

文書蓄積システム Kukura を用いた予測入力

Predictive Text Input using Document Storage System “Kukura”

小松 弘幸 高林 哲 増井 俊之*

Summary. We introduce a new predictive text input system that uses related documents for predicting the user’s next input word. Using our method, users can efficiently compose documents based on existing documents created by others. We developed a text database system called *Kukura* that stores all the texts referred to by the user and provides the information for our predictive text input system *POBox*.

1 はじめに

予測入力システムは、利用者が編集中の文書の内容に沿って候補を予測するのが望ましい。例えば利用者がメールを返信する場面では、予測入力システムはメールの内容や返信先の相手に応じた候補を予測すべきである。また利用者がウェブページを閲覧しながら文書を作成する場面では、そのウェブページに含まれる単語を予測候補として加えるべきである。しかし、従来の予測入力システムではあらかじめ用意された辞書や、利用者の入力履歴からのみ予測が行われ、利用者がどのような状況で文書を作成しているのかは考慮しない。そのため利用者は、入力したい言葉がアプリケーションの画面に表示されているにも関わらずその言葉をスムーズに予測入力できない、という歯がゆい思いをする。

この問題を解決するために、我々は日本語動的単語補完手法 *Nanashiki* (七色)[4] を開発し、予測入力への適用を行った。*Nanashiki* は現在編集中の文書から単語を抽出し、予測候補に加えるという機能を持つ。*Nanashiki* により利用者は、メールの返信時や他者が書いた文書の編集時においても、編集中のメールや文書に含まれる単語を予測候補として利用できる。しかし *Nanashiki* はウェブページなどの外部のソフトウェアが扱う文書を扱えないという制限がある。

Nanashiki の問題を解決するために、今回我々は文書蓄積システム *Kukura* (句倉) を用いた予測入力システムを提案する(図 1)。*Kukura* は利用者が閲覧した文書の保存と活用を行う。*Kukura* の予測入力への応用は、前述したウェブページや外部ソフトウェアに含まれる単語を予測可能にする。我々は *Kukura* の応用を日本語予測入力方式 *POBox*[2] に追加し、*Kukura* を利用した予測方法の評価を行った。

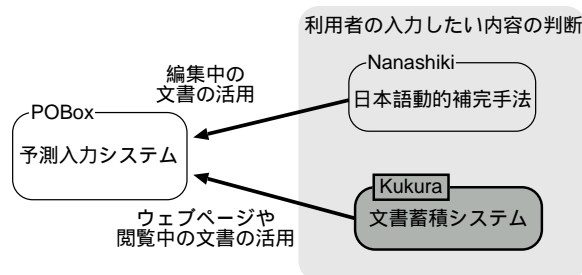


図 1. Kukura および Nanashiki と予測入力方式の関係

2 文書蓄積システム Kukura

文書蓄積システム *Kukura* の目的は、利用者が閲覧している文書を把握し予測入力などに活かすことである。*Kukura* はこの目的のために、利用者の閲覧した文書を保存し活用する。この節では *Kukura* の概要と、*Kukura* を用いた予測入力について述べる。

2.1 Kukura の概要

Kukura は利用者が閲覧した文書を蓄積し、他のアプリケーションにデータを提供する(図 2)。利用者が閲覧した文書を *Kukura* で一元的に管理することで、他のアプリケーションは容易に利用者が閲覧した文書を利用できる。*Kukura* を利用したアプリケーションの例としては、予測入力への活用の他に文書検索などが考えられる。

Kukura はウェブページやメールの他、あらゆる文書の取得を目標としている。現在の実装では、ウェブブラウザ、テキストエディタ Emacs[3] およびコマンドターミナルに対応している。実装方法については後述する。

2.2 Kukura の予測入力への応用

Kukura を予測入力に応用することで、予測入力システムはウェブページの内容や受信メールの内容など、利用者が閲覧した文書に含まれる単語を予測できるようになる。*Kukura* は蓄積した文書から単語を抽出

* Hiroyuki Komatsu, 東京工業大学情報理工学研究所数理・計算科学専攻, komatsu@taiyaki.org, Satoru Takabayashi, ソニーコンピュータサイエンス研究所, 奈良先端科学技術大学院大学情報科学研究科, satoru@cs.sony.co.jp, Toshiyuki Masui, ソニーコンピュータサイエンス研究所, masui@cs.sony.co.jp

