マルチデータベース日本語インタフェースの試作 — 日本語入力文解析処理 —

2D－1

住田一男, 真鍋俊彦, 高橋一重
(株) 東芝 研究開発センター

1. はじめに

計算機ネットワークの発達に伴い，分散した様々な情報にアクセスできる環境が整いつつあり，様々な情報へのアクセスに関するハードウェア上の障害はなくなってきた．しかし，情報の格納形式やコマンド形式などがアプリケーションやデータベース管理システム間で異なるため，実際のユーザにとっては，情報を入手できないという問題が依然として存在している [1].

様々なユーザが種々の情報を簡便に利用できることを目的として，異種・分散のデータベースを対象とした自然言語インタフェースを開発した．本稿では，ユーザから入力される自然言語の質問文を解析し，SQL 形式の中間コマンドに変換する処理について述べる．文字レベルでの類似性を判定することにより，ユーザが入力する表現をデータベース側の語彙に対応づける点に特徴がある．

2. システムの概要

システムの全体構成を図 1に図示する (破線内が試作部分)．変換ルールには，“～が～以上” といったDB 検索特有の条件提示表現を，条件式に変換するためのルールを格納している (3.3 節参照)．概念辞書には語彙とデータベーススキーマとの対応関係を，データ辞書には各データベースのスキーマと位置，種類に関する情報を格納している [2].

入力文解析部は，変換ルールと概念辞書に基づいて入力文から SQL 形式の中間コマンドを生成する．

問合せ生成部は，概念辞書とマルチ DB 情報に基づいて実際の問合せコマンドを生成し，各 DB に対応するゲートウェイを介して DB にアクセスする [2].

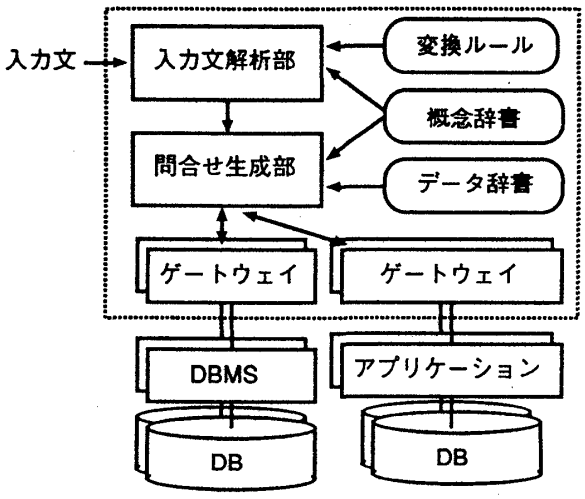

図 1: システムの全体構成

3. 入力文解析部

入力文解析処理の流れを図 2に図示する．実線矩形部は処理モジュールを，実線楕円部は辞書等のデータを，破線矩形部は入力および中間データをそれぞれ意味している．分野移行性を配慮し，詳細な世界知識を記述する方式 (例えば [3]) を避け，DB がどのような属性を持っているか等の単純な知識 (概念辞書) に基づいて処理する．

入力文の処理に先立って，同義表現変換処理のための語彙辞書と形態素解析辞書に，概念辞書内で用いられている語彙を登録している．

3.1 形態素解析

基本語彙約 6 万語に概念辞書から抽出された単語が追加された形態素辞書に基づき，入力文の単語切りを行う．例えば，図 2の入力文 (a) から，形態素列 (b) が得られる．

3.2 同義表現変換

ユーザの入力した表現を，概念辞書内で定義されている語彙に対応づける．例を図 3に示す．

A Natural Language Interface for Multi-Database System — Japanese Sentence Analysis —

Kazuo SUMITA, Toshihiko MANABE and Kazushige TAKAHASHI, Research and Development Center, Toshiba Corp.
1 Komukai-Toshiba-cho, Saiwai-ku, Kawasaki, 210, Japan.

