

ユーザのロールに応じた モバイルサービスナビゲーションシステム

User's Role Oriented Mobile Service Navigation System

深澤 佑介 長沼 武史 藤井 邦浩 倉掛 正治
Yusuke Fukazawa Takefumi Naganuma Kunihiro Fujii Shoji Kurakake
Service & Solution Development Dept., NTT DoCoMo, Inc.
{fukazawayuu, naganuma, fujii, kurakake}@nttdocomo.co.jp

Abstract—We have been developing a task-based service navigation system that offers to a user for his selection services relevant to the task the user wants to perform. We observed that tasks likely to be performed in a given situation depend on the user's role such as businessman or father. To further our research, we have constructed a role-ontology and utilized it to improve the usability of task-oriented service navigation. We have enhanced a basic task-model by associating tasks with role-concepts defined in the new role-ontology. We can generate a task-list precisely tuned to the user's current role. In addition, we can generate a personalized task-list from the task-model based on the user's task selection history. Because services are associated with tasks, our approach makes it much easier to navigate a user to the appropriate services. In this paper, we describe the construction of our role-ontology and implemented task-based service navigation system based on the constructed role-ontology.

1 Introduction

近年、携帯端末で閲覧可能なコンテンツやメディアは劇的に増加している。その多くは、待ち受け画像や着メロ、ゲーム、音楽コンテンツのダウンロードサイト等のエンターテインメント向けコンテンツである。一方で、乗り換え案内、終電案内、チケット購入、天気予報、及び災害情報等、実世界でユーザが直面する様々な問題や要望をリアルタイムで解決可能な問題解決型のコンテンツも充実してきており、携帯端末の利用者にとっては日常生活で直面する問題を場所に縛られることなく解決可能な環境が整ってきたといえる。しかしながら、現状では、問題解決向けのサービスはエンターテインメント向けのコンテンツに比べあまり利用されていない。これは、実際に問題や要求が生じたときに、ユーザ自身で必要なコンテンツをすぐ発見できないためであると考えられる。現状の携帯端末向けの情報検索サービスには、PC環境での検索技術を携帯端末上に移植したものがほと

んどであり、ユーザの実世界で発生する問題を解決するという観点で設計されたものがない。

このような背景のもと筆者らは、ユーザが実世界で抱える問題の構造を分析し、ユーザの問題の解決行動をタスクモデルとして構造化してきた [5]。タスクモデルでは、実世界の場所(ドメイン)ごとに、その場所に関連して生じる可能性が高いタスクがツリー状に構造化されている。映画館ドメインにおけるタスクモデルの一部を図1に示す。タスク間の関係は達成関係 (*is-achieved-by*) で表現されている。上位のタスクではユーザの抽象的な要求や問題を表現しており、下位のノードではより具体的な解決策を表現している。末端のノードには該当するタスクを解決可能なインターネット上のサービス(コンテンツのURI)が関連付けられている。Naganumaら [5][3] は、このタスクモデルを用いて、ユーザに要望をキーワード入力してもらい、タスクモデル内のタスクとのキーワードマッチングを行うことで、サービスを検索可能なキーワード入力型のサービスナビゲーションシステムを提案している。

しかし、ユーザの要求やユーザが抱える問題は抽象的な場合が多く、入力キーワードとして具体的に表現することを困難に感じるユーザは少ない。本稿では、タスクの検索に要する負担を軽減するため、キーワード入力をしなくてもユーザが必要とするタスクを検索できるようサービスナビゲーションシステムを拡張する。

キーワードを入力せずにタスクを検索できるようにするためには、タスクの実行場所とタスクとをあらかじめ関連付けておき、ユーザがタスクの実行場所を指定することでタスクを検索する方法がある。しかしながら、実世界で想定されるタスクの実行場所は50個以上あり、場所を選択するのに時間がかかる。そこで、本稿ではユーザが実生活を送る上で常に担っているロール(会社員、観光客)に注目する。場所の総数に比べ実世界でユーザが担うロールの数は少なく、ロールの選択に要する時間は少なくすむ。例えば、ユーザは会社に通勤して帰宅するまで、「駅」「車内」「会社」「レストラン」等の場所を通過する。場所ごとにタスクが分類されている場合、ユーザは必要なタスクやサービスをそれぞれの場所から検索する必要がある。一方、これらのタスクは「会社員」としての一