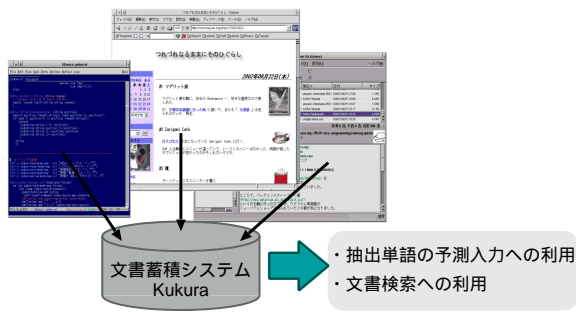


図 2. 文書蓄積システム Kukura の概要

し、予測入力システムに提供する。表 1 は Kukura が予測入力システムに提供する情報の例である。Kukura が提供する情報には、単語と単語の読みに加えて、その単語が文書に含まれる回数を表す頻度も含む。

Kukura は、名詞および未知語を予測候補とする。品詞を限定した理由は、名詞および未知語が文書の文脈を構成し、そして予測候補として有効だと考えるからである。例えば、『Kukura が提供する単語の品詞は名詞および未知語に限定される』という文章において、「が」や「および」、「さ・れる」といった助詞を予測することが効果的であるとは言えない。

Kukura は単一の名詞だけでなく、隣り合った名詞をあわせて複合語とし、ひとつの予測候補として扱う。例えば「かな漢字変換ソフトウェア」であれば、「かな漢字」と「変換」と「ソフトウェア」の各名詞だけでなく、「かな漢字変換」と「かな漢字変換ソフトウェア」という複合語も予測候補とする。複合語の処理については、第 3.6 節で扱う。

また Kukura による予測候補は、利用者に候補として提示されている段階では予測入力用辞書に登録されていない。単語の登録は、利用者が実際に入力を確定した時点でされる。そのため、文書を閲覧しただけでその文書に含まれる単語がすべて辞書に登録されて予測入力用辞書が大きく書き換えられる、ということにはならない。

3 実現

ここでは Kukura の実現方法を述べる。順に、文書の取得と蓄積方法、予測候補の作成方法、予測候補の優先順位、予測入力システムの拡張、複合語による予測候補、について説明する。

3.1 文書の取得と蓄積

現在の Kukura はウェブブラウザ、コマンドターミナル、およびテキストエディタ Emacs で扱う文書を取得する。ウェブページの取得は Kukura が提供するプロキシサーバを用いて行う。コマンドターミナルで表示された文書の取得は、script というコマンドを

用いて行う。script はコマンドターミナルの表示履歴を保存する機能を持つ。Emacs で扱う文書の取得は Emacs の拡張言語である Emacs Lisp を用いて行う。

Emacs のようにアプリケーション自身が Kukura に文書を提供する方法が望ましい文書の取得方法であるが、拡張言語を持たないアプリケーションでもプロキシサーバのようにデータの入出力をトラップすることで文書は取得可能である。

文書の蓄積は、ひとつの文書をひとつのファイルとして保存する。ファイルの分類はアプリケーションごとにメールの送信者や URL などにもとづいて行う。また、閲覧した文書を時系列で並べたリストも作成する。

3.2 予測候補の作成

予測の対象となる文書は、利用者が最近閲覧したいくつかの文書である。予測候補を作成するタイミングは、利用者が文章を閲覧した時点である。Nanashiki が予測候補を入力時に動的に生成するのに対して、Kukura は各文書に対応する予測候補をあらかじめ作成しておく。

予測候補の作成方法は取得した文書に前処理をした後に、形態素解析ソフトウェアの茶筌 [6] を用いて行う。文書の前処理では HTML 文書のタグの削除やコマンドターミナル上の文書のエスケープ記号などが除去される。その後茶筌を用いることにより文を単語に分割する。Kukura は茶筌が分割した単語群から、名詞などの予測候補として利用する品詞を抜き出し、予測候補を作成する。

3.3 予測候補の優先順位

Kukura は提示する予測候補の優先順位を、候補の出現頻度に従って決定する。出現頻度が同じときは、候補の文字数が多い方を優先する。例えば表 2 に示す予測候補では、優先順位は「サイト」、「最新ニュース」、「最新情報」、「サイト内検索」の順になる。

