

# スマートサービスの個人適応を可能にする LLMを活用した対話型ユーザニーズ抽出手法

中田 匠哉<sup>†</sup> 陳 思楠<sup>†</sup> 佐伯 幸郎<sup>††</sup> 中村 匡秀<sup>†,†††</sup>

<sup>†</sup> 神戸大学 〒 657-8501 神戸市灘区六甲台町 1-1

<sup>†††</sup> 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

<sup>††</sup> 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: <sup>†</sup>tnakata@es4.eeddept.kobe-u.ac.jp, <sup>††</sup>chensinan@gold.kobe-u.ac.jp, <sup>†††</sup>saiki.sachio@kochi-tech.ac.jp,  
<sup>††††</sup>masa-n@cmds.kobe-u.ac.jp

**あらまし** サービス個人適応に関する研究には、機械学習や対話エージェントといったアプローチが存在するが、推薦過程の説明可能性が課題となっている。従来研究では、6W1H ニーズモデルを活用した対話型ニーズ抽出システムを提案しているが、シンプルな自然言語処理では複雑なニーズ抽出が困難であった。本研究では、近年急速に活用が進む大規模言語モデル（LLM）を活用してユーザのニーズを自然言語から抽出する API の開発及び API を活用した対話型ニーズ抽出システムの構築に取り組む。評価実験として、ニーズ抽出・復元 API の出力結果の正確性や網羅性を評価することを目的として 100 件のニーズに関して検証を行った。本研究によって、LLM を活用して複雑な自然言語から高い正確性・網羅性でニーズを抽出することができるようになった。

**キーワード** 個人適応, ニーズ, LLM, 自然言語処理, 対話エージェント

## Extracting User Needs for Personalized Smart Services Using Dialogue Agent with LLM

Takuya NAKATA<sup>†</sup>, Sinan CHEN<sup>†</sup>, Sachio SAIKI<sup>††</sup>, and Masahide NAKAMURA<sup>†,†††</sup>

<sup>†</sup> Kobe University, Rokkodai-cho 1-1, Nada-ku, Kobe, Hyogo, 657-8501 Japan

<sup>†††</sup> Riken AIP, 1-4-1 Nihon-bashi, Chuo-ku, Tokyo, 103-0027 Japan

<sup>††</sup> Kochi University of Technology, 185 Tosayamadacho Miyanokuchi, Kami, Kochi, 782-8502 Japan

E-mail: <sup>†</sup>tnakata@es4.eeddept.kobe-u.ac.jp, <sup>††</sup>chensinan@gold.kobe-u.ac.jp, <sup>†††</sup>saiki.sachio@kochi-tech.ac.jp,  
<sup>††††</sup>masa-n@cmds.kobe-u.ac.jp

**Abstract** Research on service personalization involves approaches such as machine learning and dialogue agents, but the explainability of the recommendation process remains a challenge. Previous studies proposed a dialogue-based needs extraction system using the 6W1H needs model, but complex needs extraction was difficult with simple natural language processing. In this study, we focus on developing an API that extracts user needs from natural language using large-scale language models (LLMs) and constructing a dialogue-based needs extraction system using the API. We conducted verification on 100 needs to evaluate the accuracy and comprehensiveness of the output results from the needs extraction and restoration API. This study enables accurate and comprehensive extraction of needs from complex natural language using LLMs.

**Key words** Personalization, Needs, Large-scale language model, Natural language processing, Dialogue agent

### 1. はじめに

スマートサービスが社会に広く普及したことによって、サービス提供やサービス機能をユーザの嗜好に合わせて行うことで付加価値を生み出すサービス個人適応に関する研究が盛んに行

われている [1]。個人適応の研究には主に機械学習による推薦と対話による機能推薦・更新の 2 つが存在し、これらに組み合わせた会話型推薦システムに関する研究もおこなわれている [2]。しかしながら、ユーザにどのような経緯で推薦したかを説明することが困難であるという課題が存在する [3]。

従来研究では、推薦過程の説明可能性という課題を解決するために、ユーザがサービスをどのように実行したいかというニーズを where, when, who, whom, why, what, how の 7 要素で表現した 6W1H ニーズモデルを提案し、6W1H ニーズモデルを活用した対話型ユーザニーズ抽出システムの提案を行った [4]。提案の中で、ユーザのニーズ発話文から形態素解析・構文解析を用いて 6W1H 要素を抽出した。しかしながら、複雑なニーズの抽出や一部の要素の抽出が困難であるという課題が存在した。

