

## 多機種間のマニュアル概要検索のためのファジー検索拡張シェル

3K-5

freshman(Fuzzy Retrieval Extended SHell for a command MAN)<sup>1</sup>安達久博 下山豪彦<sup>2</sup>宇都宮大学 工学部 情報工学科<sup>3</sup>

## 1 はじめに

近年、ネットワーク環境下で多機種のUNIXワークステーションを利用する機会が多くなってきた。しかし、各マシンでOSのバージョンが異なるために、「同じ機能を提供するコマンド名が機種により異なる」、逆に「同じコマンド名が別の機能を提供している」という状況に遭遇する。例えば、通常rshはリモートシェルコマンドであるが、あるマシンでは別の機能を提供しており、remshという名前のコマンドがリモートシェルコマンドに対応する等の場合である。同様に、ソフトウェアの移植の際にもライブラリ関数名の違いが大きな問題になる。

一方、各マシンともオンライン・マニュアルを表示する手段として、manコマンドが提供されている。また、各コマンドの機能を一行文に要約記述したマニュアル概要文データベース(whatis)を利用し、-kオプションとして、指定された単語をマニュアル概要文に含む全てのコマンド一覧を表示する機能も提供されている。しかし、これらの機能は現在利用しているマシン上でのコマンドに対してのみ有効に働く。上記のように異なる機種間での対応するコマンドを調べる有効な手段は提供されていないのが現状であり、ユーザの負荷が大きい。

本稿では、このような問題を解消するために、多機種間のwhatisデータベースを利用し、マニュアル概要文間のファジー関係に基づいたメンバーシップ関数(類似度)により、同一機能を提供してくれる可能性の高い類似のコマンド名をユーザに提示するシステムを提案する。

## 2 マニュアル概要文の特徴

図.1にwhatisデータベースのマニュアル概要文の例を示す。データベースの形式は、コマンド(ライブラリ関数)名、セクション番号、マニュアル概要文の3項目から成る。これらの概要文(英語文)は、「計算機コマンドの機能を簡潔に表現する」という目的のため、一般の自然言語文と比べて、文中で使用される語彙(単語集合)や構文パターンが非常に限定された文集合と捉えることができる。

U	chmod(1)	- change file mode
L	fmt(1)	- simple text formatter
R	sleep(1)	- suspend execution for a time
I	stty(1)	- set terminal options
X	diff(1)	- differential file comparator
<hr/>		
	chmod(1)	- change file mode
H	adjust(1)	- simple text formatter
P	sleep(1)	- suspend execution for an interval
U	stty(1)	- set the options for a terminal port
X	diff,diffh(1)	- differential file comparator

図.1 whatis データベースの例

## 3 マニュアル概要文間の類似性を判断するメンバーシップ関数

異機種間で類似の機能を提供するコマンド名を調べる場合、先に議論した様にコマンド名の類似性からは判断できない。我々は、「コマンド機能の類似性はマニュアル概要文の類似性の度合で近似する」方針とする。

一方、「文Aと文Bが等しい」という様な明確な関係に対して、「文Aと文Bは似ている」という様なあいまいな関係は、ファジー関係として定義できる[1]。また、マニュアル概要文が文法的にも意味的にも非常に

<sup>1</sup>A Fuzzy-based Command Retrieval System on Different Computers<sup>2</sup>Hisahiro ADACHI and Hidehiko SHIMOYAMA<sup>3</sup>Department of Information Science, Utsunomiya University

制限された文集合であるという特徴を考慮し、このファジー関係のメンバーシップ関数を定義する。「英語文は、構文パターンを構成する単語の語順に強い文法的な制約がある。従って、文中に出現する単語の順序を重視（保存）する形で二つの文の単語列を比較し共通部分単語列を抽出する」を基本戦略として「最長共通部分文字列（LCS）探索問題」を単語列に拡張して類似性の尺度として導入する。