表 2. 頻度と文字数による優先順位の決定

読み	予測候補	頻度
サイシンジョウホウ	最新情報	2
サイシンニュース	最新ニュース	2
サイト	サイト	3
サイトナイケンサク	サイト内検索	1

サイト、最新ニュース、最新情報、サイト内検索の順となる

3.4 予測入力システムの拡張

日本語予測入力システム POBox の Emacs クライアント¹ に対して拡張を行った (図 3)。POBox の Emacs クライアントは、Kukura による予測候補と POBox に

¹ <http://taiyaki.org/pobox/>

表 1. Kukura が作成した予測候補の例 (一部)

読み	予測候補	頻度
カテイ	仮定	1
カナカンジ	かな漢字	7
カナカンジヘンカン	かな漢字変換	6
カナカンジヘンカンソフトウェア	かな漢字変換ソフトウェア	2
カナカンジヘンカンハウシキ	かな漢字変換方式	1
カノウ	可能	2
カノウセイ	可能性	1

よる予測候補の両方を受け取り、クライアント側で候補をマージする。

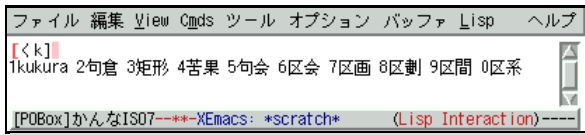


図 3. Kukura の Emacs 版 POBox への拡張

3.5 予測入力システムでの Kukura の優先順位

Kukura による予測候補は、2 つのグループに分けられて異なる優先順位が与えられる。一方のグループには優先順位が高い少数の候補が含まれ、もう一方のグループには残りの候補が含まれる。「高優先順位のグループ」は、従来の POBox が提示する予測候補よりも高い優先順位になる。そして「低優先順位のグループ」は、従来の予測候補よりも低い優先順位になる (図 4)。現在の実装では、高優先順位のグループに含まれる単語数は 2 である。

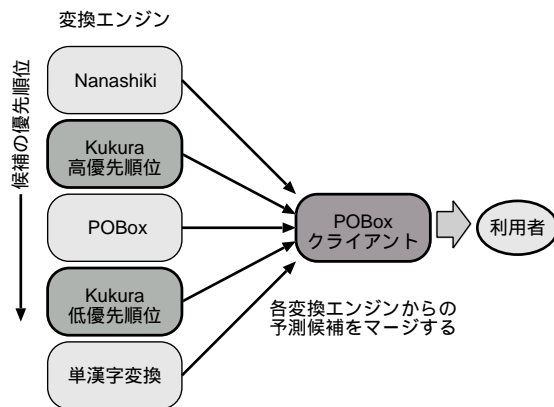


図 4. 各変換エンジンの優先順位

Kukura の優先順位を 2 つに分けた理由は、Kukura により起こり得る悪影響を軽減するためである。Kukura は広範囲の文書から予測候補を作成するため、編集

中の文書の文脈とは異なる候補を提示する恐れがある。そこで優先順位の高い候補の数を限定することにより、Kukura による悪影響を軽減させる。Kukura による予測候補の優先順位が低くても、Kukura の目的である「利用者が閲覧している文書に含まれる言葉を予測候補に加える」という目的は満たされる。

3.6 複合語による予測候補

Kukura は単一の名詞だけでなく、隣り合った名詞をあわせて複合語とし、ひとつの言葉として扱う。例えば Kukura は「文書」と「蓄積」という単語をあわせた「文書蓄積」も予測候補として扱う。複合語を候補として扱う理由は、複合語がより文脈を表しているからである。例えば、第 4.2 節の結果にある「アルバイトさん」や「料理写真日記」という複合語は一般的な単語から構成される固有名詞であり、「料理」や「写真」という個々の単語よりも複合語の方がより文脈を表している。

複合語の作成は、茶筌による品詞情報と漢字・カタカナなどの文字種をもとに行う。複合語の作成方法を以下に示す。

1. 名詞はそのまま予測候補にする
2. 名詞のほか、未知語と接頭語を複合語の要素とする
3. 同じ文字種の要素を複合語として結合する
4. 複合語どうしも複合語にして結合する

「実世界指向インタフェース」を例に説明する (図 5)。「実世界指向インタフェース」を単語に分割すると「実」、「世界」、「指向」、「インタフェース」に分けられる。このうち「実」の品詞は接頭語であり、名詞ではないため予測候補にならない。次に分割した単語を文字種をもとに結合し候補を作成する。文字種にもとづいた結合により「実世界指向」という言葉が候補になる。最後にこれまでに作成された候補をもとに、候補を作成する。「実世界指向インタフェース」という候補は「実世界指向」と「インタフェース」の 2 つの候補が結合して作成されている。

