

マニュアルテキストを用いた個人化ヘルプシステム

Personalized Help System Using Software Manuals

岩下 志乃^{*1} 伊藤 紀子^{*1} 小林 一郎^{*1*2} 杉本 徹^{*1} 菅野 道夫^{*1}
 Shino Iwashita Noriko Ito Ichiro Kobayashi Toru Sugimoto Michio Sugeno

^{*1}理化学研究所 脳科学総合研究センター
 Brain Science Institute, RIKEN

^{*2}お茶の水女子大学 理学部
 Faculty of Science, Ochanomizu University

This paper presents a method for providing an everyday language smart help using an application software manual. This system takes four phases in the process: understanding of a user's input, matching the result of understanding with manual texts of application software, planning the output texts from the matched manual text, and paraphrasing the texts to output for the user. In the planning phase a rhetorical structure of the manual text is used for determining the output text. In the paraphrasing phase, a manual text matched in the matching phase is paraphrased using the analysis result in the understanding phase. Four methods for paraphrasing concerned with a novice user are identified by a subject experiment, and rules for these methods are applied in respect to the linguistic features extracted in the understanding phase. An example scenario is shown, and the applied rules and their effects are discussed.

1. はじめに

現在、コンピュータは生活のあらゆる場面に取り入れられるようになり、人々にとって身近な存在になってきている。しかし、操作に慣れていない人は様々な場面で逆に不便を感じることも多い。そこで、専門用語などの知識が無い人でも普段使っている言葉を用いてコンピュータを操作できるようにする、日常言語コンピューティング (Everyday Language Computing: ELC) の研究が進められている [岩爪 03]。本論文では、ELC の応用例のひとつとして、ユーザと対話しながらコンピュータ操作に関するヘルプを提供するシステムを提案する。

出力元の知識ベースとして、ソフトウェアに付属のマニュアルテキストを利用する。ユーザの入力テキストとマニュアルテキストはそれぞれ、ELC における言語理解モジュールを用いて意味理解が行われ、その結果を用いてユーザ入力に合ったマニュアルテキストが検索される。検索されたマニュアルテキストから、ユーザに必要なとされているテキストが選択され、理解結果に現れる言語的特徴から適切な言い換えが行われ、ユーザに出力される。

従来の代表的なヘルプシステムとしては、黒橋らの研究 [黒橋 01] が挙げられる。まず KNP を用いてユーザ入力を解析し、発話内容の構文木と発話タイプ (ユーザの質問が「How」「What」などのどの型に当たるか) の 2 つを抽出する。これらを用いて、タグ付けされた知識ベース内の知識 (テキスト) を検索し、出力するシステムである。

比較すると、本システムの特徴は 2 つある。1 つは、知識ベース内のテキストをそのまま出力するのではなく、ユーザに合わせた形に言い換えて出力することである。本論文ではユーザを初心者とし、コンピュータに慣れていない人に分かりやすく手順を説明することを目標としている。もう 1 つは、既存のテキストを知識ベースとして利用してユーザにヘルプを提供できることである。知識ベースを作成するためには大変な労力を要するが、ソフトウェアに必ず付属しているマニュアルを利用することでその手間を省くことができる。ただし、今回

は発話タイプとして「How」を用いることを前提としている。何故なら、マニュアルテキストはある操作を行う方法について記述してあるからである。しかし、本手法は知識となるテキストさえ用意すれば、同じプロセスで全ての発話タイプに対応できると考えられる。

本論文では、ELC の枠組みを利用した対話型ヘルプシステムの構成と、処理の内容について説明を行う。修辞構造を用いたテキストの決定手法と、言い換えルールの決定手法について詳しく説明する。言い換えルールを実際のマニュアルに適用した例を示し、今後の課題を述べる。

2. 日常言語コンピューティング

2.1 概要

日常言語コンピューティング環境は、セミオティックベース、クライアント秘書、言語プロトコル、言語アプリケーションで構成されている [岩爪 03]。セミオティックベースは、選択体系機能言語学 [Halliday 94] に基づき、人間が社会的活動を通して言語を習得することに着目して開発されたデータベースであり、言語テキストの理解と生成のために有効な形で構造化されたものである [伊藤 03]。データはコンテキストベース、意味ベース、語彙文法ベース、表現ベースの 4 層で構成されている。さらに、これらに加えて電子化辞書とコーパスを持つ。