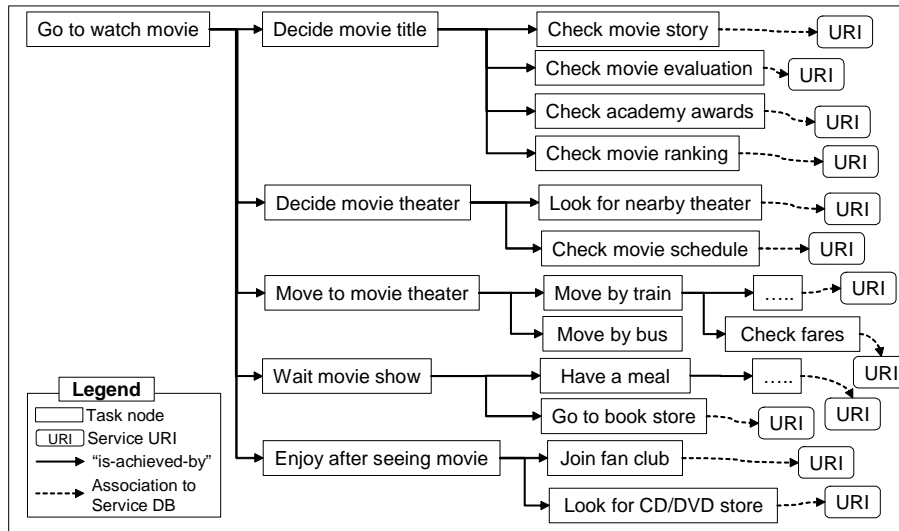


Figure 1: Extraction from task-ontology[5]

連の行動であることから、ロール「会社員」に対して必要なタスクを関連付けておくことで、ユーザのロールさえ指定しておけば場所を選択する必要はなくなる。ただし、ロールごとにどのようなタスクを実行するかはユーザによって変わる。また、一つのロールに対し対応付けられるタスクの数は一つの場所に対応付けられるタスクの数に比べ増加する。ここでは、タスクやサービスの選択履歴をユーザのロールとともに蓄積しておき、現在のロールに応じて適切なタスクやサービスの推薦することにより、必要なタスクやサービスに早く到達可能な方法を提案する。

ユーザが担い得るロールには様々な性質が報告されている [4][7]。ここでは、以下の4つの性質に着目する。1) ロールは一定ではない。例えば、仕事中は「会社員」であっても家に帰れば「父」というロールに変わりうる。2) 同時に2つ以上のロールを持つ可能性がある。例えば、電車で家族と乗っているときには「乗客」と「父」という2種類のロールを同時に持つ。3) ロールにはタスクを実行するときに担うロール(乗客、観客等)と、周囲の人間関係から決定されるロール(父、友人等)が存在する。4) ロールを担っているときには複数のタスクを実行する可能性があるが、それらのタスクの優先度はユーザによって変わる。

以降、2章ではまず上記のような様々な性質をもつロール概念を適切に利用するため、ロールオントロジーの構築について説明しその後、ロールとタスクとの関係性の記述について説明する。3章ではユーザのロールごとにタスク、及びサービスの利用履歴を蓄積しユーザのロールに応じたタスクやサービスを推薦する方法について説明し4章で結論を述べる。

2 ロールオントロジーの構築

2.1 ロール概念に関する既存研究

ロール概念は、知識表現 [9]、言語学 [2]、リレーショナルデータベース [1]、アクセスコントロール [6] 等の様々な分野において研究されてきた。知識表現の分野においてはロール概念は主にドメインオントロジーを正確に構築するために研究されてきた [9]。正確なオントロジーを正確に構築するためにはドメインからロール概念を峻別する必要があるからである [9]。

Sowa は基本概念とロール概念を明確に分離し、基本概念を「実体の本質に関する概念」、ロール概念を「他の複数の実体との関係性の上に成り立つ概念」と定義している [7]。数年後、Sowa は自らの定義を修正し、ロール概念を基本概念の下位概念であると主張した。例えば、「子供」「ペット」「カルテット」といった概念はロール概念でありかつ「人」「動物」「数」といった基本概念の下位概念であるとしている [8]。

Guarino は、Sowa の定義を発展させ、オントロジーにおける基本概念とロール概念について再定義を行った。彼の定義では、ロール概念は他の概念との関係性によって成り立つ概念であり、基本概念は他のいかなる概念からも完全に独立して成り立つ概念とされる。例えば、「人」は基本概念である。「人」が「人」であるために他の概念との関係性は必要ないからである。一方、「学生」はロール概念である。「学生」となるためには、大学に入学する必要があり、もし大学を卒業したり退学すれば「学生」であることは終わる。

Fan らは、ロール概念はイベントに大きく依存することに注目した。その上でロール概念を「ある実体がイベントに参加したときの外延的な特徴」と定義している。古崎 [10]、Sunagawa [9] らはロール概念がコンテキストに依存して決まることに注目し、ロール概念を「あるものが特定のコンテキスト