本研究では、従来研究の課題を解決し、ニーズ抽出の多様性と正確性を向上させることを目的とする。キーアイデアとして、近年爆発的に利用が普及している大規模言語モデル (LLM) の GPT モデルを活用することで、ニーズ抽出 API の構築と対話型ニーズ抽出システムの再構築に取り組む [5]。本研究のアプローチは次の通りである。

- (A1) LLM を用いたニーズ抽出・復元 API の構築
- (A2) 全体アーキテクチャの設計
- (A3) 対話フローの設計

評価実験として、提案したニーズ抽出 API とニーズ復元 API に 100 件の入力を与えることで、出力の正確性と網羅性および 6W1H 要素を十分に抽出可能かどうかを評価する。

## 2. 準備

### 2.1 個人適応

デジタル分野において個人適応 (パーソナライゼーション) とは、個人または個人の集団に対する個別の関連性を高めるために、システムの機能性、インターフェース、コンテンツなどを変更するプロセスを指す [1]。個人適応を実現する主なアプローチには、機械学習と対話がある。機械学習によるアプローチでは、ユーザの過去のサービス利用履歴に基づいて深層学習を行い、ユーザが好むサービスを予想して推薦する [6]。しかしながら、推薦過程がブラックボックスであるため、説明可能性が課題となっている。対話によるアプローチでは、ユーザとパーソナルエージェントの対話を通じて、ユーザの好みを表現するユーザモデルを更新するとともに、サービスの更新を行う [7]。対話型個人適応はスマートスピーカーやヘルスケアエージェント等で導入されている [8]。近年では、機械学習と対話を組み合わせたアプローチとして、会話型推薦システムがある。対話を通じてユーザによる質問やフィードバックを可能とするとともに、対話ログに基づいて機械学習を行うことでサービス機能の推薦を行うアプローチである。

### 2.2 従来研究：対話型ニーズ抽出システム

会話型推薦システムでは、機械学習による個人適応と同様に推薦過程の説明可能性が課題である。そこで従来研究では、対話と機械学習の中間にユーザが理解可能なニーズモデルを導入することで、システムが解釈したニーズが何であるかを説明可能にすることを目指した。具体的には、図 1 に示す個人適応システムを提案し、特にユーザニーズ抽出手法の構築に取り組んだ。提案した個人適応システムでは、対話型ユーザニーズ抽出手法によってニーズを抽出し、ニーズを活用した機械学習に

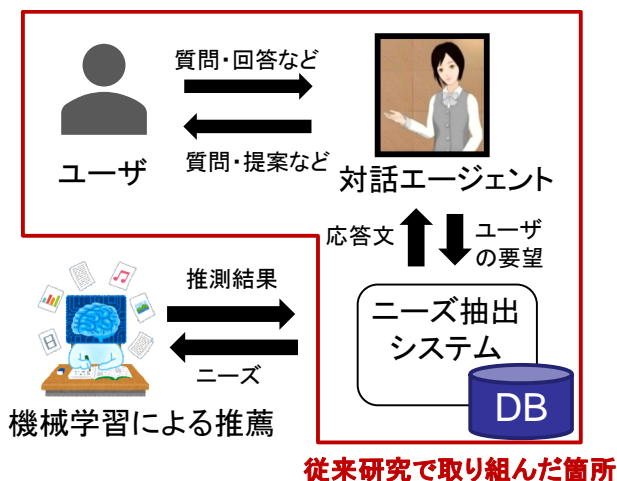


図 1: 対話型ニーズ抽出システムの全体アーキテクチャ

よってサービス推薦を行う。従来研究で構築に取り組んだ対話型ユーザニーズ抽出手法では、音声対話エージェントを通じて対話内容を取得し、ユーザが理解しやすい 6W1H 形式のニーズモデルで表されるニーズデータを自然言語処理を用いて抽出・蓄積する。6W1H ニーズモデルとは、ユーザがサービスをどのように利用したいかを表す次の 7 要素で構成されるニーズモデルである。

