適合フィードバックによる文書検索

国立情報学研究所　山田 誠二
seiji@nii.ac.jp

(財)電力中央研究所　小野田 崇
onoda@criepi.denken.or.jp

はじめに

大量のデータから有用な知識を抽出するデータマイニングが盛んに研究開発されている。このような大量のデータを扱う情報処理は、今後計算機の処理速度の向上とともに益々重要かつ身近になっていくことが予想される。一方、情報検索の分野においても大量のデータからいかに欲しい情報を取り出すかが研究されてきた。その中のひとつに、膨大な文書集合から所望の文書を検索する文書検索がある。

文書検索は、通常はユーザがクエリ（検索キーワード）を正確に記述して、そのクエリをもとに検索を行い、ユーザに検索結果を提示するが、一回の検索で所望の文書をたくさん集めることは困難である。これは、主にユーザが正確で十分な情報をもつクエリを記述することは容易ではないことによる。

よって、一度集まった文書をユーザが評価し、その評価を基に検索を修正していく必要がある。このような枠組みは、適合フィードバックと呼ばれ、興味のある文書をできるだけ多く集めたい場合に、特に有効な対話的文書検索の手法として様々な研究がなされてきた。

本稿では、対話的に文書を収集する枠組みである適合フィードバックを概観し、筆者らが行った適合フィードバックに関する最近の研究である関係学習による適合フィードバックとサポートベクターマシンによる適合フィードバックを紹介する。

適合フィードバック

適合フィードバックの具体的な枠組みの説明に入る前に、その基礎となるベクトル空間モデルについて説明する。通常、文書検索において、各文書はそれを特徴づけるベクトルで表現される場合が多い。そのようなベクトルを文書ベクトル（あるいは、特徴ベクトル）と呼ぶ。このように、文書をベクトルで特徴付ける手法を、ベクトル空間モデルと呼ぶ。もちろん、ベクトル空間モデルは、文書検索における文書の表現の一つであり、他にもブーリアンモデル、確率モデルなどが提案されている[1][2]。

ベクトル空間モデルにおける文書ベクトルは、以下の TFIDF 法[1][2]により決定される。今文書集合 D が与えられたとして、その中の各文書を $d_1,\ldots, d_N$ とし、ある文書 d 中における語 t の出現頻度を $tf(t,d)$ で表す。ここで、「語」とは単語から接尾語などを取り除いたものである。また、D において語 t の出現する文書の数を $df(t)$とする。このとき、文書 d の文書ベクトル v_dは、以下のよう定義される。なお、この文書ベクトルの次元は、D の全文書中に含まれる重複しないすべての語で構成され、その次元数を m とする。

$$v_d = (w_{t_1}^d, w_{t_2}^d, \cdots, w_{t_m}^d)$$

$$w_t^d = tf(t,d) \cdot idf(t)$$

$$idf = \log \frac{N}{df(t)} + 1$$

ここで、N は全文書数である。上式は、ある文書にだけ多く出現し、他の文書にはあまり出現しない語のベクトル値を大きくして、強調するという定性的な意味がある。この TFIDF 法による文書ベクトルの重み付けは、広く文書検索で使われている。

また、文書だけでなく、クエリ自身も文書ベクトルで記述でき、クエリベクトルと呼ばれる。具体的には、クエリ中の語が対応する次元のベクトル値を 1 に、他の次元のベクトル値を 0 にした文書ベクトルである。

このベクトル空間モデルにおいて、２つの文書間の類似度は、それらの文書ベクトルの余弦により定義される。つまり、文書ベクトルの方向が近いものほど、類似していると解釈する。

ベクトル空間モデルに基づく文書検索は、まず文書集合中の文書全てを文書ベクトルで記述する。そして、与えられたクエリをクエリベクトルに変換し、そのクエリベクトルと各文書ベクトルとの類似性を余弦で評価する。最後に、類似度の高い順にソートしたものを適合文書の候補リストとして、ユーザに提示する。

ただし、通常は、ユーザがクエリを正確かつ十分詳細に記述することは難しいので、得られた候補リストは、非適合な文書を多くその上位に含んでいる場合が多い。もし適合文書を一つだけ見つけたい場合にはそれでもいいが、できるだけたくさん見つけたい場合には、一回の検索では難しい。なお、適合文書、非適合文書とは、ユーザの所望の文書とそうでない文書を意味する。

