照合関数が線型である検索モデル一般に適用可能なレレバンスフィードバック手法

岸田和明
駿河台大学文化情報学部
kishida@surugadai.ac.jp

1.はじめに

情報検索におけるレレバンスフィードバックの手法としてはベクトル空間型モデルにおける Rocchio の手法がよく知られている。これは元の質問ベクトルに適合文献集合中の語の重みの平均を加え，さらに不適合文献集合中の語の重みの平均を差し引くことによって，元のベクトルを修正する方法である[1]。この方法は，原理的には，ベクトル空間モデルにのみ適用可能であり，なおかつ適合／不適合の 2 値での判定にのみ対応している。

それに対して，岸田[2]によって提案されたフィードバック手法は，ベクトル空間型モデルをはじめとする照合関数が線型である検索モデル一般に適用可能であり，なおかつ，文献の適合度が連続的な数値によって与えられた場合に対応した手法である。この手法の検索性能は Rocchio の手法をわずかに上回ることが実証的に示されている[2]。

本稿の目的は岸田[2]による提案手法の有効性を別の角度から再度検証することにある。すなわち，この手法が確率型検索モデルの一種である Okapi の方法[3]に対しても適用可能であることを日本語テストコレクション NTCIR-1 を使用して実証的に確認する。

2.テイラー展開に基づくレレバンスフィードバック手法

2.1 線形写像としての検索モデル

ここでは岸田[2]の提案手法を簡単に説明する。まず，適合度順出力のために検索モデルが文献 d_i に対して与える得点を s_i（$i=1,...,N$），検索質問における語 t_j の重みを b_j とする（$j=1,...,M$）。この記号を使えば，検索モデルは b_j から構成される M 次元空間から s_i によって構成される N 次元空間への写像 $f:R^{M\times 1}\rightarrow R^{N\times 1}$ ということになる。特に f を線形写像に限定し，

$$\mathbf{s}=f(\mathbf{b})=\mathbf{Ab} \tag{1}$$

とする。ここで $\mathbf{s}=(s_1,...,s_N)^T$ および $\mathbf{b}=(b_1,...,b_M)^T$ であり，$\mathbf{A}=(a_{ij})$ は $N\times M$ の行列である。

Okapi での検索モデル[3]は，

$$a_{ij}=\frac{2.2x_{ij}}{1.2(0.25+0.75l_i/\bar{l})+x_{ij}} \tag{2}$$

および，

$$b_j=x_{qj}\log\frac{N-n_j+0.5}{n_j+0.5} \tag{3}$$

と設定した場合の(1)式に相当する。ここで，

x_{ij} ：d_i における t_j の出現頻度

$l_i=\sum_{j=1}^{M}x_{ij}$ ：文献長

$\bar{l}=N^{-1}\sum_{i=1}^{N}l_i$ ：文献長の平均

x_{qj} ：質問における t_j の出現頻度

n_j ：t_j の出現文献数

である。なお，Okapi ではパラメータ設定方法にいくつかの変種があり，(2)式はその 1 つである。ベクトル空間モデルも同様に，a_{ij} および b_j を適当に設定することによって(1)式に帰着する[2]。

2.2 フィードバックの目的

得点 s_i は検索モデルによる文献 d_i の適合度についての一種の推定値であるが，これに対する真の値を r_i と表記する。この値はもちろん未知であるが，レレバンスフィードバックを活用して，ある部分集合 X における r_i の値を知ることは可能である。例えば，何らかの方法による第 1 段階の検索結果に対してその利用者が上位 10 件の文献

の適合度を回答したとすれば，この上位 10 件を X，回答された各値を r_i として考えることができる。これに対応して(1)式を，

$$\mathbf{s}_X = f_X(\mathbf{b}) = \mathbf{A}_X\mathbf{b} \qquad (4)$$

のように書き直す。$\mathbf{A}_X$ は X に含まれる n 件の文献に関する行ベクトルを $\mathbf{A}$ から抽出した $n \times M$ 行列であり，同様に $\mathbf{s}_X$ は X に含まれる文献の得点のみから成る n 次元ベクトルである。

