

# 行動履歴を利用した IoT 家電制御システムの設計

2016SE015 林知博 2016SE057 森下颯斗

指導教員：沢田篤史

## 1 はじめに

近年冷蔵庫やエアコンといった家電など、日常生活で使用する機器がインターネットに接続されるようになった。それらの機器は IoT 機器と呼ばれ、ユーザはスマートフォンなどの端末を使って IoT 機器を操作したり、IoT 機器の稼働状況や周辺状況を確認したりすることができる。外出時にもスマートフォン 1 つで家庭内の機器を操作ができる。このように機器がネットワークに接続することによってその用途の可能性を広げている。また、家電機器の管理や制御を行うことでエネルギーの節約を目的とし、近年新築住宅への導入が検討されている。

スマートホームにおける家電機器制御の問題点はユーザの行動に応じた複数機器の連携が十分でないことである。スマートホームの今後の方向性として、ユーザの操作を予測し実行する自動化が挙げられている。スマートホームでは、ユーザが個々の家電に行った操作について様々な情報を記録できる。現在では、それらの情報を必ずしも十分に活用できていない。例えば、複数機器に対する連続操作の自動化などができていない。ユーザの行動予測のための共通基盤により家電連携アプリを容易に行えるようにする必要がある。

本研究では、個人の行動パターンに合わせた複数家電操作を容易に実現するためのソフトウェア基盤の構築を目標とする。家庭内での個人の行動パターンに合わせるために操作履歴に基づく住人の次の操作予測を行う。ユーザの宅内行動には電灯を点けることや、エアコンで冷暖房を利用するなど必ず行う動作が存在する。それらの行動は時間帯や季節ごとの室温など、外的環境に依存する。同じ条件下でのユーザの行動には連続性があると考え、操作履歴をもとに高頻度の操作から連続性など関連を見つけ出すことで予測する。この予測方式を組み込んだソフトウェアを容易に構築するためのソフトウェアアーキテクチャを定義することを目的とする。なお、家庭内には複数人のユーザが存在し、連続操作が複数人で行われた場合もあるので、本研究では宅内で機器操作するユーザを一人に限定し、他のユーザの操作（ノイズ）を考慮しないものとする。

本研究では、行動履歴の構造化と履歴を利用した予測手法の提案を行う。具体的には、利用者の行動履歴の収集、予測のための仕組みの作成のためにログから N-gram の構築と確率の計算を行う。また、ソフトウェア基盤としてのアーキテクチャ設計を行う。家庭における人間の行動を家電操作の観点から分析すると、どの機器をどの順番で操作するかが重要である。機器および操作の選択を効率よく取り扱うことが必要である。さらに本研究では行動履歴を利用するために、連続したパターンの回数を利用し頻度（確

率）を算出する、N-gram を用いて、条件付き確率を算出する。

## 2 スマートホームにおける個人適応にむけて

### 2.1 スマートホーム

スマートホームとは、IoT や AI などの技術を駆使して、住む人にとって安心・安全で快適な暮らしを実現する住宅のことである。一般的には、物理的な住居そのものではなくそこに装備されたシステムを指すことが多い。多様な機器の組み合わせで住む人のニーズに合わせた利便性を提供する。Amazon の音声認識デバイス「Amazon Echo」[1] が登場して以降、IoT やスマートホームといった言葉が身近になった。玄関のドアに専用の IoT 機器を付属させることでキーレス化が可能となった。冷蔵庫やエアコンなどの家庭用電化製品に加え、宅内照明や浴室の給湯器などが IoT 家電として普及し始めたことにより、暮らしの利便性が向上した。スマートホームの今後の方向性として複数機器を連携させる操作の自動化が挙げられている。例えば、連続性の高い操作パターンに合わせて操作を行ったり、自動化した操作に対して評価を行うことで制度を向上させたりすることである。

### 2.2 スマートホームの個人適応における課題

IoT 家電の多様化により、家電間の連携が将来的に実現される。それぞれの機器操作を反映させた個人に適応する機器操作の自動化はその中でも未だ実現されておらず、今後の課題となっている。この問題に対し、山内らの研究 [6] ではイベント系列を作成することを行い、長谷川らの研究 [5] では状態遷移を用いることを行っている。