ユーザがクエリを正確に記述することは難しいが、文書を見せられて、それが自分にとって適合文書であるか、非適合文書であるかを判定することは、一般には難しくない。よって、もし検索結果の文書が適合文書か、非適合文書であるかをユーザに評価してもらえば、検索システムがその評価を利用して、さらに精度の高い検索を行うことが可能になる。このような枠組みが、適合フィードバック(relevance feedback)である。

適合フィードバックの実現には、いくつかの方法

があるが、典型的なものの一つを以下に説明していく。この手続きは、後述する対話的文書検索においても共通して用いられる。また、[U]と[S]はそれぞれユーザとシステムが行う処理であることを示している。

< 適合フィードバックの手続き >

1 [U] ユーザがクエリを入力。
2 [S] クエリベクトルを生成し、初期検索を行って結果を得る
3 [U] ユーザが検索結果の上位 N 個の文書を評価し、適合文書集合 D^+と非適合文書集合 D^-に分ける。
4 [U] ユーザが十分な適合文書が得られたと判断したら検索終了。不十分な場合は、次へ。
5 [S] D^+と D^-を使って、クエリベクトルを修正する。
6 [S] 修正されたクエリベクトルで、再検索をおこない、Step 3 へ。

一回の検索結果でユーザの評価する文書の数 N は、20〜40 に設定される。ここで重要なのは、Step 5 において、いかにクエリベクトルを修正するかである。このクエリベクトルの修正は、さまざまな方法が提案されているが、よく使われるのは下の Rocchio の式と呼ばれるものである[3]。

$$Q_{i+1} = Q_i + \frac{1}{|D^+|}\sum_j q_j^+ - \frac{1}{|D^-|}\sum_j q_j^-$$

上式において、Q_i は i 回目の検索におけるクエリベクトルであり、Q_{i+1} はそれを修正したクエリベクトルであり、$|D^+|, |D^-|$ はそれぞれ、i 回目の検索結果をユーザが評価したときの適合文書数、非適合文書数を意味する。q^+_j, q^-_jはそれぞれ、個々の適合文書、非適合文書に対応するクエリベクトル（文書ベクトルで、クエリ中の語の値を残し、他を 0 にしたもの）である。

この式は、直観的には、適合文書に含まれる語の重みはより大きく、非適合文書のそれはより小さくなるようにクエリベクトルを修正している。

以上が適合フィードバックの基本的枠組みである。クエリベクトルの修正方法を中心に、これまで様々な研究がされているが、その中でも、適合フィードバックに分類学習を応用した最近の２つの研究を以降で紹介する。

関係学習による適合フィードバック

適合フィードバックの実現方法として、現在最も良く使われるのは、ベクトル空間モデルによって表現された検索質問を、統計情報を使って自動的に修正する前節で説明した手法である。この方法の利点としては、適合度順に文書をランキングすることができる、実現が容易であるといった点が挙げられるが、表現としては次のような限界も存在する。まず、ベクトル空間モデルでは語の独立性が仮定されているため、語間の近接関係を表現することが難しい。また、同じ適合文書でもそれぞれ注目すべき単語やその組み合わせは違うと考えられるが、単一ベクトルでそれらを表現するのは無理がある。これらの情報は、いずれも文書を特徴づけるのに役立つものである。

そこで、ベクトル空間モデルでは表現できない情報を、関係学習の一手法である帰納論理プログラミング ILP （Inductive Logic Programming）[4]の手法により文書の判別ルールとして獲得する方法が提案されている[5]。得られたルールに適合する文書から優先的に順位付けを行うことで、適合文書を効率的に集めることができることを目指したものである。

前述の適合フィードバックのステップ 6 を以下の 6, 7 のステップに置き換えることにより、分類学習アルゴリズムを適合フィードバックに利用することが可能になる。分類学習(classification learning)とは、あるクラスに属する正例と属さない負例を与えられ、それらを元に判別ルールや判別関数を帰納的に学習する教師あり機械学習アルゴリズムである。本研究では、ILP をその分類学習アルゴリズムとして用いる。また、次節で紹介する対話的文書検索では、判別関数を学習するサポートベクターマシンを分類学習アルゴリズムとして用いている。

