

自然言語を媒介としたソフトウェア制御のための意味マッチング方式の検討

A Study on Semantic Matching for Natural Language Mediated Software Control and Management

岩爪道昭*¹ 小林一郎*² *¹ 高橋祐介*¹ 菅野道夫*¹
 Michiaki IWAZUME Ichiro KOBAYSHI Yusuke TAKAHASHI Michio SUGENO

*¹ 理化学研究所 脳化学総合研究センター *² 御茶の水大学 理学部
 Brain Science Institute, RIKEN Faculty of Science, Ochanomizu University

Everyday language computing (ELC) is a new computational paradigm that all people, from small children to aged persons, can access and use computing systems with his/her own everyday language. As a way to realize ELC, we proposed a framework of language-based operating system (LOS), and we are now working intensively to develop the fundamental part of it. In this paper, we report our status of research on LOS. In LOS, all kinds of information processing are done through everyday language. Language-based application is a kind of agent software that provides services through language. We explain the characteristics of the structure of language-based application programming interface (LAPI), and semantic matching process of language-based applications.

1. はじめに

我々は、ヒトの知性における言語が果たす役割を重要視し、従来の数値や形式的記号論理に基づく情報処理から、日常言語による情報処理を行う「日常言語コンピューティング (Everyday Language Computing) (以後 ELC と略記)」という新しいパラダイムを提案している[岩爪 02, 岩爪 03]。また、このパラダイムの実現形態の一つとして、「日常言語オペレーティングシステム(Language-based Operating System) (以後 LOS と略記)」の実現を目指し、実証システムの研究・開発を進めている[伊藤 03, 岩下 03, 杉本 03]。

LOS は、既存アーキテクチャ上にミドルウェア層およびアプリケーション層として実装される仮想 OS である。現在我々が取り組んでいる最小スペック版 LOS の構成と機能を図 1 に示す。

ミドルウェア層では、テキスト理解・生成のための資源であるセミオティックベース (Semiotic Base) (以後, SB と略記) とそれに基づいてテキストの処理を行うモジュール、クライアントモデル、既存アプリケーションを言語化するための言語アプリケーションインタフェース等で構成される。

一方、アプリケーション層は、実際にユーザに対して、アプリケーションサービスを提供するレイヤーであり、ユーザにとってはサービスを利用するためのインタフェースになる。具体的には、ユーザ専属のクライアント秘書や言語アプリケーションなどが挙げられる。

このように、LOS における全てのサービスは、クライアント秘書を介して、LOS 上に存在する言語アプリケーションによって提供される。言語アプリケーションは、既存のアプリケーションを言語化することで実現される。ここで、言語化とは、コンピュータ内部の全ての資源 (ソフトウェア、ハードウェア) に対して、言語 (の意味) に基づくアノテーションを施し、そのアノテーション情報に基づいて、コンピュータ環境のサービスを実行・管理することである。

我々は、これまで、既存のアプリケーションを言語で操作・管理するための言語プログラミングインタフェース (以下, LAPI) の

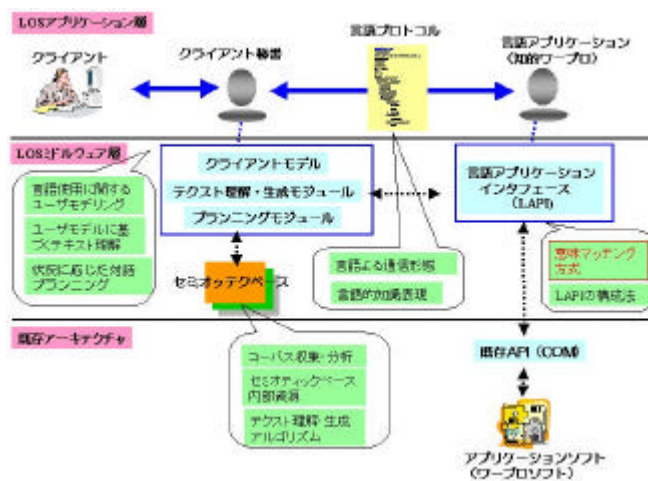


図 1 LOSの構成と機能

構成法を提案し、クライアント秘書から送られてきた要求を既存アーキテクチャにグラウンディング (テキスト二次理解¹) するための方法として、文法を利用した言い換えによる意味理解方式を提案している[岩爪 03]。

しかしながら提案した意味理解方法は、SLFT²における言語の意味情報の一部しか利用しておらず、状況層や意味層等の情報は活用しておらず、より複雑な意味を取り扱、実行可能性、頑健性などにおいて問題があった。

