

半構造データのスキーマ情報を用いた動的音声対話 システム開発

洪 仁基† 遠山 元道††

†慶應義塾大学大学院 理工学研究科 開放環境科学専攻 〒223-8522 神奈川県横浜市港北区日吉 3-14-1

††慶應義塾大学 理工学部 情報工学科 〒223-8522 神奈川県横浜市港北区日吉 3-14-1

E-mail: † hong@db.ics.keio.ac.jp, †† toyama@ics.keio.ac.jp

あらまし 近年ユビキタスコンピューティングに関する興味が高まっている。またユビキタスコンピューティングのインタフェースの1つのパラダイムである音声を用いたアプリケーションに関する研究が多く行われている。本研究では従来の音声対話システムの開発にかかる負担を減らすためデータベースを基にした音声対話システムの開発を提案する。方法としてはXMLなどの半構造データに対する構造索引を生成し、そのスキーマ情報を用いて従来の音声対話システムとは異なる動的対話生成による新たな音声対話システムを提案する。このシステムにより従来の音声対話システムの限定されたデータベースとの連動、制限されたユーザの発話などの問題点を改善できると思われる。

キーワード 半構造データ, 構造索引, 音声対話システム

The dynamic spoken dialogue system development that used schema information of semi-structured data

Inki Hong† Motomichi Toyama ††

†School of Science for OPEN and Environmental Systems, Faculty of Science and Technology, Keio
University Hiyoshi 3-14-1, Kouhoku-ku, Yokohama-shi, Kanagawa, 223-8522 Japan

††Department of Information and Computer Science, Faculty of Science and Technology, Keio
University Hiyoshi 3-14-1, Kouhoku-ku, Yokohama-shi, Kanagawa, 223-8522 Japan

E-mail: †hong@db.ics.keio.ac.jp, ††toyama@ics.keio.ac.jp

As for recent the interest for the ubiquitous computing comes to be high, and the application that uses the speech recognition is the one of the paradigm for the ubiquitous computing. In order to decrease the burden which falls on the development of the spoken dialogue system, this paper purpose the way that makes spoken dialogue system which is based on database. To make spoken dialogue system based on database, for the first, we making structure index for the semi-structured data. And make dynamic conversation that using the schema information of the structre index. Unlike previous systems that restricted on the database for the developers and on the conversation for the users, the proposed system will be improved for the problematical point.

Keyword Semi-structured data, Structure index, Spoken dialogue system

1 はじめに

近年近年ユビキタスコンピューティングに関する研究が多く行われている。音声を利用したシステムもユビキタスの新たなパラダイムの1つである。これまでの研究、開発された多くの音声対話システムはRDBに対する検索として定型化されたものであると考えられる。これに対し、より汎用的な情報検索のための研究が現在も続けられている。本研究では音声認識システムの開発における開発者の負担を減らし、近年多く使われているXMLなどの半構造データを利用可能にすることを目的とする。従来の音声対話システムでは開発者が設計した対話の流れ、すなわち開発者によって決められたオートマトンにより対話が行われ、RDBを用いた簡単な処理結果を対話に利用するといったシステムが一般的であった。しかし、このようなシステムは設計者への負担、RDBに限定されたデータベースとの連動などの問題点がある。またユーザの発話に制限があることも自由な発話を通じた音声対話システムを作る上で問題点である。本研究では上に述べた問題点を解決するためXMLなどの半構造データを用いて動的に対話文を生成、制御する方法を提案する。方法としてはXMLデータを用いて、それに対する構造索引を生成する。そして、ユーザから与えられた発話からクエリ語を抽出し、構造索引のスキーマ情報とマッチングを行う。この処理によって該当するパス情報を用いて動的にユーザに対する対話文を生成する。このような処理を通じて従来できなかった半構造データの扱いができるようになり、開発者にとっても全体的な音声対話システムの開発における負担を減らせる。

2 音声認識アプリケーション

2.1 音声利用の特徴

音声を利用したウェブアプリケーションの特徴はまずPCを持ってない場合や一時的にPCを利用できない場合に電話などを利用し、アプリケーションの使用ができる。また携帯電話やPDAなどの小型化の趨勢を考えると音声の利用で、革新的な機器の小型化ができると思われる。例えば携帯のボタンがなくてもウェブ検索や電話使用が可能になれば高性能マイクやイヤホンで構成された機器だけで携帯電話の役割を満たすことができるようになる。最後に視覚障害者や文字を知らない人など既存のPCを使用できなかった人々も音声だけでウェブの検索などができるようになる。