山内らの研究 [6] では各ユーザに合った行動履歴を木のモデルで作成し、実際に被験者と IoT 家電を設置し、IoT 機器に対するサイバー攻撃などの異常検知を行っていた。学習用データセットとテスト用データセットに分け、テスト用データセットの中でのサイバー攻撃に対して日頃のユーザの行動履歴の一連の動作の一部なのか、外部からの操作なのかを木のモデルを根から葉までを辿ることで検知している。

長谷川らの研究 [5] ではスマートベットを題材に就寝時の行動履歴をもとにベットに付属するセンサから得るバイタルデータと組み合わせた異常検知と親切な振る舞いを設計した。行動履歴から振る舞いを決定する方法として状態遷移モデルを利用している。

これらの二つの研究はイベント系列や状態遷移を利用することで行動に対する次の操作が決定的であることが言える。本研究では状態遷移を確率的に表し、ソフトウェアの基盤を作成する。

### 3 行動履歴を利用した IoT 家電制御システムの設計

#### 3.1 設計指針

本研究では行動履歴を利用した IoT 家電制御システムを設計する。行動履歴とは、宅内で一定時間内に連続して行われた機器操作の列のことである。連続して行われる操作の中には頻度の高い操作、連続性の高い操作列が存在し、これらは長期間記録した操作履歴から頻度をカウントすることで明らかになる。N-gram を用いて分割することにより、高頻度な連続操作を見つけだし柔軟な次の行動予測が可能となる。

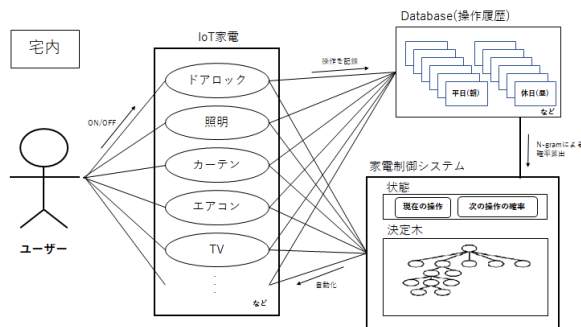


図 1 概念図

ユーザーが宅内で IoT 家電を操作した記録を Database に記録する。記録した操作履歴をもとに確率を算出し、時系列を考慮した木の構築を行う。家電制御システム内では作成した木をたどりながらリアルタイムで行われている操作と次に起こりうる操作の確率を持つようにし、閾値 (80%) を超えた場合のみ、次に起こりうる操作を自動化するよう IoT 家電に命令を出す。自動化した操作の直後に否定の操作が行われた場合は、誤った自動化であるので、N-gram のカウントを半分にするなどの処理することで、以降に誤った自動化を防止する。操作履歴をもとに N-gram、木の更新を行うタイミングは、誤った自動化が行われた場合を除き、ユーザの行動が少ない深夜のタイミングで行うものとする。

#### 3.2 行動履歴の基本構造

本研究では行動履歴を宅内で一定時間内に連続して行われた機器操作の列と定義する。また、操作予測する際の連続操作回数を 5 回までに限定して行う。これは人間が入室してから連続して行う意味のある機器操作は一般的に最大で 5 回であると考えたからである。人間の行動は時期や時間帯に影響を受けると考えたので利用する行動履歴は常に 1 ヶ月間のデータを保持し、1 日分の行動履歴を取得する毎に 1 ヶ月前の履歴を消去する。日にちは平日 (月 金) と休日 (土, 日, 祝), 時間帯は朝, 昼, 夕, 夜と分割することとした。

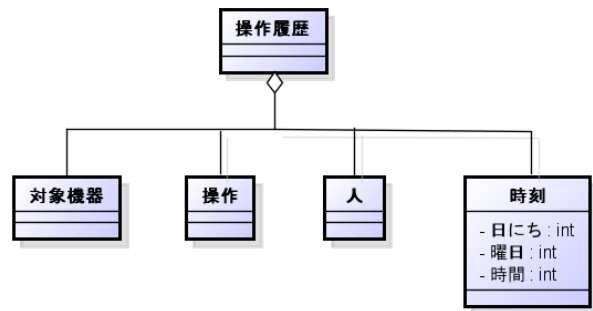


図 2 行動履歴の構成

- 対象機器  
操作の対象となる機器を示す。家庭毎で使用される IoT 機器の種類や数は異なっている。
- 操作  
機器操作の内容を示す。本研究では IoT 家電の電源を点ける、消すの 2 パターンに限定する。
- 人  
操作を行った対象の人物。
- 時刻  
機器操作の行われた時刻を示す。日にち, 時間, 曜日

