Web上の辞書を利用したメタ辞書の構築

南野　朋之† 奥村　学††
† 東京工業大学大学院 総合理工学研究科 †† 東京工業大学 精密工学研究所

1 はじめに

近年，Web上で利用できる様々な辞書が増加してきた．以前までは，個人ユーザが自分のWebページで，専門的な用語集など，限られたドメインの辞書を公開しているものが多かったが，最近では，市販されている辞書を引くことができるサービスなども充実してきており，その質，量共に，実用のレベルに達している．

Web上の辞書の良い点は，

- どこからでも利用できる
- 検索機能が利用できる
- 更新が早い
- 通常の辞書に載っていないような語に対しても，誰かが辞書を作っている場合がある

などの点である．

このようなネットワーク上の辞書を利用する上で問題となるのが，自分の知りたい語がどの辞書に登録されているかを知らなければ，調べることが出来ないという点である．特に，一般の辞書に登録されないような専門的な語や新しい語などは，登録される辞書も限られるため，このような問題が顕著に現れる．また，ネットワーク上で複数の辞書サービスが利用できる状況であれば，それら複数のサービスによってどのような記述の違いがあるかを知りたいというユーザも存在するだろう．

このような要求を満たすためには，ユーザは従来，個々の辞書サービスにアクセスし，検索したい語を探すといった作業を繰り返し行う必要があった．

本研究では，ユーザのこのような作業を軽減し，どの辞書に自分の検索したい語が存在するかを意識することなく，複数の辞書を横断的に検索することのできる辞書システム (メタ辞書) を構築する．

2 メタ辞書

2.1 メタ検索エンジンとの違い

メタ辞書と同様，Web上の複数のサービスを統合して利用できるシステムとして，これまで様々なメタ検索エンジンが開発されてきた [1][2].

この様なメタ検索エンジンとメタ辞書の最も異なる点は，メタ検索エンジンでは，個々の検索エンジンがどれも基本的にはWeb全体を対象にしているのに対し，メタ辞書システムでは，個々の辞書の検索する対象は，言語 (和英，英和など) や専門分野 (コンピュータ用語辞典など) など，まったく別な物であるという点である．

よって，個々の検索エンジンのカバー率が向上してきている現在では，ユーザにとって，メタ検索エンジンを利用することと，そのシステムが利用する検索エンジンを一つ利用することは，精度などの面を除けば，それほど変わらない．しかしながらメタ辞書は，個々の辞書の対象が全く異なっているため，ユーザがある一つの辞書を検索しても，その辞書に自分の検索した語が登録されていなければ，辞書引きすることが出来ない．

このような点から，どの辞書に自分の知りたい語が登録されているかを全く意識せず利用できるメタ辞書システムは非常に有用なシステムであると共に，現在非常に求められているシステムである．

2.2 Web上の辞書の特徴

Web上の辞書には，大きく分けて，以下の二つのタイプが存在する．

- CGIなどを利用した検索システムにより，辞書中に登録されている語を検索することができるタイプ
- HTML文書中に，直接辞書の記述がなされているタイプ

便宜上，前者のタイプの辞書を“検索型”，後者のタイプの辞書を“HTML型”と呼ぶ．どちらのタイプの辞書もWeb上に多数存在する．傾向として，検索型の辞書には大規模な辞書 (例えば，英和辞典)，HTML型の辞書には，そこにしか情報が無いような，専門的，もしくは特殊な辞書が多く存在する．

このような性質により，システムに求められるのは，このような二種類のタイプの辞書を網羅的にメタ検索できるシステムであると考えられる．

本研究では，これら二種類のタイプの辞書を，メタ検索の対象とする．

Meta-Dictionary - a system to integrate dictionaries on the WWW

† Tomoyuki NANNO (nanno@lr.pi.titech.ac.jp)
†† Manabu OKUMURA (oku@pi.titech.ac.jp)
Interdisciplinary Graduate School of Science and Engineering , Tokyo Institute of Technology(†)
Precision and Intelligence Laboratory, Tokyo Institute of Technology(††)

