サポートベクターマシンを用いた
対話的文書検索

村田 博士

博士 (情報学)

総合研究大学院大学
複合科学研究科
情報学専攻

平成 23 年度
(2011)

2012 年 3 月

本論文は総合研究大学院大学複合科学研究科情報学専攻に
博士 (情報学) 授与の要件として提出した博士論文である.

審査委員会

主査	山田 誠二 教授	総合研究大学院大学 国立情報学研究所
	相澤 彰子 教授	総合研究大学院大学 国立情報学研究所
	市瀬 龍太郎 准教授	総合研究大学院大学 国立情報学研究所
	小野田 崇 連携教授	東京工業大学 電力中央研究所
	佐藤 健 教授	総合研究大学院大学 国立情報学研究所

(主査以外は 50 音順)

Interactive Document Retrieval Based on Support Vector Machines

Hiroshi Murata

DOCTOR OF
PHILOSOPHY

Department of Informatics
School of Multidisciplinary Sciences
The Graduate University for Advanced Studies (SOKENDAI)

March, 2012

A dissertation submitted to
the Department of Informatics,
School of Multidisciplinary Sciences,
The Graduate University for Advanced Studies (SOKENDAI)
in partial fulfillment of the requirements for
the degree of Doctor of Philosophy

Advisory Committee

Prof. Seiji Yamada (Chair)	National Institute of Informatics/ The Graduate University for Advanced Studies
Prof. Akiko Aizawa	National Institute of Informatics/ The Graduate University for Advanced Studies
Assoc. Prof. Ryutaro Ichise	National Institute of Informatics/ The Graduate University for Advanced Studies
Visiting Prof. Takashi Onoda	Central Research Institute of Electric Power Industry / Tokyo Institute of Technology
Prof. Ken Sato	National Institute of Informatics/ The Graduate University for Advanced Studies

(Alphabet order of last name except chair)

内容梗概

近年の情報技術の発展に伴い，個人で扱えるテキストデータの量が急激に増加している．このような状況で，膨大なテキストデータから必要な情報を検索する機会も増え，情報検索，特に文書検索に対する期待が高まっている．文書検索では，一つの適合文書を見つければよいタスクもある一方， システムから提示された文書をユーザが評価する負荷をおってでも，できるだけ多くの適合文書を獲得したいタスクも多い．本研究で扱う検索タスクは後者であり，このようなユーザから対話的にフィードバックをかける情報検索システムは，その有効性の検証とともにさまざまな研究が展開している．

一般に， ユーザが検索意図を記述したクエリ（検索キーワード）による一回だけの検索により，多くの適合文書を獲得することは容易でない．そのため，ユーザからのフィードバックを利用して検索を繰り返すことで，できるだけ多くの適合文書を検索することが現実的である．このようなユーザからのフィードバックを利用する手法として，検索結果である文書をユーザに提示して，ユーザが適合，非適合の判定を行い，その判定結果をもとに，より精度の高い再検索を行うことを繰り返す適合フィードバック（relevance feedback）がある．

この適合フィードバックを対話的分類学習として捉え，現在最も性能の高い分類学習アルゴリズムの一つであるサポートベクターマシン (SVM : Support Vector Machines) を適用する方法が提案されている．先行研究として，分類学習としてサポートベクターマシンを適用し，比較実験により検索性能が向上することを示した研究，特に適合文書が文書データベース中に少ない場合に，サポートベクターマシンベースの対話的文書検索が有効であることを確認した研究などがあるが，これらの先行研究ではサポートベクターマシンを用いた適合フィードバックにおける提示文書の選択に対する詳細な研究は行われていない．

そこで本研究では，サポートベクターマシンを用いた適合フィードバックにおける対話的文書検索での検索性能と学習性能をともに向上させる，

ユーザへ提示する文書の選択のためのヒューリスティクスを提案し，それにより単純な文書提示や従来の適合フィードバックを凌ぐ性能向上が達成されることを実験的に示す．この文書提示のヒューリスティクスは，大規模の文書データにおける正データと負データの極端な偏りに基づくもので，サポートベクターマシンのサポートベクターを効率よく集めることができる能動学習を実現している．

文書検索のテストベッドとして広く使われている国際会議TREC（Text REtrieval Conference）のデータセットを用いて，従来の適合フィードバックで用いられるRocchioの手法と，従来のサポートベクターマシンにおける能動学習手法による提示文書選択方法について，提案手法との比較実験を行った．その結果，提案手法は，従来手法に比べて検索性能，学習性能とも同等あるいはそれ以上のパフォーマンスであることが示された．

また，このときの文書提示における，提示文書の順位付けは，その文書が適合文書にどれだけ似通っているかを判定する適合度を計算することで行う．サポートベクターマシンを用いた適合フィードバックのシステムにおいて，この適合度として判別関数と文書ベクトル間の符号付距離が用いられるが，この適合度が従来手法であるRocchioの手法などで用いられているベクトル空間モデル上でどのような特性を持つのかは明らかになっていない．そこで本研究では，サポートベクターマシンにおける距離を用いた適合度を定式化し，対話的文書検索における従来手法であるRocchioの手法との比較分析を行う．

比較分析を行った結果，Rocchioの手法におけるクエリベクトル更新式が，サポートベクターマシンに基づく適合フィードバックの重みベクトルの近似となっていること，そして，Rocchioの手法ではクエリベクトルとのコサイン類似度となっている適合度の計算式が，サポートベクターマシンに基づく適合フィードバックについては，重みベクトルとのコサイン類似度に評価対象文書ベクトルのノルムをかけたものになっていることがわかった．

この比較分析により得られた知見から，類似度が文書ベクトルモデル元来のコサイン類似度に近づく効果のあるカーネルとして，コサイン類似度に対応したコサインカーネルを提案した．提案手法の有効性を検証するために国際会議TRECのデータセットを用いた検証実験を行い，Boolean，TF，TFIDFの文書ベクトル表現について比較した結果，すべてのベクトル表現で性能が向上し，特にTFベクトル表現において，性能が大きく向上することを示した．

Abstract

The amount of text data is rapidly increasing with the development of information technology, and document retrieval is expected to become more sophisticated. The task of finding relevant documents is known as document retrieval. It can also be defined as a task to find as many relevant documents as possible, even if there is a load on the user. This latter task is the focus of our study. Document retrieval systems that use information from interactive user feedback have been studied in many ways.