クライアント秘書は言語理解生成モジュール、プランモジュール、クライアントモデルで構成され、システムが日常言語でユーザと対話するためのエージェントの役割を果たす [岩下 02]。言語理解生成モジュールでは、セミオティックベースを用いて入力テキストの理解・生成を行う。プランモジュールはテキスト理解の結果を受けて、秘書が取るべきアクションを決定する。もし決定されたアクションがアプリケーション操作である場合には、言語プロトコルの形で言語アプリケーションに情報を配信し、アプリケーションの操作が行われる。クライアントモデルは、ユーザの個人情報を格納する。

2.2 言語理解過程

言語理解処理はセミオティックベースを用いて前処理、選択体系機能言語学的 (SFL) 解析、概念解析の順に行われる [伊藤 03]。

まず、前処理として ChaSen と Cabocha を用いて形態素解

連絡先: 岩下志乃, 理化学研究所 脳科学総合研究センター 言語知能システム研究チーム, 〒 351-0198 和光市広沢 2-1, TEL: 048-462-1111(ext.7406), FAX: 048-467-6450, iwasa@brain.riken.jp

表 1: 語彙文法層インスタンス構造の例

morph	書式	設定	ツール	バー	の	フォント	で	フォント	名	を	クリック	し	ます
clause simplex	major-clause cls-elliptical cir-location material mat-doing despositive mat-elaborating mat-contact lg-concrete effective operative-voice indicative declarative conclusive ...												
group complex	SpatialLocation/ Adjunct/ Rheme1												
	nominal-group ngrp-complex ngrp-complex-non-thematized ngrp-hypotactic ngrp-expansion ngrp-elaborating ...												
group simplex	Qualifier/ Modifier				Head			Goal/ Medium/ Complement/ Rheme2		Process/ Predicator/ Rheme3			
	nominal-group ngrp-simplex non-thematic-ngrp named-individual nominal-head noun-head ...				nominal-group ngrp-simplex non-thematic-ngrp named-individual ...			nominal-group ngrp-simplex non-thematic-ngrp non-named-individual ngrp-concrete nominal-head ...		verbal-group vgrp-simplex temporal other-aux-coexist non-past vgrp-neutral vgrp-full simple-event ..			
word complex2	Thing/ Head												
	word-complex word-expansion ...												
word complex1	Modifier		Head		Relator/ Modifier		Thing/ Head		Nominal-marker/ Modifier		Event/ Head		-
	word-complex ...		word-complex ...		-		-		-		word-complex ...		-
word simplex	Modifier	Head	Modifier	Head	Relator/ Modifier	Thing/ Head	Nominal-marker/ Modifier	Thing/Head	Nominal-marker/ Modifier	Modifier	Head	Tense/ Polite-marker/ Modifier	
	word-simplex noun ...	word-simplex noun ...	word-simplex noun ...	word-simplex noun ...	word-simplex nominal-marker ...	word-simplex noun ...	word-simplex nominal-marker ...	word-neutral word-simplex nominals noun ...	word-simplex nominal-marker ...	word-simplex noun ...	word-simplex lexical-verb ...	word-simplex auxiliary-verb ...	
Morpheme	1	1	1	1	1	1	1	Head	Modifier	1	1	1	1
	base ...	base ...	base ...	base ...	base ...	base ...	base ...	base ...	suffix ...	base ...	base ...	base ...	base ...

析と係り受け解析を行い [工藤 02]、EDR 辞書に基づいた汎用辞書が検索される。その結果である SFL 特徴と EDR 概念識別子が次の SFL 解析に用いられる。SFL 解析ではセmiotickベースのそれぞれの層においてシステムネットワークの探索が行われ、インスタンス構造が生成される。表 1 に「[書式設定] ツールバーの [フォント] で、フォント名をクリックします。」の語彙文法層における解析例を示す。最後に、概念解析において、SFL 解析の結果を元に概念インスタンスフレームが作成される。

3. スマートヘルプシステムの構成

本研究で提案するヘルプシステムは、図 1 に示すようなユーザと秘書の対話型になっており、これをスマートヘルプシステムと呼ぶことにする。本システムは理解、マッチング、プランニング、言い換えの 4 つの処理過程から成る。マニュアルテキストは予めセmiotickベースによって理解され、利用可能な形で保持してある。

まず、ユーザが入力を行うと、セmiotickベースはその入力テキストの理解を行う。次に、ユーザ入力とマニュアルテキストの理解結果を用いて、質問テキストに合ったマニュアルテキストが検索される。そして、マッチングで複数のマニュアルが検索されたときに一つに決定したり、選ばれたマニュアルテキストからユーザが必要としている一部のテキストを抽出したりといった、プランニングの処理を行う。最後に、決定されたテキストがセmiotickベースとクライアントモデルの情報を用いて言い換えられ、ユーザに出力される。