- where: どこでサービスを実行するか
- when: いつサービスを実行するか
- who: 誰が主体となってサービスを実行するか
- whom: 誰を対象としてサービスを実行するか
- why: どのような理由・経緯でサービスを実行するか
- what: 具体的にサービスで何を実行するか
- how: 実行するサービスは何か

6W1H 形式のニーズモデルで表されるニーズデータを 6W1H ニーズデータと呼ぶ。例として、「スマートスピーカーに毎朝 6:30 に天気予報が雨だったらリビングでおいちゃんに聞こえるように大きな音でお知らせしてほしい」というニーズ文を 6W1H ニーズデータで表現すると次のようになる。

- where: リビング
- when: 毎朝 6:30
- who: なし
- whom: おいちゃん
- why: 天気予報が雨だったら
- what: 聞こえるように大きな音でお知らせしてほしい
- how: スマートスピーカー

従来研究では、形態素解析・構文解析を用いて名詞や助詞の意味から各要素を推測して 6W1H ニーズデータを抽出した。不足した 6W1H 要素がある場合は、ユーザに「場所はどこですか?」といった質問を繰り返し、対話の中でユーザから可能な限り詳細なニーズを聞き出す。最後に、ニーズの抽出後、システムによる抽出結果がユーザの意図した正確なものかどうかユーザに確認を行う。

### 2.3 従来研究の課題

従来研究で用いた形態素解析・構文解析によるニーズ抽出手法には限界が存在する。まず、**who** と **whom** の区別が困難である。どちらもサービスの実行に関係する人を表す要素であるが、サービス実行の主体であるか対象であるかを判別するのが困難である。また、**why** の抽出が困難である。サービスを実行したい理由や実行の条件といった意味的内容を、文法だけに基づいて抜き出すことは困難である。さらに、複雑な文章からのニーズ抽出が困難である。複数のサービス実行に関する発話では 6W1H 要素を正確に抽出することはできない。例えば、「動画サービスや音楽サービスでリビングで映画が見たいのと毎朝カラオケをしたい」というニーズ発話文からは、係受け関係だけでは **where** 要素がリビングとなるのが動画サービスか音楽サービスかわからない。

### 2.4 大規模言語モデル

大規模言語モデル (LLM) は、大量のテキストデータでトレーニングされた自然言語処理モデルである。ファインチューニングを行うことで特定の自然言語処理タスクに特化させることが可能である。GPT は、OpenAI 社が開発する LLM である [9]。テキスト生成、翻訳、文書要約など様々な自然言語処理タスクが可能である。2022 年には AI チャットサービス ChatGPT が発表され、世界的に大きな注目を集めている。ChatGPT は、GPT-3.5、GPT-4 をベースに開発されたサービスである。GPT では、プロンプトエンジニアリングによって自然言語処理の命令を工夫することで様々な処理を行うことが可能になる。代表的な手法である Few-shot Prompting は、いくつかの質問と回答のペアを例示することで回答の精度を上げる手法である。

## 3. 提案手法

### 3.1 目的とキーアイデア

本研究の目的は、バーチャルエージェントとの対話を通じて複雑な自然言語のニーズ発話から 6W1H ニーズデータを抽出することで、ニーズ抽出の多様性と正確性を向上させることである。キーアイデアは、LLM を活用して複雑な自然言語のニーズ発話から 6W1H ニーズデータを抽出する手法を構築することである。本研究のアプローチは次の通りである。

- (A1) LLM を用いたニーズ抽出・復元 API の構築
- (A2) 全体アーキテクチャの設計
- (A3) 対話フローの設計

### 3.2 (A1) LLM を用いたニーズ抽出・復元 API の構築

LLM を用いて自然言語と 6W1H ニーズデータの変換を実現するニーズ抽出 API、ニーズ再抽出 API、ニーズ復元 API の 3 種類の API を作成する。ニーズ抽出 API は、ユーザのニーズ発話文を 6W1H ニーズデータに変換することでニーズを抽出する API である。ニーズ再抽出 API は、過去に抽出した 6W1H ニーズデータで不足した 6W1H 要素を新たなユーザニーズ発話から抽出した 6W1H ニーズ要素で補完する API である。ニーズ復元 API は、6W1H ニーズデータを自然言語のニーズ文に復元する API である。