Since a user generally rarely describes a query precisely on the first attempt, an interactive approach has been proposed to modify a query vector with a user's evaluation of documents in a list of retrieved documents. This method is called relevance feedback and is used widely in information retrieval systems.

Another approach has been proposed in which relevant and irrelevant document vectors are respectively classified as positive and negative examples for a target concept based on classification learning. Some studies have proposed that Support Vector Machines (SVMs) which have an excellent ability to classify input data into two classes be applied to classification learning for relevance feedback. They did not evaluate the useful selection rule for displayed documents at each iteration to a user.

We propose a heuristics which improves learning efficiency and retrieval efficiency in interactive document retrieval for selection of displayed documents to a user. This heuristics is based on the extreme bias between positive and negative example.

We conducted experiments to evaluate the effectiveness of our proposed heuristics for active learning. We use a set of articles which is widely used in the text retrieval conference TREC. For comparison with our approach, two information retrieval methods were adopted. The first is conventional

Rocchio-based relevance feedback. The second is conventional selection rule for SVM-based active learning. Then we confirmed our proposed system outperformed other ones.

Ordering of displayed documents is accomplished by calculation of the degree of relevance in interactive document retrieval. In SVM-based interactive document retrieval, the degree of relevance is evaluated by signed distance from optimal hyperplane. It is not made clear how the signed distance on the SVMs has characteristics in Vector Space Model which is used in Rocchio-based method. We show that SVM-based retrieval has an association with conventional Rocchio-based method by comparative analysis of relevance evaluation.

As a result of their analysis, equation of weight vector of relevance feedback based on SVMs is equivalent to update equation of query vector of Rocchio-based method. The degree of relevance on SVM based method evaluates scalar product of norm of target document vector and cosine similarity of weight vector. On the other hand, the degree of relevance on Rocchio-based method evaluates cosine similarity of query vector.

From this knowledge, we propose a cosine kernel equivalent to cosine similarity that is suitable for SVM-based interactive document retrieval. The effectiveness of a method using our proposed cosine kernel was confirmed, and it was experimentally compared with conventional relevance feedback for the Boolean, term frequency (TF) and term frequency-inverse document frequency (TFIDF) representations of document vectors. The experimental results for a Text Retrieval Conference data set show that the cosine kernel is effective for all document representations, especially TF representation.

目 次

表 目 次

図 目 次

第1章　序論

1.1　本研究の背景と目的

近年の情報技術の発展に伴い，個人で扱えるテキストデータの量が急激に増加している．このような状況で，膨大なテキストデータから必要な情報を検索する機会も増え，情報検索，特に文書検索に対する期待が高まっている．文書検索では，一つの適合文書（relevant document）を見つければよいタスクもある．一方， システムから提示された文書をユーザが評価する負荷をおってでも，できるだけ多くの適合文書を獲得したいタスクも多い [酒井 06]．本研究で扱う検索タスクは後者であり，このようなユーザから対話的にフィードバックをかける情報検索システムは，その有効性の検証 [Koenemann 96] とともにさまざまな研究が展開している [Ingwersen 92].

一般に， ユーザが検索意図を記述したクエリ（検索キーワード）による一回だけの検索により，多くの適合文書を獲得することは容易でない．そのため，ユーザからのフィードバックを利用して検索を繰り返すことで，できるだけ多くの適合文書を検索することが現実的である．

このようなユーザからのフィードバックを利用する手法として，検索結果である文書をユーザに提示して，ユーザが適合文書，非適合文書の判定を行い，その判定結果をもとに，より精度の高い再検索を行うことを繰り返す適合フィードバック（relevance feedback）[Salton 71] がある．

この適合フィードバックを対話的分類学習として捉え，現在最も性能の高い分類学習アルゴリズムの一つであるサポートベクターマシン (Support Vector Machines:SVM) [Vapnik 98][小野田 07a] を適用する方法が提案されている [Drucker 02][柘植 03]．本研究では，ユーザへ提示する文書の選択のためのヒューリスティクスを提案し，それにより単純な文書提示や従来の適合フィードバックを凌ぐ性能向上が達成されることを実験的に示す．この文書提示のヒューリスティクスは，大規模の文書データにおける正データと負データの極端な偏りに基づくもので，サポートベクターマ

シンのサポートベクターを効率よく集めることができる能動学習（active learning）を実現している．

このとき，ある文書の適合文書らしさを表す適合度として判別超平面からの符号付距離を用いている．これまでの研究では，サポートベクターマシンにおける符号付距離がベクトル空間モデル上でどのような特性を持つのかは，明らかではない．そこで本研究では，サポートベクターマシンにおける距離を用いた適合度を定式化し，Rocchio の手法との比較分析を行う．

また，そこから得られた知見より，コサイン類似度により距離関係を表すコサインカーネルを導入することで，文書検索に適した学習方式を提案し，サポートベクターマシンに基づく適合フィードバックの改善を行う．さらに，文書検索用のデータセットを用いて適合フィードバックでの比較実験を行い，検索性能と学習性能における改善の効果について明らかにする．

1.2 本論文の構成と概要

第 2 章「関連研究」では，一般的な文書検索の方法について述べる．次に，本研究で文書検索に導入した学習手法であるサポートベクターマシンと能動学習について述べる．さらに，本研究で使用した適合フィードバックと，その代表的な手法である Rocchio アルゴリズムについて述べ，本研究と関連するトランスダクティブ学習とランキング学習について述べる．

第 3 章「サポートベクターマシンに基づく能動学習による対話的文書検索」では，文書検索にサポートベクターマシンを用いた場合の能動学習が，一般的なサポートベクターマシンの能動学習と異なることから，新たな能動的文書提示のヒューリスティクスを提案している．まず，能動学習による対話的文書検索の方法として，能動的文書提示について説明する．次に，能動的文書提示の方法として提案するヒューリスティクスについて，その根拠とともに述べる．さらに，比較実験を行い，その結果について考察する．

第 4 章「サポートベクターマシンに基づく対話的文書検索の比較分析」では，サポートベクターマシンに基づく対話的文書検索と，一般的な対話的文書検索手法である Rocchio アルゴリズムとの比較分析により，サポートベクターマシンに基づく文書検索の性能を向上させるカーネルを