次章では、ユーザに出力するテキストを決定するための処理と、選択されたテキストを言い換えるためのルール抽出について述べる。

4. テキストの決定

マッチングの処理によってユーザ入力に合ったマニュアルが検索された後、プランニングの処理によってユーザに出力するテキストを決定する。本研究では、2 種類のプランニングを定義している。一つは、検索された複数のマニュアルから一つを

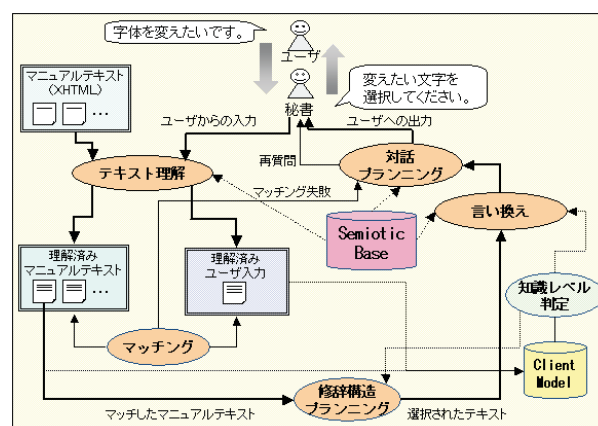


図 1: ヘルプシステムの処理過程

決定するために、ユーザへ聞き返しを行う対話プランニング、そしてもう一つは、修辞構造を用いてマニュアルテキストの中からユーザが必要としているテキストを抽出する修辞的プランニングである。それぞれについて以下に説明を行う。

4.1 対話プランニング

複数のマニュアルが検索されるということは、ユーザ入力に足りない情報が存在するためである。検索されたマニュアルにおける違いを探索し、足りない情報を推論する。その結果、ユーザに聞き返しを行う文を生成して出力する。

例として、ユーザ入力「円の色を変えたい」である場合を考える。マッチング処理を行うと、2 つのマニュアル「描画オブジェクトの輪郭の色を変更する」と「描画オブジェクトの塗りつぶしの色を変更する」が検索される。このとき、円は描画オブジェクトの低位概念であるため、2 つのマニュアルが「円の { 輪郭 or 塗りつぶし } の色を変更する」ことを表していることが推論される。この結果、「円の輪郭の色ですか？それとも塗りつぶしの色ですか？」とユーザに聞き返し、ユーザがどちらを答えるかによってマニュアルテキストが一つに決定する。

表 2: 修辞構造によるテキスト決定ルール

修辞構造	ルール
solutionhood	Nuclues のテキストのみを対象とする。
sequence	全てのテキストを順番に対象とする。
conjunction	条件節とユーザ入力のマッチングを行い 点数が閾値以上のテキストを対象とする。

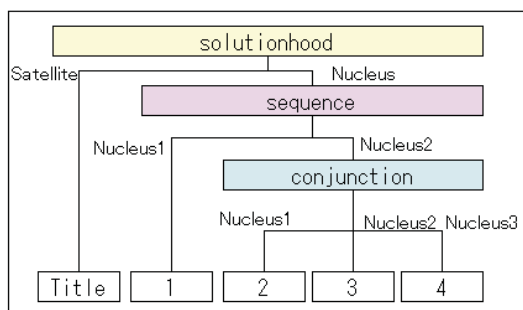


図 2: 修辞構造の例

4.2 修辞構造の利用

マニュアルテキストが一つに決定したとき、その中からユーザが必要としている部分を決定するため、本研究では修辞構造を利用したルールを設定する。ルールはユーザの知識レベルごとに決められており、例えば初心者には分かりやすい説明にするために付加的な情報をなるべく減らすようにしている。表 2 にルールの一部を示す。

例として、図 2 のような修辞構造をもったマニュアルテキストについて考える。まず、一番上の特徴が solutionhood なので、Nucleus のみが選ばれる。次の特徴が sequence なので、Nucleus1, Nucleus2 が順番に選ばれ、文 1 が決定する。次の特徴が conjunction なので、それぞれの文の条件節とユーザ入力のマッチングを取り、点数が閾値を超えたテキスト（例えば文 2）が選ばれる。