API の実装には Open AI の Chat API を用いた。モデルは

<pre>{   "where": "",   "when": "",   "who": "自分",   "whom": "自分の興味に合った人々",   "why": "",   "what": "つながってリアルタイム のコンテンツを共有する",   "how": "SNS" }</pre>	<pre>{   "where": "どこでも",   "when": "毎日",   "who": "自分",   "whom": "自分の興味に合った人々",   "why": "",   "what": "つながってリアルタイム のコンテンツを共有する",   "how": "SNS" }</pre>
--	--

(a) ニーズ抽出 API の出力例

(b) ニーズ復元 API の入力例

図 2: LLM を活用した API の 6W1H ニーズデータ入出力例

gpt-3.5-turbo で、サンプリング温度は 0.0 である。プロンプトには Few-shot Prompting を活用し、入出力例を数件与えた。API の実装に用いたプログラミング言語は Python である。

ニーズ抽出 API に入力として与えられる値は 2 つである。一つ目の入力は、ユーザがバーチャルエージェントに対して自然言語で発話した、特定のサービスに関するニーズ文である。二つ目の入力は、ニーズ文を事前に文字列検索することによって取得した **how** (実行したいサービス) の候補となるサービスの一覧である。出力として、ニーズ文から抽出した 6W1H ニーズデータが 6W1H 要素の JSON 配列形式で得られる。例えば、ニーズ文の入力として「SNS で自分の興味に合った人々とつながり、リアルタイムのコンテンツを共有したい。」という文字列を、**how** の候補の入力として「SNS」を与えた場合、図 2a の結果が得られる。出力は単独の JSON 形式ではなく JSON 配列形式であるため複数の 6W1H 要素の JSON が取得できる。1 つの発言からあるサービスに関する複数のニーズや、複数のサービスに関するニーズを抽出することもできる。複数抽出の例として、ニーズ文にヘルスケアアプリと料理アプリに関する要求が含まれている場合に、**how** の候補の入力として「ヘルスケアアプリ」「料理アプリ」を与えた場合を考える。最も複雑な出力ではヘルスケアアプリに関する複数のニーズと料理アプリに関する複数のニーズが合わさった JSON 配列が得られる。

ニーズ再抽出 API に入力して与えられる値は 3 つである。一つ目の入力と二つ目の入力はニーズ抽出 API と同じく、ニーズ文および **how** の候補一覧である。三つ目の入力は、過去に抽出した単一の 6W1H ニーズデータである。出力として、ニーズ文の内容と過去の 6W1H ニーズデータの内容を全て踏まえたうえで新たにユーザニーズを 6W1H 要素の JSON 配列形式で抽出したものが得られる。例えば、入力として与える 6W1H ニーズデータが **where** と **when** を含まない場合に、合わせて与える入力を「毎日リビングで使いたい」というニーズ文とすると、出力として **where** が「リビング」、**when** が「毎日」となり他要素は入力と同じ内容となった 6W1H ニーズデータが得られる。ただし、出力結果が複数の 6W1H ニーズデータである場合は、各 6W1H ニーズデータが入力の 6W1H ニーズデータを更新して得られたニーズであるか全く新規のニーズであるかを判定する必要がある。判定のアイデアとして、例えば JSON を文字列とみなしてレーヴェンシュタイン距離が最も近いものを更新ニーズとみなすことができる。