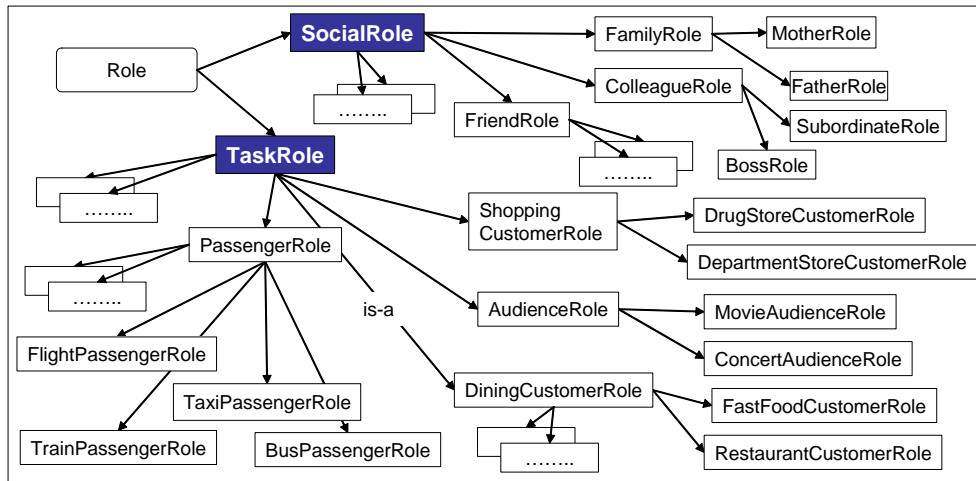


Figure 2: Role-ontology for task-based service navigation system

のもとで果たす役割を捉えて概念化したものである」と定義している。Sunagawaらはコンテキストとロール間の関係は *part-of* 関係と *participate-in* 関係によって表現されるとし、それらを取り扱い可能なオントロジ構築ツール「法造」を開発している。

このように、オントロジにおけるロール概念の研究は、基本概念との定義の違い、ロール概念と基本概念との関係性の表現等、基本概念との峻別という観点で述べられることが多い。しかしながら、社会活動におけるロールをシステム的に取り扱っている研究はほとんどなされていない。

2.2 ロールオントロジの構築

本稿では、次の2種類のロール概念を定義する。行動を共にしている人や周囲にいる人々との人間関係に応じて決まるロールと、ユーザの実世界行動(タスク)から決まるロールを定義する。ここでは、前者のロールを *social-role* と呼び、後者のロールを *task-role* と呼ぶ。 *task-role* と *social-role* は同時に担うことができる。例えば、ユーザが電車に同僚と乗っているとき、ユーザは「TrainPassenger Role」と「Colleague Role」の2つのロールを同時に担う。また、ユーザが子供とデパートに買い物に出かけたとき、ユーザは「ShoppingCustomer Role」と「Father Role」の2つのロールを同時に担う。

Social-role や *task-role* に属するロールはそれぞれ独立ではなく包含関係がある。例えば、 *social-role* では「Colleague role」は「Boss role」「Subordinate role」を包含している。 *Task-role* の場合には、「Passenger Role」は「TrainPassenger Role」を包含している。このようなロール間の包含関係を一般的なオントロジで用いられる *is-a* 関係を用いて表現する。また、ロール間に包含関係があるとき、親ロールを担ったときに実行するタスクの集合は子ロールを担ったときのタスクの集合を包含する。

ロールオントロジの一部を図2に示す。最上位概念となる「role」は、上記で定義した2つのロール概念「task-role」、及び「social-role」がリレーション *is-a* によって関連付けられている。さらに、 *task-role* には「Passenger Role」「Audience Role」「ShoppingCustomer Role」及び、「Dining Customer Role」などタスクに依存して定まるロールがサブ概念として定義されており、「social-role」には、「FamilyRole」「FriendRole」「Colleague Role」などの人との社会的関係によって定まるロールがサブ概念として定義されている。

2.3 タスクとロールの関係性記述

ユーザの現在のロールからユーザに必要なタスクを取得するためには、タスクモデル内に定義されているタスクとロールオントロジ内に定義されているロールとの関係性をあらかじめ表現しておく必要がある。ここで、 *social-role* はそれを担うユーザによって実行するタスクが大きく変わることから、ここでは *task-role* とタスクの関係性を表現する。タスクはそのタスクを実行するときに担うロールをもつユーザによって実行されることから、 *is-played-by* リレーションを新たに定義し、タスクと *task-role* との関係性を表現する。

図4に、タスクと *task-role* を関連付けた例を示す。図中において「Go to watch movie」は、 *task-role* 「MovieAudience Role」と関連付けられている。また、「Move to movie theater」は、「Passenger Role」と関連付けられている。従って、ユーザが現在行っている、または行おうとしているタスクが分かれば、その時点でのユーザの *task-role* も分かるようになる。また、図4においてロールと関連付けられていないタスクノードは、親ノードに関連付けられているロールと同一のロールと関連付けられている。

