

双方向対話を活用した個人適応型スマートシステムの設計と実現

中田 匠哉[†] 佐伯 幸郎^{††} 中村 匡秀^{†,††}

[†] 神戸大学 〒 657-8501 神戸市灘区六甲台町 1-1

^{†††} 理化学研究所・革新知能統合研究センター 〒 103-0027 東京都中央区日本橋 1-4-1

^{††} 高知工科大学 〒 782-8502 高知県香美市土佐山田町宮ノ口 185

E-mail: [†]tnakata@es4.edept.kobe-u.ac.jp, ^{††}saiki.sachio@kochi-tech.ac.jp, ^{†††}masa-n@cmds.kobe-u.ac.jp

あらまし 普及と複雑化が急速に進むスマートシステムにおいて、ユーザが要望通りに設定を構築することは容易ではない。我々の先行研究では、対話によってサービス設定を自動化することを目的として、対話型個人適応フレームワーク SSIP の概念設計を行った。しかし、対話の技術的困難さから具体的な SSIP の実現には至っていなかった。LLM の発展により対話技術が容易化されたことを受け、本研究では SSIP の具体的な技術設計に取り組む。SSIP の主要な要素として双方向対話・ニーズ抽出・機能決定アルゴリズム・サービス自動設定を切り出し、特定の技術によらない汎用的なフレームワークとして実現する。ケーススタディでは提案した設計手法に従って SSIP の実装を行う。本研究によって、スマートシステムの対話型個人適応の実現手法が明らかになった。

キーワード 個人適応, スマートシステム, 自己適応システム, 対話, ニーズ

Design and Implementation of Smart Systems with Interactive Personalization

Takuya NAKATA[†], Sachio SAIKI^{††}, and Masahide NAKAMURA^{†,††}

[†] Kobe University, 1-1 Rokkodai-cho, Nada-ku, Kobe, Hyogo, 657-8501 Japan

^{†††} Riken AIP, 1-4-1 Nihon-bashi, Chuo-ku, Tokyo, 103-0027 Japan

^{††} Kochi University of Technology, 185 Tosayamadacho Miyanokuchi, Kami, Kochi, 782-8502 Japan

E-mail: [†]tnakata@es4.edept.kobe-u.ac.jp, ^{††}saiki.sachio@kochi-tech.ac.jp, ^{†††}masa-n@cmds.kobe-u.ac.jp

Abstract In smart systems, which are rapidly increasing in use and complexity, it is not easy for users to build configurations as they desire. In our prior study, we conceptually designed an interactive personalization framework, Smart System with Interactive Personalization (SSIP), with the aim of automating service configuration through dialogue. However, due to the technical difficulty of dialogue, we have not been able to realize a concrete SSIP. This study addresses the technical design of SSIP. The main elements of SSIP are interactive dialogue, needs elicitation, function determination algorithm, and automatic service configuration, and are implemented as a general framework that does not depend on specific technologies. In the case study, SSIP is implemented according to the proposed design methodology. This study reveals a method for realizing interactive personal adaptation of smart systems.

Key words Personalization, Smart system, Self-adaptive system, Dialogue, Need

1. はじめに

近年、スマートシステムの普及が進んでいる。スマート家電・IoT などの物理的なスマート製品が多く利用されるようになった。様々な赤外線リモコンの代用が可能なスマートリモコンや、スマートロック・スマートタグなどの手軽な商品も登場しており、冷蔵庫・掃除機・エアコンなど、大型家電のスマート化も進んでいる [1], [2]。SNS・メール・動画サイトなどの Web アプリケーション・スマートフォンアプリケーションもスマートシステムの一部である。スマートシステムの操作はデバイスによ

て可能であり、スマートフォンで製品やアプリケーションの連携設定が可能である。スマートスピーカーでは、キーワードを話すことで機能が実行でき、スマートスピーカーがおすすめの機能を提案する [3], [4]。このように、スマートシステムの利便性が向上する一方で、スマートシステムを構成する要素が増え、システムの全体像が複雑化している [5]。そのため、ユーザが自身の要望をどうやって実現すればよいか分りにくくなっており、ユーザの手動設定によらないサービス個人適応が求められている。我々の先行研究では、個人適応フレームワーク Smart System with Interactive Personalization (SSIP) の概念設計を行っ