ニーズ復元 API に入力して与えられる値は、6W1H ニーズ

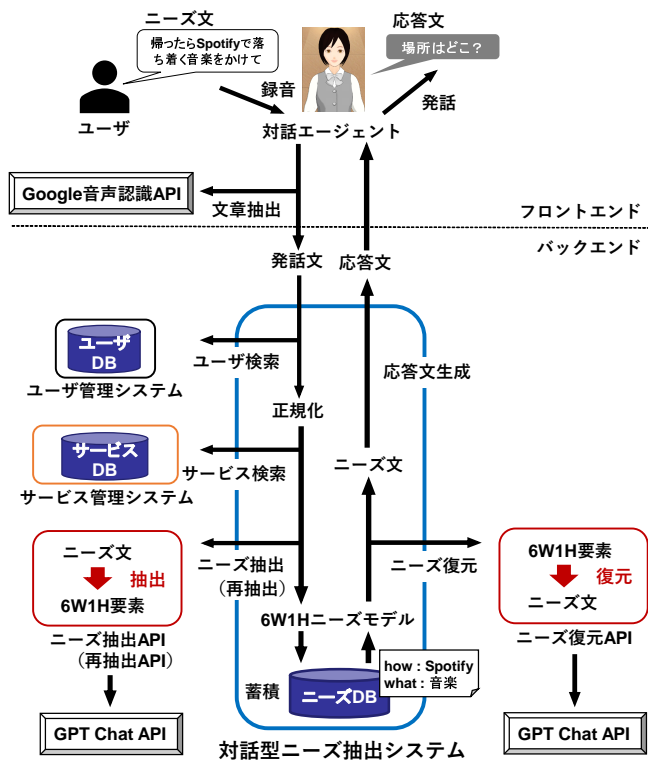


図 3: ニーズ抽出システムの全体アーキテクチャと動作フロー

データのみである。出力として、ニーズを表現する自然言語の文章が得られる。例えば、図 2b に示す 6W1H ニーズデータを入力に与えた場合、出力として「毎日、どこでも自分の興味に合った人々とつながってリアルタイムのコンテンツを共有するために、SNS を利用したい。」という文章が得られる。

### 3.3 (A2) 全体アーキテクチャの設計

LLM を活用した API を組み込んだ、対話型ユーザーニーズ抽出システムの全体アーキテクチャを図 3 に示す。フロントエンドアプリケーションは、バーチャル対話エージェントと Google 音声認識 API で構成される。バックエンドアプリケーションは、対話型ニーズ抽出システムを中核としてユーザー管理システム・サービス管理システム・ニーズ抽出 API・ニーズ復元 API・Chat API といった複数のシステム・API を組み合わせる。システムによって抽出される 6W1H ニーズデータは、対話型ニーズ抽出システムを介してニーズデータベースに格納される。

システムの処理は、図 3 の矢印の順に実行される。まず、ユーザーが特定のサービスに対するニーズ文を音声で発話する。音声は対話エージェントによって録音される。録音された音声は Google 音声認識 API によってテキスト形式のニーズ文に変換される。その後、ニーズ文は対話型ニーズ抽出システムに渡される。

対話型ニーズ抽出システムは、まずユーザー管理システムにニーズ文を発言したユーザーがシステム登録済みのユーザーかどうかを問い合わせて判定する。発話文の文字列は、漢字とひらがなはカタカナに、全角文字は半角文字に、小文字は大文字に変換することで正規化される。正規化文字列と、サービス管理シ

ステムから取得した登録済みサービス名の一覧を比較することで、ニーズ文が実行ターゲットとする how (サービス名) の候補を抽出する。その後、正規化前の発話文と how の候補をニーズ抽出 API またはニーズ再抽出 API に入力として渡し、出力として 6W1H ニーズデータを得る。ニーズ抽出 API (再抽出 API) は内部で GPT モデルを使った Chat API を呼び出している。抽出した 6W1H ニーズデータは MySQL データベースであるニーズデータベースに蓄積される。

応答文の生成フローは発話文の内容やニーズデータの抽出結果によって複雑に分岐するため、概要を述べる。まず、ユーザーへのニーズ確認などでニーズ文が必要な場合は、ニーズ復元 API に 6W1H ニーズデータを入力として渡し、出力としてニーズ文を得る。その後、6W1H ニーズデータとニーズ文の内容に基づいて条件分岐を行い、応答文を生成する。生成された応答文は対話エージェントへと渡される。最後に、対話エージェントはユーザーに向けて応答文を発話する。

### 3.4 (A3) 対話フローの設計

対話フローには、(D1) 新規ニーズ発話への応答と (D2) ニーズ抽出中の追加発話への応答の 2 種類存在する。あるサービスに対する 1 つのニーズを抽出するための対話の繰り返しを 1 サイクルとし、サイクルの初めのユーザー発話に対して (D1) フローが、2 回目以降のユーザー発話に対して (D2) フローが適用される。

(D1) 新規ニーズ発話への応答フローの動作ステップを次に示す。

- Step 1. 受け取った発話文からニーズ抽出 API で 6W1H ニーズデータを抽出する。
- Step 2. 不足した 6W1H 要素があればユーザーに追加情報を要求する。
- Step 3. 不足した 6W1H 要素がなければニーズ復元 API でニーズ文を復元し、正しくニーズを抽出できているかユーザーに確認する。

