

HTMLのフォーム入力のための文法の自動生成とSALTによる実装

住吉 貴志 河原 達也 奥乃 博

京都大学 情報学研究科

〒 606-8501 京都市 左京区 吉田本町

e-mail: sumiyosi@kuis.kyoto-u.ac.jp

あらまし WWW のページの多くでユーザが入力するフォームが用いられているが、小型端末上での入力手段としては音声があると考えられる。そのような音声入力のためのタグとして SALT が提案されているが、認識に用いる文法を人手で記述・指定する必要がある。そこで本研究では、任意のフォームページに対して文法を生成し、あいまいな発話に対して応答を生成する機能を付与するシステムを SALT と JavaScript を用いて構築した。また、フォーム項目の一つであるテキストフィールドの語彙を自動的に得るために、テキストフィールドの周辺文字列からカテゴリを自動判別する方法についても検討した。

Automatic Generation of Speech Grammars for HTML Forms and Speech Interface Implementation with SALT

Takashi SUMIYOSHI Tatsuya KAWAHARA Hiroshi G. OKUNO

School of Informatics, Kyoto University, Kyoto 606-8501, Japan

e-mail: sumiyosi@kuis.kyoto-u.ac.jp

Abstract Forms are used in user interfaces in many WWW pages, and speech is a useful input method for portable devices. SALT has been proposed for speech-input tag definition, however grammars for speech recognition still need to be specified and they are generally hand-crafted. Therefore, we design and develop a system using SALT and JavaScript that automatically generates grammars for any form, and prompts the user in cases of ambiguous input. Moreover, we investigate a method to classify the text fields into categories using their surrounding text to automatically generate the appropriate vocabulary.

1 序論

近年、携帯電話、PDA などの小型の情報端末が普及し、無線通信技術によってどこにいてもインターネット上の情報にアクセスできる環境が整いつつある。WWW のドキュメントを記述するのに広く用いられている HTML には、静的な情報を記述するだけでなく、ユーザからの入力を受け付けサーバに送信できるフォーム機能も組み込まれている。画面などのディスプレイ装置が利用できる端末ではペンなどのポインティングデバイスによる入力方式が採用されている。しかしながら、ペンによる文章の入力は負担であるため、音声を用いた入力の実現が期待される。

このような流れの中で、HTML のフォーム項目に対して音声による入力を可能にする HTML の拡張規格として、Speech Application Language Tags (SALT) が提案されている [1][2]。しかしながら、フォームごとに音声認識に使用する文法を用意する必要があり、それには専門的な知識を要するため、SALT の付与にはコストがかかる。特に既存の HTML フォームの膨大な量を考慮すると、適切な文法を自動的に用意できることが望ましい。

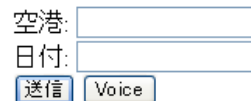
本研究では、HTML のフォーム入力に関する音声認識について検討し、文法の自動生成と、それを用いた SALT サーバを構築する。特に、HTML に記述されている情報を用いて、文法・語彙の自動生成に必要なカテゴリ判別についても検討する。

2 音声入力ブラウザに関する技術

2.1 先行研究

HTML のフォーム項目の音声入力に関しては、音声認識 Web ブラウザの研究と関係が深い [3][4]。これらはページ切り換えなどのブラウザ制御コマンドとページのリンク先への移動を音声で行なうシステムであり、リンクとして定義されている文字列を元に文法を作成するという手法が採られている。

フォーム入力を音声で行うための研究も行われている [5]。[6] では、あらかじめページ毎に音声認識用の情報を用意することでフォーム入力を可能にしているため、自由度は高いものの手作業を要する。[7] は、フォーム項目の一つであるセレクトボックスに対して、段階的な対話によってフォームを埋めて



```
<form>
  空港: <input name="Airport" type="text"><br>
  日付: <input name="Date" type="text"><br>
  <input name="Submit" type="submit" value="送信">
  <input type="button" value="Voice" onclick="Reco.Start()">
  <SALT:listen id="Reco" xml:lang="ja" onreco="Reco.Stop()">
  <SALT:grammar id="Gram">
    <rule>
      <list>
        <list proptype="AP">
          <phrase valstr="伊丹">伊丹</phrase>
          <phrase valstr="羽田">羽田</phrase>
          : (略)
        </list>
      </list>
    </rule>
  </SALT:grammar>
  <SALT:bind targetelement="Airport" value="//AP" />
  <SALT:bind targetelement="Date" value="//DT" />
  </SALT:listen>
</form>
```