6 [S] 適合文書集合 D^+と非適合文書集合 D^-を用いて、分類学習アルゴリズムにより、判別ルール（あるいは、判別関数）を作る。
7 [S] 全文書を検索ルールを満たす文書集合 A と満たさない文書集合 B に分け、それぞれの文書集合中の文書に検索ベクトルを用いて順位を付ける。A を上位集合、B を下位集合としたものを検索結果として返し、Step3 へ。

検索ルールの分類学習を組み込んだ対話的文書検索の手続きの概念図を図１に示す。また、本研究では、手続き上には明記されていないが、クエリのキーワードを追加するクエリ拡張も使っている。

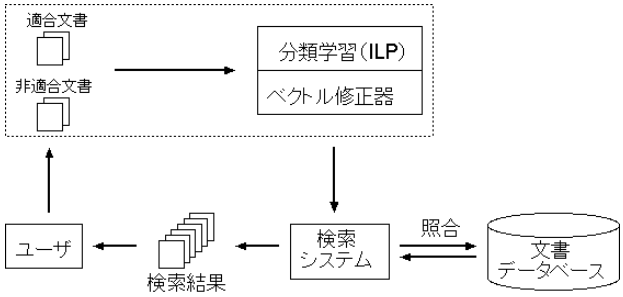

図１ 関係学習による適合フィードバック

検索ルールの生成は、適合文書を正例文書、非適合文書を負例文書とした分類学習問題として扱う。検索ルールは、ホーン節で表現する。また、ルールのボディ部は、以下の述語を用いて構成される。

- ap(A, word) : word が文書 A に現れる。
- near(A, word1, word2) : word1 と word2 が文書 A に現れ、順不同で 5 単語以内に近接して存在。

near 関係は単語間の近接関係を表すもので、共起関係よりも制約の強い条件となる。ILP では分割統治的に仮説を生成することが多く、その結果正例集合は複数の仮説で被覆される。本研究でもこの戦略を用いているため、検索ルールは多くの場合複数のルールから構成される。したがって以下のように 2 つのルールによって構成される場合、適合文書は 2 つのルールの内、どちらかを満たせば良いことになる。

rel(A) :- ap(A, mammal), near(A, species, protect).
rel(A) :- ap(A, species), near(A, mammal, protect).

判別ルール学習アルゴリズムの詳細[5]は割愛するが、判別ルールを生成するための手続きは、ルールを一つずつ生成し、ルール集合に追加する作業を繰り返す。新しいルールが一つ生成されると、それによって被覆される文書が正例文書集合から取り除かれるので、ルールが生成される度には減少していき、最終的に空集合となれば手続きが終了となる。また、ルールは空のボディ部にリテラルを一つずつ、追加していき、負例を一つも含まなくなると完成となる。追加するリテラルは、条件候補リテラル集合の中から選ばれるが、その際の評価基準には、重み付け情報利得を用いる。

情報利得を用いた探索は、効率的である反面、山登り法であるため、探索が行き詰まることがある。本手法では、このような場合にバックトラックを行う。バックトラックが行われるのは、負例をまだ含んでいるのに選択できるリテラルがなくなったとき、つまり、過去のいずれかの時点に戻ってリテラルの選択をやり直す必要が生じた場合である。

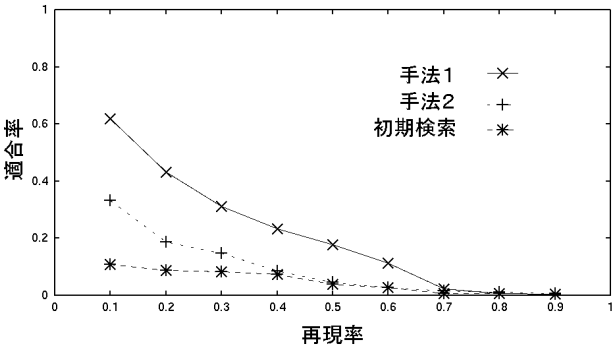

図 2 関係学習による対話的文書検索システム

検索対象用の文書データベースとして、文書検索の分野でテストベッドとして広く使われている TREC（Text REtrieval Conference）[6]が提供するデータベースの中から英字新聞記事（The Los Angels Times、 約 13 万記事、1 記事当りの平均単語数 526 語）を使って、評価実験を行った。その結果の再現率‐適合率曲線[2]のグラフ（４回目のフィードバックにおける平均値）を図２に示す。この図で、手法１は提案手法であり、手法２は従来の適合フィードバックである。図からわかるように、提案手法がもっとも高い性能を示している