この手法には「実世界」が予測候補にならないなど、いくつかの課題が残る。しかしすべての組み合

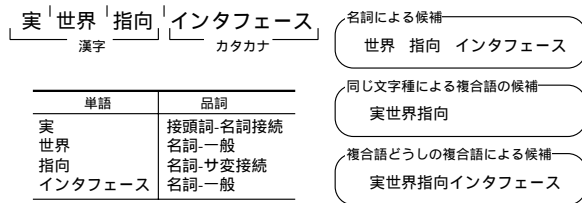


図 5. 複合語の作成例

わせを有効にすると、膨大な候補が作成されて予測の精度が落ちるため、適度な絞り込みを行っている。より高精度な複合語の作成方法は今後の課題とする。

4 評価

Kukura が作成した予測候補の妥当性および、実際に Kukura を利用して文書作成を行った結果を検証した。

4.1 Kukura による予測精度の評価

Kukura が適切に働く場面においての、Kukura による候補の予測精度を評価した。Kukura が適切に働く場面とは、利用者が資料を閲覧しながらその資料に対する文書を作成する時のように、Kukura が扱う文書と編集時の文書の文脈が一致する場面である。

評価方法

利用者がある文書を参考にしながら新たに文書を作成する場面を想定して、Kukura による予測候補の精度について評価を行った。評価は、元となる文書である閲覧文書と、その文書を参考にして作成した作成文書を用いて機械的に行った。例えば「新聞記事」と「その記事の概要」などが、閲覧文書と作成文書にあたる。

評価はまず、Kukura により閲覧文書と作成文書のそれぞれから予測候補を作成し、予測候補の単語数を求めた。そして、閲覧文書と作成文書に共通する予測候補の数と、共通する候補の数を作成文書による候補の数で割った数字を利用率として評価した。

結果

閲覧文書と作成文書として下記の内容の文書を 2 つずつ使用した。

1. 新聞記事と、その記事の概要
2. 新製品のプレスリリースと、その新製品の紹介記事
3. 映画のあらすじと、その映画の感想

計測の結果を表 3 に示す。表の通りおおむね高い利用率を示しており、一定の精度があると考えられる。

表 3. 閲覧・作成文書から Kukura が作成した予測候補数

文書内容	閲覧	作成	共通	利用率
新聞記事 1	161	53	19	35.8%
新聞記事 2	1147	449	181	40.3%
新製品 1	156	60	34	56.7%
新製品 2	516	84	57	67.9%
映画 1	505	248	123	49.6%
映画 2	198	172	22	12.8%

4.2 実際に利用した評価

Kukura を用いた予測入力システムを実際に利用して、入力履歴を記録した。そして、この入力履歴をもとに評価を行った。

評価方法

「入力した単語」、「Kukura による予測かどうか」、「利用者が実際に入力した文字列」の 3 点の記録をもとに評価を行った。「Kukura による予測かどうか」の判定では、以下に示す 3 点の条件によって判定した。

1. 「入力した単語」が Kukura による予測候補に含まれている。
2. 「利用者が実際に入力した文字列」が「入力した単語」の読みよりも短い。
3. 「利用者が実際に入力した文字列」と単語の読みの長さは同じであるが、入力した単語が従来の予測入力システムによる予測候補には含まれていない。

2 を行う理由は「予測候補」の入力に利用者が「yosokukouho」と入力していれば、たとえ Kukura の予測候補に含まれていたとしても、Kukura が効果的に機能したとは言えないからである。²

また、単語入力にかかったキーストローク数を評価するために、入力した単語のローマ字表記での長さ、利用者がその単語を入力するために実際に入力した文字列の長さを用いた。ローマ字表記での長さはヘボン式を用いて算出している。ただし条件 2 および 3 の判定は、ヘボン式以外のローマ字表記の揺れも手作業により吸収した上でやっている。

結果

開発者のひとりによる入力履歴をもとに評価を行った。入力した 1,367 単語のうち、Kukura による予測に含まれる単語は 141 単語であった。Kukura が予測した単語のローマ字表記による文字長と、利用者