3　メタ辞書システム

3.1　ユーザインターフェース

図 1 は，本研究で実装したメタ辞書システムのスクリーンショットである．このシステムは，二つのフレームで構成されている．左側のフレームには，検索クエリを入力するテキストボックスと，検索する辞書のカテゴリを指定するプルダウンメニューが存在する．右側のフレームには，最初，システムの使用方法と登録されている辞書の一覧が表示される．

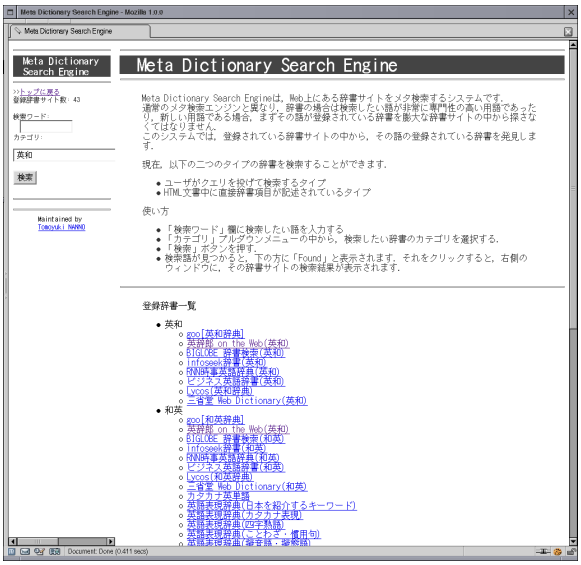

図 1: システムのスクリーンショット

図 2 は，クエリとして「apple」，辞書のカテゴリとして「英和」を指定して検索した場合の出力である．

図 2: 検索結果の例

ユーザが指定したカテゴリに含まれる辞書の一覧と共に，もし，その辞書の検索結果に「apple」が含まれる場合には「Found」，含まれなかった場合には「Not Found」が表示される．「Found」の部分にはリンクが張られていて，リンクをクリックすると，図 3 の様に，

図 3:　「Found」をクリックした場合

実際その辞書を検索した結果が右側のフレームに表示される．

3.2　システム構成

本システムのシステム図を図 4 に示す．

本システムでは，検索する辞書情報をデータベースとしてシステム外部に持っている．その際，各辞書がどのような語を含んでいるといったような情報は一切持たず，ユーザからの検索要求があった際に，各辞書ページの検索を行う．このデータベースを書き換えることで，対象とする辞書を追加する事ができる．

以下にデータベースと，処理を行うモジュールについて詳述する．

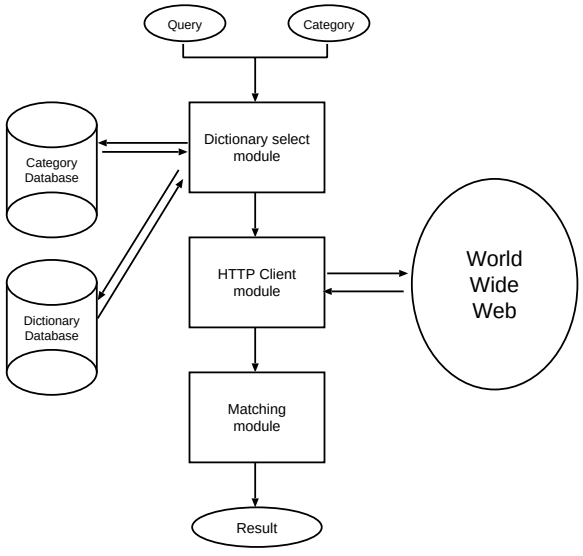

図 4: システム構成

3.2.1 Category Database

カテゴリデータベースは,「id, カテゴリ名」の形式で記述する．カテゴリは，ユーザがメタ検索する対象を絞り込む際に使用する．

```
0, 英和辞書
1, 和英辞書
2, 国語辞書
3, 新語辞典
4, 百科辞典
...
```