提案する．まず，サポートベクターマシンに基づく適合フィードバックと Rocchio アルゴリズムの比較分析について述べる．次に，比較分析結果から得られた知見より，性能向上が期待できるコサインカーネルを提案する．さらに，比較実験を行い，その結果について考察する．

最後に第 5 章「結論」で本論文の成果をまとめる．

第2章　関連研究

2.1　文書検索

2.1.1　検索モデル

通常の文書検索では，キーワードをクエリ（質問）として入力することで文書を探し出す．このとき，キーワードと各文書が適合しているかどうか，またどのくらい適合しているのかを計算する必要がある．このような計算は，検索された文書の並び順を決定するランキングアルゴリズムによって行われる．この並び順が上位の文書が，より適合していると判断される．検索モデルはランキングアルゴリズムの前提となるものであり，何が適合して何が適合しないのかを決定する．

情報検索のモデルでは，それぞれの文書が索引語 (index term) と呼ばれる代表的なキーワードのセットによって表されると考える．通常，索引語は単語であり，文書の主要な内容を表している．そのため，索引語は，文書内容の索引や要約に用いられる．一般に索引語は，自身に意味を持つ名詞である．また，形容詞，副詞，接続詞などは，それらが補完的に使用されるため，索引語としては一般に不向きである．文書に対する索引語の集合が与えられたとき，文書内容を表現するためには，すべての語が一様に有用というわけではない．文書の内容を表すためには，語の重要性を定義することが必要となる．たとえば，10 万の文書の集合があるとき，10 万文書すべてに現れる単語は，索引語としてまったく役に立たない．一方，5 文書だけに現れる単語があれば，ユーザが興味を持つ文書の領域を狭めることができるため，有用であるといえる．それゆえに，文書の内容を表現するためには，各索引語の適合性を明らかにする必要がある．これは，各索引語の重みとして表現することができる．

t_i を索引語，d_j を文書，$w_{i,j} \geq 0$ を (t_i, d_j) の重みとする．このとき，$w_{i,j}$ は文書内容に対する索引語の重要性を定量化して表している．ここで，$w_{i,j} = 0$ は，文書 d_j に索引語 t_i が存在しないことを意味する．また，

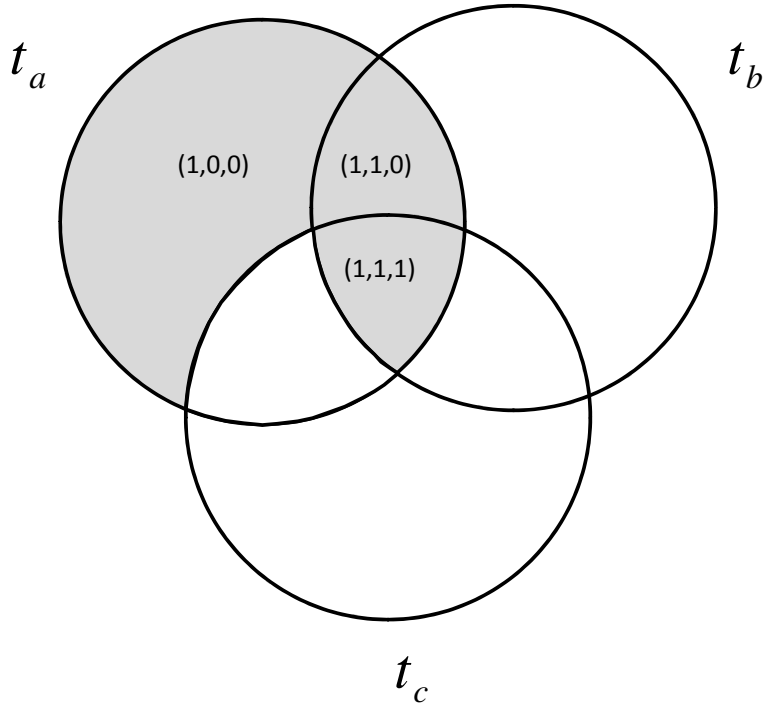


図 2.1: クエリが $[q = t_a \wedge (t_b \vee \neg t_c)]$ のときの 3 要素の状態

索引語の数を k とすると，文書 d_j は $\mathbf{x}_j = (w_{1,j}, w_{2,j}, \ldots, w_{k,j})$ の索引語ベクトルで表現できる．

以下では，代表的な検索モデルについて述べていく．

2.1.2 ブーリアンモデル

ブーリアンモデルは，集合論とブール代数に基づく単純な検索モデルである．集合の概念は直感的であるため，ブーリアンモデルは一般のユーザにも理解しやすく，さらに，明確な意味論を持つブール表現として，クエリを表現できる．

しかし，ブーリアンモデルには大きな欠点がある．一つは，その検索が 2 値の判定基準に基づいていること，つまり，文書が適合しているか否かで評価され，順序付けの考えがないことである．２つ目は，一般のユーザにはブール表現でクエリを表すのが難しいということである．

ブーリアンモデルでは，索引語が存在するか否かを考える．結果として，索引語の重みは $w_{i,j} \in \{0, 1\}$ のようにすべて 2 値で表される．クエリ q は，

索引語に対する NOT，AND，OR の 3 つの演算による結合からなる．そのため，クエリはブール表現の加法標準形 (disjunctive normal form:DNF) のベクトルで表すことができる．たとえば，クエリが $[q = t_a \wedge (t_b \vee \neg t_c)]$ のとき，加法標準形のクエリベクトルは $\mathbf{Q}_{dnf} = (1,1,1) \vee (1,1,0) \vee (1,0,0)$ となる．ここで，それぞれの成分は，3 要素からなる 2 値ベクトル (t_a, t_b, t_c) である．図 2.1 にクエリの 3 要素の状態を示す．

g_i を索引語 t_i に関する重みをかえす関数とし，$\mathbf{Q}_{cc}$ を $\mathbf{Q}_{dnf}$ の要素とする．そのとき，ブーリアンモデルでの類似度は，

$$\mathrm{sim}(d_j, q) = \begin{cases} 1 & \text{if } \exists \mathbf{Q}_{cc} \mid (\mathbf{Q}_{cc} \in \mathbf{Q}_{dnf}) \wedge (\forall_{t_i}, g_i(\mathbf{x}_j) = g_i(\mathbf{Q}_{cc})) \\ 0 & \text{その他} \end{cases}$$