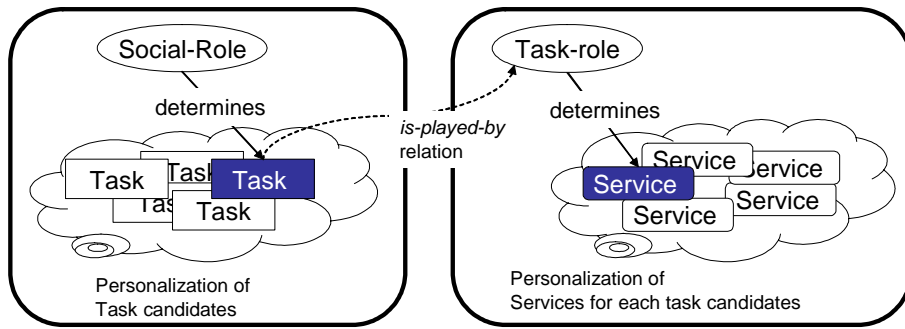


Figure 3: Concept of personalization of task-list and their recommended services

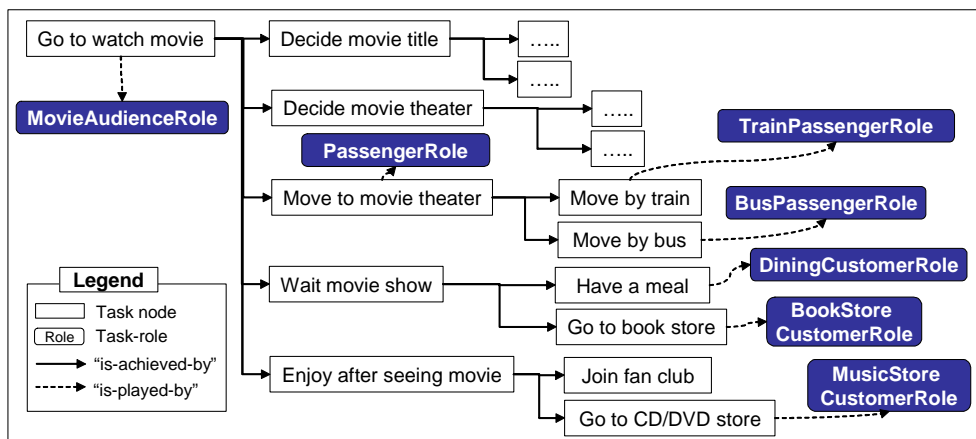


Figure 4: Enhancement of task-ontology using role-concept defined in role-ontology

2.4 Social-role と task-role に応じたタスクとサービスの優先度付け

本稿ではロール概念として social-role と task-role の2種類を定義しているが、それぞれサービスナビゲーションシステムにおいて果たす役割は異なる。social-role は、ユーザが実世界でどのような行動をするかの意思決定に影響を与える。一方、task-role は、なんらかのタスクを実行中に担うロールであり、そのタスクを実行するためにどのサービスを使うかの意思決定に影響を与える（図3）。

次に、タスクやサービスを推薦する際の social-role と task-role の利用方法について述べる。通常、推薦するタスクやサービスの優先度はそれらの過去の選択頻度に基づいて決める。しかしながら、前述のように実世界においてユーザは複数の social-role や task-role を切り替えてタスクを実行するため、蓄積された選択履歴は複数の social-role や task-role を担ったときに実行したタスクやサービスが混在している。このような選択履歴を基にタスクやサービスの優先度を決めると、ユーザの現在のロールでは不必要なタスクやサービスが混在して推薦されてしまう。例えば、「Colleague role」のときに、「Father role」で頻繁に実行する「Go to theme park」が推薦されてしまう可能性がある。この問題を解決するため、ここではタスクの選択

履歴を social-role とともに蓄積し、サービスの選択履歴を task-role とともに蓄積しておく。タスク検索の際には、ユーザから現在の social-role、及び task-role を取得する。次に、現在の social-role のもとで過去に選択されたタスクを取得し選択回数が多い順に並び替える。さらに、並び替えられたタスクについてそれぞれ task-role を取得し、それぞれの task-role のもとで選択されたサービスを選択回数が多い順に複数個推薦する。これにより、ユーザのロールに応じて必要なタスクやサービスのみを推薦することができタスクやサービスの検索に要する時間を軽減することができる。

3 ロール概念を利用したサービスナビゲーションシステム

本章では、ロール概念を利用したサービスナビゲーションシステムについて説明する。

3.1 システム全体の流れ

以下に基本的なロール概念に基づくサービスナビゲーションの全体の流れについて説明する（図5）。まず、ユーザの現在の task-role、及び social-role を取得する。次に、取得した task-role と関連付け

