発音に基づく英単語検索システム

035705H 新垣芽依美　　指導教官：宮城隼夫

1　はじめに

海外のテレビ番組等を字幕なしで観ていると, 自分の知らない単語を耳にすることがある. その単語を辞書で調べようとするとき, スペルがわからないためそれが難しい場合がある. 英語はスペルと発音が一致しない単語を多く持つ言語であり, 英語を母国語としない人にとって, スペルの不明瞭な単語を検索することは容易ではないためである.

ある程度スペルが予想できれば, 電子辞書などに搭載されているスペルチェック機能を用いて目的の単語を検索することが可能である. しかし, スペルがまったくわからない場合, 目的の単語を見つけるのに時間がかかる, あるいは結局見つけられないということもある.

従って, 発音を利用して英単語の検索を行えば, より効率よく単語を見つけ出せると考えられる.

発音を利用した検索システムには, 発音記号を入力して単語を検索できる Pronunciation Search[1] がある。しかしこのシステムは入力した発音記号列と完全に一致する単語が辞書中になければ, 何も提示せず検索を終了する. そのため発音記号について熟知していなければ使いこなすことが難しい.

そこで同研究室の金城氏 [2] は, 正しく発音記号列を入力できなかった場合においても, その入力記号列に対して Edit Distance による変換操作を行うことで, 入力記号列に近い発音を持つ単語を探し出す英単語検索の手法を提案した.

本研究では, その手法を組み込んだ英単語検索システムの構築を目的とする.

2　基本概念

2.1　発音記号

発音記号は, 正式には国際音声記号（IPA : International Phonetic Alphabet）[3] と呼ばれる. これはあらゆる言語の音声を文字で表記すべく, 国際音声学会が定めた音声記号である.

2.2　Edit Distance

文字列の“ 近さ ”を示す尺度のひとつとして Edit Discance（以下 ED とする）[4] という考え方が使われている. ED は, ある文字列から別の文字列に変形するのに必要な変換操作（削除, 挿入, 置換）の最小回数である. 例えば, “ breath ” と“ breathe ”の ED は, ’e ’を 1 回挿入すればよいので 1 となる.

3　単語検索処理

入力された発音記号列に近い音を持つ単語を検索する処理手順を示す.

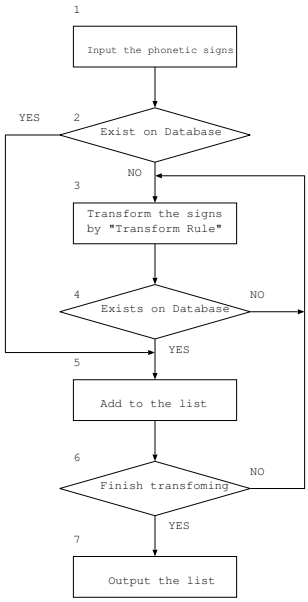

図 1: 検索処理の流れ

手順 1　ユーザは聞き取った英単語を, 発音記号を使用して入力する.

手順 2　**手順 1** で入力された発音記号列をデータベースと照合し、完全に一致するものがあればリストに格納する.

手順 3　**手順 1** で入力された発音記号列を, 変換ルールに基づいて変換操作する.

手順 4　**手順 1** で入力された発音記号列と, **手順 3** の操作により生成された発音記号列をデータベースと照合する.

手順 5　**手順 3** を行った結果, データベースに存在が確認された発音記号をそれに該当する英単語とともに検索結果候補リストに格納する.

手順 6 すべての変換ルールを実行し終えるまで**手順 3** から**手順 5** の操作を繰り返す.

手順 7 リストを表示して終了する.

4 システムの構築

4.1 開発・実験環境

OS Mac OS X 10.3.9

CPU 800 MHz PowerPC G4

メモリ 640MB

DBMS PostgreSQL 8.1.4

JAVA J2SE 1.4.2_09

4.2 入力用発音記号

英語には, 日本語の ’ア ’に近い音が約５種類あり, 日本人がそれらを聞き分けることは容易ではない [5]. 本研究では, ユーザが発音記号を選択および入力する際の負担を軽減するため, ５種類の母音 [æ/ʌ/ɑ/ə/ɛ] を単一の [a] として扱う.