結局，フィードバックの目的とは，

$$\tilde{\mathbf{b}} = \arg\min_{\mathbf{b}} \phi(\mathbf{r}_X, f_X(\mathbf{b})) \qquad (5)$$

となるベクトル $\tilde{\mathbf{b}}$ を求めることにほかならない。ここで ϕ は 2 つのベクトル間の距離を測る何らかの関数であり，$\mathbf{r}_X$ は n 件の文献集合 X に対して利用者が与えた適合度の値から成るベクトルである。

2.3 テイラー展開による解法

(5)式を解くために，まず，$f_X(\tilde{\mathbf{b}})$ を初期質問ベクトル $\mathbf{b}$ のまわりでテイラー展開し，

$$f_X(\tilde{\mathbf{b}}) = f_X(\mathbf{b}) + \frac{\partial f_X(\mathbf{b})}{\partial \mathbf{b}^T}(\tilde{\mathbf{b}} - \mathbf{b}) + K$$

とする。次に，(5)式が満たされるように，$f_X(\tilde{\mathbf{b}})$ が $\mathbf{r}_X$ に等しくなると仮定する。この仮定の下に(4)式を使うと，結果的に，

$$\tilde{\mathbf{b}} = \mathbf{b} + \mathbf{A}_X^{-1}(\mathbf{r}_X - \mathbf{s}_X) \qquad (6)$$

を得る[2]。ここで $\mathbf{A}_X^{-1}$ は $\mathbf{A}_X$ の逆行列を意味するが，$\mathbf{A}_X$ は正方行列ではないので普通の逆行列ではない。このために特異値分解を利用する。つまり，

$$\mathbf{A}_X^T = \mathbf{U}\Lambda\mathbf{V}^T$$

と分解すれば，(6)式を

$$\tilde{\mathbf{b}} = \mathbf{b} + \mathbf{U}\Lambda^{-1}\mathbf{V}^T(\mathbf{r}_X - \mathbf{s}_X) \qquad (7)$$

とかくことができる[2]。これが岸田[2]による提案手法の最終形である。

3. 検索実験による実証分析

3.1 分析の目的と使用データ

分析の主目的は，岸田[2]による手法が確率型検索モデルの例である Okapi の手法の検索性能を実際に向上させることを確認することにある。同時に，ベクトル空間型モデルに対する適用も試みる。

使用するデータは日本語テストコレクション NTCIR-1[4]である。これは学会発表の標題や抄録から成る約 33 万件のレコードから構成されている。今回は No.31～83 までの 53 件の検索課題を使う。なお，検索課題は<title>と<description>フィールドのみを用いる（すなわち，「短い検索質問」）。

3.2 実験の手順

実験の手順は次のとおり。

①第 1 次の検索を Okapi(1)～(3)式と，ベクトル空間モデルで実行する。実行結果をそれぞれ ORGPRB，ORGVEC と表記する。

②各実行結果の上位 n 件の適合／不適合の情報を適合判定ファイルから抽出し，提案手法を実行する。それらの結果を Okapi については TYLPRB，ベクトル空間型については TYLVEC とかく。また，ベクトル空間型に対しては Rocchio の方法も実行し，ROCCHI と表記する。

なお，②でのフィードバックに利用する文献数は $n = 10$，$n = 20$ の 2 通りとする。

また，日本語テキストからの語句の自動識別方法は岸田[2]での実験と同じである。すなわち，機械可読辞書との最長一致で語句を切り出し，その上で隣接する 2 つの語句を複合語として機械的に組み合わせた。さらに，ベクトル空間型モデルおよび Rocchio の方法の詳細についても，岸田[2]での実験と同様である。

