図9 加速度センサデータの品質

本研究では、アプリケーションの入力データの品質の差がアプリケーションに及ぼす影響を評価することを目的としているので、入力データである動画データと加速度センサデータの2種類のデータの品質を10段階にし(図8)、10段階のデータをそれぞれ入力データとした時に、アプリケーションにどのような影響を与えるのかを検討する。

動画データの品質については、最高の品質である10fps<sup>(注5)</sup>から最低の品質である1fpsまでの10段階の品質のデータを入力データとした。

加速度センサデータの品質については、SunSPOTによって1秒間に100回程取得される加速度データをどの程度までコマ落としするかをデータの品質とする。動画データの最高の品質が10fpsなので、加速度センサデータについても、最高の品質である10回/秒から最低の品質である1回/秒をそれぞれ入力データとした。

品質が異なる加速度センサデータのグラフを図9に示した。一番上のグラフが、最高の品質である10回/秒で、下のグラフに行くほど、品質が悪くなり、グラフが粗くなって行くのがわかる。

このように差を付けた2種類の入力データに対し、以下の2つの評価実験を行い、アプリケーションがどの程度低品質の入力データまで正解を出力し続けられるかを実験した。

(1) 評価実験1: 加速度センサデータの品質を10回/秒に固定し、動画データの品質のみを下げる。

(2) 評価実験2: 動画データの品質を10fpsに固定し、加速度センサデータの品質のみを下げる。

(3) 評価実験3: 動画データの品質と加速度センサデータの品質を共に下げる。

## 8. 評価実験結果と考察

### 8.1 正解の言語化回数

表2 正解の言語化回数(アプリケーション拡張前)

|     |    |    |    |    |    |    |    |    |    |    |
|-----|----|----|----|----|----|----|----|----|----|----|
| 閾値1 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 10 |
| 回数  | 5  | 13 | 5  | 9  | 16 | 6  | 15 | 9  | 9  | 10 |
| 閾値1 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 20 |
| 回数  | 3  | 7  | 4  | 5  | 8  | 1  | 7  | 5  | 3  | 3  |
| 閾値1 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 30 |
| 回数  | 0  | 5  | 0  | 2  | 5  | 1  | 4  | 1  | 2  | 1  |
| 閾値1 | 41 | 42 | 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 |
| 回数  | 0  | 3  | 0  | 1  | 3  | 0  | 2  | 0  | 0  | 1  |

表3 正解の言語化回数(アプリケーション拡張後)

|        |        |     |     |     |     |     |     |
|--------|--------|-----|-----|-----|-----|-----|-----|
| 閾値1=10 | 閾値1=11 |     |     |     |     |     |     |
| 閾値\閾値  | 閾値\閾値  |     |     |     |     |     |     |
| 0.1    | 861    | 661 | 461 | 0.1 | 661 | 661 | 461 |
| 0.2    | 551    | 351 | 221 | 0.2 | 551 | 351 | 221 |
| 0.3    | 451    | 231 | 221 | 0.3 | 451 | 221 | 221 |
| 0.4    | 231    | 231 | 111 | 0.4 | 231 | 221 | 111 |
| 0.5    | 221    | 121 | 011 | 0.5 | 221 | 121 | 011 |
| 閾値1=12 | 閾値1=13 |     |     |     |     |     |     |
| 閾値\閾値  | 閾値\閾値  |     |     |     |     |     |     |
| 0.1    | 661    | 561 | 451 | 0.1 | 561 | 551 | 441 |
| 0.2    | 561    | 351 | 221 | 0.2 | 561 | 351 | 221 |
| 0.3    | 451    | 221 | 221 | 0.3 | 451 | 221 | 221 |
| 0.4    | 231    | 221 | 111 | 0.4 | 231 | 221 | 111 |
| 0.5    | 221    | 121 | 011 | 0.5 | 221 | 121 | 011 |
| 閾値1=14 |        |     |     |     |     |     |     |
| 閾値\閾値  |        |     |     |     |     |     |     |
| 0.1    | 551    | 451 | 451 |     |     |     |     |
| 0.2    | 451    | 341 | 221 |     |     |     |     |
| 0.3    | 451    | 221 | 221 |     |     |     |     |
| 0.4    | 221    | 221 | 111 |     |     |     |     |
| 0.5    | 221    | 121 | 011 |     |     |     |     |