た[6]。SSIPは、対話に基づいてユーザの要望をサービスに自動反映するフレームワークである。しかし、高度な対話が技術的に容易でなかったため実現に至っていなかった。他の先行研究において対話型ニーズ抽出システムの研究開発を行うなど、SSIPの部分的な実現は達成されていた。

本研究では、SSIP全体を実現するために、SSIPフレームワークの設計を技術的に具体化する。対話型ニーズ抽出システムを抽象化してSSIPに組み込むなど、特定の技術によらない汎用的な設計手法を構築する。提案手法では、SSIPを構成する主要要素として、双方向対話・ニーズ抽出・機能決定アルゴリズム・サービス自動設定を定義し、各要素について詳細に設計手法を定義・議論する。すなわち、ユーザとエージェントが対話を行い、要求抽出や機能提案を通じてサービス設定を自動化するシステムを設計する手法を明らかにする。ケーススタディでは、提案した設計手法に従って実際にSSIPを構築することで、SSIPの拡張性や発展性を明らかにする。本研究によって、スマートシステムの対話型個人適応が具体的に実現可能になる。

2. 準備

2.1 サービス個人適応

議論を明確化するために、サービス・機能・サービス個人適応の3つの用語について本論文中での意味を定義する。サービスの定義は、アプリケーションが提供するユーザ体験である[7]。サービスはユーザごとにサービス設定を待つものとする。機能の定義は、ユーザ体験の詳細な仕様である。機能はサービス設定によって変化するものとする。サービス個人適応の定義は、サービス設定を要望通りに変更し、ユーザを満足させることである。

従来のサービス個人適応は、ユーザ自身によって手動で行われてきた。ユーザが自身の要望を解釈し、サービス設定方法を自身で考え実行する手法である。従来手法の課題は3つある。ユーザは自分のニーズを個人設定に反映させる方法が分からず、提供されるサービスの内容を把握できず、そもそも自分自身のニーズを正確に把握していない。また、スマートシステムにおけるサービス個人適応に焦点を当てると、手動での個人適応がさらに困難になっている。スマートシステムはIoT機器やスマートフォンなどの構成要素が増えて全体像が複雑化しており、自分の要望をどうやって実現すればいいかユーザ視点で難解になっている。スマートシステム構築の難解さから、AR技術でオートメーション構築を補助する技術も研究されている[5]。そのため、ユーザによる手動設定以外のサービス個人適応が求められている。

2.2 先行研究1：SSIP[6]

先行研究では、対話によって個人適応手法を実現するフレームワークSSIPを概念設計した。先行研究において、従来の手動個人適応における課題を解決するSSIPの3要件を定義した。

- (要件 R1) ユーザごとに個人設定を定義でき、システムはその設定に応じたサービスを提供すること
- (要件 R2) 運用中継続的に、ユーザと次の2種類の対話をする：〈対話 C1〉要求抽出対話、〈対話 C2〉機能提案対話

- (要件 R3) システムは、運用中継続的に、対話の内容に基づいて個人設定を更新し、サービスに反映すること

簡易的なSSIPの例として、スマートスピーカーがある。Amazon Alexaでは、ユーザをアカウント管理し、キーワードによる簡易対話によって機能呼び出しや機能提案を実現する[3]。SSIPの先行研究では、意味解釈に基づく対話技術の技術的困難さから高度なSSIPの実装を断念した。近年、大規模言語モデル(LLM)の発展によって自然言語の意味解釈技術が飛躍的に進歩した[8]。そのため、LLMを用いて高度なSSIPを実現できる可能性が出てきたため、本研究の着手に至った。

2.3 先行研究2：対話型ニーズ抽出システム[9]

先行研究では、SSIPにおける〈対話 C1〉要求抽出対話を実現する技術として、対話型ニーズ抽出システムを研究開発した。要求を6W1Hで表現することで、エージェントとの対話によって自動的にユーザニーズを抽出・蓄積することを可能にする。6W1Hニーズデータモデルを以下の通り定義した。