本論文では、SLFT における4つの層 (表現、語彙・文法、意味、状況) の情報利用した意味マッチングの方式について検討する。

連絡先: 岩爪道昭, 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.7403), Fax: 048-467-6450,
 e-mail: iwazume@brain.riken.go.jp

¹ クライアント秘書におけるセミオティックベースを用いたユーザ発話理解は、テキスト一次理解となる。言語アプリケーション側では、セミオティックベースの資源を常に共有しているとは限らないと仮定している。

² Systemic Functional Linguistic (選択体系機能言語理論) [Halliday 94]

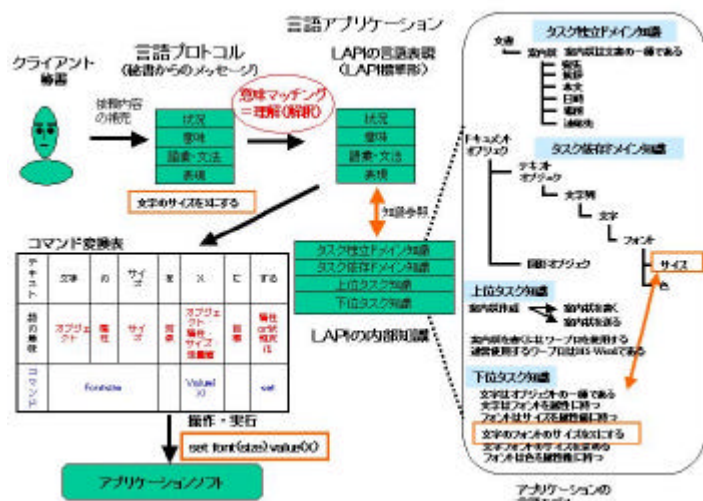


図2 言語アプリケーションの構成

2. アプリケーションの言語化とLAPI

2.1 構成

言語アプリケーション及びLAPIの構成を図2に示す。LAPIは、言語プロトコルによるクライアント秘書や他のエージェントからのメッセージを受信・理解するための外部インタフェース、日常言語で表現された内容を実行可能な形式に展開する意味マッチングモジュール³、展開された自然言語表現系列を対象アプリケーションの実装レベル(コマンド)に変換する内部インタフェースによって構成されている[岩爪03]。

2.2 LAPI 内部知識

LAPI 内部の知識は大別すると、**タスク依存ドメイン知識**、**タスク独立ドメイン知識**、**タスク知識**によって構成されており、外部からの要求を実行可能な実装レベルの形式に展開するための知識や、アプリケーションの実行結果を外部へ報告する際に、形式表現から言語表現へ変換するために用いられる(詳細は[岩爪03]を参照)。

2.3 言語アプリケーションのアクション分類

意味マッチング方式を検討する前に、ここでは、言語アプリケーションが実行するアクションのタイプについて整理する。言語アプリケーションのアクションは、**物理アクション**と**応答アクション**に大別される。

(1) 物理アクション (ドメイン・アクション)

物理アクションは、対象アプリケーションソフトウェアの制御・実行に関するものである(例:「文字のサイズを16ptに設定する」,「招待状のテンプレートを開く」など)。

また、物理アクションには、ソフトウェア内部の状態を確認する**ステータス同定アクション**(例:「現在のタイトルレの文字の色を調べる」)なども存在する。

(2) 応答アクション (インタラクション・アクション)

応答アクションは、ユーザや他のエージェントとの対話インタラクションに関するもので、**同定**(例:「誰に送る招待状ですか?」), **不確定情報の確認**(例:「文字の色は黒でいいですか?」), **報告**(例:「文字の色を赤にしました」), **実行予告**(例:「MS-Wordを起動します。」), **了解**(例:「わかりました」)などがある⁴。

3. 意味マッチング方式の検討

3.1 LAPI 標準形

2.3節のアクション分類をさらに具体化したものが**LAPI標準形**である。LAPI標準形は、対象アプリケーションの操作・制御に関する標準的な自然言語表現である。言い換えると、LAPI標準形は、対象アプリケーションの言語モデルの一部であり、メソッド知識を汎化したものであると言える。

ワープロソフトにおけるLAPI標準形の例を以下に示す。

- 例1:<オブジェクト>の<属性>を<値>に設定する。
- 例2:<フォント>の<属性>を<値>に設定する。
- 例3:<フォント>の<サイズ>を<16pt>に設定する。
- 例4:<フォント>の<サイズ>を<大きい>に設定する。