と表される．もし $\mathrm{sim}(d_j, q) = 1$ ならば，クエリ q に対して文書 d_j は適合しており，そうでなければ，適合していない．

ブーリアンモデルは文書が適合しているか否かを判定するが，クエリの一部が適合する場合を想定していない．たとえば，$\mathbf{x}_j = (0,1,0)$ の文書は t_b の索引語を含むが，クエリ $[q = t_a \wedge (t_b \vee \neg t_c)]$ には適合しない．

2.1.3 ベクトル空間モデル

ベクトル空間モデルは，文書とクエリの索引語に重みを与えて，クエリと文書間の類似度を計算する．類似度が高い順に検索文書を並べ替えることにより，クエリの語に部分的に適合する度合いを表すことができる．ベクトル空間モデルで順序づけされた答えはブーリアンモデルによる答えと比べて，より正確なものとなる．

ベクトル空間モデルでは，検索語と文書の組 (t_i, d_j) に対する重み $w_{i,j}$ は正の値となる．さらに，クエリの索引語にも重みづけすることができる．(t_i, q) の重みを $w_{i,q}$ とすると，クエリベクトル $\mathbf{Q}$ は，索引語の数が k のとき，$\mathbf{Q} = (w_{1,q}, w_{2,q}, \ldots, w_{k,q})$ となる．また，前述のように，文書ベクトル $\mathbf{x}_j$ は $\mathbf{x}_j = (w_{1,j}, w_{2,j}, \ldots, w_{k,j})$ で表される．

ベクトル空間モデルでは，文書 d_j とクエリ q の類似度の評価を，図 2.2 に示すように，ベクトル $\mathbf{x}_j$ と $\mathbf{Q}$ の間の角度 θ のコサインで表現する．

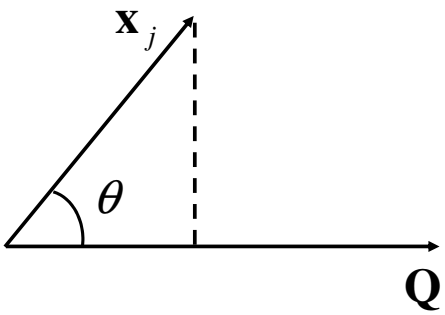


図 2.2: 文書ベクトル $\mathbf{x}_j$ とクエリベクトル $\mathbf{Q}$ の類似度 $\cos\theta$

$$\begin{aligned}\mathrm{sim}(d_j, q) &= \cos\theta \\ &= \frac{\mathbf{x}_j \cdot \mathbf{Q}}{\|\mathbf{x}_j\| \times \|\mathbf{Q}\|} \\ &= \frac{\sum_{i=1}^{t} w_{i,j} \times w_{i,q}}{\sqrt{\sum_{i=1}^{t} w_{i,j}^2} \times \sqrt{\sum_{i=1}^{t} w_{i,q}^2}}\end{aligned}$$

ここで $\|\mathbf{x}_j\|$ と $\|\mathbf{Q}\|$ は，文書ベクトルとクエリベクトルのノルムである．$\|\mathbf{Q}\|$ は，すべての文書に共通なので，順位付けに影響しない．

$w_{i,j} \geq 0, w_{i,q} \geq 0$ なので，$\mathrm{sim}(d_j, q)$ は 0 から+1 までの値をとる．ベクトル空間モデルでは，このクエリに対する類似度に従って文書をランク付けする．これにより，クエリの一部だけが適合している文書でも検索することが可能となる．このとき，索引語にどのような重みを与えるかを決定する必要がある．

索引語の重みはさまざまな方法で計算することができる．最も単純なものは，ベクトル空間モデルの類似度計算に，ブーリアンモデルの文書ベクトルの考え方を適用するものである．これにより，ブーリアンモデルとは異なり，クエリの一部だけが適合している文書も検索可能となる．

しかし，重みが 2 値であるため，索引語どうしの重要度の違いを表すことができない．そこで，用いられる方法が，文書 d_j 内の語 t_i の頻度によって重みを表す，TF（term frequency）である．こうすることで，文書の内容を表現するのにどのくらいよくその語が用いられるかを表現することができる．

さらに，多くの文書に現れる語の重要性を弱めることで，適合文書と非適合文書の区別を容易にすることを目的として，全文書中の語 t_i を含む文書の頻度の逆数をとった，IDF（inverse document frequency）を TF にかけた，TFIDF がよく用いられている．

ベクトル空間モデルの利点をまとめると，次のようになる．

- 語の重みづけにより検索性能が向上する
- クエリに部分的に適合する文書を検索できる
- コサイン類似度により文書の並べ替えができる

欠点としては，理論上，索引語は相互に独立であるという仮定に基づいていることである．したがって，多くの局所的な語の依存関係によって，性能が落ちる場合がある．

2.2 サポートベクターマシン

サポートベクターマシン (Support Vector Machines:SVM) は，2 つのクラスの判別を行う機械学習手法である．

入力されるデータ $\mathbf{x}$ を，n 次元の特徴空間における点としたとき，そのデータにはクラスラベル y として -1 か $+1$ がつくとする．サポートベクターマシンは，人間によって既に判別済みの N 個のデータ $\mathbf{x}_i$ とクラスラベル y_i の組 $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$ に基づいて，未知の入力データ $\mathbf{x}$ に対して，クラスラベル y を精度良く推定する，判別関数 $f(\mathbf{x})$ のパラメータを決定する．

この入力データとラベルの組の集合 $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$ を，学習用サンプル， または， 訓練サンプルと呼ぶ．

サポートベクターマシンは，線形判別器の一種であり，判別関数 $f(\mathbf{x})$ と判別結果 y は以下のようになる．

$$
\begin{aligned}
f(\mathbf{x}) &= \mathbf{w} \cdot \mathbf{x} + b \qquad (2.1)\\
y &= \mathrm{sgn}(f(\mathbf{x})) = \begin{cases} +1 & \text{if } f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b \geq 0 \\ -1 & \text{if } f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b < 0 \end{cases}
\end{aligned}
$$