(注3): 2行目のCPTを参考にする

(注4): 3行目のCPTを参考にする

(注5): frame per second

評価実験では、2台のネットワークカメラでドア、机、椅子を撮影し、加速度センサ端末をドアと椅子に設置した状態で「人がドアを開ける」、「人が机を拭く」、「人がいすにすわる」という言語化判定を行った。

表2と表3にアプリケーションの拡張前[2]と拡張後の言語化回数を示した。表2は、「人がドアを開ける」のみの言語化を行った時の言語化回数を示したもので、正解の言語化回数は2回である。赤く網かけをしてある部分では、正解の言語化回数である「2」が得られているが、適切なタイミングで言語化されたのは、オレンジの吹出しの通り、それぞれ、「1回」、「0回」、「1回」であった。つまり、拡張前のアプリケーションでは、適切なタイミングで正解の言語化回数を得ることができていなかった。

これに対し、表3は、「人がいすにすわる」、「人がドアを開ける」、「人が机を拭く」の3つの行動について言語化を行った時の言語化回数を示したものである。表中の3桁の数字は、左から順に、いす、ドア、机の言語化回数となっている。拡張前のアプリケーションについては、閾値1のみであったのに対し、拡張後のアプリケーションでは、加速度センサデータを追加したので、加速度センサデータ処理用の閾値である閾値2と閾値3が表の縦軸と横軸になっている。3は閾値1を10-14、閾値2を0.1-0.5、閾値3を1-3まで変化させてアプリケーションを実行した結果である。表の右下に行く程、言語化の基準が厳しくなるので、言語化回数が減っている。網かけの部分に正解の言語化回数である221が得られた所である。

先行研究で用いられた拡張前のアプリケーションでは、言語化のタイミング、回数について、なかなか正解が得られなかったが、本研究でアプリケーションを拡張した後は、表3の通り、閾値を調節する事で、適切なタイミングで正解の言語化回数を得られるようになった。

## 8.2 入力データ品質の下限值

更に、表3の網かけの部分（正解の言語化回数を得られた所）に対して、閾値1、閾値2、閾値3を維持したままの状態ですべての動作について評価実験1~3を行った。図10のグラフの縦軸はデータ品質、横軸は表3の網かけ部分の左側についている数字と対応している。左側の青い棒グラフが評価実験1、右側の赤い棒グラフが評価実験2の結果を表している。グラフが短いほど、低品質の入力データでもアプリケーションが正解を出力できたという事を意味している。

図10において、青い棒グラフ（評価実験1）と赤い棒グラフ（評価実験2）の長さを比べると、ほとんどのグラフで赤い棒グラフが他よりも短い。つまり、動画データと加速度センサデータの品質では、加速度センサデータの方がある程度低い品質でも正解が得られ、それに対し、動画データはある程度高い品質が要求されるという事がわかる。

緑の棒グラフ（評価実験3）は他のグラフよりも長いので、全ての入力データである動画データと加速度センサデータの両方のデータの品質を下げる場合は、評価実験1と2、つまり、入力データのうち動画データのみ、または加速度センサデータのみを品質を下げる場合よりも、ある程度高品質の入力データ