図 1: SALT を用いた HTML フォームの例

いくことでタスクを遂行するシステムである。[8] では、テキストフィールドに対して N-gram 文法を用いて入力する手法が採用されている。

2.2 SALT

SALT[1] は、HTML に音声入力や音声プロンプトの情報を付与するタグを拡張するための規格である。例を図 1 に示す。

SALT は HTML の `<form>` タグ内に記述する。`<listen>` タグが一つの認識単位 (オブジェクト) を表し、その中に `<grammar>` タグで認識に用いる文法を、`<bind>` タグで認識結果とフォーム項目との対応関係を指定する。この例では、ユーザが「Voice」というボタンをクリック (タップ) すると、Reco という ID をもつ `<listen>` オブジェクトをアクティブにし、システムはユーザの発話を受理できる状態になる。ユーザが「伊丹」と発話すると、「AP= 伊丹」という情報に相当する XML (SML) が発行され、`<bind>` タグで指定された内容に従って、Airport という名前のテキストフィールドに「伊丹」という文字列が入力される。

このように、SALT に対応した Web ブラウザを用いることで、SALT が記述されているフォームに

音声で入力することができる。

VoiceXML とは異なり、SALT には対話制御の枠組みはないが、JavaScript などのスクリプト言語によって SALT のオブジェクトを制御することが可能である。これにより複雑な対話制御や、電話など音声のみの端末でのブラウジングにも対応できる。

3 フォーム入力のための文法生成

3.1 フォームの分類

本研究では音声入力が有効であると考えられる検索条件入力、個人情報入力などを目的とするフォームを対象とする。このようなフォームは、主にテキストフィールド、セレクトボックス、ボタンという 3 種類のフォーム項目から構成されている。

テキストフィールド: キーボード、音声により任意の文字列を入力

セレクトボックス: メニュー形式で、複数の子要素から選択

ボタン: 複数項目からの選択や、検索開始などのアクションを起こす

このうちセレクトボックスとボタンは、項目に対して文字列が定義されている場合が多いので、その文字列をそのまま音声認識用文法に利用することで対処できるが、テキストフィールドには語彙が記述されていないため、文法を単純に生成できない。当該部分にディクテーション用の N-gram を用いることも考えられる [8] が、認識性能は十分ではなく、多くの計算量を消費するという問題もある。

そこで本研究では、カテゴリという概念を用いる。テキストフィールドのようなフォーム項目は都道府県、名前、駅名などといった何らかのカテゴリに分類可能なものが多いと考えられ、カテゴリが決まればおのずと語彙も推定することが可能である。カテゴリの種類と語彙の関係をデータベースとして持つておくことで、カテゴリが正しく与えられればそこから語彙を得られる。

カテゴリの自動判別については 5 章で検討する。



図 2: 情報検索フォームの例

3.2 文法の生成

実際に各項目に文法を用意する必要があるが、ユーザの発話様式は多様であり、それらのすべてに対応することは困難であるため、何らかの仮定をする必要がある。ここでは、ユーザは次のようなフォーム項目の種類によって定められる内容句を発話すると仮定する。

テキストフィールド: カテゴリに対応する文法

セレクトボックス: 各選択肢の文字列

ボタン: ボタンに割り当てられている文字列

内容句には、[4] と同様に、句を形態素解析して得られる部分文字列を受理する文法を用いた。すなわち、任意の助詞以外の形態素から始まり、それ以降の任意の助詞以外の形態素、あるいは最後の形態素で終わる文字列を受理可能とした。

さらに、項目の周辺の文字列で構成される項目指定句を導入し、「(項目指定句) が (内容句)」という発話形式で項目の明示的な指定を可能にする。項目指定句は内容句と同様に、形態素単位の部分文字列を受理できるようにした。

このような方法により、例えば図 2 の情報検索フォームに対して図 3 に示すような文法が生成される。トップレベル文法の前半は、省略可能な項目指定句部分であり、後半は内容句である。句の形態素列の一部を受理できる文法をサブルールを用いて定義している。テキストフィールドの場合、内容句部分はカテゴリに対応する外部ルールの参照となる。この例では「出発駅が東京」といった文を受理し、駅名についてはCG_station.xml というファイルを参照している。