#### 3.3 N-gram を用いた行動予測手法

##### 3.3.1 N-gram

N-gram モデルは、確率・統計的自然言語処理の分野で最も広範に使われている言語モデルである。N-gram では対象となる文書を文字単位の記号列と考え、隣接した N 個の記号毎の出現頻度 (度数と呼ぶ) を集計する。隣接した 1 個の記号からなる記号列 (つまり, 1 文字) 毎の度数を uni-gram, 2 個の記号を bi-gram, 3 個の記号ならを tri-gram と呼ぶ。一般に N-gram モデルとは単語の生起が直前の (N-1) 単語にのみ依存すると考えた言語モデルである。

##### 3.3.2 行動予測方式

本研究では日付や時間など条件ごとの行動履歴 1 ヶ月分のデータを利用し、連続する最大回数 5 回までを考慮すべく、1~5gram をそれぞれ作成する。学習データから 1~4 回の連続操作の後に起こりうる操作を条件付き確率 [4] を求め、確率が大きい操作を予測するという方法である。

条件付き確率は、学習データ中に出現する操作列の相対頻度から求めることができる。取得した家電操作履歴の列を  $w_1^n = w_1 \cdot \dots \cdot w_n$  とする。例として、 $w_1$ (照明をつける) $w_2$ (エアコンを付ける) という連続操作の後に  $w_3$ (テレビを付ける) という操作が行われる確率を計算する場合、bi-gram で  $w_1 w_2$  の出現頻度, tri-gram で  $w_1 w_2 w_3$  の出現頻度を算出する。出現確率の計算は以下の式を用いることとする。

$$P(w_3|w_1w_2) = \frac{P(w_3 \cap w_1w_2)}{P(w_1w_2)}$$

$$P(w_3 \cap w_1w_2) = \frac{w_1w_2w_3 \text{の総数}}{\text{tri-gram の総数}}$$

$$P(w_1w_2) = \frac{w_1w_2 \text{の総数}}{\text{bi-gram の総数}}$$

### 3.4 アーキテクチャ設計

本研究では宅内で一定時間内に行われた機器操作の履歴を記録する。この中で頻度の高い操作や連続性の高い操作列を明らかにするために N-gram を用いる。また、予測にはどの機器をどの順番で操作するかが重要であり、機器および操作の選択を効率良く取り扱う必要があるため機器操作を木構造で扱う。自動化の際は、新たな機器操作が行われたら木構造のノードをたどり、あればひとつ先のノードをたどる。なければ待機する。各ノードが持つ条件付き確率で計算した確率と閾値を比較することで閾値以上ならば先のノードの操作を自動化する。このような観点からアーキテクチャを設計する。

#### 3.4.1 デザインパターン

デザインパターン [2] とはソフトウェア設計時の典型的な問題に対し、その解決策に名前を付けて整理し、再利用しやすいようにカタログ化したものである。我々の設計したアーキテクチャでは、N-gram を利用し木構造を構築するので Builder パターン、Composite パターンを、木構造の各ノードをたどり閾値と比較するので Interpreter パターン適用している。

- Builder パターン

Builder パターンではオブジェクトの生成過程を抽象化することによって、動的なオブジェクトの生成を可能にする。また、Director クラスでは Builder クラスのインタフェースを使って、Product オブジェクトを段階的に組み立てる正しい順序を管理している。Builder パターンを利用するメリットはオブジェクト生成過程と生成手段を局所化できる点である。Director クラスにはオブジェクト生成家庭のコードだけが、ConcreteBuilder クラスにはオブジェクトの生成手段のコードのみが記述されることにより、それぞれに関するコードを凝縮することができ、独立して修正・拡張することができる。