が要求されるという妥当な結果が得られた。

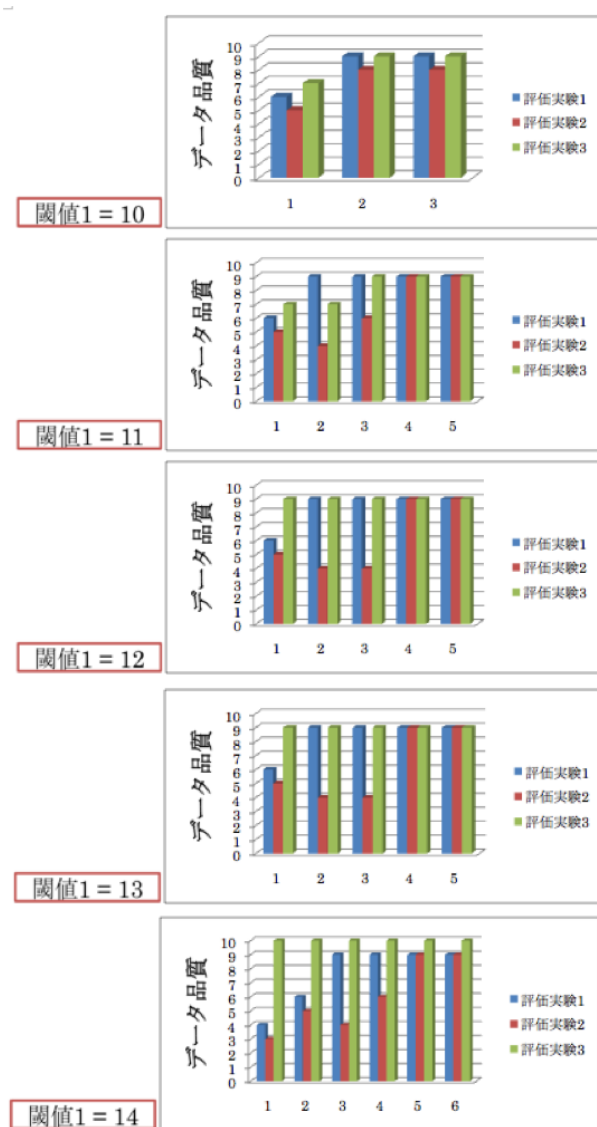


図10 最低限必要な入力データの品質

## 9. 実環境における評価実験

### 9.1 実験住宅 OchaHouse



図11 実験住宅 OchaHouse

研究室で以上の評価実験を行った後、実環境である実験住宅 OchaHouse<sup>(注6)</sup>[5]で収集した実データについて評価実験を行った。

(注6): お茶の水女子大学ユビキタスコンピューティング実験住宅: 人間の生活に身近な住宅において、人間と情報環境が調和に相互作用を行うための基盤技術の開発を目的とし、情報科学科、生活科学部が中心となって研究に利用している。

## 9.2 正解の言語化回数と入力データ品質の下限値

表 4 正解の言語化回数（実環境）

| 閾値1 = 10  |   |   |   | 閾値1 = 11  |   |   |   |
|-----------|---|---|---|-----------|---|---|---|
| 閾値2 \ 閾値3 | 1 | 2 | 3 | 閾値2 \ 閾値3 | 1 | 2 | 3 |
| 0.1       | 2 | 0 | 0 | 0.1       | 2 | 0 | 0 |
| 0.2       | 1 | 0 | 0 | 0.2       | 1 | 0 | 0 |
| 0.3       | 1 | 0 | 0 | 0.3       | 1 | 0 | 0 |
| 0.4       | 1 | 0 | 0 | 0.4       | 1 | 0 | 0 |
| 0.5       | 0 | 0 | 0 | 0.5       | 0 | 0 | 0 |

| 閾値1 = 12  |   |   |   | 閾値1 = 13  |   |   |   |
|-----------|---|---|---|-----------|---|---|---|
| 閾値2 \ 閾値3 | 1 | 2 | 3 | 閾値2 \ 閾値3 | 1 | 2 | 3 |
| 0.1       | 2 | 0 | 0 | 0.1       | 2 | 0 | 0 |
| 0.2       | 1 | 0 | 0 | 0.2       | 1 | 0 | 0 |
| 0.3       | 1 | 0 | 0 | 0.3       | 1 | 0 | 0 |
| 0.4       | 1 | 0 | 0 | 0.4       | 1 | 0 | 0 |
| 0.5       | 0 | 0 | 0 | 0.5       | 0 | 0 | 0 |

| 閾値1 = 14  |   |   |   |
|-----------|---|---|---|
| 閾値2 \ 閾値3 | 1 | 2 | 3 |
| 0.1       | 2 | 0 | 0 |
| 0.2       | 1 | 0 | 0 |
| 0.3       | 1 | 0 | 0 |
| 0.4       | 1 | 0 | 0 |
| 0.5       | 0 | 0 | 0 |