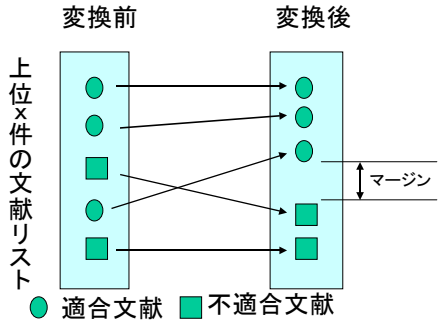

図 1　$\mathbf{r}_X$ の計算方法

3.3 適合判定結果の変換

NTCIR-1 では，適合判定結果を最終的には適合／不適合の 2 値に直して，適合判定ファイルを提供している。したがって，岸

田[2]の手法を適用するには，2 値での判定結果を $\mathbf{r}_X$ に変換しなければならない。その変換法の基本的な考え方を図 1 に示す。

図 1 の左側は変換前の上位 x 件の文献リストであり，第 1 次検索の得点 s_i の順序で並べられている。このリストを一次関数

$$\hat{r}_i = As_i + B \tag{8}$$

によって計算される $\hat{r}_i$ で並べ替え，すべての適合文献が不適合文献よりも上位に位置づけられるようにする。この $\hat{r}_i$ を r_i の推定値とする。

(8)式のパラメータ A と B は，

$$\begin{cases} r^1_{\max} = As^1_{\max} + B \\ r^1_{\min} = As^1_{\min} + B \end{cases} \tag{9}$$

の連立方程式の解として求めることができる。ここで，$r^1_{\max}$ と $r^1_{\min}$ はそれぞれ適合文献集合における r_i の最大値と最小値，$s^1_{\max}$ と $s^1_{\min}$ は適合文献集合中における s_i の最大値と最小値である。(9)式を解くと，

$$\hat{A} = (r^1_{\max} - r^1_{\min}) / (s^1_{\max} - s^1_{\min}) \tag{10}$$

$$\hat{B} = (s^1_{\max} r^1_{\min} - r^1_{\max} s^1_{\min}) / (s^1_{\max} - s^1_{\min}) \tag{11}$$

である。

以上の(9)〜(11)式は適合文献についての $\hat{r}_i$ を求めるための方程式にすぎないが，不適合文献に対しても同様に方程式を設定することができる。すなわち，各最大値・最小値をそれぞれ不適合文献における値に置換すればよい。これを $r^0_{\max}$ ，$r^0_{\min}$，$s^0_{\max}$ $s^0_{\min}$ と表記する。

(10), (11)式中の s_i についての最大値と最小値は第 1 次の検索結果から自動的に決まる。一方，r_i についてはあらかじめ設定しておく必要がある。

まず，ベクトル空間モデルの場合には，照合関数として利用される余弦の性質から，$r^1_{\max} = 1.0$ および $r^0_{\min} = 0.0$ とするのが合理的であろう。次に，例えば「マージン」（図 1 参照）$r^1_{\min} - r^0_{\max}$ を 0.2 に設定すれば，自動的に，$r^1_{\min} = 0.6$ および $r^0_{\max} = 0.4$ のように決まる。本稿ではこの数値を用いる。

一方，Okapi の場合には，r_i の上限と下限を設定するのが難しく，本稿では，$r^1_{\max} = 2 \times s^1_{\max}$，$r^1_{\min} = s^1_{\max}$，$r^0_{\min} = 0.0$ とする。Okapi による s_i は負になる可能性もあるが，ここでは 0.0 を最小値と考えた。$r^0_{\min}$ については第 1 次の検索結果の s_i のレインジ（範囲）の 1/2 を s_i の最小値に加算したものとする。つまり，

$$r^0_{\min} = \min(s^1_{\min}, s^0_{\min}) + [\max(s^1_{\max}, s^0_{\max}) - \min(s^1_{\min}, s^0_{\min})]/2$$

である。

3.4 実験結果

NTCIR-1 の日本語レコードの総数は 332,918 件，平均文献長（$\bar{l}$）は 118.0 であった。

各実行結果の平均精度の平均（mean average precision）を表 1 に示す。まず，第 1 次検索については，Okapi の方法で 0.268，ベクトル空間型で 0.228 であり，Okapi の方がわずかに上回った。表 1 ではこれらをベースライン（基準）として，フィードバックの各手法が平均精度の平均を増加させた割合を示してある。