ここで、<・>で囲まれた部分は、操作対象オブジェクトのクラス名、サブクラス名、属性名、値などを指し示している。

例1は、最も汎化されたLAPI標準形の1つである。例2は、具体的なオブジェクトクラスが指定された表現形(クラス表現)になっている。一方、例3は、全てのスロットが埋められたLAPI標準形のインスタンス表現である。このインスタンス表現まで具体化されると、内部インタフェースに受け渡され、コマンド変換表に基づいて、実際にアプリケーションが操作・実行される。例4は、例3と類似した表現であるが、<値>のスロットに、定性値として「大きい」という言語表現が入っているケースである。クライアント秘書を介して言語アプリケーションに送られてくる指示内容は、「文字もう少し大きくして欲しい」といった定性的かつ曖昧な表現になるケースが少なくない。そこで、例4のような言語による定性値の表現を媒介して、最終的に例3のインスタンス表現に具体化することになる。

スロット値の定性表現と定量表現との間のマッピングは、相対的で状況依存である場合も少ないが、本論文では、簡単のために、ファジイメンバシップ関数のような比較的単純な写像関係を想定している。

LAPI標準形は、SBによって解析された入力文と同様に、コンテキスト、意味、語彙文法、表現の4つのレイヤーで構造化されている。

しかし、言語アプリケーションは、SBの利用を必ずしも想定していない⁵。また、クライアント秘書も言語アプリケーションが提供するサービスの詳細や対象アプリケーションの構造・機能に関する詳細については一切感知しない。したがって、クライアント

³ [岩爪03]では、プランニングモジュールとしている通り、意味マッチングに必要な推論、プランニング等の機能も含まれる。

⁴ FIPA ACL[FIPA 97]における遂行文(performative)と類似した概念であるが、本稿では、より具体的なアプリケーションサービスのカテゴリを想定し、その範囲内で汎用性のあるアクションのタイプを整理している。

⁵ SBが十分整備されていない現時点では、言語アプリケーションにおけるSBの利用は、開発者にとって負担が大きく現実的でない。

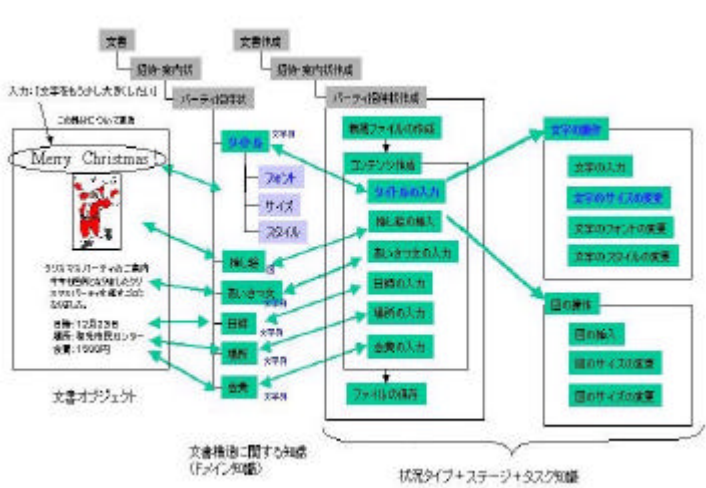


図3 状況とタスク構造

秘書を介して送られてくるユーザの要求文 (入力文) と LAPI の内部言語表現では、付与されているアノテーションの表記ラベルやその基になっているモデルが異なっているケースも考えられる。

本研究における意味マッチングとは、内部知識モデルや表層のラベル名が完全に一致しない外部からの要求 (入力文) から LAPI 標準形のインスタンスを生成するプロセスとして従えられる。これは、言語アプリケーションは自己の資源を最大に利用しながら LAPI 標準形空きスロットをできる限り埋めていくユニフィケーションプロセスとみなすこともできる。このような考え方は、**アクティブマッチング**と呼ばれる動的な意味処理である [井佐原 95]。 [井佐原 95] との相違点は、本研究では、SLFT の 4 層の構造を考慮に入れたマッチングにあると考えられる。

3.2 意味マッチングに利用する言語特徴

入力文には、SB 用いて解析された、表現、語彙文法、意味、概念、コンテキストに関する様々な言語特徴情報が付与されている (詳細は [伊藤 03], [杉本 03] を参照)。しかし、これらの情報は全て等価な意味を持つ訳でなく、対象のドメインやタスクによっては、その一部を利用するだけ十分なケースも少なくない。

本論文では、既存ワープロソフトを言語化した知的ワープロによる文書作成支援タスクを、応用事例としているので、最低限ワープロソフトの制御・実行が実現できる範囲で SB 各層の情報を利用することにする。以下に、意味マッチングに利用する言語特徴情報を示す。