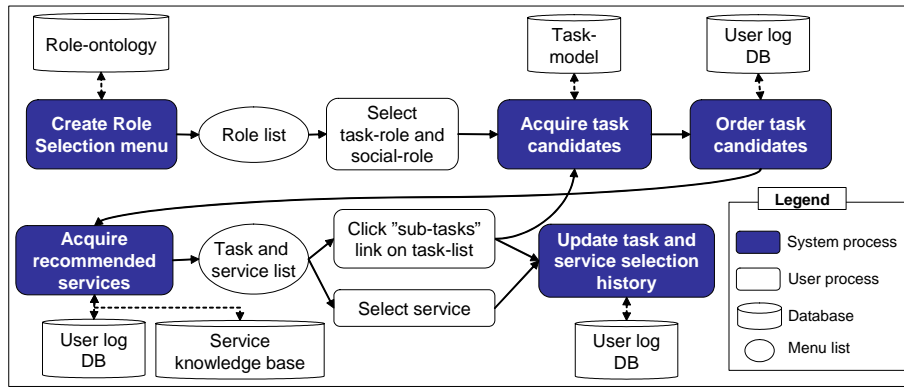


Figure 5: Overall processes of role-based service navigation

られているタスクをタスクの候補として取得する。さらに、タスク選択履歴を参照しユーザの social-role に応じてタスクの候補間に優先度を付け並び替える。次に、サービス選択履歴に基づきそれぞれのタスクの候補に対しユーザの task-role に応じて推薦サービスを決定する。最後に、並び替えられたタスクと推薦サービスがともにタスクリストとしてユーザに提示される。サービス利用時、もしくはサブタスクの候補を取得した時には、その履歴がタスク選択履歴、サービス選択履歴に蓄積される。

3.2 ユーザロールの獲得

ロールの指定画面について説明する。この画面がサービスナビゲーションシステムのトップ画面となる。ロール指定画面の例を図 6(a) に示す。social-role の選択画面のメニュー項目には、ロールオントロジー内で定義されている social-role の下位概念から構成されている。同様に、task-role の選択画面は、ロールオントロジー内で定義されている task-role の下位概念から構成されている。ロール選択リスト内のロールメニューは、ユーザ自身で編集することができる。ロール選択メニュー内の「edit your menu」リンクをクリックすると、より詳細なロールを選択することができロール指定画面上のロールを追加、削除できる。

ロール指定画面において social-role は必ず指定する必要があるが、task-role は必ずしも指定する必要はない。過去に同じ social-role を指定したときに利用したタスクを再度利用したい場合には、social-role のみを指定してもらえばよい。その結果、過去に同じ social-role を指定したときに利用したタスクが選択頻度順に並び替えられて提示される。一方、過去に同じ social-role を指定したときには一度も利用したことがないタスクを利用したい場合には、そのタスクを実行するときの task-role も同時に指定してもらおう。その結果、指定した task-role に関連付けられたタスクをタスクモデルから取得しユーザに提示する。

3.3 タスク候補の取得

本節では、タスクの候補の取得方法について述べる。タスクの候補は、ロール選択後、及び sub-tasks リンククリック後に取得される (図 5)。

ロール指定画面 (図 6(a)) で social-role と task-role の両方が指定された場合、指定された task-role、もしくは選択された task-role の下位の task-role と関連付けられたタスクを拡張タスクモデルから検索しタスクの候補とする。例えば、図 6(a) で「Audience Role」を選択した場合、「Audience Role」の下位概念である「MovieAudience Role」「PlayAudience Role」「ConcertAudience Role」及び「GameAudience Role」と関連付けられたタスク「Go to watch movie」「Go to see a play」「Go to a concert」、及び「Go to see a game」がタスクの候補となる。取得したタスクの候補を並び替えタスクリストが生成される (図 6(b))。並び替えの方法については、3.5 節にて述べる。タスクの候補にはそれぞれサービスが推薦されるが、推薦されるサービスの決定方法については 3.6 節にて述べる。

一方、ロール指定画面 (図 6(a)) で social-role のみが指定された場合、次節で説明するタスクの選択履歴内で過去に同じ social-role を指定したときに利用したタスクをタスクの候補として取得する。

また、図 6(b) にて sub-tasks リンクをクリックした場合、選択されたタスクのサブタスクをタスクモデルから取得しタスクの候補とする。例えば、「Go to watch movie」の sub-tasks リンクをクリックした場合、そのサブタスク「Decide movie title」「Decide movie theater」「Move to movie theater」及び「Enjoy after seeing movie」が検索されタスクの候補となる。生成されたタスクリストを図 6(c) に示す。