全般的には，上位 10 件よりも，上位 20 件の適合／不適合の情報を使ったほうが平均精度が高い。次に，Okapi に着目すると，上位 10 件で 0.396，上位 20 件で 0.450 と

表 1 各実行結果の Mean Average Precision **(53 検索課題)**

	確率型（Okapi）		ベクトル空間型		
	ORGPRB	TYLPRB	ORGVEC	ROCCHI	TYLVEC
第一次検索 (ベースライン)	.268	-	.228	-	-
フィードバック 上位 10 文献	-	.396 (+48.0%)	-	.376 (+65.2%)	.378 (+66.3%)
フィードバック 上位 20 文献	-	.450 (+68.1%)	-	.434 (+90.4%)	.459 (+101.6%)

いう数値が得られている。これは，ベクトル空間型モデルでの Rocchio の手法をわずかに上回る数値である。このことから岸田[2]の手法を確率型検索モデルに適用する試みはいちおう成功したといえる。

しかしベースラインからの増加率は，Okapi の場合，48.0%と 68.1%にすぎず，ベクトル空間型に岸田[2]の手法を適用した場合の 66.3%，101.6%に劣る。この結果，ベースラインは Okapi のほうが高いにもかかわらず，上位 20 件のフィードバックの結果はベクトル空間型が上回ることになっている。

4.考察とまとめ

今回の実験では 2 値的な適合判定のデータを使って，岸田[2]の提案手法が確率型検索モデルに実際に適用可能であることを示した。しかし，その増加率は，ベクトル空間型モデルに適用した場合に比べて，それほど高くはなかった。この原因としては，$\hat{r}_i$ の設定方法の問題が考えられる。

もし，利用者から各文献の適合の程度が数値によって与えられれば，r_i の推定という操作は不要である。今回の実験では 2 値の適合判定結果のみが使用可能という状況から，この推定が必要になったのにすぎない。しかし，現実的には，適合の程度を合理的に数値に置き換える作業はそれほど容易ではなく，2 値での適合判定のほうがむしろ現実的という状況も多いであろう。とすれば，本稿における r_i の推定方法はむしろ本質的な役割を果たすことになる。

本稿のこの方法の重要な点は r_i の推定に第一次検索での得点 s_i を利用していることにある。この結果として s_i がフィードバック情報の中に含まれてくる。Rocchio の方法では s_i は使われないので，使用するフィードバック情報という観点からは，この点が岸田[2]の手法と Rocchio の方法との大きな相違ということになる。直感的には，得点 s_i を使用したほうが望ましい。例えば，適合文献集合中で同程度の重みの平均を持つ 2 つの語 A，B があったとして，もし語 A のほうが得点の高い適合文献により多く出現していたならば，語 A は語 B よりも重要であると予想される。

このように考えると，岸田[2]の方法における r_i の推定は，決して便宜的なものではなく，むしろ本質的な役割を担っていることになる。すでに述べたように，今回，Okapi での改善の程度が低かったということはこの r_i の推定方法が不十分だった可能性がある。この点の改善が今後の大きな課題である。

謝辞 貴重なテストコレクションを準備され，研究上の使用を許可された学術情報センター（現：国立情報学研究所）の皆様に感謝します。

参照文献

[1]Rocchio, J.J. Jr. “Relevance feedback in information retrieval”. *The SMART Retrieval System: Experiments in Automatic Document Processing*. G. Salton ed., Englewood Cliffs, New Jersey, Prentice-Hall, 1971, p.313-323.

[2]岸田和明. “文献の適合度に関する目標値に基づくフィードバック手法”.『情報処理学会研究報告（2001-FI-61,2001-NL-142）』 Vol.2001，No.21，p.189-196 (2001).

[3]Robertson, et al. “Okapi at TERC-3”. *Overview of the Third Text Retrieval Conference (TREC-3)* D.K. Harman ed., Gaithersburg, MD, National Institute of Standards and Technology, 1995, p.109-126.

[4]http://research.nii.ac.jp/~ntcadm/index-ja.html