- Composite パターン

Composite パターンでは、オブジェクトを木構造に組み立てるのが特徴である。ファイルとディレクトリなどのような、木構造を伴う再帰的なデータ構造を表すことを可能にする。Composite パターンにおいて登場するオブジェクトは「枝」と「葉」であり、これらを表現するオブジェクトをクライアントから一様に扱えるようにする。Composite パターンを利用するメリットとしてまずクライアント側の操作が容易になる点が挙げられる。木構造の「枝」の部分

にも「葉」の部分にも同じ処理でアクセスすることが可能である。また、新しい枝を簡単に追加できることもメリットといえる。枝葉を同一視することができるため、他の Component クラスや Leaf クラスを修正することなく、新しい枝を追加することができる。

- Interpreter パターン

Interpreter パターンは何らかのフォーマットで書かれたファイルの中身を解析し、解析した結果から得られた手順に基づいた処理を実現するために最適なパターン。Interpreter パターンを利用するメリットとして規則の追加や変更が可能であることが挙げられる。

#### 3.4.2 アーキテクチャ

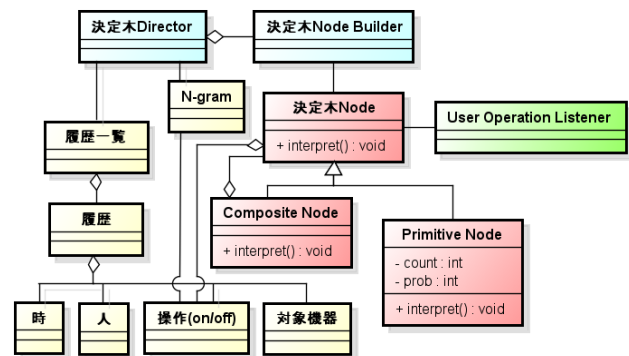


図3 本研究を考慮したアーキテクチャ

本研究では宅内で一定時間内に行われた機器操作の履歴を記録する。時間、対象人物、操作、対象機器が記録されている履歴を、日にち（平日、土日祝）や時間帯（朝、昼、夕、夜）など条件ごとに集約したものが履歴一覧である。この中で頻度の高い操作や連続性の高い操作列を明らかにするために N-gram を用いる。

操作履歴に対して N-gram でカウント、確率算出した後に同じ手順で何度も木構造を構築させることや、決定木が持つノードの表現形式を変更したい場合などの踏まえ本研究では Builder パターンを用いる。履歴一覧や N-gram での確率計算などの情報を持つ決定木 Director が決定木 NodeBuilder に対し木の作成の指示を出す。決定木 NodeBuilder では RootNode の作成、CompositeNode、PrimitiveNode の作成、削除を行う。Director では実際に確率算出した操作パターンを反映させていく。この結果、操作パターンを考慮した決定木が生成される。

本研究では操作履歴を N-gram 化、確率算出した後に操作列を木構造で表現する。木のすべてのノードは実際の操作と事前の操作列かその操作が起こりうる確率を所持する CompositeNode は内容物として複数の PrimitiveNode や CompositeNode の所持が可能である。枝葉同一の処理である点や、新しく IoT 機器が増設された場合の操作の追加に柔軟に対応できる点、つまり階層構造や木構造で表現されるオブジェクトの取り扱いを容易にする点から、

Composite パターンを用いることとした。

本研究では自動化を行う際に、木構造を表現した Composite パターンと構造的に同様な Interpreter パターンを用いる。「処理」(木をたどり閾値と比較するなど)と処理対象(操作列)と分けた場合、「処理対象」は「処理」を踏まえた結果を必要とするため Composite パターンと同様の構造になっている。IoT 機器から得られた操作が木構造のノードにあるならひとつ先のノードをたどり、なければ待機する。各ノードが持つ条件付き確率と閾値を比較し閾値以上であれば自動化するという処理を行う。また、UserOperationListener は Client の役割を持っており、Interpreter パターンを適用したクラスを利用し処理する。

## 4 考察

本研究では各ユーザの行動履歴を利用した IoT 家電システムの設計をした。ユーザから機器操作するのではなく、システムから能動的に家電を制御することにより、ユーザの利便性が向上することが本研究の利点であると考えられる。