3.3 発話からの項目の決定と絞り込み

ユーザの発話から項目指定句と内容句を認識した結果から、ユーザがどの項目を発話したかを求める

```

<GRAMMAR langid="411">
<RULE name="top" toplevel="active">
<0>
<!-- 項目指定句 -->
<L>
<RULEREF name="SR_1_0" />
<RULEREF name="SR_1_1" />
<RULEREF name="SR_2_0" />

: (省略)

</L>
<P>が</P>
</0>
<!-- 内容句 -->
<L>
<!-- カテゴリ「駅名」 -->
<RULEREF propname="CG_station" name="CG_station"
url=" ../category/CG_station.xml" />
<!-- カテゴリ「駅名」 -->
<RULEREF propname="CG_station" name="CG_station"
url=" ../category/CG_station.xml" />
<!-- ボタン「検索」 -->
<RULEREF name="SR_22_0" />
</L>
</RULE>
<!-- 形態素の部分受理用ルール -->
<RULE name="SR_1_0">
<P propname="WORD_0" propid="0" val="1"/>出発/シユツ
バツ;</P>

<0><RULEREF name="SR_1_1" /></0>
</RULE>
<RULE name="SR_1_1">
<P propname="WORD_1" propid="1" val="1"/>駅/エキ;</P>
</RULE>

: (省略)
</GRAMMAR>

```

図 3: 文法生成例

ことができる。しかし、例えば図 5 のようにカテゴリが駅名と判断されたテキストフィールドが 2 つあり、修飾語がそれぞれ出発駅と到着駅であるような場合、「駅が大阪」という発話ではどちらのテキストフィールドに対する発話かが同定できない。このような場合は、絞り込んだ候補から 1 つを選択させるダイアログをユーザに提示し、指定してもらう。

4 文法・SALT 自動生成システム

前章で述べた手法を用いて、任意の HTML フォームに対して文法を自動生成し、SALT を付与するシステムを設計・実装した。全体構成を図 4 に示す。

ユーザはアクセスしたい URL を SALT サーバに送信する。SALT サーバはその URL にアクセスして HTML を取得し、カテゴリデータベースを用いてカテゴリ判別を行い、適切な語彙と文法を生成し、HTML に対して SALT と制御用の JavaScript を付与する。これらの結果は直接ユーザ端末に送られる。

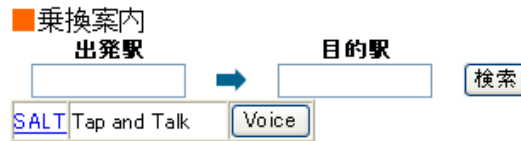


図 5: SALT インターフェース

ただしカテゴリ判別は後述のように 100%正しく行えるわけではないので、図の破線のように、あらかじめ文法・SALT 生成モジュールを用いて得られる結果に、トップレベル文法のカテゴリ文法の参照部分を人手で記入し、WWW サーバに再配置することもできる。この場合、ユーザは SALT サーバを介することなくページに直接アクセスできる。

SALT 生成モジュールによって、フォームの近くに SALT インターフェースが付与される (図 5)。ユーザはこれを用いて音声入力開始の合図 (タップ) を送ることができる。SALT サーバによって付与された SALT の記述に従い、音声認識モジュールが起動され、指定された文法が SALT サーバから読み込まれるが、このときカテゴリ文法への参照が含まれていると、その文法も読み込んで動的に結合される。

ユーザの発話終了に伴い音声認識が成功すると、SALT サーバによって付与された制御用の JavaScript が実行され、認識結果に基づきテキストフィールドへの入力やボタンのイベント発火などの処理を行う。あいまいな発話に対しては候補を提示し、選択をユーザに求める。

本システムの実装においては、ユーザ端末の OS に Windows XP を、SALT 対応ブラウザとして、Internet Explorer に Internet Explorer Speech Add-in¹ を組み込んだものを、音声認識モジュールには Julius for SAPI[10] を用いた。Internet Explorer Speech Add-in と Julius for SAPI は、Speech API(SAPI) を用いて文法や認識結果などの情報をやりとりする。

¹ Microsoft .NET Speech SDK Version 1.0 Beta 2 SDK [9] に付属 (ver. 1.0 Beta 2)