3.4 タスク・サービス選択履歴の蓄積

タスク選択履歴、及びサービス選択履歴は、タスクリストで sub-tasks リンクをクリック後、及びサービス選択後に更新される (図 5)。Sub-tasks リンクをクリックした場合には、選択されたタスクをユーザの social-role と関連付けてタスク選択履歴

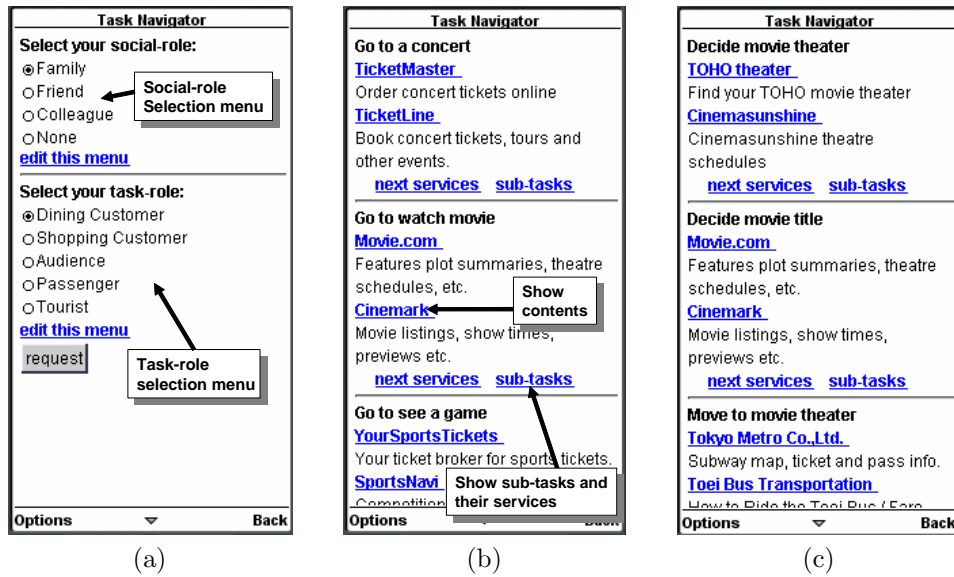


Figure 6: Screenshots of service navigation;(a)role-selection menu, (b) task-list and their recommended services when user selected “Audience Role” from the role-selection menu shown in (a), (c)task-list and recommended services when user clicks “sub-tasks” link of “Go to watch movie” from the task-list shown in (b)

Table 1: An example of task selection history

Social-role	Selected task	Freq.
FriendRole	Go to watch movie	4
	Enjoy after seeing movie	4
	Look for CD/DVD store	8
FamilyRole	Go to a concert	4
	Move to a concert hall	4

Table 2: An example of service selection history

Task-role	Selected service	Freq.
MovieAudience Role	HMV.com	4
	iTunesMusicStore	4
ConcertAudience Role	Tokyo Metro Co., Ltd.	2
	Toei Bus Transportation	2

に蓄積する。一方、サービスを選択した場合には、選択したサービスのタスクを social-role と関連付けてタスク選択履歴に蓄積し、選択されたサービスと task-role とを関連付けてサービス選択履歴に蓄積する。

以下に例を示す。図 6(b) にて「Go to watch movie」の sub-task リンクをクリックした場合には、タスク「Go to watch movie」を social-role 「FamilyRole」と関連付けて蓄積する(表 1)。

この時点でユーザの task-role は「AudienceRole」から、「Go to watch movie」の task-role である「MovieAudience Role」に変化する。次に、ユーザが「Look for CD/DVD store」のサービス「HMV.com」を選択した場合は、「HMV.com」と task-role 「MovieAudience Role」を、「Look for CD/DVD store」と social-role 「FamilyRole」とを関連付けて蓄積する(表 2)。

3.5 タスク選択履歴に基づくタスク候補の並び替え

3.2 節で述べた方法に従ってタスクの候補を取得した後、タスクの候補の並び替えを行う。まず、タスク選択履歴(表 1)を参照し、全てのタスクの候補について現在の social-role のもとで選択された回数を取得し、選択回数の多い順に並び替える。もし、初めて指定された social-role でタスク選択履歴が蓄積されていない場合には、ロールオントロジー内の social-role 間の親子関係に基づき関連する social-role のもとで蓄積されたタスク選択履歴を利用する。

3.6 サービス選択履歴に基づく推薦サービスの決定

本節では、前節で決定したタスクの候補に対して推薦するサービスの決定方法について述べる。従来のサービスナビゲーションシステムでは、末端のサービスに辿り着くまでタスクの選択を繰り返す必要があり、タスク選択の途中でサービスの提示することができなかった。ここでは、タスクの候補を提示する時点でサービスも一緒に提示する方法