### 4.1 提案したアーキテクチャに関する考察

本研究ではアーキテクチャの設計において、生成、構造、振る舞いをそれぞれ Builder パターン、Composite パターン、Interpreter パターンを用いて表現した。まず Builder パターンとインスタンスを生成する Factory パターンと類似している。Factory パターンは生成に必要なパラメータを秘匿しており、動的に引数が変わる場合などに力を発揮する Builder パターンを用いることとした。時系列データ木構造取り扱うことから、本研究では新たな操作の追加、誤った自動化に対してのノードの削除など柔軟に対応すべく Composite パターンを用いている。最後に振る舞いに関して、Composite パターンを作成した場合に適応できる Interpreter パターンを利用した。比較するデザインパターンとして Visitor パターンなどが挙げられる。

### 4.2 関連研究との比較

関連研究との比較として状態遷移を確率的に表し、ソフトウェアの基盤を作成している点が挙げられる。

山内らの研究 [6] では各ユーザに合った行動履歴を木のモデルで作成し、日頃のユーザの行動履歴の一連の動作の一部なのか、外部からの操作なのかを木のモデルを根から葉までを辿ることで検知していた。根から葉まで辿った場合、必ず異常と検知されてしまう点に対し、本研究ではユーザ自ら否定の動作をした場合に閾値を減らす処理がされるのでユーザが望む動作をシステムから能動的に行うことが可能になる。

長谷川らの研究 [5] ではスマートベットを題材に就寝時の行動履歴をもとにベットに付属するセンサから得るバイタルデータと組み合わせた異常検知と親切的な振る舞いを設計した。行動履歴から振る舞いを決定する方法として状態遷移モデルを利用しており、次の行動の決め方が決定的で

ある。それに対し本研究では、N-gram の更新を 1 日毎に行うことで過去の履歴から新しい状況に柔軟に適応することが可能である。

## 5 おわりに

近年冷蔵庫やエアコンといった家電など、日常生活で使用する機器がインターネットに接続されるようになった。ユーザはスマートフォンなどの端末を使って IoT 機器を操作したり、IoT 機器の稼働状況や周辺状況を確認したりすることができる。スマートホームの今後の方向性として、ユーザの操作を予測し実行する自動化が挙げられている。スマートホームにおける家電機器制御の問題点はユーザの行動に応じた複数機器の連携が十分でないことである。本研究では、個人の行動パターンに合わせた複数家電操作を容易に実現するためのソフトウェア基盤の構築を目標とする。その中で家庭内での個人の行動パターンに合わせるために操作履歴に基づく住人の次の操作予測を行う。本研究では、行動履歴の構造化と履歴を利用した予測手法の提案を行った。具体的には、予測のための仕組みの作成のためにログから N-gram の構築と確率の計算、ソフトウェア基盤としてのアーキテクチャ設計である。ユーザから機器操作するのではなく、システムから能動的に家電を制御することにより、ユーザの利便性が向上することが本研究の利点であると考えられる。実装まではいたらなかったため今後は実際に試作品を制作して妥当性を検証する必要があると考えられる。

## 参考文献

- [1] AmazonEcho: Easy operation of IoT home appliances, <https://www.amazon.co.jp/b?ie=UTF8&node=5364343051>, 2017.
- [2] Eric Gamma, オブジェクト指向における再利用のためのデザインパターン (改訂版), ソフトバンククリエイティブ, 1994.
- [3] 川原圭博, 司化, 猪鹿倉知広, 登内敏夫, 森川博之, 青山友紀, ”行動履歴と制約条件を考慮した情報家電制御機構”, 情報処理学会研究報告 (IPSJ SIG Technical Reports), 2006 号, 14(MBL-36 UBI-10) ページ, 2006.
- [4] 北研二, 確率的言語モデル, 東京大学出版社, 2006.
- [5] 長谷川結佳, 間瀬友美子, ”行動履歴を利用したコンテキスト指向組み込みシステムのソフトウェアアーキテクチャに関する研究”, 南山大学理工学部卒業論文要旨集, 2018.
- [6] 山内雅明, 大下裕一, 村田正幸, 上田健介, 加藤嘉明, ”スマートホーム IoT におけるユーザ行動の学習に基づく異常検知手法”, 信学技報, vol.117, no.353, IN2017-58, pp.73-78, 2017.