サポートベクターマシンによる適合フィードバック

本節では、図１における分類学習器（分類学習アルゴリズム）として、ILP の代わりに、高性能な判別関数の学習アルゴリズムとして近年注目されているサポートベクターマシン(以下、SVM)を用いた研究[7]を紹介する。なお、システムの処理手続きは、前節のものと同じである。

まずは、SVM について概観する。学習サンプル $(\mathbf{z}_1, y_1),\ldots,(\mathbf{z}_l, y_l), \mathbf{z}_i \in F, y_l \in \{\pm 1\}$ が与えられ、次式を満たす判別関数 $f_{w,b} = \mathrm{sgn}((w \cdot z) + b)$ を推定する問題を考える。

$$f_{\mathbf{w},b}(\mathbf{z}_i) = y_i, \quad i = 1,\ldots,l \tag{1}$$

ここで、学習サンプル $(\mathbf{z}_i, y_i)$ は次式のように、入力空間での学習サンプル $(\mathbf{x}_i, y_i)$ を高次元特徴空間上に写像したサンプルであるとする。

$$\mathbf{z}_i = \Phi(\mathbf{x}_i) \tag{2}$$

式(1)で表現される判別関数が存在する場合に以下の制約を考える。

$$y_i \cdot ((\mathbf{z}_i \cdot \mathbf{w}) + b) \geq 1, \quad i = 1,\ldots,l \tag{3}$$

$(\mathbf{w}, b), (-\mathbf{w}, -b)$ のように $\mathbf{w}$ と b の方向の違いにより、同じ超平面判別関数が 2 つ存在することとなる。しかし、式(1)と式(3)によって判別関数は一意に定めることができる。

汎化能力の高い判別関数は式(3)で表現される制約条件の下、次式を最小化することで推定できる[8]。

$$\tau(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}\|^2 \tag{4}$$

この凸最適化問題を解くため、式(4)の Lagrangian を計算すると

$$L(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{l} \alpha_i (y_i ((\mathbf{z}_i \cdot w) + b) - 1), \tag{5}$$

ここで、$\alpha_i \geq 0$ は Lagrange 乗数である。この Lagrangian を α_i について最大化し、$\mathbf{w}$ と b について最小化する。パラメータ $\mathbf{w}$ と b における L の導関数は、鞍点において次式を満たす。

$$\frac{\partial}{\partial b} L(\mathbf{w}, b, \boldsymbol{\alpha}) = 0, \quad \frac{\partial}{\partial \mathbf{w}} L(\mathbf{w}, b, \boldsymbol{\alpha}) = 0. \tag{6}$$

式(6)から次式が成立する。

$$\sum_{i=1}^{l} \alpha_i y_i = 0, \tag{7}$$

$$\mathbf{w} = \sum_{i=1}^{l} \alpha_i y_i \mathbf{z}_i \tag{8}$$

結局、$\mathbf{w}$ は学習サンプルの展開式となる。$\mathbf{w}$ の解はただ一つに決まるが、Lagrange 乗数 α_i はその必要が

ない。式（3）を表現し直した次式の制約条件に対して非ゼロでなくてはならない。

$$\alpha_i \cdot [y_i((\mathbf{z}_i \cdot \mathbf{w}) + b) - 1] = 0, \quad i = 1, \cdots, l. \quad (9)$$

$\alpha_i > 0$ を有するパターン $\mathbf{z}_i$ をサポートベクターと呼ぶ。サポートベクター以外の学習サンプルは凸最適化問題の解法には関係のないものとなる。つまり、サポートベクター以外の学習サンプルは式（3）の制約条件を自動的に満たし、式（8）の展開項には現れないのである。