2.2 音声対話システム

音声を利用したウェブアプリケーションは既存のシステムとは色々な面で異なる特徴を持っている。まず従来のウェブのように空間的な配置を利用できないため時間的な配置を利用する。一般的にウェブで使うための文書は本の5割程度にサイズを減らし使用するが、音声を利用する場合にはさらにサイズを減らす必要がある。

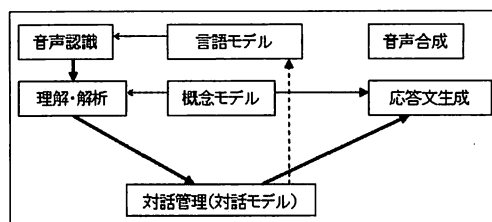


図 1: 音声対話システムの構成

古典的方法論としては限定されたタスクドメイン(例えば電車の切符予約など)を仮定し、これに対して音声認識を利用する有限状態オートマトン(FSA)の形式文法や意味理解に用いられる規則、文法を設計者が記述するのが一般的な方法である。実用化された音声対話システム(カーナビなど)のほとんどはユーザの発話に制限をおける場合が一般的である。これは音声対話システムが空間的な配置ではなく時間的な配置を利用し、ユーザとの対話をリードする必要があるからである。従ってシステムのユーザビリティを高めるためにはユーザの発話に対し、システムが適切に応答するようにする方法論が要求されている。本研究ではユーザの発話に制限を設けず、自由な発話による対話ができるようにするためにXMLなどの半構造データのスキーマ情報を用いて動的にユーザの必要に応じた対話文を生成する音声対話システムの開発を目的としている。

3 構造索引

3.1 関連研究

本研究では半構造データの音声対話システムへの適用のため構造索引のスキーマ情報を利用する。本章ではこれまでの構造索引に関する重要な研究を紹介し、筆者が提案した簡略で正確な構造索引であるCO-indexに関して説明する。

3.1.1 データモデル

XMLなどの半構造データは入れ子構造のオブジェクトモデルで記述することができる。Labeled graph $G=(V_G, E_G, root, \Sigma_G, label, Oid, value)$ は図2のように表す。 E_G の各edgeはobject-subobject, object-valueの関係を表す。 V_G の中でSimpleノードはoutgoing edgeを持たず、そしてvalue functionを通じて値が与えられる。各ノードはlabel functionによりラベルが付けられ、これらはOid functionによって区別される。このデータモデルはsub-objectを持っているcomplex objectと値を持っているatomic objectで構成されている。

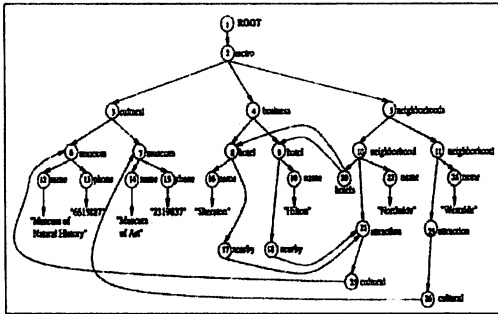


図 2: サンプルデータグラフ

3.2 CO-Index

今まで提案されてきた構造索引 [1], [2], [3] は様々な問題点がある。まず問題点となっているのは索引のサイズである。索引のサイズを減らすとクエリに対する答の正確率は低くなる。つまり、索引のサイズを減らすとパスに対しての正しい評価ができなくなるということである。本研究は上記のような理由、すなわち、索引のサイズを減らすとクエリに対する答の正確率が低くなるといった問題点を改善するためのCO-Indexを用いる。現在まで提案されてきた構造索引の中ではA(0)-index[3]が最小のサイズである。しかし、パスの正しい評価のためには1-index[1]が一番良いと考えられる。従って、本研究では1-indexを元にし、索引のサイズをA(0)-indexと同等まで減らすため、新たにCO-indexを生成し、対話に用いる。

