頻度情報を用いた
Anthyのかな漢字変換精度向上の提案

柳田誠也

Uim conference　2004年11月27日

Anthyと商用変換エンジンの比較

・文節の切り方は商用変換エンジンと比べても遜色ない

例：わたしはきのうでんしゃにのってとうきょうへいった。

Anthy-5900:　わたしは　きのう　でんしゃ　にのって
　　とうきょうへ　いった。
MS-IME 2002:　　わたしは　きのう　でんしゃに　のって
　　とうきょうへ　いった。

・しかしながら、同音異義語の処理が不得意

Anthy-5900:　　わたしは機能電車似のって東京へいった。
MS-IME 2002:　私は昨日電車に乗って東京へ行った。

↓

同音異義語をうまく処理できればAnthyの変換精度はより向上する

→商用エンジンでは用例変換で同音異義語を処理している。

用例変換の実装:統計的に扱う

わたしはきのうでんしゃにのってとうきょうへいった。

	昨日				行った
	機能		載って		言った
私は	帰納	電車に	乗って	東京へ	入った 。
	気嚢				逝った
					炒った

・「電車」の後には「乗る」のくる確率が高い
・「東京」の後には「行く」のくる確率が高い

↓

あらかじめ単語対の出現頻度を調べておき、
変換候補選択にその情報を利用する

頻度情報の収集手順(1)

- 単語ID生成処理　…… 手作業、初回のみ
  mecab登録単語に4バイトの通し番号をつける。

- 元データの収集　…… 手作業
  頻度情報を収集する対象の文書をできるだけ多く集める。

- 元データの前処理　…… nkf利用
  元データの漢字コードを統一し、半角カタカナ・半角記号は
  全角に統一する。なお英字の統一はmecab内で行われる。

- 形態素解析処理　…… mecab利用
  元データをmecabで形態素解析する。

  私 は 昨日 電車 に 乗っ て 東京 へ 行っ た 。

  私　　名詞, 代名詞, 一般, *, *, *, 私, ワタシ, ワタシ
  は　　助詞, 係助詞, *, *, *, *, は, ハ, ワ
  昨日　　名詞, 副詞可能, *, *, *, *, 昨日, キノウ, キノー
  電車　　名詞, 一般, *, *, *, *, 電車, デンシャ, デンシャ
  に　　助詞, 格助詞, 一般, *, *, *, に, ニ, ニ
  (略)

頻度情報の収集手順(2)

・フィルター処理　　…… 新規開発、Perl
　形態素解析結果から助詞、記号、および未知語を取り除く。
　活用がある場合は基本形に戻し、
　文頭・文末は特殊記号EOSとする。

　EOS 私 昨日 電車 乗る 東京 行く た EOS

・入力データ作成処理　……新規開発、Perl
　処理結果に単語IDを付加する。

　EOS 私 昨日 電車 乗る 東京 行く た EOS
　255 1 2 3 4 5 6 7 255
　(単語IDは説明のためにつけた仮のもの)

・頻度データ作成処理　…… 新規開発、CまたはPerl
　(1)配列要素数が単語ID×単語IDの二次元配列を用意する。
　(2)入力データの各々の単語について以下を行う。
　　(2-1)入力データから1単語読み取る。これを注目単語とする。
　　(2-2)注目単語の次の単語を読み取る
　　(2-3)配列[注目単語ID][次の単語ID]++

処理に必要な資源

・ディスク容量
　入力データ用　　…数GB、入力データは多ければ多いほどよい
　頻度情報格納用 …1TB～2TB(無圧縮の場合)
　　頻度情報は圧縮すれば数十GBに収まる

・CPU
　速いに越したことはないがBSD/Linuxが動けばよい

・メモリ
　最低512MB

・作業期間
　数ヶ月？

頻度収集処理自体は容易に分散処理可能
↓
PCを数台用意し、入力データを共用すれば
短期間で作業終了