5. 言い換え

5.1 言い換えルール抽出

マニュアルテキストをユーザに合わせた形に言い換えるため必要なことは、どのような言い換えがユーザにとって必要であるかを抽出することと、それらの言い換えがどのような言語的特徴のときに現れるかをルール化することである。まず、ユーザがワープロ操作をするときにヘルプを提供するという場面では実際にどのような対話が行われるかを抽出するため、被験者実験を行った。ユーザは指定の文書を作成するタスクを行い、操作方法について分からないことがあるときには、別室にいるインストラクターに随時マイクを通して質問を行う。ユーザはワープロ操作の初心者、インストラクターはワープロ操作上級者で、35 組についてそれぞれ 1 時間ずつ実験を行った。対話音声を書き起こしたテキストからインストラクターが操作の説明をしている回答部分のみを抜き出し、その特徴を分析した。

5.2 言い換えルール

被験者実験の結果とマニュアルテキストの分析結果から、分割、補完、ムード変化、類義語選択の 4 つの言い換えを定義し

た。それぞれの言語的特徴と言い換え方法を以下に説明する。

分割

被験者実験の結果、インストラクターの発話はまず操作するボタンやメニューバーなどオブジェクトの位置を説明し、ユーザにその場所の確認をさせた後に操作方法を説明している場合が大半を占めた。そこで、マニュアルテキスト内に場所を示す節がある場合には、それをひとつの文として独立させて説明し、その後操作の説明をすることが考えられる。

場所を表す名詞句には、意味ベースのインスタンス構造にの SFL ルール *SpatialLocation* が現れる。この句を独立させ、「～があります」という文を生成する。次に、操作の説明を行うため、元の文における分割された句を「そこ」という指示詞に変換し、「そこで～」という文を生成する。

補完

マニュアルテキストでは、手順を説明した後に結果を説明していない場合が多い。しかし、初心者は操作を行った後にどのような結果になるのかが分からないと安心して操作を行うことができない。そこで、マニュアルテキストの最後に結果を補完することを考える。マニュアルに書かれた操作を行うと、結果としてそのタイトルの状態になる。何故なら、タイトルはその操作を行う目標であり、目標は最終的な結果となるからである。そこで、操作説明の最後の文を「原因」とし、タイトル文が「結果」となるように 2 つの文を繋げる。つまり、補完後は「(原因)して、(結果)」という文が生成される。

補完を行う際には、タイトル文において冗長となる句を省くことが必要である。ここでは、語彙文法ベースにおいて Modifier (修飾語句) の特徴を持つ句は冗長であると判断し、省略する。

ムード変化

マニュアルテキストは書き言葉で独白的であるのに比べ、インストラクターの説明は話し言葉で対話的である。マニュアルテキストからユーザへの説明に変換すると、語彙文法ベースにおけるムードが変化することにより言い回しが変化する。

本研究では 2 種類のムード変化を定義する。一つは埋め込み文、もう一つは語尾である。埋め込み文に関するムード変化は、文が「変更する文字」のような埋め込み文の構造をもつ場合に行われる。このとき、ムードの特徴が *optative* となり、「～たい」という語が付加される。この例の場合は「変更したい文字」と変化する。語尾に関するムード変化では、ムードの特徴が *imperative* となり、語尾が「～ます」から「～ください」へと言い換えられる。

類義語選択

同じ概念の言葉を説明するときでも、ユーザによって違う言い回しを使う場合は多い。そのため、ユーザに合った語彙を選択することは重要である。特に、マニュアルテキストはある程度の専門用語知識を前提として作られているため、それらをユーザにとって分かりやすい言葉に変換することが必要不可欠である。

本論文では、語彙をユーザの使用頻度、専門度によって選択する。初心者ユーザは、本人が使う頻度が高い語彙と専門度の低い語彙は理解しやすいと考えられるので、これに当てはまる語彙を使用する。また、専門用語を説明する辞書を用意し、マニュアルテキスト内に専門用語が用いられている場合には、辞書内の説明に言い換える。

表 3: ユーザと秘書の対話例

	対話	言い換え
ユーザ	字体を変えたいです。	
秘書	変えたい文字を選択してください。	ムード変化, 類義語
	ツールバーに MS 明朝と書いてあるところがあります。	分割, 類義語
	そこでフォント名をクリックして、字体を変えてください。	補完, 類義語, ムード変化

表 4: マニュアル「フォントを変更する」

手順	テキスト
1	変更する文字列を選択します。
2	[書式設定] ツールバーの [フォント] で、フォント名をクリックします。

5.3 ルール適用例

表 3 に示す具体的な対話例を用いて、言い換えのシミュレーションを行う。言い換えの順番としては、大きな範囲に影響が及ぶものからルールを適用すると効率が良いので、分割 補完 ムード変化 類義語の順番で行う。ユーザの質問に対して、表 4 に示すマニュアルテキストが検索されたとすると、手順 1、手順 2 の順番で、それぞれ表 3 における秘書の発話のように言い換えられる。