ここで，$\mathbf{w}$ は線形判別器の重みベクトルと呼ばれるパラメータであり，b はバイアスと呼ばれる．

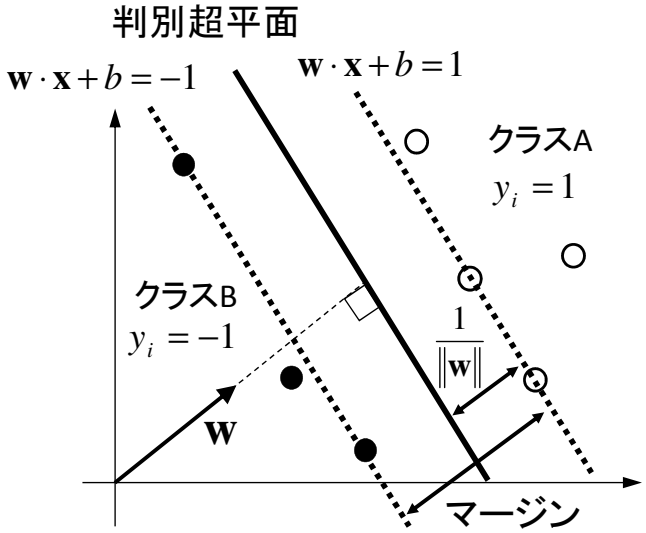


図 2.3: サポートベクターマシンの概念図

与えられた N 個の事例 $\{(\mathbf{x}_i, y_i)|i = 1, \cdots, N\}$ すべてを正しく判別できる線形判別関数 $f(\mathbf{x})$ が存在する場合を線形分離可能と呼ぶ．例えば，$\mathbf{x}$ の次元数 n が訓練サンプルの数 N よりも大きく，同一の $\mathbf{x}$ に対して同一の判定結果 y となる場合は， 必ず，訓練サンプルは超平面 $f(\mathbf{x}) = 0$ によって線形分離可能となる．

ここで，図 2.3 のように，2 つの異なるクラスの訓練サンプルが $n-1$ 次元の超平面で線形分離可能となっているとする．この場合，訓練サンプルを完全に判別できる超平面は無数に存在している．このような超平面の中で，未知の入力データを判別するのに最も優れているものが，よい判別関数となる．

サポートベクターマシンは，2 つのクラスの真ん中を通る判別関数を最も優れているものとし，その評価関数として，超平面と訓練サンプルとの最小距離であるマージンを用いて，これを最大化するように判別関数を決定する．

ここで， 図 2.3 のようなマージン $MGN(\psi)$ を以下のように定義する．判別関数 $f(\mathbf{x}) = \mathbf{w}\cdot\mathbf{x}+b$ と，訓練サンプル $\psi = \{(\mathbf{x}_i, y_i)|i = 1, \cdots, N\}$ が与えられた時，マージン $MGN(\psi)$ は， 特徴空間において判別面 $f(\mathbf{x}) = 0$ に最も近い正例， 負例への距離である．

$$MGN(\psi) \equiv \min_{(\mathbf{x}_i,y_i)\in\psi} \frac{|\mathbf{w}\cdot\mathbf{x}+b|}{\|\mathbf{w}\|} = \min_{(\mathbf{x}_i,y_i)\in\psi} \frac{y_i(\mathbf{w}\cdot\mathbf{x}+b)}{\|\mathbf{w}\|}$$

判別関数が線形分離可能なので，$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b$ に対して次式が成立する．

$$y_i \, f(\mathbf{x}_i) > 0$$

マージンは $MGN(\psi) = \min y_i(\mathbf{w} \cdot \mathbf{x} + b)/\|\mathbf{w}\|$ であり，$\mathbf{w}$ と b を定数倍しても不変である．このため， $\min y_i(\mathbf{w} \cdot \mathbf{x}_i + b) = 1$ の制約を加えることにより $MGN(\psi) = 1/\|\mathbf{w}\|$ となる．これにより，$MGN(\psi)$ の最大化，つまり $\|\mathbf{w}\|$ の最小化は以下により定式化できる．

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2} \|\mathbf{w}\|^2 \\ \text{subject to} \quad & y_i \left(\mathbf{w} \cdot \mathbf{x}_i + b\right) \geq 1 \quad (i = 1, \cdots, N) \end{aligned} \tag{2.2}$$

ラグランジュ未定乗数を $\alpha_i \geq 0$ とすると，

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2} \|\mathbf{w}\|^2 + \sum_i \alpha_i \left(1 - y_i(\mathbf{w} \cdot \mathbf{x}_i + b)\right)$$

を得る．この最適化問題を解くには，$\mathbf{w}, b$ に対して L を最小化し，α に対して L を最大化する．最適解において，L の勾配が 0 となるので，

$$\frac{\partial L}{\partial \mathbf{w}} = 0, \frac{\partial L}{\partial b} = 0$$

となり，次の関係が導かれる．

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x}_i \tag{2.3}$$

$$\sum_i \alpha_i y_i = 0 \tag{2.4}$$

これらから，次の双対問題が得られる．

$$\begin{aligned} \max_{\alpha} \quad & \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \\ \text{subject to} \quad & \alpha_i \geq 0 \quad (i = 1, \cdots, N), \sum_i \alpha_i y_i = 0 \end{aligned} \tag{2.5}$$

この二次計画問題を解くことで，最適な α_i が計算される．このとき，多くの α_i が 0 となり，$\alpha_i \neq 0$ となるものが，最小距離のサンプル，つま

り判別超平面をサポートするベクトルであるサポートベクターとなる．また，判別関数の最適なパラメータ $\mathbf{w}$ と b は，式 (2.1) と (2.3) から計算できる．

訓練サンプルが線形分離可能でない場合，上記の最適化ができなくなる．そこで，多少の判別誤りは許すように制約条件を緩めた，ソフトマージンと呼ばれる方法が用いられる．

制約条件を緩めるには，スラック変数 $\xi_i > 0$ を導入し，制約条件を次のように変える．

$$y_i\, f(\mathbf{x}_i) \geq 1 - \xi_i$$

こうすることで， $y_i\, f(\mathbf{x}_i) \geq 1$ とならない状態も許容する．さらに，制約違反 ξ_i が極力小さくなるように，ペナルティー項 $C\sum_i(\xi_i)$ を導入する．これにより，最適化問題は次のように変更される．