- how：使いたいサービスは何か
- why：なぜしたいか
- when / where：いつ／どこでしたいか
- what：サービスで具体的に何をしたいか
- who / whom：誰が主体で／誰に対してしたいか

エージェントとのニーズ抽出対話では、最初にどのサービスを利用したいかを特定する。6W1H要素を穴埋めする問い返し対話を繰り返し、要望についてユーザ自身でじっくり考え直す機会を提供する。抽出したニーズが正しいかどうか、ユーザ自身が判断する。Webアプリ上のシンプルなチャットボット対話、ヴァーチャル・エージェント(VA)によるフレンドリーな会話などエージェントの形式は問わない。

2.4 SSIP実現の現状と技術的課題

SSIPの3要件を実現するために解決すべき課題を整理する。SSIPの(要件 R1)では、個人識別と設定の分離が必要だが、技術的には困難ではない。(要件 R2)では〈対話 C1〉と〈対話 C2〉を同時に実現する必要があるため、2種類の対話を含む全体の対話フローの設計が必要となる。〈対話 C1〉要求抽出対話は先行研究2：対話型ニーズ抽出システムで実現したが、ニーズ抽出を一般的に議論できていない。〈対話 C2〉機能提案対話を実現するためには、どのようなアルゴリズムで機能提案するかを明らかにする必要がある。(要件 R3)では個人設定の更新とサービスへの反映が必要であるため、どのように対話内容を設定とサービスに反映するかを明らかにする必要がある。本研究では以下の4つの課題の解決に取り組む。

- (課題 P1) 〈対話 C1〉と〈対話 C2〉の2種類の対話を含む全体の対話フローの設計
- (課題 P2) ニーズ抽出の一般化
- (課題 P3) どのようなアルゴリズムで機能提案するか
- (課題 P4) どのように対話内容を設定とサービスに反映するか

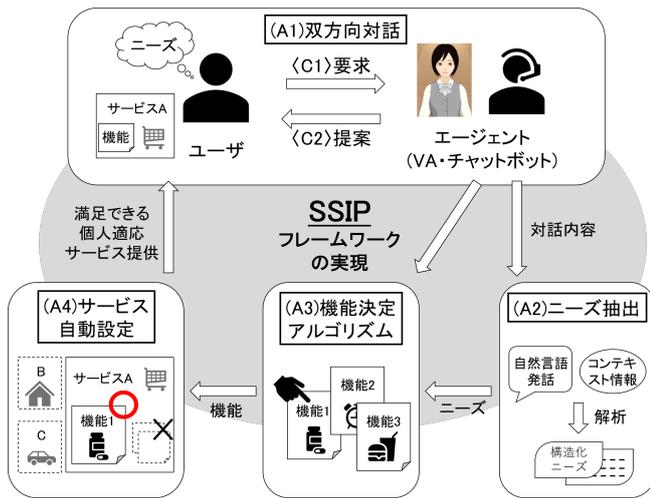


図 1: SSIP フレームワーク設計アーキテクチャ

3. 提案手法

3.1 目的とキーアイデア

本研究の目的は、ユーザのサービス要望を対話に基づいて自動的に実現するスマートシステムの実装手法を確立することである。キーアイデアとして、先行研究 1 における SSIP フレームワークの概念設計をもとにして、SSIP フレームワークを技術的に具体化・詳細化する。また、先行研究 2 の対話型ニーズ抽出システムを SSIP フレームワークの中で一般的に議論しなおす。さらに、先行研究で未実現の〈対話 C2〉を含むいくつかの要素の実装手法を議論する。対話内容の生成・解釈に LLM を活用することで、ユーザと自然な対話が可能で実用的な対話型個人適応システムを実現する。具体的な実装例はケーススタディにおいて与える。本研究は以下の 4 つのアプローチに取り組み、(課題 P1) から (課題 P4) を解決する。

- (A1) 双方向対話
- (A2) ニーズ抽出
- (A3) 機能決定アルゴリズム
- (A4) サービス自動設定

3.2 全体アーキテクチャ

提案手法の全体アーキテクチャを図 1 に示す。(A1) から (A4) は SSIP のコアとなる要素であるが、特定の技術を指すものではない。様々な技術を各要素に用いることで、柔軟な SSIP の開発を可能とすることがフレームワークの目的である。

