予測入力の拡張

Improvement of Predictive Text Input System

小松 弘幸 [1)]
Hiroyuki KOMATSU

1) 東京工業大学　情報理工学研究科　数理・計算科学専攻（〒152-8552 東京都目黒区大岡山 2-12-1 東京工業大学 松岡研究室 西 7 号館 1F 102 号室　E-mail:komatsu@taiyaki.org）

ABSTRACT. Predictive text input systems reduce the cost of input by predicting user's input using the knowledge of natural languages and the history of user's operations. In recent years, the predictive text input systems have become popular especially for mobile phones and mobile information appliances.
Our project develops a new predictive text input system PRIME as free software and proposes new methods to predict inputing words.

1　背景

従来のかな漢字変換方式に代わる新しい日本語入力方式として、POBox などに代表される予測入力方式がある。予測入力方式は携帯電話や PDA などの日本語入力装置として、現在では広く用いられるようになった。しかし予測入力方式は、候補の予測方法、入力インタフェースにおいてまだ充分洗練されているとは言えない。

現在の予測入力方式では、入力候補として予測し提示している単語の取捨選択および優先順位の決定は、文章入力の内容によらず一定である。例えばメールでのやりとりの場合、利用者が使用したい言葉は送信相手によって変化する。そのため、予測入力方式は提示する単語の候補を変化させるべきである。しかし従来の方式では、あらかじめ用意された単語辞書もしくは利用者の学習辞書に基づいて優先順位が決定され提示されているため、柔軟な入力候補の予測が困難であった。

2　目的

私は Emacs 上で動作する予測入力方式を実装し、フリーソフトウェアとして一般に公開している[*1](図 1)。現時点での実装でも充分実用的であるが、まだ実装上の改善すべき点が残されている。

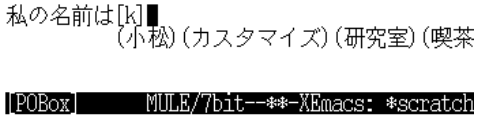

図 1: 一般に公開している予測入力システム

本プロジェクトの目的は、フリーソフトウェアとして実用的に利用可能な予測入力システムを作成するとともに、新しい予測方法 [1−2] を提案することである。

[*1] http://www.taiyaki.org/pobox/

3　開発内容、および特徴

(1) 予測入力システム本体

本プロジェクトの本題は、予測入力システムの作成である。既存のフリーライセンスの予測入力システムとしては POBox がある。POBox は優れた予測入力システムであるが、改良の余地も大きい。本プロジェクトによる予測入力システムの POBox に対する大きな特徴は

- 単文節変換
- 複数変換エンジン
- 単語辞書の抜本的変更
- ローマ字変換エンジン
- 様々な接続方法をサポート

である。以下、それぞれの特徴について述べていく。

a) 単文節変換

本予測入力システムでは、かな漢字辞書の単語には品詞情報が付与されている。そのため、各品詞の特徴に基づいた語尾活用などの単文節変換機能を実現した (図 2)。例えば、「走る」という単語には「ラ行五段」という品詞情報が与えれているため、「走らない」、「走りましょう」といった変換が可能である。従来の POBox では語幹と活用を分けて入力する必要があった。すなわち、「走」＋「らない」の 2 回に分けた入力が必要であった。

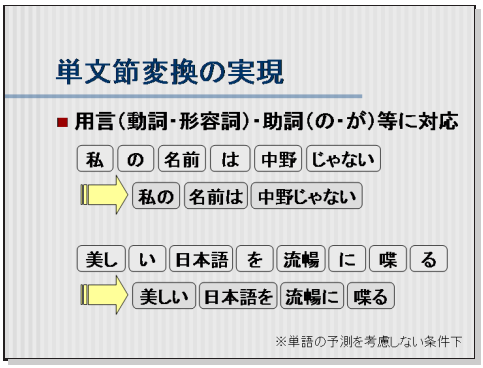

図 2: 単文節変換の実現

b) **複数変換エンジン**

複数の変換エンジンが同時に利用可能であれば、利用者の状況に応じた使い分けが可能である。例えば、専門用語辞書の ON/OFF や追加機能の利用を容易に実現できる。

従来の POBox でも複数変換エンジンは利用可能であった。しかし、変換サーバでは複数変換エンジン機能はサポートしておらず、変換クライアント側で強引にサポートするいびつな実装であった。本プロジェクトでは複数変換エンジンを再設計し、変換サーバ側で複数の変換エンジンを利用可能にした。