ここで単語列拡張 LCS とは、2つの単語列  $A = a_1 a_2 \dots a_m$  と  $B = b_1 b_2 \dots b_n$  に対して、両者に共通かつ最長の部分単語列を導出することである。ここで、部分単語列とは、元の単語列から単語の出現順序を変えずに抽出したもので、連続している必要はない。LCS は以下に示した DP 法を利用し効率良く探索することができる [2]。

$$LCS(A, B) = c(m, n) \quad (1)$$

$$c(i, j) = \begin{cases} c(i, j) = 0 & \text{if } i = 0 \vee j = 0 \\ c(i-1, j-1) + 1 & \text{if } a_i = b_j \quad (1 \leq i \leq m \wedge 1 \leq j \leq n) \\ \max\{c(i, j-1), c(i-1, j)\} & \text{if } a_i \neq b_j \quad (1 \leq i \leq m \wedge 1 \leq j \leq n) \end{cases}$$

一方、2つの概要文 A, B の単語数をそれぞれ  $Length(A)$ ,  $Length(B)$  とし、LCS の長さを  $LCS(A, B)$  とした場合、マニュアル概要文間のファジー関係を規定するメンバーシップ関数  $R(A, B)$  は次式で定義できる。但し、冠詞は単語の比較の対象外とする。

$$R(A, B) = \frac{LCS(A, B)}{Length(A) + Length(B) - LCS(A, B)} \quad (0 \leq R(A, B) \leq 1) \quad (2)$$

## 4 検索手順と実現例

本システムは、日頃使用している A 社の計算機とビギナーである B 社の計算機の 2 つの whatis データベースを利用して、以下に示す手順で入力された A 社の計算機でのコマンド名のマニュアル概要文に類似した B 社のマニュアル概要文をファジー検索し、ユーザの指示によりマニュアル表示を行う。図 2 は、DEC 社の ULTRIX 4.3 上に実装した実現例を示している。ここで、-hp オプションは日頃使っている計算機が HP-UX であることを表わしている。

- (1) コマンド名を比較し同じならば (2) へ、違うならば (3) へ
- (2) 概要文を比較し同じならば (4) へ、違うならば (3) へ
- (3) 概要文の類似度を計算し、検索結果を表示し、選択されたら (4) へ
- (4) 指定されたコマンド名を man コマンドへ引数として渡す

```

freshman -hp
freshman man rensh
HP          : rensh : execute from a remote shell
1:DEC :0.500000: rsh : remote shell
2:DEC :0.200000: rlogin : remote login
3:DEC :0.200000: msh : MH shell
4:DEC :0.166667: pixstats : analyze program execution
5:DEC :0.166667: rcp : remote file copy
何番のマニュアルをご覧になりますか? █

```

## 5 実験と検討

図 2 freshman の実行例

ULTRIX 4.3(DEC) と HP-UX 6.5(HP) における whatis データベースのセクション番号 (1) のコマンドについて、コマンド名と概要文に等しいものを除いたデータについて比較実験を行なった。

HP-UX のデータ 282 文をキーにして ULTRIX のデータ 258 文と比較したところ、メンバーシップ関数のグレードが 0.5 以上で検索した結果、該当する概要文が上位に入る割合が約 90% となった。残りの約 10% のコマンドのグレードが低い原因としては、1. 文中に使用されている単語の意味は同じであるが、単語が違う。2. 複数のコマンドを新規に 1 つの概要文に対応させたために全く違う文（単語列）が生成されている。3. 略語とフルスペルの違い。などが挙げられる。

## 6 おわりに

本稿では、多機種間で類似した機能（概要文）を持つコマンドをファジー関係に基づき類似検索するシステム freshman の一構成方法を提案した。今後、マニュアル概要文の単語間のシソーラスを利用することで類似度計算の機能を拡張し、より最適な検索結果を得る必要がある。

## 参考文献

- [1] 坂和正敏: ファジー理論の基礎と応用, 森北出版, pp.30-43, (1989).
- [2] Thomas H and et al: Introduction to Algorithms, The MIT Press, pp.314-320 (1990).