まず、(A1) の双方向対話によってユーザの要求と提案への返答を得る。〈対話 C1〉要求対話を行う場合は (A2) のニーズ抽出を行う。(A3) のアルゴリズムによる機能決定を経て (A4) のサービス自動設定を行い、ユーザが満足する個人適応サービスを提供する。各要素のバックグラウンド技術として、エージェントによるユーザとの対話技術が必要となる。対話内容の生成・解釈には LLM を代表とする自然言語処理技術を活用することで行う。

3.3 (A1) 双方向対話

〈対話 C1〉と〈対話 C2〉を含む双方向の個人適応対話を実

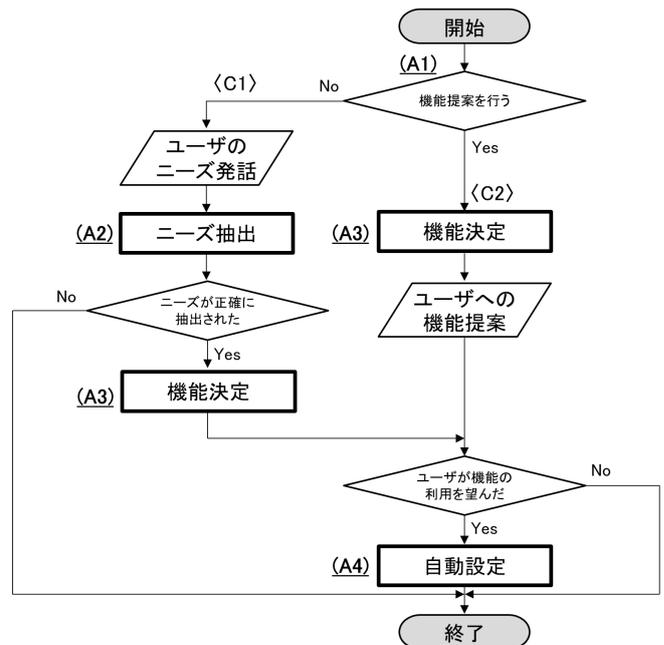


図 2: 個人適応フロー

現する必要がある。各対話の具体的な処理フローを定義する。また、2 種類の対話をどのように開始するかを設計する。図 2 に〈対話 C1〉と〈対話 C2〉を含む全体の処理フローを示す。〈対話 C1〉要求抽出対話は、ユーザ主導の対話である。ユーザの発話をエージェントが聞き取ってニーズを抽出する対話で、先行研究 2 の対話型ニーズ抽出システムが実現技術の一例である。〈対話 C1〉による個人適応の処理フローは、(1) 〈対話 C1〉、(2) ニーズに合致する機能の決定、(3) サービス自動設定である。フローの中断条件は、正確なニーズ抽出に失敗した場合および、提案機能の利用を望まない場合である。〈対話 C2〉機能提案対話は、エージェント主導の対話である。エージェントによる機能提案に対してユーザが利用するか答える対話である。〈対話 C2〉による個人適応の処理フローは、(1) ユーザに適切な機能の決定、(2) 〈対話 C2〉、(3) サービス自動設定である。フローの中断条件は、ユーザが提案機能の利用を望まない場合である。個人適応対話はサービス自動設定後に終了する。個人適応対話終了後の処理は未定義であり、各 SSIP にゆだねられる。例えば、対話全体を終了する、その他の対話に繋げる、再度個人適応対話を開始するなどがある。

〈対話 C1〉と〈対話 C2〉のどちらの対話をいつ開始すべきかが明確でないため、対話の開始条件を定義する必要がある。まず、ユーザによる能動的発話がある場合を考える。発話に要求が含まれる場合、〈対話 C1〉を開始する。発話が機能提案を求める内容である場合、〈対話 C2〉を開始する。どちらの対話も開始出来ない場合の処理は未定義であり、各 SSIP 実装にゆだねられる。次に、ユーザによる発話がない場合は、開始条件の具体的な設計が必要である。満たすべき条件は 3 つある。(開始条件 1) は、ユーザがエージェントと話せる状態にあることである。例えば、エージェントが動作するアプリケーションが開いている状態や、常設エージェントの知覚でユーザの存在が