変換サーバ側で複数の変換エンジンを利用することにより、

- 素直な実現方法
- 一元的な単語の学習
- クライアントの容易な作成

を実現した。

c) **単語辞書の抜本的変更**

かな漢字変換用の単語辞書に含まれる内容を、

- 単語
- 単語の読み
- 品詞
- 優先順位

とした。これまでの単語辞書に含まれていた内容は

- 単語
- 単語の読みのローマ字表記

であった (図 3)。

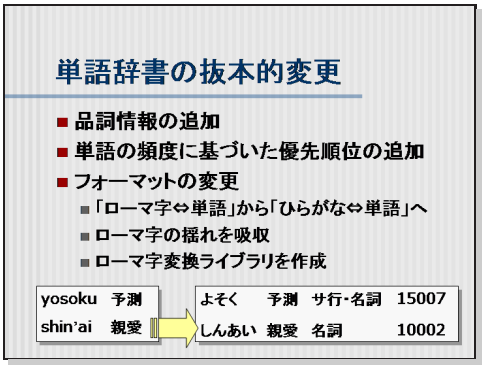

図 3: 単語辞書の抜本的変更

品詞を加えることにより、前述した単文節変換を実現可能にし、品詞情報に基づいたきめの細かい予測を可能にした。また、優先順位の情報からよく利用される単語から候補として提示可能にした。従来の辞書の場合、利用者によって学習されていない単語は、予測候補は辞書の収録順に提示されていた。

加えて、「単語の読み」の表記方法をローマ字からひらがな等に変更した。具体的には、これまでは「予測」という単語のよみは「yosoku」として登録されていたが、この表記方法を「よそく」とひらがな表記に改めた。

単語の読みをローマ字で表記する利点は、単語のインクリメンタルな検索が容易な点である。つまり、「y」と入力するだけで「や」、「ゆ」、「よ」、および英単語の「y」から始まる単語を検索できる。

しかし、ローマ字表記には欠点として表記の揺れの問題が存在する。例えば「親愛」であれば「shin'ai」と登録されているため、利用者が「sinnai」と入力しても「親愛」を検索することはできない。従来の方法では「shin'ai」に加えて「shinnai」、「sin'ai」および「sinnai」という考え得るすべての方法を独立して登録していた。

本プロジェクトでは、かな漢字変換辞書の単語の読みにはローマ字表記ではなく、ひらがな等のいわゆる普通の読みを採用した。単語の読みを素直に表記することにより、ローマ字表記の揺れを抑え辞書をコンパクトにすることを可能した。ローマ字表記の利点であったインクリメンタル検索を当方式で実現する方法については、次節で説明する。

d) **ローマ字変換エンジン**

前節で述べたとおり、かな漢字変換辞書の形式をローマ字表記からひらがな表記に変更したために、そのままでは単語のインクリメンタル検索を行えない。そこで、本プロジェクトでは、ひらがな表記でもインクリメンタル検索を可能するローマ字ひらがな変換エンジンを作成した (図 4)。

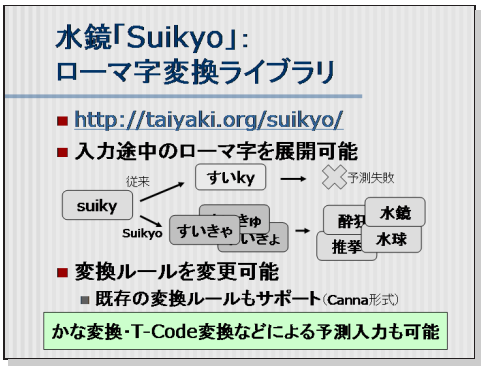

図 4: ローマ字変換エンジン

例えば「suiky」という入力に対して従来のローマ字ひらがな変換エンジンでは「すい ky」のみを変換結果としていたが本プロジェクトの成果物は「すい ky」に加えて「すいきゃ」や「すいきゅ」など取りうる結果をすべて返すことが可能である。

また変換方法は、別のファイルに記述された変換ルールを元に行われているため、ローマ字かな変換以外にも利用可能である。更に、既存のかな漢字システムである Canna と同じ変換ルールのフォーマットを採用しているため、これまでの資産を有効活用することが可能である。実際、ローマ字かな変換に加えて、かなキーボード用の変換ルールや、T-code 入力、AZIK 入力、Dvorak 配列用などの変換ルールが利用可能である。