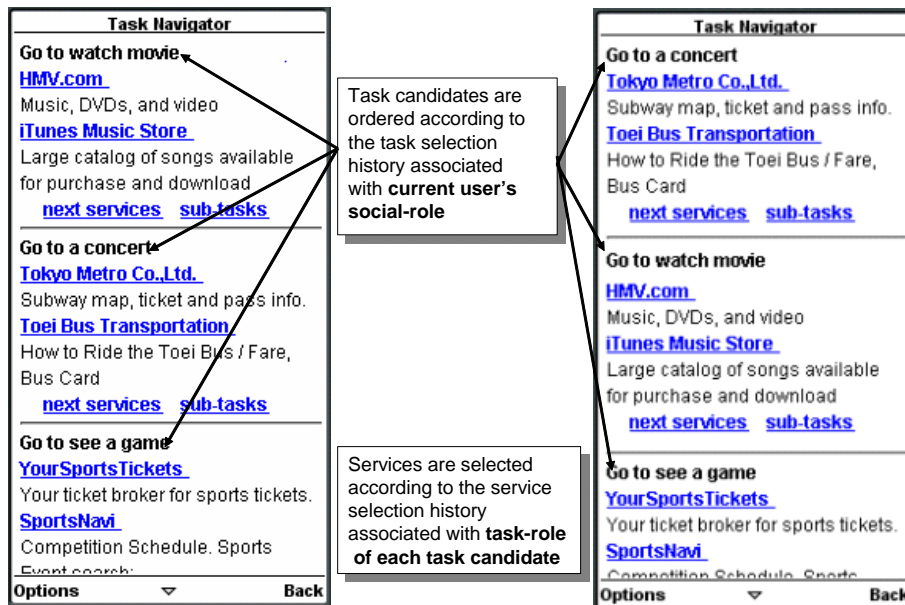


Figure 7: Screenshots of personalized task-list and their recommended services; left:task-list and services tuned for a user with family, right:task-list and services tuned for a user with friend

を提案する。タスクの候補に対してサービスを推薦する際、タスクと関連する膨大なサービスの候補の中から、タスクをメインに解決可能なサービスに絞り込んでユーザに提示する必要がある。ここでは、タスクの候補 A に対して、画面上で推薦するサービスの集合 S を求める方法について述べる。まず、task-role と関連付けられたタスクモデルを参照しタスク A の task-role (ATR とする) を取得する。もし、サービス選択履歴 (表 2) において ATR と関連付けられて蓄積されたサービスがある場合、当該サービスの中で利用頻度の高い上位 2 つのサービスを提示する。

一方、サービス選択履歴内に ATR と関連付けられて蓄積されたサービスがない場合には、タスク A のサブタスクのうち、サービスと関連付けられているタスク ST1 をすべて取得する。タスク ST1 に対し関連付けられている task-role を調べ、ATR、もしくは ATR の子ロールと一致するタスク (ST2 とする) を全て取得する。最後に、ST2 と関連付けられたサービスをタスクの候補 A に対して、画面上で推薦するサービスとする。これによって、サービスの選択履歴がない場合でもタスクをメインに解決可能なサービスに絞り込んで提示することができる。例えば、「Go to watch movie」タスクを実行しているときには映画に関連する情報を提示することができ「Move to a theater」のときには移動に関するサービスを提示することができる (図 6(b), (c))。

3.7 利用例

以下に、ロール指定画面において「Family Role」及び「Audience Role」を選択した場合のタスク候補の並び替え、及びサービスの推薦例を示す。まず、

task-role 「Audience Role」からタスクの候補「Go to watch movie」「Go to see a play」「Go to a concert」及び「Go to see a game」がタスクの候補として取得される (詳細は 3.3 節)。タスクの候補を取得後、表 1 に示すタスク選択履歴を参照し、「Family Role」の元でタスクの候補のうちの一つ「Go to watch movie」のみが蓄積されていることが分かる。従って、タスクの候補を並び替える際には「Go to watch movie」が優先的に上位に並び替えられる。次に、それぞれのタスクの候補に対し推薦サービスを決定する。例として「Go to watch movie」に対して推薦するサービスを決定する。サービス選択履歴 (表 2) を参照し「MovieAudience Role」と関連付けて蓄積されているサービスの中で最も多く利用されている「HMV.com」「iTunesMusicStore」が推薦される。一方、「Go to see a game」の場合には、「GameAudienceRole」と関連付けられて蓄積されたサービスがないため、タスクモデルの中で「Go to see a game」のサービスと関連付けられている下位のタスクのうち、「GameAudienceRole」と一致するサービス「YourSportsTickets」「SportsNavi」を推薦する。生成されたタスクリストを図 7(left) に示す。