式(5)の Lagragian に式(7)、式(8)の条件を代入すると、元の最適化問題の双対問題となる次の凸最適化問題を得ることができる。

$$\max_{\alpha} \sum_{i=}^{l} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{l} \alpha_i \alpha_j y_i y_j (\mathbf{z}_i \cdot \mathbf{z}_j)$$
$$\text{subject to } \alpha_i \geq 0, \ \ i = 1, \ldots, l, \ \ \sum_{i=1}^{l} \alpha_i y_i = 0. \quad (10)$$

式(8)の展開式を判別関数の式(1)に代入することによって、式(1)の判別関数を、分類されるサンプルとサポートベクターとの内積で評価される次式に書き換えることができる。

$$f(\mathbf{z}) = \text{sgn}\left(\sum_{i=1}^{l} \alpha_i y_i (\mathbf{z} \cdot \mathbf{z}_i) + b \right) \quad (11)$$

以上より、式(10)で表現される凸 2 次計画問題を解くことで、判別関数 $f_{\mathbf{w},b}(\mathbf{z}) = \text{sgn}((\mathbf{w} \cdot \mathbf{z}) + b)$ を得ることができる。式(10)の 2 次計画問題を解き、高次元空間上で線形判別関数を求める方法を SVM と呼ぶ。SVM によって得られる判別関数の例を図 3 に示す。○(適合)と●(非適合)は各々異なるラベルを有する学習サンプルを表す。図中、破線上の学習サンプルが、$\alpha_i > 0$ を有するサポートベクターである。

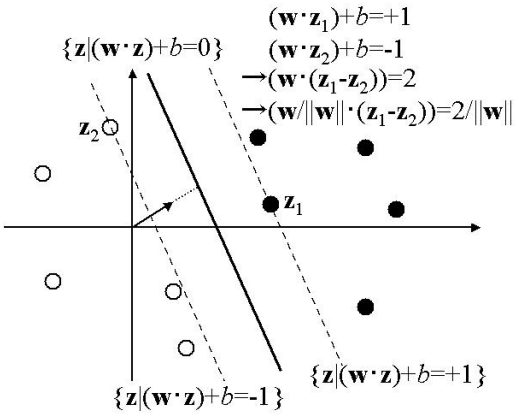

図 3 SVM の例

上述の SVM を適合フィードバックに適用した結果について述べる。適用データとして、文書検索に関する国際会議 TREC で広く使用されているデータの中の英字新聞記事(前節で使ったものと同じ)を使用した。このデータには、検索要求文とその要求に適合する文書集合が提供されている。

文書ベクトルは、文献[9]を参考に TFIDF で算出し、その文書ベクトルを SVM への入力サンプルとした。一般に SVM で判別関数を決定する際、カーネルトリック[10]を用いる方法が使われるが、我々が扱う文書ベクトルは非常に語数が多いので、学習サンプルを高次元特徴空間へ写像して分離できるようにする必要がない。そこで、文書ベクトル空間上での線形分離により判別関数を決定した。SVM の学習には、LibSVM を使用した。

SVM の学習に用いる文書数を 20 文書、すなわちユーザが評価を行う文書数を 20 文書とした検索実験を行った。適用結果を図 2 に再現率-適合率曲線で示す。図 2 は、SVM による学習を 4 回行った後の結果を示している。つまり、ユーザが 4 回 20 文書の評価を行った後の結果である。各回にユーザが評価する 20 文書は、SVM が決定した判別関数からの距離が遠く、適合領域に入る上位 20 文書で構成されている。図 4 には、SVM による適合フィードバック(太実線)の能力を比較するため、 Rocchio-based フィードバック手法(点線：従来手法)とフィードバックをしない場合(細実線)の結果も示した。

図 4 より、SVM によるフィードバック手法は、フィードバックを行わない場合よりも検索性能が高いことがわかる。また、従来の Rocchio-based フィードバック手法と比較しても、検索性能が高いことがわかる。

また、表 1 にフィードバック回数と SVM によるフィードバック手法および Rocchio-based フィードバック手法の平均適合率を示す。表 1 より、Rocchio-based フィードバック手法は、フィードバック数が増加しても平均適合率の向上が望めない。しかし、SVM によるフィードバック手法は、フィードバック回数の増加に伴い、平均適合率が向上していることがわかる。