本プロジェクトで作成したローマ字変換エンジンは、既に独立して「水鏡 (Suikyo)」 という名前で公開を行っている。

e) **様々な接続方法をサポート**

本プロジェクトで作成した予測入力システムは、変換クライアントとの接続方法を複数用意している。複数の接続方法を用意することで、変換クライアントは予測入力システムと容易かつ柔軟に接続できる。接続方法は以下の通りである。

- Ruby クラス
- 標準入出力
- TCP/IP
- Unix ソケット

現時点では Ruby クラスは、後述する XIM サーバとの接続に使用されている。また、標準入出力は、Emacs クライアントとの接続に使用されている。TCP/IP と Unix ソケットでの接続は、本質的なセキュリティの問題を含んでいる。そのため、これらの接続方法は実用するアプリケーションでは使用すべきではない。しかし実装コストを考慮

した場合、TCP/IP と Unix ソケットは有効な接続方法である。クライアントのプロトタイピング用として、これらの接続方法は用意している。

(2) 新しい予測方法

新しい予測方法として、文書蓄積システム「句倉 (Kukura)」を用いた予測入力を作成した [1]。文書蓄積システム「句倉 (Kukura)」は、利用者がこれまでに利用したすべての文書を自動的に蓄積し活用することを目指すシステムである。予測入力システムが Kukura を活用することにより、利用者がこれまでに利用した文書内に含まれてる新語や未知語・固有名詞を、予測することが可能になる。

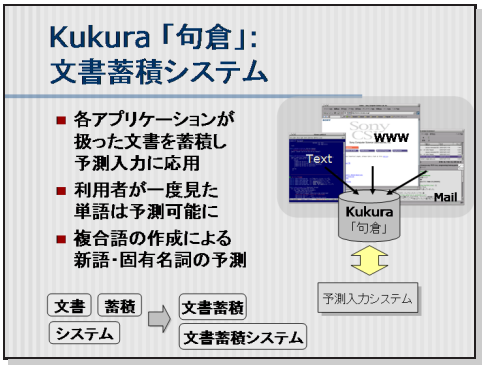

図 5: 文書蓄積システム「句倉 (Kukura)」

例えば「未踏ソフトウェア創造事業」という単語は、固有名詞であり一般的な単語辞書に含まれている可能性は低い。従来の予測入力システムでは「未踏ソフトウェア創造事業」は「未踏」、「ソフトウェア」、「創造」、「事業」と 4 つの単語に分けて入力する必要があった。対して Kukura を用いた予測入力システムでは、「未踏ソフトウェア創造事業」という単語を利用者がメールやブラウザなどで閲覧していれば、ひとつの単語として予測可能にしている。

現段階では、Kukura は利用者が閲覧したメールとブラウザの内容を蓄積し活用することが可能である。また Kukura を活用した予測方法は研究論文として、「文書蓄積システム Kukura を用いた予測入力」という題名で発表している。

(3) Emacs 以外への実装

Emacs 以外でも利用可能なクライアントの作成として、XIM サーバを作成した (図 6)。XIM サーバは、UNIX で広く用いられている X ウインドウシステム向けの文字入力システムである。

予測入力システム用の XIM サーバは既存のシステムである kinput2 にスクリプト言語 Ruby を組み込むことにより実現した (図 7)。

kinput2 は XIM サーバとして広く使われているシステムである。この kinput2 をベースにすることにより、利用者は新しい XIM サーバを違和感なく利用できる。

また、予測入力システム用の変更を直接 kinput2 に組み込むのではなく kinput2 には Ruby を組み込んだ。これにより予測入力システム以外でも Ruby のコードを書くだけで kinput2 を利用した XIM サーバを容易に作成することが可能になる。逆に、kinput2 以外の XIM サーバでも Ruby を組み込めば、予測入力システムを利用出来るようになる。