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_i \xi_i \\ \text{subject to} \quad & y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1 - \xi_i,\ \xi_i \geq 0,\ \ i = 1, \ldots, N \end{aligned} \tag{2.6}$$

ここで，C は制約条件の緩和を制御するパラメータである．

このように最適化問題を変更すると，ラグランジュ未定乗数 α に関する問題 L も次のように変更される．

$$L(\mathbf{w}, b, \xi_i, \alpha) = \frac{1}{2}\,\|\mathbf{w}\|^2 + C\sum_i \xi_i + \sum_i \alpha_i\,(1 - \xi_i - y_i(\mathbf{w}\cdot\mathbf{x}_i + b))$$

このとき得られる，双対問題は次のようになる．

$$\begin{aligned} \max_{\alpha} \quad & \sum_i \alpha_i - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \mathbf{x}_i\cdot\mathbf{x}_j \\ \text{subject to} \quad & 0 \leq \alpha_i \leq C \quad (i = 1, \cdots, N), \sum_i \alpha_i y_i = 0 \end{aligned} \tag{2.7}$$

このように，ソフトマージンの双対問題は，ハードマージンの式 (2.5) と比較して，制約条件が変更されるだけとなる．

2.2.1 カーネル

サポートベクターマシンを非線形判別器に拡張するため，図 2.4 のように，n 次元の入力 $\mathbf{x}$ に対して，m 次元の特徴ベクトルを与える非線形な

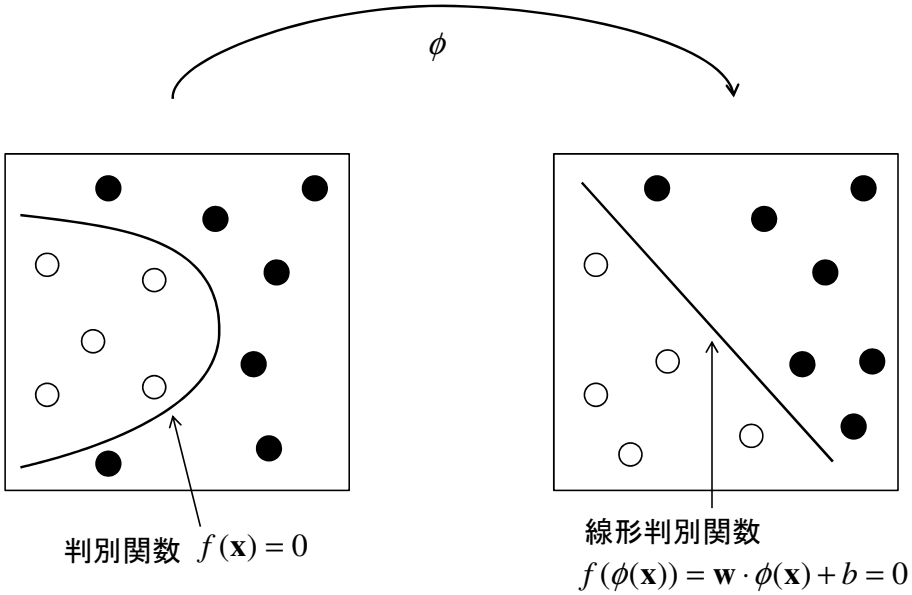


図 2.4: 非線形写像による線形分離

写像 $\phi : \mathcal{R}^n \mapsto \mathcal{R}^m$ を用いて，より高次元の空間に写像して線形判別することを考える．

このとき，サポートベクターマシンの判別関数は，式 (2.1) より

$$f(\phi(\mathbf{x})) = \mathbf{w} \cdot \phi(\mathbf{x}) + b$$

となり，式 (2.3) より

$$f(\phi(\mathbf{x})) = \sum_i \alpha_i y_i \phi(\mathbf{x}) \cdot \phi(\mathbf{x}_i) + b$$

となる．ここで，非線形特徴ベクトル $\phi(\mathbf{x})$ の内積をカーネル $K(\mathbf{x}, \mathbf{x}') = \phi(\mathbf{x}) \cdot \phi(\mathbf{x}')$ とする．これにより，判別関数は

$$f(\phi(\mathbf{x})) = \sum_i \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + b$$

となり，最適化問題についても，式 (2.7) より

$$\begin{aligned} \max_{\alpha} \quad & \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \\ \text{subject to} \quad & 0 \le \alpha \le C \quad (i = 1, \cdots, N), \sum_i \alpha_i y_i = 0 \end{aligned}$$

となる．このように，カーネル関数 K によって，ϕ を求めることなく判別することができる．この解法をカーネルトリックと呼ぶ．

カーネルトリックは，実際に ϕ が存在するカーネルが与えられたときに有効となる．ここで，カーネルが正定値関数であるときには．ϕ が存在する．代表的なカーネルとして以下が挙げられる．

線形カーネル

$$K(\mathbf{x}, \mathbf{x}') = \mathbf{x} \cdot \mathbf{x}'$$

p 次多項式カーネル

$$K(\mathbf{x}, \mathbf{x}') = (a + \mathbf{x} \cdot \mathbf{x}')^p$$

RBF カーネル

$$K(\mathbf{x}, \mathbf{x}') = \exp(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{\sigma^2})$$

2.2.2 サポートベクターマシンの学習理論

教師あり学習アルゴリズムの目的は，訓練対象となるサンプルを用いて未知のサンプルの誤判別率を最小にすることである．この能力を汎化能力と呼ぶ．ここでは，サポートベクターマシンの汎化能力の評価方法を説明することで，どのような学習を行っているかを理論的に述べる．

入力データとラベルの組 $(\mathbf{x}, y)$ は，ある確率分布 $P(\mathbf{x}, y)$ から独立に生成されていると仮定する．つまり，訓練サンプルも未知のテストサンプルもこの確率分布から得られていると仮定する．また，判別関数を $f(\mathbf{x})$ としたとき，損失関数 $Q(\mathbf{x}, y)$ を次のように定義する．

$$Q(\mathbf{x}, y) = \begin{cases} 0 & yf(\mathbf{x}) \geq 0 \\ 1 & yf(\mathbf{x}) < 0 \end{cases}$$