(D2) ニーズ抽出中の追加発話へ応答フローでは、同じ 6W1H ニーズデータを継続して更新できるように共通の対話 ID を使ってニーズデータベースから検索を行う。応答フローの動作ステップを次に示す。

- Step 1. 過去に抽出した 6W1H ニーズデータを対話 ID をもとにデータベースから取得する。
- Step 2. 受け取った追加発話文と過去の 6W1H ニーズデータからニーズ再抽出 API で新たな 6W1H 要素を再抽出する。
- Step 3. 複数の 6W1H ニーズデータが再抽出された場合は、過去の 6W1H ニーズデータと最もレーヴェンシュタイン距離が近い 6W1H ニーズデータを計算によって求め、求めた 6W1H ニーズデータを今回の対話で更新された 6W1H ニーズデータとみなす。
- Step 4. 応答文に関しては (D1) の Step 2. と Step 3. と同様に生成する。

## 4. ニーズ抽出・復元 API の評価実験

(A1) のニーズ抽出・復元 API について、以下のリサーチク

エスジョンを確認することを目的として評価実験を行った。

RQ1 ニーズ抽出 API とニーズ復元 API において、ニーズの変換を正確かつ網羅的に行えているか。

RQ2 ニーズ抽出 API において、6W1H 要素を十分に抽出できているか。

#### 4.1 実験準備

ニーズ抽出 API の評価実験では、API に入力する実験用データとしてニーズ発話を ChatGPT を用いて生成した。生成した 100 通りのニーズ文は、学生・高齢者・社会人など様々な立場を想定して ChatGPT が生成した。ニーズ文に含まれるサービスは合計 28 種類存在した。API の基本的な品質を調べるために、各ニーズ文はサービスが対象とするサービスは 1 つのみとし、複数サービスを対象としたニーズ文は入力として使用しなかった。評価は、API の出力結果を著者が分析し、特性と条件を調べた。

ニーズ復元 API の評価実験では、API に入力する実験用データとして 6W1H ニーズデータを ChatGPT を用いて生成した。生成した 100 通りの 6W1H ニーズデータは、学生・高齢者・社会人など様々な立場を想定して ChatGPT が生成した。6W1H ニーズデータの how 要素には合計 21 種類のサービスが含まれていた。評価は、API の出力結果を著者が分析し、特性を調べた。

#### 4.2 実験結果

ニーズ抽出 API の評価実験の結果を図 4 と図 5 に示す。図 4 の結果は、3 種類の特性について分析したものである。内容正確性は、抽出結果に虚偽の内容が含まれていないかを示す。内容網羅性は、発話文の重要な情報を全て抽出できているかを示す。分類正確性は、抽出した各ニーズ要素が適切な 6W1H 要素に分類されているかを示す。例えば、where に抽出されるべき「リビング」という要素が when に分類されている場合、分類正確性を満たさない出力結果と評価した。また、全ての特性を同時に満たす出力結果についてもグラフで示した。

図 5 の結果は、どれだけ多くの種類の 6W1H 要素を抽出できたかについて分析したものである。詳細条件の項目では、サービス実行の詳細な条件を表現する where, when, who, whom のうち、いずれか 1 つ以上を抽出できているかどうかを分析した。総合条件の項目では、サービス実行の総合的な条件を文章で表現する why が抽出できているかどうかを分析した。また、詳細条件と総合条件の両方を抽出できた出力結果と、どちらか一方を抽出できた出力結果をグラフで示した。

ニーズ復元 API の評価実験の結果を図 6 に示す。内容正確性は、復元結果に虚偽の内容が含まれていないかを示す。内容網羅性は、6W1H 要素の重要な情報を全て網羅できているかを示す。また、全ての特性を同時に満たす出力結果についてもグラフで示した。

#### 4.3 実験の考察

実験結果から RQ1 と RQ2 について考察を行う。まず、RQ1 のニーズの変換を正確かつ網羅的に行えているかについて考察する。ニーズ抽出 API では、内容正確性は 100%であり、抽出結果に一切の虚偽はなく正確であった。内容網羅性は 96%