実環境における評価実験では、研究室における評価実験と同様、2台のネットワークカメラと加速度センサ SunSPOT からデータを収集した。言語化対象は「人がドアを開ける」のみで、SunSPOT はドアに取り付けた。

なお、表 4 の網かけの部分为正解の言語化回数が得られた所であり、閾値 1 が 10 ~ 14 のそれぞれの表で縦に 3 つ並んだ網かけを上から 1, 2, 3 とし、この番号は、図 12 の横軸の数字と対応している。

実環境では、言語化対象が 1 つのみであるので、評価実験 1 ~ 3 の棒グラフの長さの関係が図 10 とは異なる。これは、環境と言語化対象の行動数によって、アプリケーションの要求する入力データの品質が異なるという事を示している。

以上の 2 つの環境における最低限必要なデータ品質を次の章で比較する。

## 10. アプリケーションの要求条件に基づく入力データの品質

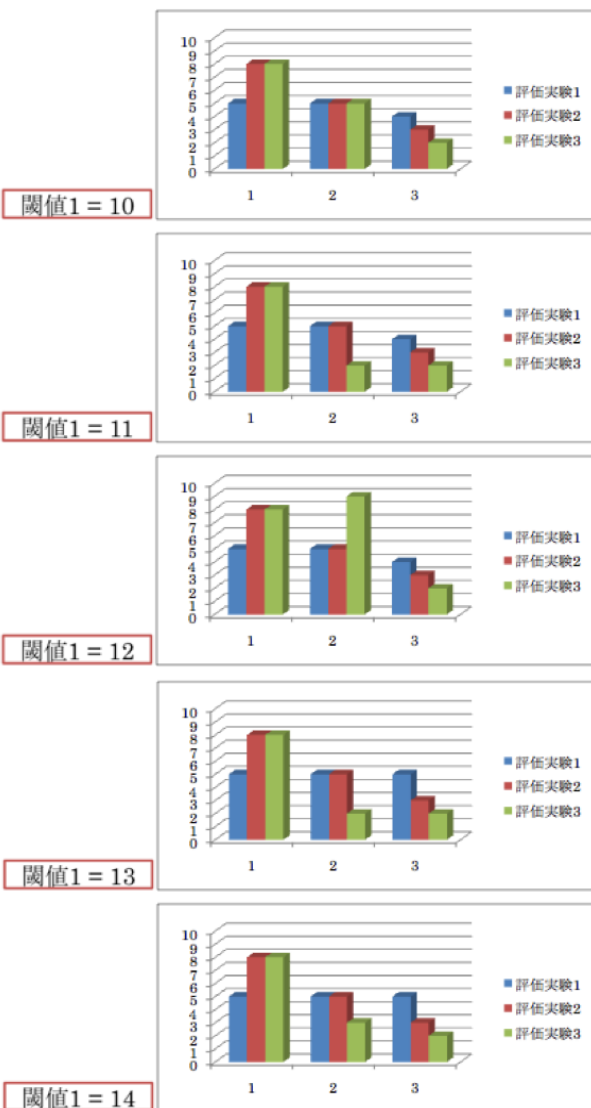


図 12 最低限必要な入力データの品質（実環境）

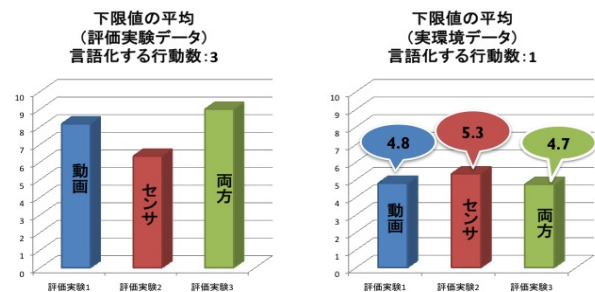


図 13 2 つの環境における入力データの下限値の平均

評価実験データと実環境データで行った評価実験 1 ~ 3 の棒グラフの値の評価実験毎の平均値を取ったものを図 13 に示した。青い棒グラフは動画データの品質のみを落とした場合、赤い棒グラフは加速度センサデータの品質のみを落とした場合、緑の棒グラフは動画データと加速度センサデータの両方のデータの品質を落とした場合のアプリケーションが正解を得るのに最低限必要な入力データの品質を表している。

全体的に評価実験データのグラフの方が長いので、言語化対象となる行動数がより多い方が、アプリケーションの要求する入力データの品質が高品質である事がわかる。

また、評価実験データでは、赤い棒グラフ（評価実験 2）が青と緑の棒グラフよりも短いのにに対し、実環境データでは、赤い棒グラフ（評価実験 2）が青と緑の棒グラフよりも長い。これは、評価実験データでは、ドアといすの 2 つのみに SunSPOT を取付け、机には取付けていなかったために、言語化の際に加速度センサデータはある程度低品質でもアプリケーションが耐えられ、一方、実環境データでは、SunSPOT を取付けたドアのみの言語化を行ったために、他の入力データよりも加速度センサデータの品質が重要であったという事を意味している。

## 11. まとめ

本研究では、既存の「人の行動を言語化するシステム」に対

