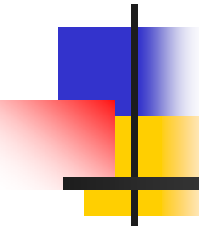


文書クラスタリングの技法

7. Leader-follower アルゴリズムの応用
8. Crouch のアルゴリズム
9. CBC アルゴリズム

Leader-followerアルゴリズム

- 特徴
- クラスタ個数の代わりに文書とクラスタの類似度の閾値を設定しておく
- 分類対象を処理していく中で閾値に基づいて自動的にクラスタが形成されていく

単純なleader-follower法①

（1）閾値θsを設定して最初の文書d_1を読み、最初のクラスタとする。

→クラスタの個数Lは1になる。

→iを初期化しておく。

（2）i←i+1とし文書diを読む。すべての文書を読み終わっていれば終了。そうでない場合は（3）に進む。

単純なleader-follower法②

（3）文書ベクトルdiとその時点での全てのクラスタベクトルckとの類似度s(di,ck)を計算する。

もし閾値を越えればdiをクラスタckに加え、ベクトルckを更新する。

もし文書がどのクラスタにも加わらなければ新しいクラスタとする。

→L←L+1,cL←di

この後（2）に戻る。

単純なleader-follower法③

- 問題点
- θsの設定が課題となっている
- Lが大きくなると計算量が増加する
- クラスタベクトルの格納に多くのメインメモリを消費する

Crouchのアルゴリズム ①

（1）クラスタの設定

（2）クラスタへの文書の割り当て

という二段階からなる。

段階（1）

・ベクトルckを更新する際に各語の変動係数を計算してそれに基づき各クラスタを表現する語の集合（core terms）を選別する。

Crouchのアルゴリズム ②

- core termsの重さの平均に基づいてベクトルckを更新する。

段階（2）

- クラスタの重複が許され、一件の文書が複数のクラスタに属することが可能である。

利点

- クラスタベクトルを格納する領域が少ない
- 類似度sの計算時間が短い

CBCアルゴリズム①

段階Ⅰ　文書ごとに類似度の高い20文書を求める

段階Ⅱ　段階1で計算された類似度かcommitteeリストを求める。

段階Ⅲ　各文書をもっとも類似したcommitteeに割り当てる。

＊ここでcommitteeとは各クラスタの核となるいわば「代表者」であり、段階2で求められる。

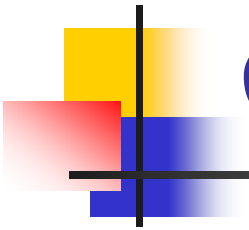
CBCアルゴリズム②

段階Ⅱ

1．2つの閾値θ1, θ2を決める。3種のリストLc、Lk,Lrを用意してLkをcommitteeリストとする。

また、未処理の文書をリストEに格納する。

2．リストLcを空にし、Eに含まれる文書で次の処理を行う。



CBCアルゴリズム③

２－１　その文書と類似した20文書に対して群平均クラスタリングを適用する。

２－２　その結果生成されたクラスタCごとに得点|C| × avgsim(C)を計算する。

・|C|はそのクラスタに含まれる文書数

・avgsimはC中の文書の組ごとの類似度の平均

CBCアルゴリズム④

２－３　得点の最も高いクラスタをリストLcに追加登録する。

３　リストLcを得点の降順に並べ替える

４　リストLkを空にする（初期化）

５　リストLc中の上位から順に書くCについて次の処理を行う。

CBCアルゴリズム⑤

５－１　クラスタCの重心とその時点でリストLkに含まれている全てのcommitteeの重心との類似度を計算する。

５－２　それらの類似度が全て閾値θ１を下回っている場合、CをcomitteeのリストLkに追加する。

６　リストLkが空ならば処理を終了する。

CBCアルゴリズム⑥

７　再びEに含まれる文書ごとに以下の処理を行う。

７－１　もしその文書とすべてのcommitteeとの類似度が閾値θ２を下回るならばその文書をリストLrに追加する。

８　もしリストLrが空ならば処理を終了する。

そうでなければLkの内容を保存し、LrをEとして２の処理に戻る。

CBCアルゴリズム⑦

出力結果

段階8で保存されたリストLkの和集合

段階Ⅲ　和集合に含まれる各committeeの重心に対する各文書の類似度を計算し、類似度の最も高いcommitteeに割り当てる。

CBCアルゴリズム⑧

利点

各クラスタの核をcommitteeとして求めることによって「周辺的」な文書がクラスタの重心の計算に不当な影響を及ぼすことを防いでいる。