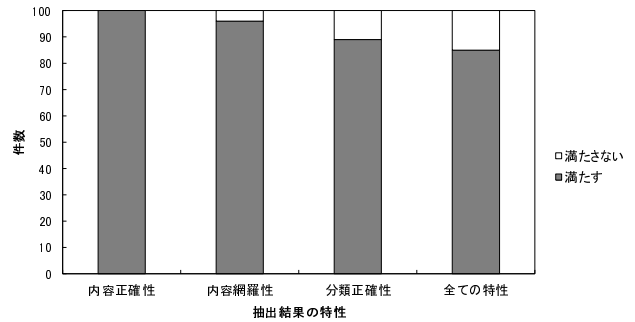


図 4: ニーズ抽出 API による抽出結果の特性

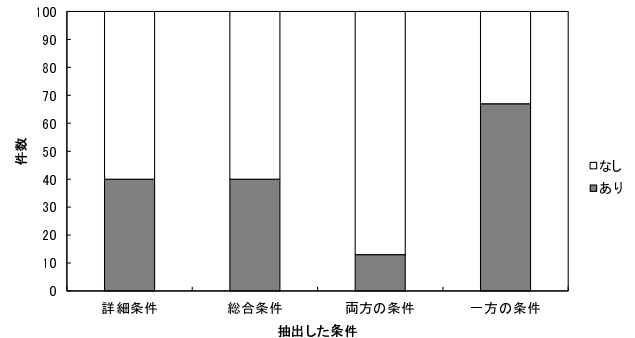


図 5: ニーズ抽出 API によって抽出した条件

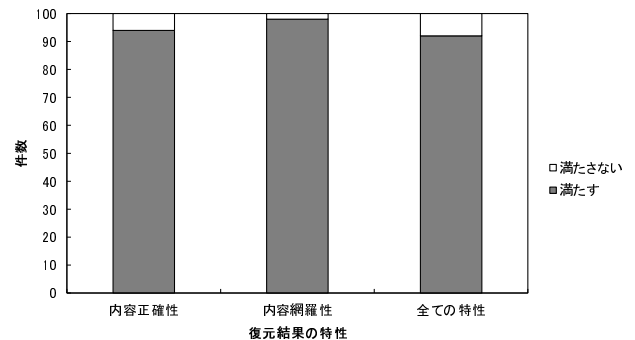


図 6: ニーズ復元 API による復元結果の特性

あったが、欠落していた内容は「便利にショッピングしたい」「楽しいひとときを過ごしたい」といった核心的ではない付加的な情報であった。分類正確性は 89%であり、特に when などの詳細条件や what に分類すべき内容を誤って why に分類したものが多かった。ニーズ復元 API では、内容正確性は 94%であり、why と what の内容を逆転して文章化しているものがあった。内容網羅性は 98%であり、複数の要素で内容が似通っている場合に文章化すべき情報が欠落することがあった。また、抽出 API では 85%、復元 API では 92%が全ての特性を満たしている。これらの結果から、対話型ニーズ抽出システムでユーザーにニーズが正しく抽出できたか確認しつつニーズ抽出を行うユースケースにおいては、ニーズ抽出・復元 API は高い正確性と網羅性を持つと言える。

次に、RQ2 の 6W1H 要素を十分に抽出できているかについて考察する。ニーズ抽出 API において、詳細条件を抽出できたも

のが40%, 総合条件を抽出できたものが40%であった。また、詳細条件または総合条件の一方が抽出できたものが67%であった。この結果から、一度のニーズ抽出において十分な6W1H要素抽出が出来ていると言えない。しかしながら、ニーズ文から抽出可能な6W1H要素の種類はニーズ文の内容にも依存する。また、対話を繰り返す中で不足した6W1H条件をユーザに尋ねるといったユースケースにおいては、ニーズ抽出APIは十分に6W1H要素を抽出できる性能を持つと言える。

## 5. 考 察

LLMをニーズ発話文と6W1Hニーズデータとの相互変換に活用することで、複雑な自然言語の発話からニーズを抽出することができた。具体的には、1つのニーズ発話文から複数のニーズを抽出できるようになったほか、従来研究では困難だったwhoとwhyを抽出できるようになった。API評価実験の結果から、1度のニーズ抽出によって正確にニーズを抽出できることが分かった。ただし、各要素の分類や多様性についての特性は不確定であり、不足した6W1H要素を尋ねる対話が重要であることがわかった。ニーズ抽出対話を実現するために、ユーザ検索システム、サービス検索システム、ニーズ抽出・復元APIを活用し、処理を分散させたマイクロサービスアーキテクチャとしてニーズ抽出システムを実現した。今後の課題として、システム全体の性能評価実験と、実際のユーザによる利用時のシステム品質評価に取り組む必要があると考えている。