センシングできる状態である。(開始条件 2) は、対話に適切なタイミングであることである。例えば、ユーザが起床している時間帯や、ユーザが仕事をしていない時間帯、前回の対話からしばらく時間が経過している状態である。(開始条件 3) は、ユーザもしくはエージェントに対話をする動機があることである。例えば、新規機能追加や季節変動など環境の変化が生じ、それに伴ってユーザの利用したい機能が変化する可能性がある状態や、設計者がユーザの要求を聞きたい場合、ユーザに機能を使って欲しい場合である。

3.4 (A2) ニーズ抽出

ユーザの要求を抽出する処理は一般的な定義が曖昧なため、ニーズ抽出とは何かを整理する必要がある。そのため、要求・ニーズについて定義し、ニーズ抽出対話設計の指針を検討する。ユーザの要求を推測できる情報は、様々な情報源の中に含まれる。〈対話 C1〉のユーザ発話がどのような情報源でどのような情報を含むかは、発話の種類によって変化する。テキストベースの発話の場合は自然言語情報が含まれる。音声が付く場合は声のトーンや抑揚などに要求が表現される可能性がある。映像が付く場合は表情・視線・ジェスチャーといった現実の対話における非言語コミュニケーションの情報が要求の推測に活用できる。ユーザ発話以外の情報源としては、ライフログ・サービス利用ログ・環境情報などのユーザをとりまくコンテキスト情報がある。コンテキストは SSIP がアクセスできるもののみ要求の推測に利用できる。

本研究におけるニーズの定義は、雑多な情報源に含まれる要求を、構造的に抽出したデータとする。要求のどういった側面を捉えるかはデータ構造の設計に依存する。例えば、先行研究 2 の 6W1H データモデルは、要求のユーザ自身が理解できる側面に着目する。また、対話設計に依存して得られるユーザ発話は大きく変化する。発話の種類によって情報源が変化するだけでなく、対話の流れによって発話の内容・回数も変化する。そのため、どのようなユーザ発話が欲しいかをニーズ構造に基づいて決定することで、どのような対話が適切かを設計する指針となる。さらに、要求は時間や環境に従って変化するため、ニーズ抽出対話は継続的に行う必要がある。ニーズデータがどの頻度で必要かを検討し、対話の頻度を設計する必要がある。

3.5 (A3) 機能決定アルゴリズム

機能決定アルゴリズムの実現手法が明らかでないため、機能決定の目的・アルゴリズム・入出力について詳細に検討する。機能決定の目的は、ユーザが満足する機能を決定することである。〈対話 C1〉要求抽出対話では、現在のユーザニーズに合致する機能を決定する。〈対話 C2〉機能提案対話では、〈対話 C1〉と異なりニーズという条件が存在しないため、より緩い条件でユーザに適切な機能を決定する。機能決定アルゴリズムは、ランキング問題や制約付き最適化問題に帰着すると考えられる [10], [11]。ランキング問題としては、機能リストをユーザが満足する順に優先順位を付けて並び替える問題といえる。制約付き最適化問題としては、料金・時間・利用環境などの制約のもとに最もユーザが満足する機能を求める問題といえる。ユーザの満足度を直接数値化することは難しいため、間接的な指標

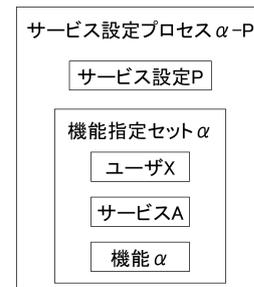


図 3: サービス設定プロセスの定義

が必要である。例えば、機能の使用頻度・直近に利用した機能との類似度・機能利用時間などが挙げられる。

アルゴリズムに求められる出力は、決定した機能 α と機能 α を持つサービス A である。さらに、機能決定過程に説明可能背が求められる場合は、アルゴリズムが機能 α に決定した理由を出力すべきである。次に、アルゴリズムに求められる入力について整理する。まず、機能決定を行う前提として、決定候補であるサービスと機能の一覧が必要である。また、機能を利用するユーザに関する情報として、ユーザが誰か・ユーザのサービス利用状況・ユーザのサービス利用履歴・ユーザのニーズ履歴が挙げられる。SSIP がセンサなどで得られたコンテキスト情報をも利用可能な場合、アルゴリズムの入力として含めることができる。〈対話 C1〉におけるアルゴリズムでは現在のユーザニーズも入力として利用できる。