(4) 同音異義語辞書

かな漢字変換において、同音異義語の候補から目的の言葉を選択するという動作は、利用者の負担となる。例えば

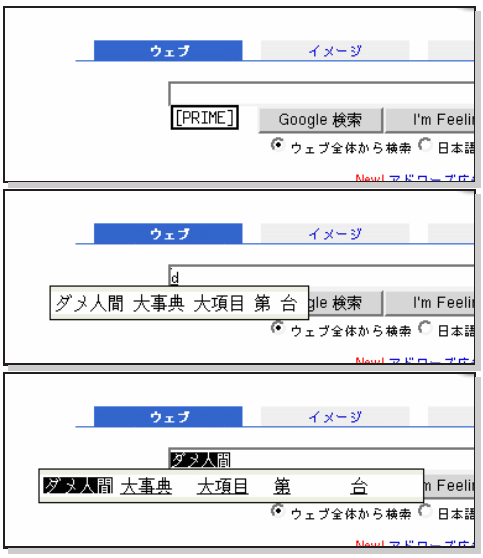

図 6: XIM サーバの動作例

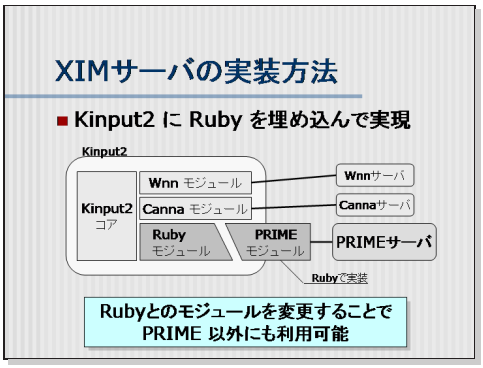

図 7: kinput2 + Ruby による実装

「写真を」に続いて「とる」を「撮る」に変換する場合、「取る」、「採る」といった同音異義語から、利用者は選択する必要がある。予測入力システムは言葉の共起関係を元に同音異義語に優先順位を付けているが、完全ではなく最終的には利用者の判断に委ねられる。そのため、本プロジェクトでは同音異義語辞書を作成し、利用者が同音異義語の中から適切な候補を選択する際の参考にできるようにした。

同音異義語辞書は、市販のかな漢字変換システムでは広く採り入れられているが、フリーライセンスのものは存在しなかった。本プロジェクトで作成した同音異義語辞書はフリーなライセンスの下で配布可能であり、本プロジェクトの本題である予測入力システム以外にも広く利用可能である。

図 8 に同音異義語辞書の一部を例として示す。

4 今後の展望

今後も同様に予測入力システムの開発を継続する。本プロジェクトにより単文節変換は実現された。今後は一般のかな漢字変換システムと同様な連文節変換の実現を目指す。また、現段階では主に Unix 環境向けのシステム内容であるので、Windows の標準環境でも利用可能なシステムを構築する。更に、本プロジェクトの成果物に対する一般利用者向けのドキュメントおよび配布パッケージを整備する。

```
お
#  お お カ行五段
   お 置 カ行五段 その位置にとどめる。「グラスを置く、担当者を置く」
#! お 於 カ行五段（記述なし）

   お 押 サ行五段（一般的）ものに触れて力を入れる。「背中を押す、念を押す」
   お 推 サ行五段 推進させる。「彼を議長に推す」
   お 圧 サ行五段（「押す」も使う）圧迫する。「気迫に圧される」
   お 捺 サ行五段（「押す」も使う）捺印する。「判を捺す」

#  お お ラ行五段
   お 折 ラ行五段 まげる。「小枝を折る、骨を折る（苦労する）」
   お 織 ラ行五段 布をつくる。「布を織る。」
   お 居 ラ行五段 いる。「彼はどこにおられますか、勉強しております」

   お 追 ワ行五段（「逐う」と区別なし）目標をつかまえようとする。「犯人を追う」
   お 負 ワ行五段 ひきうける。「荷物を負う、責任を負う」
   お 逐 ワ行五段（「追う」と区別なし）目標をつかまえようとする。「犯人を逐う」
```

図 8: 同音異義語辞書の例

5 参加企業及び機関

プロジェクト実地管理組織として、日本エンジェルズ・インベストメント株式会社が参加した。

6 参考文献

[1] 小松 弘幸, 高林 哲, 増井 俊之: 文書蓄積システム Kukura を用いた予測入力 WISS2002, pp. 43-47, December, 2002.

[2] 小松 弘幸, 高林 哲, 増井 俊之: 日本語動的単語補完方式 Nanashiki を活用した予測入力, インタラクティブシステムとソフトウェア IX: 日本ソフトウェア科学会 WISS2001, pp. 67-74, December, 2001.