3.2.1 CO-Index の概念

構造データは次のような object で構成されている。

- Atomic object : 値を含んでいる object

- Complex object : subobject を含んでいる object

構造要約の目的であるデータの構造をユーザに知らせるといふ観点から見ると、そのデータ構造だけでも良いと考えられる。従って、索引をComplex object だけで生成し、その索引にある条件を与える時にAtomic object を用いる方法である。XMLデータの場合、タグの中で実際の値 (integer, real, string, img など) を含んでいるタグに対して、その親のタグはAtomic object に対するポインタを持つようにし、実際の索引にはAtomic object が現れないようにする方法である。特にAuction、個人プロフィール、教職員情報などのデータに対し有効であると思われる。このCO-indexとA(0)-indexをデータグラフから生成し、比較すると次の図3、4のようになる。ここで注目すべきところは、CO-indexは1-indexを元にしたため正しいパスの評価ができるのにもかかわらず、そのサイズはA(0)-indexとあまり変わらないという点である。

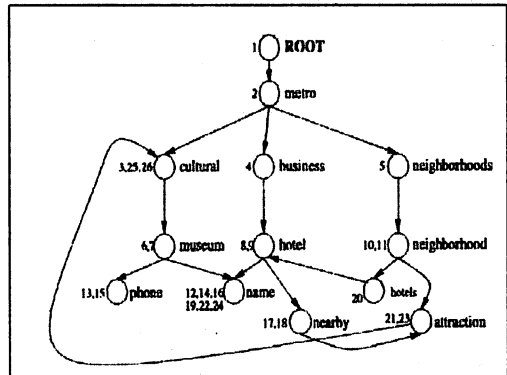


図 3: A(0)-index

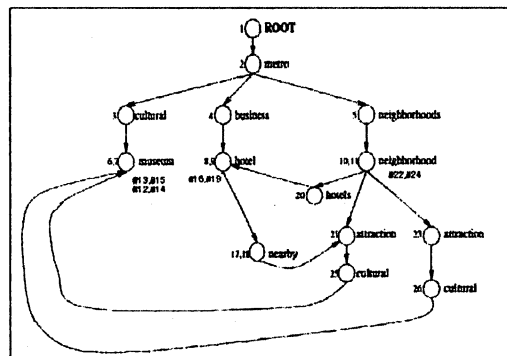


図 4: CO-index

3.2.2 アルゴリズム

CO-index の計算アルゴリズムは bisimulation 計算に対する標準アルゴリズムの変形である。まず compute bisimulation の重要な概念を説明する。グラフノードのある集合に対する stability は、あるノード集合 A に対して Succ(A) と表す。これは次の関係である集合を意味する

{ v | there is a node u in A with an edge from u to v }

定義: 与えられたデータグラフの 2 つの集合 A, B に対して次の条件を満たすと A は B に対して Stable という

- A は Succ(B) の subset である
- A と Succ(B) は disjoint である。

もし 2 つのノード (or node set) A, B がある場合、B に対して stable である A を作る。この場合 $A \cap \text{Succ}(B)$ と $A - \text{Succ}(B)$ によって分割する方法を用いる。まず bisimulation の copy を生成し、この copy から、bisimulation の同等クラスに対して stable である限り同等クラスを分割する。この接近には $O(nm)$ の時間と $O(m)$ の空間がかかる。ここで m はグラフ G におけるエッジの数である。bisimulation を計算すると compute CO-index の手順によって Complex object index が生成される。

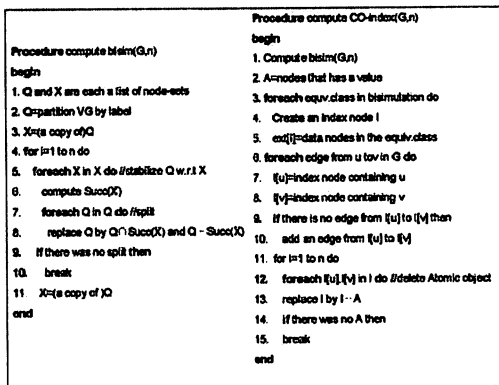


図 5: CO-index の計算

3.2.3 データ索引ファイル

CO-index はソースデータに含まれている Atomic object を索引に表示しないという面から索引には含まれないがその Atomic object の管理はしなくてはならない。そのためデータ索引ファイルを生成する。ソースデータから生成された CO-index は

データの構造をユーザに知らせるのには十分と考えられる。しかし、実際クエリを実行するためには Atomic object の情報を管理する必要がある。データ索引ファイルはデータファイルに格納されているタグ名、タグ属性を要素名としてリスト化して、各要素名がデータファイルに格納されている位置と関連付けられている索引ファイルである。構造索引からそのタグの位置が value object 含まれてない時は object 位置リストからソースデータへの位置が分かるようになり、もし value object を含んでいる場合は object 位置リストと value object リストを union することにより、ソースデータの検索が可能になる。