機能決定アルゴリズムは、以上の目的・アルゴリズム・入出力を明らかにしたうえで設計すべきである。特に、何をもってユーザが満足する機能を決定できたとするかの目的設定が重要となる。

3.6 (A4) サービス自動設定

機能をサービス設定に反映する手法を明らかにするために、サービス設定の正常な動作を適切に保証する手法を検討する。まず、図 3 に示すように、ユーザ X・サービス A・機能 α の 3 つの情報をまとめたデータを機能指定セット α と定義する。機能指定セット α を与えられたときに、サービス設定 P を適用してユーザ X がサービス A において機能 α を利用できるにするサービス設定プロセス α -P を考え、これを自動化する。このとき、サービス設定プロセスが正常に動作することをどのように保証するかが問題となる。すなわち、機能指定セット α とサービス設定 P の適切な紐づけと、サービス設定 P の適切な動作をそれぞれ保証する必要がある。

紐づけはサービス側または SSIP システム側で行う。サービス側で紐づける場合を図 4 に示す。サービス設定プロセス α -P は、機能指定セット α をサービス外部の SSIP から与えるとサービス内部で機能指定セット α に紐づいたサービス設定 P が適用される処理となる。SSIP からサービス設定プロセスを実行するために Web API などの専用のインターフェイスが必要となる。メリットは、サービス内で完結するため、サービス設定プロセス α -P の動作責任が明確化されることである。デメリットは、サービス設定プロセス α -P を導入・更新するためにサービスの改修が必要になることである。

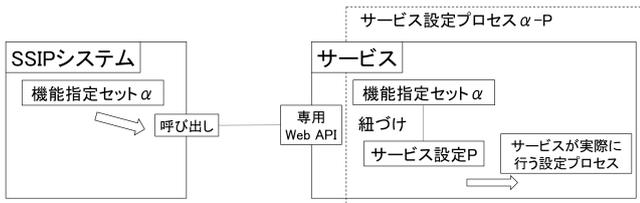


図 4: サービス側紐づけ

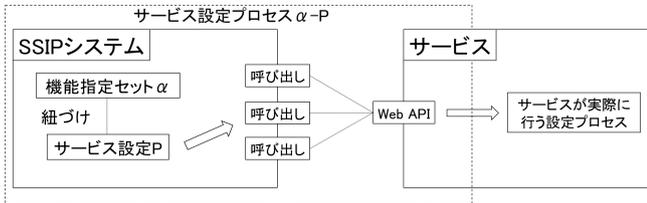


図 5: SSIP システム側紐づけ

SSIP 側で紐づける場合を図 5 に示す。サービス設定プロセス α -P は、サービス外部から実行可能なサービス設定インターフェイスを組み合わせて構築された機能 α を実現するためのサービス設定 P を適用する処理となる。サービス設定プロセス α -P のメイン処理はサービス外部の SSIP など構築する必要がある。メリットは、サービスの改修が不要であることである。デメリットとして、サービス設定プロセス α -P の動作保証および責任が曖昧になること、外部からサービス設定が可能なインターフェイスが十分にあるという前提が必要となることが挙げられる。

2 つの紐づけを比較すると、サービスの改修が可能である場合は、サービス側で機能指定セット α とサービス設定 P の紐づけを行うべきであることが分かる。しかしながら、サービスの改修が可能でない場合も多く、その場合はシステム側で紐づけを行い、なるべく正常にサービス設定プロセス α -P が動作することを丁寧に確認する必要がある。サービス設定の自動化は、このようにサービス側またはシステム側で紐づけたサービス設定プロセスを、機能指定を入力として受け取ったときに実行することで実現できる。

4. ケーススタディ

ケーススタディとして、SSIP の軽量な実装例を与える。(A1) から (A4) の各アプローチをそれぞれ簡易的に具体化する。実装アーキテクチャを図 6 に示す。主要なコンポーネントは、(A1) を実現するユーザインターフェイス (UI)、(A2) から (A4) を実現する SSIP 基盤システム、ユーザが利用する各種サービスである。