して、「データ収集」、「データ処理」、「情報解析」の3階層を強化する事で、データ品質評価フレームワークを実現した後、このデータ品質評価フレームワークを用いて、ライフログ解析アプリケーションでは、あまり考慮されて来なかった入力データの品質評価を行った。

評価実験の結果、既存のシステムに対し、加速度センサデータのノードとベイズ判定を適用した事により、言語化のタイミングが改善され、従来のアプリケーションと比較して、安定した言語化回数の正当率が得られるようになったという事がわかった。また、正解の言語化回数を得られた場合について、入力データの品質を下げていくと、アプリケーションがどこまで正解を出力できるかを評価した結果、複数の入力データの品質を下げるよりも、1つの入力データの品質を下げた方が、アプリケーションに及ぼす影響が少ないという妥当な結果が得られた。

更に、実環境で収集された実データについても評価実験を行うことで、環境によってアプリケーションが要求する入力データの品質の差を比較した。

## 12. 今後の課題

本論文では、本研究で提案した「データ収集」、「データ処理」、「情報解析」の3層で構成されるデータ品質評価フレームワークに対し、「人の行動を言語化するアプリケーション」を適用して評価実験したが、実データを用いて、他の処理をするアプリケーションについても評価実験をしていきたい。

## 謝 辞

本研究を進めるにあたり、大変有用なアドバイスを頂いたお茶の水女子大学の小林一郎教授、小林研究室(M1) 落合恵理香さんに深く感謝いたします。

## 文 献

- [1] 落合恵理香, 小林一郎, 特定空間における人の行動予測モデルに基づく言語化への取り組み, The 24th Annual Conference of the Japanese Society for Artificial Intelligence, 2010
- [2] 岩木紗恵子, 村瀬勉, 小口正人, ライフログのためのセンサデータと画像ストリーム処理ミドルウェアの構築, DEIM Forum2010, B3-1, 2010年3月
- [3] サン・マイクロシステムズ-SunSPOT 無線センサーネットワークでバイス  
<http://www.jp.sun.com/products/software/sunspot>
- [4] PLANEX CS-WMV04N  
<http://www.planex.co.jp/product/camera/cs-wmv04n/>
- [5] ochaHouse PukiWiki  
<http://ochahouse.com/>