本研究では次の 4 つのデータ索引ファイルを生成する。

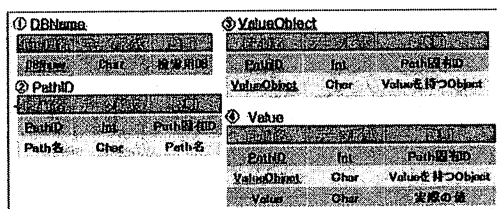


図 6: 4 つのデータ索引ファイル

4 動的音声対話システム

4.1 システム概要

本章では半構造データのパス情報を利用し、動的な対話を生成する方法について述べる。本システムはユーザの発話に制限を設けず、自由な発話による対話を可能にすることを目的とする。従って移動式携帯端末を対象にし、音声を利用したウェブ検索の際に XML などの半構造データに対する構造索引を生成し、それに従うスキーマ情報を利用、動的に対話を生成する。ユーザにおけるより適切な対話制御、応答生成のためユーザモデルを導入する。そして、自然語を用いたクエリの効率的な処理を考慮し、構造索引により生成されたスキーマ情報を利用、CO-Index のスキーマをユーザに提供したり、与えられたクエリに対して生成されたクエリ語とスキーマとのマッチングを行い必要なパス情報をフィルタリングする。このパス情報を用いてユーザのデータ構造に対する知識度による対話制御、すなわちユーザ親和的な対話処理を行う。

4.2 ユーザモデル

4.2.1 ユーザモデルの分類と対話戦略

ユーザによって、利用するシステムのデータベースに対する知識度は異なる。従ってシステム使用時ユーザのデータベースに対する知識度により対話の流れを決める必要がある。本システムではデータベースに対する知識がまったくないユーザとある程度データ構造に対する知識があるユーザに分け、対話モードを設計する。また構造索引に対するパスの重要度、与えられたクエリの分析による対話制御などユーザの特性に合わせた Dynamic な対話生成を目的とする。

```
System: Kにデータから検索を行います。次の項目から必要な項目を選んで下さい。理工学部、経済学部、医学部...
User: 理工学部
System: 理工学部で宜しいですか
User: はい
System: 理工学部にはJ,SD,K...があります。必要な項目を選んで下さい。
User: J
System: Jで宜しいですか
.....
```

図 7: 一般モードの例

```
System: Kにデータから検索を行います。検索語を話して下さい
User: 理工学部の中澤教授の情報が欲しい
System: 理工学部、中澤、教授で宜しいですか
User: はい
System: 理工学部、中澤、教授で検索した結果2つのパスがあります。次の項目から該当する項目を選んで下さい。J,SD,K...
User: J
System: 理工学部のJの教授の中澤で宜しいですか
User: はい
System: 該当するパスには次の項目があります。所属地区、学歴、取得学位
.....
```

図 8: 検索モード例

- 検索モード : データの構造をある程度知っているユーザ用でクエリ文の解析から対話を制御
- 一般モード : データ構造をまったく知らないユーザ用で重要度が高いパス順に対話を制御

4.3 Matching Function

本章ではユーザによって与えられたクエリ文から抽出したクエリ語と構造索引のスキーマ情報との比較し、もっとも適切なパスを取り出すための Matching Function について述べる。説明のため慶応義塾大学の教職員情報データベースである K-ris を用いる。簡略なデータモデルは図 9 に示す。

4.3.1 Signature の利用

検索モードの場合、与えられたクエリ語によって得られるパスがいくつかある時そのパスに含ま

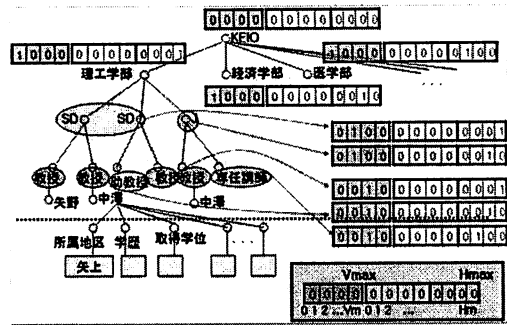


図 9: K-ris のデータサンプル

れているノードの中でクエリ語でないものに対し Signature を用いて正確なパスであるか検討する。図 9 のように各ノードごとに Signature を作る。Tree 構造の深さで Vmax が決まり、幅により Hmax がきまる。Grouping されている各ノードは独立の Signature を持つ。例えば図 10 のようにユーザからのクエリが与えられたらまずクエリ語である理工学部、中澤、教授を抽出し、スキーマとのマッチングを行う。各クエリ語の Signature を or 演算で計算すると確認すべき所が見つかり、その部分はユーザにもう 1 回確認を行い正確なパスを決める。