- (1) コンテキスト
 - 状況タイプ
 - ステージ, ムーブ
 - 概念辞書 - クラス概念, インスタンス概念
- (2) 意味
 - 発話機能
- (3) 語彙文法及び表現
 - 語句単位の言い回し (キーワード)

3.3 コンテキストを考慮した処理

先行詞が明示されていないような照応解析や指示対象が省略されている場合には、コンテキストを考慮した意味処理が必要不可欠である。

例えば、図3では、「文字をもう少し大きくしたい」という発話において、「タイトル」のという表現が省略されている。この省略情報を補うためには、ドキュメントの操作履歴から、直近の操作オブジェクトを特定したり、そのオブジェクトが、ドメイン知識階層のどれに対応し、どのようなタスク構造と結びついているか関連付けられている必要がある。

対話によって発生する場面場面の状況は⁶、ただ一度きりのものであり、理論上は無制限個存在する。ただし、ワープロによる文書作成のように限定されたドメインタスクに限定すれば、それらはカテゴリ化可能であり、典型的なプロトタイプが存在すると考えられる。このプロトタイプは状況タイプとそれに関連するタスク構造を基軸として、表現される。

すなわち、ワープロタスクにおける状況タイプの最上位ノードを「文書作成 (write-document)」とし、これより上位の状況タイプに関しては言語アプリケーション (知的ワープロ) は感知しない。また、タスクの構造やメソッド単体でも状況 (ステージ) を構成することにする。タスク構造にラベルを付与することで内部の詳細な構造をカプセル化・インデックス化 (タスク・メソッドなどの動詞概念を名詞化・名詞句化) することで、SB はあらかじめ全ての状況タイプを用意しなくてもよい。

4. 意味マッチングの流れ

意味マッチングのプロセスは、(1)ステージ同定、(2)関連知識の検索・同定、(3)LAPI 標準形呼び出し、(4)インスタンス生成 (ユニフィケーション) に分けられる。

以下の例文を用いて、意味マッチングのプロセスについて説明する。

例: 「文字をもう少し強調したい」

(1) ステージ同定

入力文のステージ情報をもとに現在のステージを同定する。図4の例では、「文書作成 (write-document)」というラベルのみを共有していると仮定している。ここから以下の部分のステージ情報に関しては、LAPI 内部のタスク知識構造を流用する。

(2) 関連知識の検索・同定

ステージ情報に関連付けられている LAPI 内部のドメイン知識を検索する。図4では、コンテキスト層の概念辞書ラベルのクラス階層のみ記載しているが、実際には、これらの知識は全て4層の言語特徴表現が付与された自然言語によって記述されている。この例では、入力文の「強調する (emphasizing)」という情報から、強調に関連するフォントの属性 (サブタイプ) として「サイズ (size)」および「ボールド (bold)」が同定される。

⁶ 状況意味論における状況は、この意味に近い。一方、SFLT における状況タイプは、状況 (対話の場面) を構成するプロトタイプのようなものであると考えられる。

(3) LAPI 標準形 (クラス) 呼び出し

関連知識が同定されると、指示対象となっているオブジェクトの属性操作に関する LAPI 標準形 (クラス表現) が呼び出される。図 4 では、

<フォント>の<属性>を<値>に設定するという LAPI 標準形が呼び出されている。

(4) インスタンス生成 (ユニフィケーション)

入力文、同定された内部知識の情報を利用して、呼び出された(3)で呼び出された LAPI 標準形の空きスロットに値を埋めていく。図 4 の例では、文字強調に関する属性を持つサブタイプとして「サイズ(size)」と「ボールド(bold)」が選ばれており、それぞれに対してインスタンスを生成する。例えば「サイズ(size)」の場合は、「強調(emphasis)」と「大きさ(qualitative-size)」の二つ属性間には、

強調(emphasis)=1 大きさ(qualitative)=大きい(large)

という写像関数が用意されており、LAPI 標準形の<値>のスロットには、<大きい(large)>が埋められる。同様に、<ボールド>に関するインスタンスの<値>は、<太い(thick)>という定性値が代入される。さらに、これらの定性値は、定量値へ変換する写像関数によって、それぞれ<16 pt>、<1>という定量値が与えられる。

5. まとめ

本稿では、自然言語を媒介としたソフトウェアの制御 実行のための意味マッチング手法を提案した。本手法は、ソフトウェアの資源全てに対して、人間及び機会がともにアクセス 理解可能な自然言語表現とその意味特徴情報を付与 (言語化) し、その自然言語を解釈することで、関連するサービスやその組み合わせを実行することを狙っている。現在、知的ワープロシステムを応用対象としてプロトタイプの開発を進めており、COM(Component Object Model)を介して、MS-Word と Java ワ