² 例外もある。Kukura には「科学」と「化学」のような同音語の候補の優先順位を変える効果もあるため、読みをすべて入力していても Kukura が有効なときもある。しかし今回の評価では扱わない。

よるキーストローク数を表4に示す。Kukuraによる単語の文字長は、平均長が約8.7文字、最大長が24文字であった。これに対して利用者が実際に入力した文字長は、平均長が約3.4文字、最大長が8文字であった。

表4. 入力単語長(ローマ字表記)とキーストローク数

	平均	中間	最大	最小
入力した単語長	8.7	9	24	3
キーストローク数	3.5	4	8	1

実際に入力されたKukuraの予測候補の一部を表5に挙げる。この表が示すとおり、固有名詞などの普段使われない単語が効果的に予測されている。

この表の「同音語辞書」や「校正ツール」は、『同音語辞書を用いて校正ツールを作る』といった内容のメールに含まれていた言葉である。また「アダージョ」という固有名詞は「料理写真日記」というウェブページに含まれていた。これらの単語は、返信メールやウェブページの作者へのメールを作成する時に、初めて入力された。これらの場面においてKukuraによる受信メールやウェブページに含まれる単語の予測は有効に働き、利用者は過去に入力したことのない単語であっても、目的に応じた単語を入力できた。

表5. Kukuraの予測例

入力単語	実際に入力した文字
アダージョ	ada
髪質	kami
アルバイトさん	aru
同音語辞書	douo
校正ツール	kou
プロモーションビデオ	pu
料理写真日記	ryo
かな漢字変換ソフトウェア	kana

5 関連研究

5.1 Nanashiki (七色)

我々が開発した日本語動的単語補完手法Nanashiki(七色)[4]も、利用者が編集・閲覧中の文書を活用して予測候補を作成する。予測入力におけるKukuraとNanashikiの違いは予測候補を作成するタイミングにある。Kukuraによる予測候補は、利用者が文書を閲覧した時点であらかじめ生成される。これに対してNanashikiによる予測候補は、利用者のキー入力のたびに動的に生成される。そのためNanashikiは、現在編集中的文書など内容が頻繁に変更される文書に対して効果的である。逆にウェブページなどの内容の変更がされない文書に対しては、あらかじめ候

補を作成するKukuraの方が実行速度の点において有利である。NanashikiとKukuraをあわせて用いることでより効果的な予測入力の実現できる。

5.2 Japanist

関連する文書に含まれる単語を予測入力に活用する手法は、富士通の予測入力システムJapanist[5]でも利用されている。Japanistは利用者が指定した文書ファイルから学習を行い予測入力用の辞書を作成する。我々の手法との違いは、Japanistは利用者が明示的に文書を指定する必要があるのに対して、Kukuraは自動的に関連する文書を利用する点である。

5.3 履歴の活用

利用者が閲覧した文書を活用するKukuraの手法は、利用者の活動履歴を利用する手法のひとつといえる。活動履歴を利用する手法は様々なインタフェースにおいて利用されている[1]。Kukuraは利用者の活動履歴自体を利用するのではなく、利用者の活動から派生した情報を扱っている点がユニークであるといえる。つまりKukuraは利用者が過去に入力した単語の利用という直接的な行動の利用ではなく、利用者の文書閲覧という行動から派生した、その閲覧文書に含まれる単語を利用している。

6 おわりに

我々は、利用者の閲覧している文書を把握し予測入力に活用するための文書蓄積システムKukura(句倉)を提案した。そしてKukuraにより予測入力システムを拡張しその有効性を確認した。今後の課題はKukuraによって文書の取得可能なアプリケーションの拡充と、より効率的な複合語の作成手法である。我々は利用者の目的を察する予測入力システムの実現を目指している。

参考文献

- [1] H. Lieberman. *Your Wish is My Command: Programming By Example*. Morgan Kaufmann, February 2001.
- [2] T. Masui. An efficient text input method for pen-based computers. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '98)*, pp. 328–335. Addison-Wesley, April 1998.
- [3] R. Stallman. *GNU Emacs Manual*. Free Software Foundation, 2000.
- [4] 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. 近代科学社, 2001.
- [5] 富士通株式会社. Japanist, 2000. <http://software.fujitsu.com/jp/japanist/>.
- [6] 松本裕治. 形態素解析システム「茶筌」. *情報処理*, Vol. 41, No. 11, pp. 1208–1214, 2000.