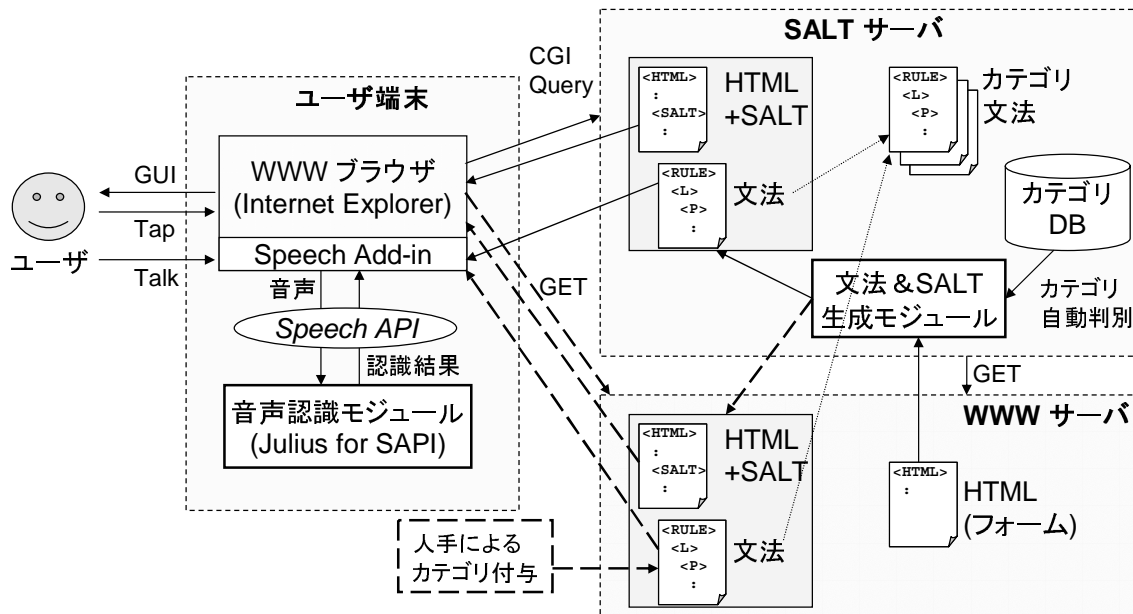


図 4: SALT 自動生成システムの構成

5 カテゴリの自動判別

5.1 周辺文字列を用いたカテゴリの自動判別

このシステムでは、テキストフィールドに対してカテゴリを付与する必要があるが、このカテゴリが自動的に求められるのが望ましい。

カテゴリ判別には、項目の周辺にある文字列を利用することが考えられるが、HTML によって書かれた Web ページでは、情報が タグなどによって階層的な構造を与えられていたり、<table> タグなどがレイアウトのために用いられていたりするため、単に前後の文字列を参照するのみでは不十分である。そこで、階層構造やレイアウトの情報も考慮して、階層的上位関係にあるもの、2次元平面的に上下などに近接するものについて関連が高いと判断する。

カテゴリは、国名、都道府県などといった一般的なカテゴリと、氏名、住所などといった個人的なカテゴリが存在する。個人的なカテゴリに対応する文法は、あらかじめ個人ごとに作成する必要がある。これらの一般的なカテゴリと個人的なカテゴリを統合したものをカテゴリデータベースと呼ぶ。カテゴリデータベースのレコードは次のようになっている。

(カテゴリ ID , キーワード集合 , 文法)

カテゴリ	キーワード
都道府県	都道府県
職業	職業
駅名	駅
国名	国
年	年
月	月
日	日
年齢	年齢, 歳, 才
郵便番号	郵便, 〒
メールアドレス	メール, アドレス, mail
名前	名前, 氏名, 名, name
住所	住所
電話番号	電話, TEL, FAX
検索語	検索, キーワード, yahoo, lycos, google, excite, infoseek, search

表 1: 判別に用いたカテゴリデータベース

キーワードはそのカテゴリを表現するのに用いられる代表的なものである。ここでは、カテゴリの種類とキーワードは、一般的に Web で使用されると考えられるものを選定した。表 1 にその一覧を示す。