ープロ(フリーウェア)の2つのソフトウェアが操作可能になっている[杉本 03]。今後は意味マッチング方式の形式化、知的ヘルプシステムとの統合、他のドメイン、タスクへの提供、異なるアプリケーション間による問題解決等に取り組む予定である。

参考文献

[岩爪 02] 岩爪, 小林, 伊藤, 高橋, 菅野: 日常言語コンピューティング - 言語現象の社会的, 機能的解明と日常言語オペレーティングシステムの実現に向けて -, 人工知能学会論文誌 近未来チャレンジ特集, Vol.17, No.2, pp.181-185, 2002.
 [岩爪 03] 岩爪, 小林, 杉本, 岩下, 高橋, 伊藤, 菅野: 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 近未来チャレンジ特集, Vol.18, No.1, pp.45-56, 2003.
 [小林 02] 小林, 岩爪, 杉本, 岩下, 小澤, 菅野: 自然言語をコンピュータの通信プロトコルにする研究, ファジィ学会誌 Vol.14, No.5, pp.491-502, 2002.
 [伊藤 03] 伊藤, 杉本, 高橋, 小林: 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 2003 年度人工知能学会全国大会, 3B1-01, 2003.
 [岩下 03] 岩下, 小林, 伊藤, 杉本, 菅野: セミオティックベースにおけるコンテキスト層の検討 2003 年度人工知能学会全国大会, 3B1-03, 2003.
 [杉本 03] 杉本, 岩爪, 小林, 伊藤, 高橋, 岩下, 菅野: セミオティックベースを使った日常言語アプリケーションシステム 2003 年度人工知能学会全国大会, 3G2-06, 2003.
 [Halliday 94] M. A. K. Halliday: An introduction to functional grammar, 2nd ed. London: Edward Arnold, 1994.
 [井佐原 95] 井佐原 概念辞書を用いた日本語理解システムに関する研究, 電子技術総合研究所研究報告, No.972, 1995.
 [FIPA 97] FIPA'97 Specification Part2: Agent Communication Language, <http://www.fipa.org>, 1997

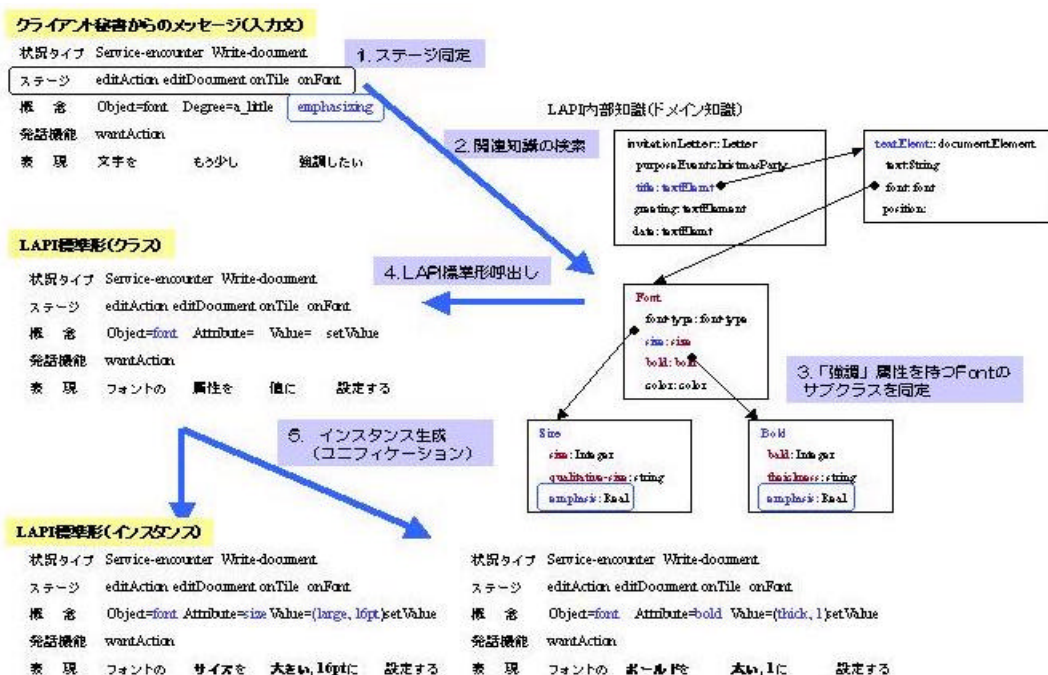


図 4 意味マッチング例