3.2.2 Dictionary Database

辞書データベースは，辞書のタイプによってデータベースの記述方法が異なる．

検索型の辞書の場合

以下の内容をデータベースに記述する．

- type：検索型の場合 “0” (HTML 型は “1”)
- ID：通し番号
- 名前：辞書の名前
- カテゴリ番号：カテゴリデータベースと対応する，その辞書の含まれるカテゴリ番号
- server 名：接続するサーバ名
- port 番号：接続するサーバのポート番号
- URL：辞書のトップページの URL
- 文字コード：クエリを URL エンコードする際の文字コード
- クエリ前：クエリより前にある URL
- クエリ後：クエリより後にある URL
- not found：not found の場合を判断する文字列

例として，goo[英和辞典] * の場合，

```
------
0
0
goo[英和辞典]
0
dictionary.goo.ne.jp
80
http://dictionary.goo.ne.jp/cgi-bin/ej-top.cgi
EUC-JP
/cgi-bin/dict_search.cgi?MT=
&sw=0
検索結果に該当するものが見当たりません。
------
```

となる．

HTML 型の辞書の場合

以下の内容をデータベースに記述する．

- type：HTML 型は “1”
- ID：通し番号
- 名前：辞書の名前
- カテゴリ番号：カテゴリデータベース中の番号
- server 名：接続するサーバ名
- port 番号：接続するポート番号
- URL：辞書のトップページの URL
- match pre：辞書のエントリがある部分の前に相当する正規表現
- match post：辞書のエントリがある部分の後に相当する正規表現
- 固定 URL：複数の辞書ページに共通する URL 部分
- 変化 URL：複数の辞書ページで異なる部分 (複数指定可能)

「match pre」と「match post」で囲まれる正規表現で抽出される部分によって HTML 文書中に含まれる辞書のエントリを抽出する．また，HTML 型辞書の特徴として，頭文字などによって，辞書が複数ページに分割されているケースも多い．そのような辞書に対しては，例えば辞書が “/test1/index1.html” と “/test1/index2.html” に分割されている場合,「固定 URL 部分」を例えば “/test1/” とし,「変化 URL 部分」を “index1.html,index2.html” とすることで，複数ページに分割された辞書を登録することが出来る．

例えば，中日現代用語辞典† の場合，

```
------
1
1013
中日現代用語辞典
9
www.qiuyue.com
80
http://www.qiuyue.com/
<DT>
$
/
shingopin.htm
------
```

となる．

3.2.3 Dictionary Select Module

辞書選択モジュールは，ユーザの入力に含まれる辞書カテゴリから，検索する辞書の集合を決定する．また，辞書にクエリを送信する際，どの文字コードで送信するかが辞書ごとに異なるため，辞書データベースを参照し，クエリの文字コードを変更し，URL エンコード‡する．現在システムが対応している文字コードは，以下の通り．

* goo[英 和 辞 典] http://dictionary.goo.ne.jp/cgi-bin/ej-top.cgi

† 中日現代用語辞典 http://www.qiuyue.com/

‡ URL 中に含まれる文字列のうち，英数字以外のものを“ %xx ”（xx は文字コードを 16 進数表記したもの）という書式に変換したものであり，主にリンク先の URL 表記などに用いられる．当然，同じ文字でも文字コードが異なると，エンコードの結果は異なる．

- EUC-JP
- Shift_JIS
- JIS
- utf-8

3.2.4 HTTP Client Module

HTTP クライアントモジュールは，検索するべき辞書の URL 一覧を受け取り，それぞれの Web サーバにリクエストを送信する．その際，5 秒のタイムアウトを設定し，Web サーバがダウンしているなど，接続ができない場合は，検索の対象から除外する．