(A1) の具体化では、エージェントとして VA を利用した。〈対話 C1〉〈対話 C2〉の開始条件は、ユーザが対話を望むタイミングとした。ユーザが対話したい時にボタンを押すことで対話が始まる。この条件は、(A1) で定義した 3 つの開始条件を同時に満たす。(A2) の具体化では、先行研究 2 の対話型ニーズ抽出システムを採用した。ニーズ構造として 6W1H データモデルを採用し、6W1H 要素を問いかける対話設計を行った。

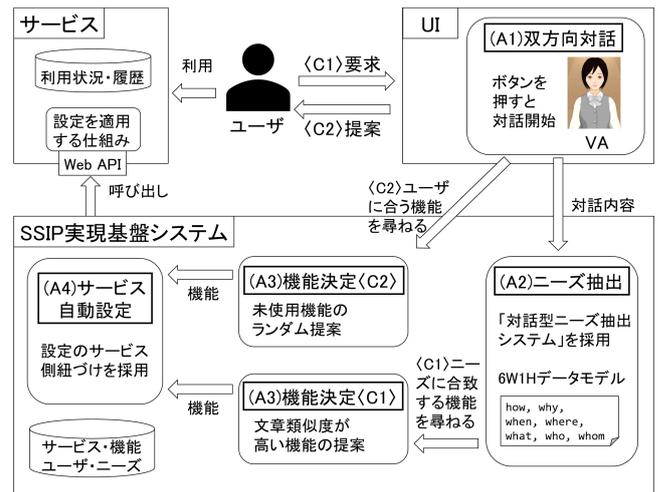


図 6: ケーススタディ版 SSIP アーキテクチャ

(A3) の具体化に当たって、2 つの機能決定アルゴリズムを設計した。まず、〈対話 C2〉機能提案対話における緩い条件下でのアルゴリズムを実装した。入力としてユーザのサービス利用状況および履歴を用い、未使用機能からランダムに 1 つ機能を決定した。次に、〈対話 C1〉要求抽出対話における特定のニーズに機能を合致させるアルゴリズムを実装した。入力として機能の一覧とリアルタイムに取得したユーザニーズを用い、文章類似度が最も高い機能を決定した。(A4) の具体化では、機能指定セットとサービス設定の紐づけをサービス側で実装し、SSIP 基盤システムから Web API を呼び出すことでサービス設定を自動化した。

ケーススタディ版 SSIP における個人適応フローを整理する。〈対話 C1〉を含む個人適応フローは、(1) ユーザのボタン押下、(2) ユーザによる要求発話、(3) VA によるニーズ確認、(4) ユーザによるニーズ抽出成否の返答、(5) VA による機能提案、(6) ユーザによる提案の承諾と拒否、(7) サービス自動設定である。〈対話 C2〉を含む個人適応フローは、(1) ユーザのボタン押下、(2) VA による機能提案、(3) ユーザによる提案の承諾と拒否、(4) サービス自動設定である。

5. 考 察

本研究では、個人適応フローを定義することで、対話からサービス自動設定までの具体的な処理を明らかにした。また、SSIP の中心的な要素となる (A1) から (A4) の要素を抽象化して切り出したことで、SSIP の実現という大きな課題を、各処理単位での既存技術の応用や新規技術の開発へと分割することに成功した。また、各要素の設計手法を明らかにし、ケーススタディで実装例を与えた。設計手法の限界として、ユーザの満足度を評価する仕組みがないこと、ユーザのプライバシーをどう守るかが明らかでないことが挙げられる。今後の研究では、これらを考慮した手法の拡張が必要となる。

ケーススタディでは SSIP の実装を試みたが、今回与えた実装はあくまで軽量版であり、今後 SSIP は様々な発展が期待できる。どのような改良が可能かという観点から、ケーススタディ