周辺文字列に対して優先順位の高いものから順に形態素単位でキーワード集合と比較し、部分一致した場合にその項目のカテゴリであると判別する。なお、実際の利用を考えると、必ずしもカテゴリを一つに絞り込む必要はないため、カテゴリ候補を複数

ディレクトリ	B	C	E	S	合計
総数	29	38	23	24	114
正解数	27	35	21	20	103
誤答数	1	2	2	2	7
無答数	1	1	0	2	4
正解率 (%)	93.1	92.1	91.3	83.3	90.4

B: ビジネスと経済 C: コンピュータとインターネット
E: エンターテインメント S: 自然科学と技術

表 2: 実験結果

保持する。

5.2 判別実験

このカテゴリ判別手法を検証するために実験を行った。Web でのフォームの実際の使われ方を調べるため、また評価用データとして、ディレクトリ型ポータルサイトである www.yahoo.co.jp の 4 つのディレクトリに分類されている Web サイトのトップページを収集した。

フォームを含むページから分野毎にランダムに選択したものを対象に判別実験を行った。出現するテキストフィールドに本手法を適用して得られたカテゴリと、手動で正解を付与したカテゴリとを比較し、判別精度を求めた。実験結果を表 2 に示す。

5.3 考察

今回の実験では、判別精度は 90.4% であり、正解カテゴリはすべて第一候補に挙げられていた。この結果から、ある程度のカテゴリデータベースを用いることで、本手法により実用的なカテゴリ判別精度が得られることがわかった。

また、収集したサンプルのテキストフィールドの多くは「検索語」「メールアドレス」というカテゴリであった。今回実験のために収集したデータサンプルは、特に本研究で対象とする情報検索型のページ以外のものも多数含まれていたため、カテゴリの種類が少なく、カテゴリに大幅な偏りがみられた。今後、情報検索型のページを主な対象にした実験も行う予定である。

6 結論

周辺文字列からフォーム項目のカテゴリを判別し、HTML フォームに対して文法を生成する方法を提案した。これに基づいて、標準的な仕様である SALT と JavaScript を用いて既存の HTML へ音声インターフェースを付与できるシステムを構築した。

検索語などの一部のカテゴリに関しては、文法が必ずしも適切であるとは限らないため、部分的に N-gram を利用することで対処するという方法が考えられる。型番などの記号的な入力や、テキストエリアに対する長文の入力などは、本研究では有効な解決策を見いだすことができなかった。

今後、このフォーム入力機能を含めた音声認識 Web ブラウザとしての評価を行う予定である。

参考文献

- [1] SALT Forum: *SALT Forum* (2001). <http://www.saltforum.org/>.
- [2] Wang, K.: SALT: a Spoken Language Interface for Web-Based Multimodal Dialog Systems, *Proc. ICSLP* (2002).
- [3] 桂浦誠, 中村哲, 鹿野清宏: 音声キーワードによるネットサーフィンの実現, 情報処理学会研究報告, 98-SLP-20-12 (1998).
- [4] 甲斐充彦, 中野崇広, 中川聖一: 音声認識サーバ-SPOJUS-を利用した WWW ブラウザの音声操作システム, 情報処理学会研究報告, 98-SLP-20-14 (1998).
- [5] Issar, S.: A Speech Interface for Forms on WWW, *Proc. Eurospeech '97*, Rhodes, Greece, pp. 22-25 (1997).
- [6] 近藤和弘, チャールズヘンブル: 音声認識を用いた WWW ブラウザとその評価, 電子情報通信学会論文誌, Vol. J81-DII, No. 2, pp. 257-267 (1998).
- [7] 中野崇広, 甲斐充彦, 中川聖一: WWW 上のフォーム型情報検索サービスのための音声インタフェースの検討, 情報処理学会研究報告, 99-SLP-25-1 (1999).
- [8] 中野崇広, 甲斐充彦, 中川聖一: WWW 上のテキスト入力フォームのための任意文字列入力の音声インタフェース, 情報処理学会第 62 回全国大会 2001.3, 1L-7 (2001).
- [9] Microsoft: *Microsoft .NET Speech Technologies* (2002). <http://www.microsoft.com/speech/>.
- [10] 住吉貴志, 李晃伸, 河原達也: 音声認識エンジン Julius/Julian の API 実装, 情報処理学会研究報告, 2001-SLP-37-16 (2001). <http://julius.sourceforge.jp/>.