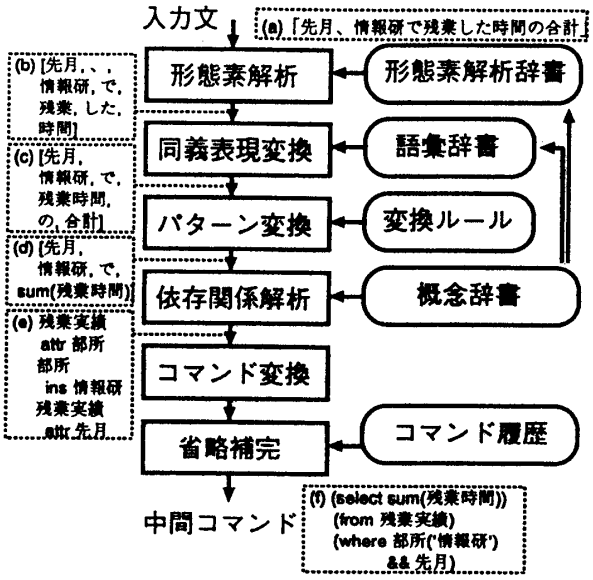

図 2: 入力文解析部の処理の流れ

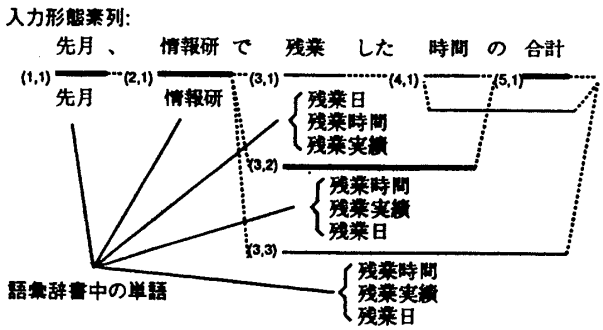

図 3: 類似語列の検出

処理の流れは以下の通りである.

1 名詞句を構成する部分形態素列を求める
2 1で求めた部分形態素列ごとに語彙辞書に格納されている語との類似度を求める
3 2の組合せのうち類似度の総和が最大になるパスを求める
4 3で求めたパスを構成する部分形態素列から求められた類似語を採用する

図2の例文の内 "残業" 以降の形態素列を考えると, 名詞句を構成する部分形態素列として, "残業", "残業, した, 時間", "残業, した, 時間, の, 合計" 等, 6 部分列が得られる. これらのそれぞれの部分形態素列について, 類似する語が求めることができる (例えば, "残業, した, 時間" に対して "残業時間" や "残業実績" など).

図 3に示すように, 部分形態素列の組合せはラティス構造で表現できる. このラティス構造から最大の類似度の組合せとなるパスを求める. なお, 部分形態素列と語彙辞書に格納されている語との類似度は, 文字の一致数により算出し, 部分形態素列の形態素数により正規化している.

3.3 パターン変換

"残業時間が 60H 以上" や "従業員番号が 850321" といった DB 中の属性値の範囲を指示する表現や, "〜の合計" といった集約関数に対応する表現を条件式に変換する. この処理は, 表層表現と対応する条件式とを対として記述した変換ルールに基づいたルールベースな処理である. 例えば, 図 2の形態素列 (c) 中の "残業時間, の, 合計" は, (d) に示すように "sum(残業時間)" に変換される.

3.4 依存関係解析

概念辞書に基づいて, 語と語がどのような関係にあるかを解析する. 図 2において, "attr" はエンティティー属性関係, "ins" は属性−属性値関係を示す.

3.5 コマンド変換, 省略補完

これらのステップで, SQL 形式に変換する. コマンド履歴には, それまでに入力された入力文から得られた中間コマンドを格納している. このコマンド履歴に基づいて, 省略を含む文が入力された場合に, 省略を補完する (図 2の (f)).

4. おわりに

文字レベルの類似性に基づいて, ユーザが入力する表現を, データベースの定義で用いられている語彙へ対応づけることにより, 簡便な知識 (概念辞書) に基づくロバストな入力文解析処理を実現した. 類似語彙へマッピングするため, ユーザが予想していないような中間コマンドを生成する可能性がある. このような場合に対処するため, ユーザとのインタラクション機能を組み込んでいく.

参考文献

[1] 例えば bit 別冊, ACM Computing Surveys'90 pp.37-85.
[2] 第 51 回情全大, 2D-2.
[3] IEICE Trans., Vol.E75-D, No.4, pp.352-362.