版 SSIP の限界を述べる。まず、軽量版の実装であるにもかかわらず、多くの技術を利用しており実装コストが重い。OSS の組み合わせなどでより軽量な実装が可能な手法を模索する必要がある。(A1) では、開始条件が完全にユーザ依存であり、エージェント主導の対話が存在しない。コンテキスト情報などを利用する工夫を行うことで、自動的にエージェントから話しかける対話も実現可能であると考えられる。(A2) および (A3) では、自然言語とサービス利用ログのみで個人適応を試みている。音声・映像情報やコンテキスト情報を活用することで、さらに満足度を高めることができる可能性がある。また、(A3) で実装した機能決定アルゴリズムは非常にシンプルなものである。提案手法に基づいてより丁寧に問題設定を行ったアルゴリズムを構築することで性能向上が期待できる。このように、SSIP はケーススタディによって実装が完了したわけではないため、今後も発展・改良版の SSIP を模索していく必要がある。

6. ま と め

本研究では、スマートシステムにおけるユーザのサービスに関する要望を自動的に実現することを目的として、我々が先行研究で提案した対話型個人適応フレームワーク SSIP の実装手法の確立に取り組んだ。具体的には、SSIP の主要要素として (A1) 双方向対話、(A2) ニーズ抽出、(A3) 機能決定アルゴリズム、(A4) サービス自動設定を定義し、各要素の設計手法を技術的な観点から具体的に提案した。ケーススタディでは、提案した設計手法に基づく簡易な SSIP を実現し、具体的な実装への道筋を示した。本研究によって、SSIP をベースにした対話型個人適応スマートシステムの開発手法が明確化し、今後研究開発していくべき 4 つの要素が明らかになった。今後の研究では、ニーズ抽出手法および機能決定アルゴリズムの高度化、軽量版 SSIP の構築、満足度評価の導入を通じて、よりユーザの満足度を向上させることのできる個人適応基盤の研究開発に取り組む予定である。

謝辞 本研究の一部は JSPS 科研費 JP20H05706, JP22H03699, JP22K19653, JP23H03401, JP23H03694, JP23K17006, および、立石科学技術振興財団の研究助成を受けて行われている。

文 献

- [1] “Nature remo,” <https://shop.nature.global/collections/nature-remo>. (Accessed on 10/17/2024).
- [2] S. Mamonov and R. Benbunan-Fich, “Unlocking the smart home: exploring key factors affecting the smart lock adoption intention,” *Inf. Technol. People*, vol.34, no.2, pp.835–861, Jan. 2021.
- [3] “Alexa,” <https://www.amazon.co.jp/meet-alexa>. (Accessed on 10/17/2024).
- [4] S. Aiolfi, “How shopping habits change with artificial intelligence: smart speakers’ usage intention,” *Int. J. Retail Distrib. Manag.*, vol.51, no.9/10, pp.1288–1312, Jan. 2023.
- [5] F.P. Raffaele Ariano, Marco Manca and C. Santoro, “Smartphone-based augmented reality for end-user creation of home automations,” *Behav. Inf. Technol.*, vol.42, no.1, pp.124–140, 2023.
- [6] T. Nakata, S. Saiki, and M. Nakamura, “Characterizing smart systems with interactive personalization,” 22nd IEEE-ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel Distributed Computing (SNPD2021), pp.8–14, Nov. 2021.

- [7] T. Nakata, S. Chen, S. Saiki, and M. Nakamura, “Exploring a unified service data model for needs-based service recommendations,” *Proceedings of International Conference on Green and Human Information Technology 2024 (ICGHIT20204)*, pp.355–358, Jan. 2024.
- [8] OpenAI, J. Achiam, and S.A. et al., “Gpt-4 technical report,” 2024. <https://arxiv.org/abs/2303.08774>
- [9] T. Nakata, S. Chen, S. Saiki, and M. Nakamura, “Employing large language models for dialogue-based personalized needs extraction in smart services,” *Computer Science & Information Technology (CS & IT)*, vol.13, pp.21–33, Dec. 2023.
- [10] M. Morik, A. Singh, J. Hong, and T. Joachims, “Controlling fairness and bias in dynamic learning-to-rank,” *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.429–438, Association for Computing Machinery, 2020.
- [11] A. Kumar, G. Wu, M.Z. Ali, R. Mallipeddi, P.N. Suganthan, and S. Das, “A test-suite of non-convex constrained optimization problems from the real-world and some baseline results,” *Swarm Evol. Comput.*, vol.56, p.100693, 2020.