まず、表 4 における手順 1 について言い換えを行う。手順 1 の理解結果から、分割と補完は、条件となる特徴が無い場合行われず、ムード変化により、「変更する文字列」が「変更したい文字列」に、「選択します」が「選択してください」にそれぞれ言い換えられる。最後に、類義語選択により、「変更する」が「変える」に、「文字列」が「文字」に言い換えられ、最終的に「変えたい文字を選択してください。」という文がユーザに対して出力される。

次に、手順 2 について言い換えを行う。手順 2 の理解結果例は表 1 に示したとおりである。まず「[書式設定] ツールバーの [フォント] で」の句がオブジェクトの場所を表す特徴 *SpatialLocation* を持つため、まずこの句を独立させて「[書式設定] ツールバーの [フォント] があります」という文を生成する。この文に対して、補完は条件を満たさないで行わない。また、「あります」のような存在を表す文はムードが変化しても言い回しが変わらないため、ムード変化による言い換えは行われず、類義語変化は、「[書式設定] ツールバーの [フォント]」という専門用語が、専門用語辞書を用いることにより「ツールバーに MS 明朝と書いてあるところ」と置き換わり、「ツールバーに MS 明朝と書いてあるところがあります。」という文が生成される。

場所を表す文を生成した後は、そのオブジェクトに対する操作を説明する文を生成する。手順 2 の文において前半部で説明された句を「そこ」に言い換えると、「そこでフォント名をクリックする」となる。これは最後の文であるので、補完条件を満たす。補完される文はタイトルの文から生成される。Modifier の特徴を持つ節が省略されるので、補完される文は「フォントを変更する」という文となる。よって、補完された文は「そこでフォント名をクリックして、フォントを変更する」

となる。最後に、手順 1 と同様にムード変化と類義語選択を行うと、「そこでフォント名をクリックして、字体を変えてください。」と言い換えられ、この文が出力される。

6. おわりに

本論文では、日常言語コンピューティングの枠組みを用いて、既存のマニュアルテキストを基にしてユーザに分かりやすいヘルプを提供するシステムの提案を行った。検索されたマニュアルテキストから出力テキストを決定するため、修辞構造に基づいた推論を行う。また、言い換えのルール抽出には、実際にユーザとインストラクターが質問/応答する対話実験を行い、そのテキストを分析した結果を用いた。実際のマニュアルテキストに対して言い換えルールを適用し、言い換えが適切に行われることを確認した。

今後の課題としては、対話プランニング手法の改良が挙げられる。例えば、説明の順番をもっと分かりやすいように並べ替えて出力したり、出力したヘルプに関して再度ユーザが質問した場合に、対応できるようにすることが考えられる。また、マニュアルテキスト以外の言語的知識を利用することにより、現在のような「How」型だけでなく他の発話タイプの質問に回答することができると思われる。

参考文献

- [岩爪 03] 岩爪 道昭, 小林 一郎, 杉本 徹, 岩下 志乃, 高橋 祐介, 伊藤 紀子, 菅野 道夫: “日常言語コンピューティング (第 2 報) - 日常言語に基づく計算機資源の管理・実行環境を目指して-”, 人工知能学会論文誌, Vol.18, No.1, pp.45-56 (2003).
- [黒橋 01] 黒橋 禎夫, 日笠 亘, 藤井 綱貴: “入力質問と知識ベースとの柔軟なマッチングに基づく対話的ヘルプシステム”, 2001 年情報学シンポジウム論文集 (2001).
- [Halliday 94] Halliday, M. A. K.: An Introduction to Functional Grammar (2nd ed.), London, Edward Arnold (1994).
- [伊藤 03] 伊藤 紀子, 杉本 徹, 高橋 祐介, 小林 一郎: “セミオティックベースを使ったテキスト処理アルゴリズム”, 第 17 回人工知能学会全国大会, 3B1-01 (2003).
- [岩下 02] 岩下 志乃, 岩爪 道昭, 小林 一郎, 杉本 徹, 菅野 道夫: “日常言語コンピューティングにおけるクライアントモデルの提案”, 第 16 回人工知能学会全国大会, 3B1-03 (2002).
- [工藤 02] 工藤 拓, 松本 祐治: “チャンキングの段階適用による日本語係り受け解析”, 情報処理学会論文誌, Vol.43, No.6, pp.1834-1842 (2002).