## 6. ま と め

本研究の目的は、バーチャルエージェントとの対話によって複雑な自然言語のニーズ発話文から6W1Hニーズデータを抽出する手法を構築することで、ニーズ抽出の多様性と正確性を向上させることである。キーアイデアとして、LLMを活用して複雑な自然言語のニーズ発話文から6W1Hニーズデータを抽出する手法を構築した。具体的には、LLMを使った自然言語と6W1Hニーズデータの抽出・再抽出・復元APIの3つの相互変換APIを設計・開発した。また、提案APIを用いた対話型ニーズ抽出システムの全体アーキテクチャを設計した。そして、APIにLLMを用いることで複雑化した対話フローを整理して再設計した。

さらに、ニーズ抽出APIとニーズ復元APIに100件のテストデータを入力する評価実験を行った。結果、抽出APIは85%、復元APIは92%が正確性と網羅性を満たす出力が得られた。本研究によって、LLMを活用して複雑な自然言語から高い正確性・網羅性で6W1Hニーズデータを抽出することができるようになった。将来研究として、より実践的な利用での対話型ニーズ抽出システムの評価を検討している。

**謝辞** 本研究の一部はJSPS科研費JP19H01138, JP20H05706, JP20H04014, JP20K11059, JP22H03699, JP19K02973, 若手研究23K17006の助成を受けて行われている。

## 文 献

- [1] H. Fan and M.S. Poole, "What is personalization? perspectives on the design and implementation of personalization in information sys-

tems," *Journal of Organizational Computing and Electronic Commerce*, vol.16, no.3-4, pp.179-202, 2006.

- [2] D. Jannach, A. Manzoor, W. Cai, and L. Chen, "A survey on conversational recommender systems," *ACM Computing Surveys*, vol.54, pp.1-36, 05 2021.
- [3] K. Hollis, L. Soualmia, and B. Séroussi, "Artificial intelligence in health informatics: Hype or reality?," *Yearbook of Medical Informatics*, vol.28, pp.003-004, 08 2019.
- [4] 中田匠哉, 陳 思楠, 佐伯幸郎, 中村匡秀, "音声対話エージェントを活用したサービス個人適応に向けたユーザニーズ抽出手法の検討," 電子情報通信学会技術研究報告, 第122巻, pp.13-18, Jan. 2023. 石川, ITビジネスプラザ武蔵.
- [5] M. Cascella, J. Montomoli, V. Bellini, and E. Bignami, "Evaluating the feasibility of chatgpt in healthcare: An analysis of multiple clinical and research scenarios," *Journal of Medical Systems*, vol.47, no.1, p.33, March 2023. <https://doi.org/10.1007/s10916-023-01925-4>
- [6] D. Goldenberg, K. Kofman, J. Albert, S. Mizrahi, A. Horowitz, and I. Teinema, "Personalization in practice: Methods and applications," pp.1123-1126, *WSDM '21*, 03 2021.
- [7] A.B. Kocaballi, S. Berkovsky, J. Quiroz, L. Laranjo, H.L. Tong, D. Rezazadegan, A. Briatore, and E. Coiera, "The personalization of conversational agents in health care: Systematic review," *Journal of Medical Internet Research*, vol.21, p.e15360, Nov. 2019.
- [8] H. Ozono, S. Chen, and M. Nakamura, "Encouraging elderly self-care by integrating speech dialogue agent and wearable device," 8th International Conference, ITAP 2022, Held as Part of the 24th HCI International Conference, HCII 2022, vol.LNCS 13331, pp.52-70, May 2022.
- [9] S. Bubeck, V. Chandrasekaran, R. Eldan, J. Gehrke, E. Horvitz, E. Kamar, P. Lee, Y.T. Lee, Y. Li, S. Lundberg, H. Nori, H. Palangi, M.T. Ribeiro, and Y. Zhang, "Sparks of artificial general intelligence: Early experiments with gpt-4," 2023.