5 構成

システムは Java プログラムとデータベース（PostgreSQL）で構築する.

プログラムは, 削除, 置換, 挿入それぞれの変換操作のクラスファイルとメインクラスから成る.

メインクラスでは入力された発音記号の読み込み, データベースとの接続, 候補となる単語のリストへの格納およびリストの出力を行う.

5.1 辞書データベース

5.1.1 辞書データ

- English-German Free dictionary
  eng-deu.dict.dz, eng-deu.index

Freedict.org[6] が配布している辞書データのうち, 発音記号を含み, 収録語の多いデータを利用する.

上記辞書データは dict フォーマットで記述されている。DICT（Dictionary Server Protocol）[7] はサーバから辞書を引くためのプロトコルである.

5.1.2 データベースの構築

上記辞書データは英単語を見出し語として使用している. 本研究では発音記号を見出し語とする検索シスステム構築するため, データをアンフォーマットして発音記号を取り出す.

また, 発音記号をコマンドラインから扱えるようにするため, 各記号を次の表に従ってアルファベットや数字に置き換える.

表 1: 子音

p(p)	b(b)	t(t)	d(d)	k(k)
g(g)	f(f)	v(v)	T(θ)	D(ð)
s(s)	z(z)	S(ʃ)	Z(ʒ)	h(h)
C(tʃ)	B(dʒ)	m(m)	n(n)	N(ŋ)
l(l)	r(r)	j(j)	w(w)	

表 2: 母音

a(æ/ʌ/ɑ/ə/ɛ)	i(i)	u(u)	e(e)	o(ɔ)
A(ɑː)	I(iː)	U(uː)		O(ɔː)
1(ai)	2(au)	3(ei)	4(ɔi)	5(ou)
6(iər)	7(uər)	8(eər)	9(ɔər)	R(əːr)

6 変換操作

削除, 置換, 挿入それぞれの変換操作をクラスとして Java プログラムを作成する.

6.1 削除

- 入力記号列に母音が２つある場合, そのどちらか一方を削除する（入力記号数３の場合のみ）.
- 入力文字列の語頭以外の母音に対して削除操作を行う.

6.2 置換

- 入力記号列の音素を, 金城氏 [2] の提案した置換対応表に従って操作する.

6.3 挿入

- 入力記号列に子音の連続がある場合, 子音間に母音を挿入する.
- 破裂音 [p/b/t/d/k/g] を語尾に挿入する.
- 破裂音 [p/b/t/d/k/g] を各子音の直前に挿入する.
- 入力記号列に [ɔ/ɔː/ou] がある場合, それらの直後に [l] を挿入する.

7　まとめ

本研究では, 発音記号を用いたスペルチェッカシステムの目指している.

今後の予定は, メインクラスおよびそれぞれの変換操作に対応したクラスファイルを作成する. また, ユーザにとって発音記号の入力および出力された検索結果の参照が容易となるためのインタフェースを構築する.

参考文献

[1] Longman Dictionary : Pronunciation Search, http://www.longman.com/ldoce/, 2005

[2] 金城めぐみ, 『発音に基づいた英単語検索システム』, 平成 17 年度琉球大学工学部情報工学科卒業論文, 2006

[3] IPA 国際音声字母, http://www.coelang.tufs.ac.jp/ipa/

[4] 田中穂積, 『自然言語処理 -基礎と応用-』, 大阪教育図書, 1999

[5] 小池生夫, 『英語のヒアリングとその指導』, 大修館書店, 1993

[6] FreeDict.org, http://www.freedict.org/en/, 2006.11.23

[7] 訳用辞書システム, http://www.tradic.jp/dict, 2006.11.23