全ての接続が終了すると，Web サーバから，データを読み込む．その際，効率をよくするために，全ての接続先からパラレルにデータを読み込む．

3.2.5 Matching Module

検索型の辞書に対しては，辞書に含まれているかどうかを辞書データベースに登録されている「not found の場合を判断する文字列」を使用して判断する．「not found の場合を判断する文字列」によって判断する理由は，見つかった場合の出力が同じ辞書内でも多様であるからである．例えば，語義が一つしかない場合は，直接辞書の記述を表示するが，複数ある場合は，各語義にリンクが張られているだけの場合もある．それに対して，見つからなかった場合の表示は常に同じである．

HTML 型の辞書に対しては，辞書データベースに登録されている正規表現を使用して，辞書のエントリ部分を抽出し，その中にクエリが含まれるかどうかを判断する．

検索語が見つかった辞書に対しては，クリックした時に右側のフレームに表示されるように検索結果を表示する URL をリンクする．

4 関連研究

“ネットワーク英日・日英辞書メタ検索システム”[3] は，ネットワーク上の辞書をメタ検索できるシステムである．しかしながら，その辞書に該当する語があるかどうかの判定は行っておらず，また，HTML 文書に直接記述があるタイプの辞書は検索することが出来ない．

“OneLook”[4] は，どの辞書に該当する語があるかどうかの判定を行う．しかし，その際，全ての辞書に関して，あらかじめインデックス付けをしておき，それを利用することで，検索語があるかどうかを判定している．本研究では，直接辞書を見に行き，実際その語があるかどうかの判定をするため，辞書の更新により早く対応することが出来る．

5 おわりに

本研究では，自分の調べたい語がどの辞書に存在するかをユーザが意識せずに，辞書を検索することの出来るメタ辞書システムを構築した．通常の辞書に登録されないような専門性の高い語や，辞書に登録されるまでに時間のかかる新しい語などを検索したい場合，このような Web 上の辞書を利用することで，非常に快適な検索を行うことが出来るようになる．

また，使用する辞書に関する知識をユーザに問わないという点で，例えば,「日本語-アラビア語」の辞書が存在しなかった場合に,「日本語-英語-アラビア語」で検索を行うなどといった，複数の辞書サービスのシーケンシャルな利用をより容易にするだろう．

今後の課題としては，辞書の登録方法をより簡単にする方法を検討中である．HTML に直接記述された辞書のエントリ部分を抽出する正規表現を人手で記述することは，非常に大変な作業である．場合によっては，単一の正規表現であらゆるエントリ部分をマッチさせることは不可能かもしれない．しかしながら，より多くの辞書をメタ検索可能にするためには，このような辞書の追加の作業がどうしても必要になる．

そこで，今後，このエントリ抽出部分に，HTML 文書をあたかもデータベースの様に扱うことを目的としている wrapper[5][6] に関する研究を適用することを計画している．

また，半自動的な辞書ページの発見，登録などにも今後取り組んでいく予定ある．ディレクトリ型検索エンジンや辞書のリンク集などを手がかりに，システムが辞書ページの候補を提示し，対話的にデータベースに格納すべきデータの入力を支援するシステムを現在検討中である．

参考文献

[1] “Mecha Search”, http://bach.scitec.kobe-u.ac.jp/metcha/

[2] “metaclrawler”, http://www.metacrawler.com/index.html

[3] 電脳翻訳工房びんごばんご, “ネットワーク英日・日英辞書メタ検索システム”, http://www2n.biglobe.ne.jp/~TM_03822/honyaku/webdict.htm

[4] OneLook, http://www.onelook.com/

[5] Joachim Hammer, Hector Garcia-Molina, Junghoo Cho, Arturo Crespo, Rohan Aranha, “Extracting Semistructured Information from the Web”, In Proceedings of the Workshop on Management of Semistructured Data, held in conjunction with ACM SIGMOD’97, pages 18-25, May 1997.

[6] William W. Cohen, Matthew Hurst, Lee S. Jensen, “A Flexible Learning System for Wrapping Tables and Lists in HTML Documents”, The Eleventh International World Wide Web Conference (WWW2002), http://www2002.org/CDROM/refereed/355/index.html