これにより，テストサンプルが無限に存在する場合，誤判別率は次のように表せる．

$$R(f) = \int \int Q(\mathbf{x}, y) P(\mathbf{x}, y) d\mathbf{x} dy$$

この誤判別率 $R(f)$ を期待リスク (expected risk) と呼ぶ．学習は，判別関数の集合 $\mathcal{F}$ が与えられたとき，期待リスク $R(f)$ を最小化する判別関数

$f \in \mathcal{F}$ を見つける問題として定式化できる．しかし，確率分布 $P(\mathbf{x}, y)$ が未知なので，期待リスク $R(f)$ を計算することはできない．そこで，訓練サンプルに対する誤判別率である経験的リスク (emprical risk)

$$R_{emp}(f) = \frac{1}{N} \sum_{i=1}^{N} Q(\mathbf{x}_i, y_i)$$

を扱うことになる．ここで N は訓練サンプル数である．この経験的リスク R_{emp} を最小化する方法は，経験的リスク最小化 (Emprical Risk Minimization) と呼ばれる．

経験的リスク最小化における学習においても，判別関数 f は，与えられた関数の集合 $\mathcal{F}$ から見つけられるとすると，関数集合 $\mathcal{F}$ をどのように設定するかで，学習結果が変わってくる．そこで，経験的リスク最小化において，期待リスクをできるだけ小さくするための $\mathcal{F}$ の設定方法を考える．

期待リスクをできるだけ小さくするために，$R(f)$ と $R_{emp}(f)$ の誤差の評価を行うが，このとき，集合 $\mathcal{F}$ の VC(Vapnic and Chervonenkis) 次元 h を導入する．集合 $\mathcal{F}$ の VC 次元 h とは，点にどのようなラベルをつけても，集合に属する関数によって分離可能な最大の点の数であり，集合の容量 (capacity) を表す．VC 次元 h により，$h < N$ ならば，$1 - \delta$ の確率で次式が成立する [Vapnik 98].

$$R(f) \leq R_{emp}(f) + \sqrt{\frac{h(\log \frac{2N}{h} + 1) - \log \frac{\delta}{4}}{N}}$$

したがって，期待リスクを小さくするためには，VC 次元 h を小さい値に抑えればよい．つまり，関数集合 $\mathcal{F}$ として小さい集合を選べば，VC 次元が小さく抑えられるため，期待リスクも小さくなる．

この定理を用いるために，構造的リスク最小化 (Structural Risk Minimization) という方法を考える．この方法では，VC 次元が単調増加する複数の関数集合

$$\mathcal{F}_1 \subset \mathcal{F}_2 \subset \cdots \subset \mathcal{F}_k$$

を考え，このうちのどれを選んで経験的リスク最小化を行えば，最も期待リスクを小さくできるかを考える．

サポートベクターマシンの関数集合は，線形関数の集合であり次のように定義される．

$$\mathcal{F}_\omega = \{f | f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b, \|\mathbf{w}\| \leq \omega\}$$

訓練サンプルの入力ベクトル $\mathbf{x}$ が半径 D の超球に含まれていると仮定すると，$\mathcal{F}_\omega$ に対応する判別関数の集合の VC 次元の上限は，

$$h \leq \min(D^2\omega^2, n) + 1$$

で与えられる．ここで，n は特徴空間の次元である．

サポートベクターマシンは，訓練サンプルを完全に判別するという制約のもとで，$\|\mathbf{w}\|$ を最小化する関数を選ぶ．これは，経験的リスクを 0 にする関数の中で，最も小さい $\mathcal{F}$ に含まれているものを選ぶことに対応している．これを幾何的に解釈すると，二つのクラスの真ん中に判別面を置くことに相当する．このような学習の枠組みを PAC(Probably Approximately Correct) 学習と呼ぶ．

2.3 能動学習

教師あり学習を行う場合において，入力データ $\mathbf{x}$ は容易に得られても，学習に必要なラベル y は容易に得られないことがよく起こる．

たとえば，設備診断のような場合，設備に取り付けられた各種センサーの測定値 $\mathbf{x}$ は大量に得られるが，設備が正常か異常かといった判定結果 y は，専門家の判断を必要とするため容易には得られない．また，文書の分類についても，個々の文書データ $\mathbf{x}$ が大量に得られても，各文書の分類結果 y を得るためには，人間の判定が必要となる．

このように，入力データ $\mathbf{x}$ が大量にあるにもかかわらず，そのラベル y を得るコストが高いときに教師あり学習を行うには，学習結果から得られる分類器の性能を効果的に高くするようなサンプルを人間に提示して，そのラベルを付けてもらうようなアルゴリズムが必要となる．このような学習の枠組みを能動学習 [Cohn 96] と呼ぶ．

能動学習の例として，Query by Committee[Seung 92] と uncertainty sampling[Lewis 94] について述べる．

2.3.1 Query by Committee

Query by Committee[Seung 92] は，同じ訓練データを学習させた $2k$ 個の分類器を用意して，それらの半数が正と判定し，それ以外が負と判定するような入力データを人間に提示するアルゴリズムである．ここでのクエリ（質問）は，文書検索におけるキーワードの意味ではなく，人間に

ラベルをつけてもらうために提示するサンプルのことである．分類器が重みベクトルによるパラメトリック学習モデルで表されるとすると，学習は，すべての訓練データを完全に分類できる分類器の重みベクトルの集合であるバージョン空間上から，ランダムに $2k$ 個の重みベクトルを選ぶことによって行われる．