一方、ロール指定画面にて「Friend」及び「Audience Role」を選択した場合、上記の例と同じタスクの候補が取得される。タスクの候補を取得後、表 1 に示すタスク選択履歴を参照し、「FriendRole」の元でタスクの候補のうちの一つ「Go to a concert」のみが蓄積されていることが分かる。従って、タスク並び替えの際に「Go to a concert」が優先的に上位に並び替えられる。次に、それぞれのタスクの候補に対し推薦サービスを決定する。例として「Go to a concert」に対する推薦サービスを決定する。サービス選択履歴

(表2)を参照し、「ConcertAudience Role」と関連付けて蓄積されているサービスの中で最も多く利用されている「Tokyo Metro Co., Ltd.」「Toei Bus Transportation」が推薦される。生成されたタスクリストを図7(right)に示す。

また、ロール選択メニュー内の「edit your menu」リンクからロールメニューを編集し、social-role「Brother Role」を初めて選択したとする。この場合、ロールオントロジー内の social-role 間の関係性から上位概念である「Family Role」のもとで蓄積されたタスク選択履歴を利用しタスクの候補を並び替えることができる。

4 結論

我々は、ユーザが実世界で抱える問題の構造を分析し、ユーザの問題の解決行動(タスク)をタスクモデルとして構造化してきた。本稿では、ユーザが実生活で担うロールによって実世界における行動が大きく変わり得る点に着目し、ユーザのロールに応じて必要なタスクを検索可能なサービスナビゲーションシステムを提案した。このとき、タスクの実行場所に比べ実世界でユーザが担うロールの数は少なく、一つのロールに対し対応付けられるタスクの数は多くなるという問題が発生する。この問題に対しては、タスクやサービスの選択履歴をユーザのロール(social-role や task-role)とともに蓄積しておき、現在のロールに応じてタスクを並び替える方法、及びタスクの候補に対してサービスを推薦する方法を提案した。結果として、必要とするタスクやサービスに到達するまでの時間を短縮する効果が期待できる。

ただし、現状ではロールに応じたサービスナビゲーション手法としては下記の点で不十分である。ロールオントロジー内で利用したリレーションは *is-a* のみであり、Sunagawa らが提案する *participate-in* リレーション等によりロールオントロジーをさらに充実させ、タスクやサービスの推論性能を向上する必要がある。さらに、ユーザのコンテキスト(場所、時間等)からユーザのロールを推定する機構や、ユーザのロールの変化を動的に取得する機構を検討する必要がある。

References

- [1] E.F. Codd. A relational model of data for large shared data banks. *Communications of the ACM*, Vol. 13, No. 6, pp. 377–387, 1970.
- [2] C.J. Fillmore. Types of lexical information, ausztige abgedruckt. In R. Dirven and G. Radden, editors, *Fillmore's Case Grammar: A Reader*. Julius Groos Verlag, 1987.
- [3] Y. Fukazawa, T. Naganuma, K. Fujii, and S. Kurakake. A framework for task retrieval in task-oriented service navigation system. In R. Meersman et al., editor, *Int. Work-*

shop on Web Semantics, LNCS 3762, pp. 876–885, 2005.

- [4] C. Masolo, L. Vieu, E. Bottazzi, C. Catenacci, R. Ferrario, A. Gangemi, and N. Guarino. Social roles and their descriptions. In *Proc. of the 9th Int. Conf. on the Principles of Knowledge Representation and Reasoning*, pp. 267–277, 2004.
- [5] T. Naganuma and S. Kurakake. Task knowledge based retrieval for service relevant to mobile user's activity. In Y. Gil et al., editor, *4th International Semantic Web Conference: ISWC 2005, LNCS 3729*, pp. 959–973, 2005.
- [6] R. S. Sandhu, E.J. Coyne, H.L. Feinstein, and C.E. Youman. Role-based access control models. *IEEE Computer*, Vol. 29, No. 2, pp. 38–47, 1996.
- [7] J.F. Sowa. *Conceptual Structures: Information Processing in Mind and Machine*. Addison-Wesley, New York, 1984.
- [8] J.F. Sowa. Using a lexicon of canonical graphs in a semantic interpreter. In M.W. Evens, editor, *Relational Models of the Lexicon: Representing Knowledge in Semantic Networks*, pp. 113–137. Cambridge University Press, 1988.
- [9] E. Sunagawa, K. Kozaki, Y. Kitamura, and R. Mizoguchi. Organizing role-concepts in ontology development environment: Hozo. In *AI Technical Report*, Vol. 4 of 1, pp. 453–468. Artificial Intelligence Research Group, I.S.I.R., Osaka Univ., 2004.
- [10] 古崎晃司, 來村徳信, 池田満, 溝口理一郎. 「ロール」および「関係」に関する基礎的考察に基づくオントロジー記述環境の開発. *人工知能学会論文誌*, Vol. 17, No. 3, pp. 196–208, 2005.