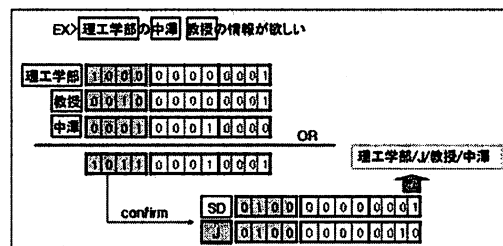


図 10: Signature 利用の例

4.3.2 重みの利用

Signature を用いると正確なパスが得られるが、そのためのユーザとの対話数が増える問題点、そしてそのための体系的なコストが増えてしまう問題点がある。これに対し重みを用いると対象にないパスをシステム側で推薦してくれるのである。重みはグルーピングされたパスの数を考慮し、もっとも多くのパスがグルーピングされたパスを重みが高いと判断し、ユーザに先に提供する。重みの例は図 11 に示す。

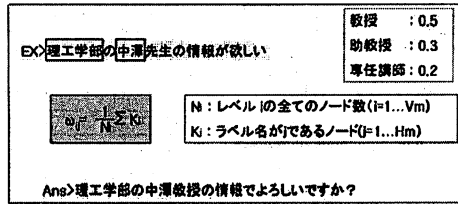


図 11: 重みの利用例

4.4 システム設計

対話ターン数を減らすことは本研究の重要なポイントの1つである。データに対するパス数を減らすことにより対話ターン数と処理速度の向上が期待できる。そして本研究では全体のシステムの中で音声認識、クエリ語抽出を除いたデータ処理、対話文生成にポイントをあてる。その理由としては現在色々な市販の音声認識エンジンが商品化されていて、また VoiceXML 処理のために多くのゲートウェイサービスを行っている所があるからである。

一般的な処理の流れを説明する。まずアプリケーションのメタ情報や全体で使う特性、変数などを定義しサービスを開始する。ユーザは接続後検索したいDBをDBNameから選択する。システムでは選択されたDBに対しCO-indexを生成する。次にユーザのクエリ語に対しスキーマとのマッチングを通じてもっとも適切なパスをユーザに返す。あとそのパスに該当するValue Objectをenumerate機能を用いて選択するようにする。最後に選択されたValue Objectの値をユーザに返し終了する。このような処理は次に述べる方法によって行われる。

XMLに対するCO-indexはまずJavaのSAXを用いてCO-Bisimilarityの概念に基づきグラフを要約する。要約した結果は3つのデータ索引ファイルとして貯蔵され、ユーザの検索語に応じたパス検索などに用いられる。そして、基本的な音声対話システムとしての処理のため9つのVoiceXMLファイルを作成し、VoiceXMLのケーとウェイサービスの1つであるBeVocalを通じて実験を行った。DBとの検索結果は動的なVoiceXMLファイルを生成し処理する。システムにおけるARS(Automatic Speech Recognition)やTTS(Text To Speech)出力はBeVocalのケーとウェイサービスにより処理される。

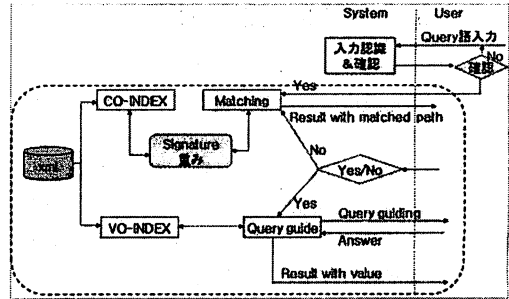


図 12: システム図

表 1: XML Data

A	Auction 0.1	11.597KB	206130 nodes
B	Auction 0.01	1.155KB	21048 nodes
C	Auction 0.001	116KB	2086 nodes
D	Tokyomap 13101	2.775KB	93801 nodes
E	K-ris sample	34KB	1043 nodes

5 評価実験

今回の実験ではまずXMLデータに対する構造索引の生成し、1-indexとCO-indexのパス数を比較する。実験に用いられるXMLデータは表1のようであり、JavaのSAXを利用し各データに対するパス数を比較する。

