4K－11

WWWサーチエンジンの検索結果のクラスタリングによるカテゴリ分け

夏目貴史　森 真史　石塚 満
東京大学工学部電子情報工学科

1 はじめに

近年、WWWにおける情報の量は膨大なものとなっている。その中から、ユーザが必要な情報を探し当てるために使用するものとしてはサーチエンジンがあげられる．そのサーチエンジンを使用する際、場合によっては検索にヒットしたリンク先（以下ファイルと呼ぶ。リンク先のHTMLファイルの意。）が何百と出てくる。この場合、ファイルを絞り込むために二次的なキーワードを入力するといったような操作をする必要がある。こういった操作を加えることなく自動的に、ユーザが利用しやすいような絞り込む方法を提供する方法について本稿では述べる。

2 検索情報のカテゴリ分け

ユーザに絞り込むガイドラインを与える方法としては、二次キーワードの提示や、検索にヒットしたファイルのカテゴリ分けということが考えられる。二次キーワードの提示とは、検索にヒットしたファイルの持つ単語からキーワードになりそうな単語の候補を抽出し、ユーザに提示する。そして、選択されたキーワードと最初にユーザが入力したキーワードを使ってサーチエンジンでAND検索をすることによって絞り込んでいく方法である。ファイルのカテゴリ分けとは、ユーザがまず第一番目のキーワードをサーチエンジンに入力し、その検索にヒットしたファイルをカテゴリ分けしてユーザに提示し、ユーザが自分が望む情報に近いカテゴリを選んでいくことによりファイルを絞り込んでいく方法である。ここではファイルのカテゴリ分けの方法を採用する。その理由は、

- 一つのカテゴリに含まれる要素の数を制限することにより二度三度繰り返す手間を省くことができる。

からである。

Takashi NATSUME,Masafumi MORI,Mitsuru ISHIZUKA
Dept. of Information and Communication Engineering,
Faculty of Engineering, The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo, 113, JAPAN

3 最大距離アルゴリズムによるクラスタリング

3.1 クラスタリングの方法

まず、検索にヒットしたファイルに含まれている頻出語（以下ワードと呼ぶ）を抽出する。そのうち二つ以上のファイルに含まれているワードとその数を調べ、それを軸とし、n 次元空間を生成する。これをファイルの特徴ベクトルとする。それを最大距離アルゴリズム [1] によってクラスタリングする。

3.2 最大距離アルゴリズムの問題点

最大距離アルゴリズムによってクラスタリングを試みたが、うまくクラスタが生成されないことがわかった。これは、

- ひとつのクラスタにたくさんのファイルが固まってしまうことがある。
- クラスタの要素の数にかなりばらつきがでる。
- 同じワードが含まれていないファイルが同じクラスタに分類されることがある。

ということである。そこで別の方法を取ることにした。この方法については次節で述べる。

4 カテゴリ分けの方針

まずカテゴリ分けを行なうにあたって、方針を定めることにした。以下のような方針でカテゴリを作成するものとする。

- 一つのカテゴリに含まれる要素の数が多過ぎないこと。
- カテゴリの要素同士の関連がかなり高いこと。
- 一つのファイルが複数のカテゴリに含まれることがあってもよい。

5 カテゴリ分けのアルゴリズム

最大距離アルゴリズムの場合と同じく特徴ベクトルをもとめるのだが、ここではあるワードが含まれていれば1いなければ0として n 次元空間を生成することにする。これをファイルの特徴ベクトルとする。ファイルとファイルの距離（関連の高さ）は特徴ベクトルの内積をとり、その値が大きい程関連が高い（距離が近い）とすることにする。そして、以下のような手順でカテゴリ分けする。

Step.1 ファイルの除外

特徴ベクトルの要素がすべて0のファイルをカテゴリ分けの対象から除外する。これは特徴ベクトルの要素がすべて0であるとどのファイルと内積をとっても0にしかならない、つまり関連がないということになるからである。

Step.2 ファイルの代表

特徴ベクトルが同じファイルは一つを除いて取り除く。つまり一つのファイルに複数のファイルの代表をさせるということである。これは、サーチエンジンの検索結果には同じリンク先が二回以上登録されていたり、ミラーサイトが存在していたりするためである。

Step.3 基準としたファイルから一番近いファイルの発見

Step.1 と Step.2 を経て m 個のファイルが残ったとして、それらを File.1, ...,File.m と表現する。先ず、File.1 を基準として一番近いファイルを見つけ出す。

Step.4 ひとつのカテゴリの作成

一番近いファイルが、ある閾値より近ければ基準としたファイルとそのファイルは同じカテゴリに属するものとする。これらを仮のカテゴリとする。そして、次に仮のカテゴリと一番近いファイルを見つけ、それがある閾値より近ければ仮のカテゴリに加える。そしてまた一番近いファイルを見つけ.... という具合に、カテゴリの作成を行なう。仮のカテゴリとファイルの距離は、仮のカテゴリの各要素とファイルの距離の平均とする。

Step.5 複数のカテゴリの作成

File.2 から File.m をそれぞれ基準として Step.3 から Step.4 を繰り返す。こうして複数のカテゴリが作成される。

Step.6 ユーザに提示するカテゴリの選択

Step.5 までで複数のカテゴリが作成されたわけだが、これをすべてユーザに提示するとユーザはたくさんのものの中から選ばなければならなくなり、ユーザにとって使いやすいものとはならないので、適当な数に抑えてやる必要がある。それは

- 要素の数
- カテゴリの要素同士の距離の近さ
- 希少なワードがふくまれているかどうか

といったような基準で適当な数を選び出すものとする。

Step.7 ファイルの付加

Step.2 で取り除いたファイルについては、代表させているファイルのところへ付け加える。

Step.8 どのカテゴリにも含まれないファイルの扱い

どのカテゴリにも含まれないファイルはその他のファイルとしてユーザに提示する。

6 おわりに

本稿ではサーチエンジンの検索結果のカテゴリ分けの方法について提案した。今後は、この方法を、辞書を利用するなどより正確にカテゴリ分けできるように拡張をはかっていきたい。

参考文献

[1] 長尾真「画像認識論」コロナ社, 1983, pp.114-126

[2] 長尾真「パターン情報処理」コロナ社, 1983, pp.115-116