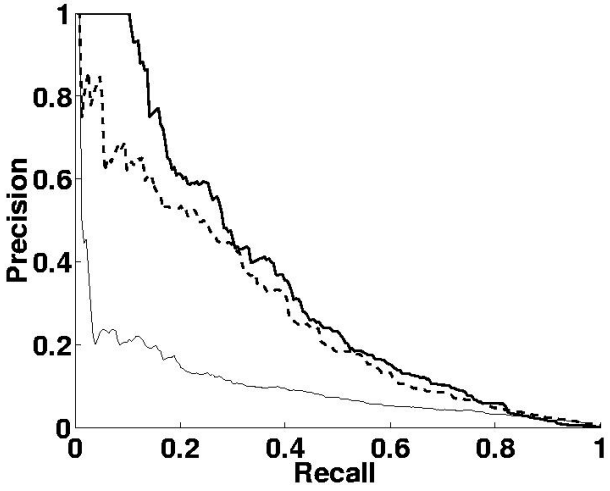

図 4 再現率-適合率曲線:太実線が SVM、点線が Rocchio、細実線がフィードバックなしを表す。

表 1 フィードバック回数と平均適合率

フィードバック回数	平均適合率	
	Rocchio	SVM
1	0.225	0.263
2	0.250	0.350
3	0.235	0.613
4	0.235	0.638

まとめ

大量の文書からユーザの欲しい文書をできるだけ多く検索するための、対話的文書検索の枠組みである適合フィードバックについて、その基本的手続きから、分類学習を適用した先端的な研究例までを解説した。適合フィードバックでは、クエリベクトルの修正によりある意味で分類学習が実現されているが、その分類学習アルゴリズムを、記号ベースの機械学習である関係学習と連続値ベースの優れた分類学習アルゴリズムであるサポートベクターマシンに変更したシステムの研究例を紹介し、先端的な研究の方向性を示した。

今後、大規模文書からの文書検索がより広い応用範囲をもつにつれて、対話的に文書検索を行う適合フィードバックはますます重要になると考えられる。本稿により、対話的文書検索、適合フィードバックについて読者の理解が深まれば幸である。

参考文献

[1] 徳永健伸；情報検索と言語処理, 東京大学出版 (1999).

[2] R.B. Yates, B.R. Neto; Modern Information Retrieval、, Addison Wesley (1999).

[3] J.J. Rocchio; Relevance feedback in information retrieval, 313-323, The Smart system - experiments in automatic document processing, Prentice Hall Inc. (1971).

[4] 古川康一, 尾崎知伸, 植野 研; 帰納論理プログラミング, 共立出版 (2001).

[5] 岡部正幸, 山田誠二；関係学習を用いた対話的文書検索，人工知能学会誌，Vol.16, No.1, F (2001).

[6] Voohees, E.M., Harman, D.: Overview of the Seventh Text REtrival Conference(TREC-7), Proceedings of the Seventh Text REtrieval Conference, NIST Special Publication (1999).

[7] 小野田崇，村田博士，山田誠二：情報検索における能動学習，第 128 回情報処理学会「知能と複雑系」研究会，93-98 (2002).

[8] V. Vapnik, The Nature of Statistical Learning Theory, Springer (1995).

[9] R. Schapire, Y. Singer, A. Singhal, Boosting and Rocchio Applied to Text Filtering, Proceedings of the Twenty-First Annual International ACM SIGIR, pp. 215-223 (1998).

[10] 小野田崇, Introduction to Large Margin Classifiers, 人工知能学会誌, Vol.17, No.1, pp. 21-30 (2002).

やまだ せいじ YAMADA, Seiji

Web からの情報収集、ヒューマンエージェントインタラクション、知能ロボットなどを幅広く研究している。

連絡先 〒101-8430 東京都千代田区一ツ橋 2-1-2　国立情報学研究所

電話 03-4212-2562　URL http://research.nii.ac.jp/~seiji/

おのだ たかし ONODA, Takashi

機械学習全般に興味を持ち，学習アルゴリズムの開発や学習アルゴリズムの数理的側面の研究を行っている．また，学習アルゴリズムの様々な分野への適用研究も進めている．

連絡先 〒201-8511 東京都狛江市岩戸北2-11-1　(財)電力中央研究所 情報研究所

電話 03-3480-2111