表1のデータに対し説明すると、A,B,CはBenchmarkのAuction XML dataを0.1, 0.01, 0.001のスケールで分けたものであり、Dは東京地図情報データの13101番のデータである。そしてEのK-ris sampleデータは筆者が<http://k-ris.keio.ac.jp>を参照して簡略に作ったものである。

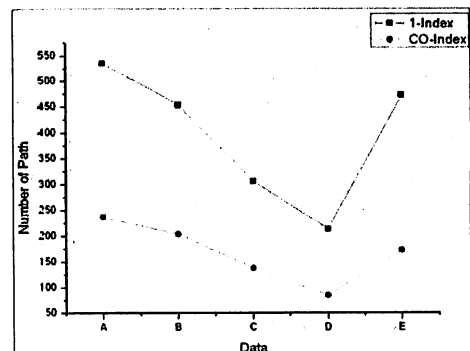


図 13: 各データに対するパス数

表 2: ターン数

	クエリ語 2	クエリ語 3
Signature	5 回	4 回
重み	4 回	3 回

K-ris データから教職員のプロフィールを検索する場合を例に上げると従来の方法を用いて音声認識エラー、無応答エラーなどによる処理を除き、すべて順調に成功した場合システムとユーザの間に 6 回の対話ターン数が発生する。これに対し、Signature と重みを利用した場合のターン数は表 2 のようになる。

6 むすび

今回の研究では音声対話システムの開発における設計者の負担を減らすため構造索引のスキーマ情報を利用した、動的な対話制御の提案した。CO-index を用いることにより簡略で正確なスキーマを得ることができ、これを利用して動的にユーザにあったパスを用いて対話文を生成することで開発者への負担、ユーザへの便利さが提供できるとしている。しかし、一般に使われている XML データをそのまま利用するには対話自体の不自然さが問題となり、自然な対話文の生成を今後の課題として考えている。そしてスキーマ情報を基に音声対話システムで認識率を高めるために使用する Grammar の自動生成についても今後検討していきたいと思っている。

7 参考文献

- [1] Tova Milo, Dan Susiu. Index Structures for Path Expressions. 7th International Conference on Database Theory(ICDT), pp.277-295, January 10-12, 1999
- [2] Raghav Kaushik, Philip Bohannon, Jeffrey F Naughton and Henry F Korth. Covering indexes for branching path queries. In Proceedings of SIGMOD, pp.133-144, 2002
- [3] Raghav Kaushik, Pradeep Shenoy, Philip Bohannon, Ehud Gudes. Exploiting local similarity for efficient indexing of paths in graph structured data. In Proceedings of ICDE, pp.129-140, 2002
- [4] Jason McHugh, Serge Abiteboul, Roy Gold-

man, Dallas Quass, and Jennifer Windom. Lore:A Database Management System for Semistructured Data. In Proceedings of SIGMOD, pp.54-66, 1997

- [5] Roy Goldman, Jennifer Widom. DataGuide: Enabling Query Formulation and Optimization in Semistructured Database. Proceedings of the 23rd International Conference on Very Large Data Bases(VLDB), pp.436-445, August 25-29, 1997
- [6] Jason McHugh, Jennifer Windom, Serge Abiteboul, Qingshan Luo, Anand Rajaraman. Indexing Semistructured Data. Technical Report, Stanford University, February, 1998
- [7] Raghav Kaushik, Philip Bohannon, Jeffrey F Naughton, Pradeep Shenoy. Updates for Structure Indexes. Proceeding of the 28th VLDB Conference, Hong Kong, China, pp.239-250, 2002
- [8] Masakazu Hattori, Katuhiko Nomura, Takuya Kanawa, Naomichi Sueda. Retrieval optimization technique using a query graph in XML database. Transactions of Information Processing Society of Japan(IPSJ), Vol.43, pp.1-15, 2002
- [9] Kazunori Komatani, Shinichi Ueno, Tatsuya Kawahara Hiroshi G. Okuno. Flexible Guidance Generation using User Model in Spoken Dialogue Systems. Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, pp.256-263, July 2003
- [10] Bob Carpenter, Sasha Caskey, Krishna Dayanidhi, Caroline Drouin and Roberto Pieraccini. A Portable, Server-Side Dialog Framework for VoiceXML. Proceedings of ICASSP, pp.2705-2708, 2002
- [11] Anoop K.Sinha, Scott R.Klemmer, Jack Chen, Kames A.Landay, Cindy Chen. Suede : A Wizard of Oz Prototyping tool for speech user interface. Proceedings of the 13th annual ACM symposium on User interface software and technology, pp.1-10, Nov 2000