「教師」となる目的関数を $y_0(\mathbf{x})$ とし,「生徒」であるパラメトリック学習モデルを $y(\mathbf{w}:\mathbf{x})$ とする．このとき，教師と生徒はいずれも $\{\pm 1\}$ となるブール評価関数である．すべての $\mathbf{x}$ に対して，$y_0(\mathbf{x}) = y(\mathbf{w}_0:\mathbf{x})$ となるような $\mathbf{w}_0$ が存在する，つまり，生徒は教師を実現可能であると仮定する．入力ベクトル $\mathbf{x}_i$ は，教師出力 $y_i = y_0(\mathbf{x}_i)$ の符号によって，正例または負例と呼ばれる．入出力の組 $\psi_i = (\mathbf{x}_i, y_i)$ からなる訓練集合により，次式のバージョン空間 W_N が定義される．

$$W_N = \mathbf{w} : y(\mathbf{w}:\mathbf{x}_i) = y_i, i = 1, \ldots, N$$

バージョン空間 W_N は，訓練集合を分類できるすべての $\mathbf{w}$ の集合である．もし，$\mathbf{w}$ の事前分布 $P(\mathbf{w})$ が平坦ならば，事後分布はバージョン空間内で一定である．これは，次のように書くことができる．

$$P(\mathbf{w}|\psi_1, \ldots, \psi_N) = \begin{cases} V_N^{-1}, & \mathbf{w} \in W_N, \\ 0, & \text{その他} \end{cases} \tag{2.8}$$

ここで，V_N はバージョン空間 W_N の体積である．ここからは，この事後分布から $\mathbf{w}$ をランダムに選ぶ Gibbs 訓練アルゴリズムを考える．

教師に質問する入力 $\mathbf{x}_{N+1}$ を選ぶ方法がクエリアルゴリズムである．一般には，次の条件付き確率で書くことができる．

$$P(\mathbf{x}_{N+1}|\psi_1, \ldots, \psi_N) \tag{2.9}$$

入力 $\mathbf{x}_{N+1}$ はこの分布から選ばれ，教師からラベル y_{N+1} をつけられる．この結果が訓練集合に付け加えられる．

重み $\mathbf{w}$ の事後分布の式 (2.8) のエントロピーは次式で表される．

$$E = \log V_N \tag{2.10}$$

エントロピーは $\mathbf{w}$ についての不確実性を定量化するので，クエリ $N+1$ からの情報利得はエントロピーの減少として，次のように定義できる．

$$I_{N+1} = -\Delta E = -\log \chi_{N+1} \tag{2.11}$$

ここで，体積比を次のように定義する．

$$\chi_{N+1} \equiv \frac{V_{N+1}}{V_N} \tag{2.12}$$

情報利得 I_{N+1} はクエリ列 $\{\psi_1, \ldots, \psi_{N+1}\}$ に依存する．この依存性を消去するため，平均化を行う．クエリ列 $\psi_1, \ldots, \psi_{N+1}$ を一定にすると，平均は，N 個の例に一致する全教師ベクトル $\mathbf{w}^0$ と，クエリアルゴリズムによって与えられるすべての入力 $\mathbf{x}^{N+1}$ について行う．したがって，平均情報利得は次の式で与えられる．

$$\langle I_{N+1} \rangle = -\langle \log \chi_{N+1} \rangle_{\mathbf{w}^0, \mathbf{x}^{N+1}} \tag{2.13}$$

同様に，体積比の確率分布を次のように計算できる．

$$P(\chi_{N+1} | \psi_1, \ldots, \psi_N) = \langle \delta(\chi_{N+1} - \frac{V_{N+1}}{V_N}) \rangle_{\mathbf{w}^0, \mathbf{x}^{N+1}} \tag{2.14}$$

これらの量はクエリ列 $\psi_1, \ldots, \psi_{N+1}$ へ依存していることに注意する．

入力 $\mathbf{x}_{N+1}$ によってバージョン空間 W_N を 2 つに分ける

$$W^+ = \{\mathbf{w} \in W_N : y(\mathbf{w} : \mathbf{x}_{N+1}) = +1\} \tag{2.15}$$

$$W^- = \{\mathbf{w} \in W_N : y(\mathbf{w} : \mathbf{x}_{N+1}) = -1\} \tag{2.16}$$

$\mathbf{w}^0$ の事後分布についての平均から，式 (2.13) の平均情報利得は次の式によって得られる

$$\langle I_{N+1} \rangle = \langle -\frac{V^+}{V_N} \log \frac{V^+}{V_N} - \frac{V^-}{V_N} \log \frac{V^-}{V_N} \rangle_{\mathbf{x}_{N+1}} \tag{2.17}$$

ここで $V^\pm$ は $W^\pm$ の体積であり，$\mathbf{x}_{N+1}$ に依存する．教師がクエリに答えた後，y_{N+1} は確実にわかる．答えが来る前は，y_{N+1} の値は不確定である：ベイズでの解釈では，確率 V^+/V_N で $+1$，確率 V^-/V_N で -1 である．この分布のエントロピーはまさにクエリの情報価値であり，式 (2.17) の平均の中身の表現となる．平均情報利得は，$V^+ = V^-$ の $\mathbf{x}_{N+1}$ によって，つまりバージョン空間を半分に分けるクエリによって最大化される．

Query by Committee は，簡単なモデルによる理論的な考察から，クエリの数が増えるにつれて，情報利得は有限の値に近づき，汎化誤差は指数関数的に減少することが示されている．しかし，バージョン空間による理想的な学習条件のもとで，簡単なモデルを用いるものであるため，このような性能が現実のタスクで必ず得られるとはいえない．

2.3.2 uncertainty sampling

Lewis ら [Lewis 94] は，確率的な学習を行う単一の分類器を用いる場合について，判定結果が最も不確実な，つまり確率が 0.5 に近いサンプルを選択する不確実性に基づくサンプリング (uncertainty sampling) を提案した．

サンプリングアルゴリズムは次のようになる．

1. 初期分類器を作成する
2. 教師によるラベル付けが可能な間，次を行う
   (a) ラベル付けされていない例に対して現在の分類器を適用する
   (b) どのクラスに属するかが最も確実でない ℓ 個の例を見つける
   (c) 教師が ℓ 個の例にラベル付けする
   (d) すべてのラベル付けられた例で新たな分類器を訓練する

ここで ℓ は，理想的には 1 であるが，例へのスコア付けや選択にコストがかかる場合は，大きな値を取る．

この戦略は，サンプルに対する不確実性を最も減少させ，情報利得が最大になるようにサンプルを選択する．しかしながら，分類器が持つパラメータの不確実性を最も減少させるとは限らない．

2.4 適合フィードバック

2.4.1 適合フィードバック

通常，検索を行う場合には，ユーザがクエリ（検索キーワード）を記述して，そのクエリをもとにして検索を行い，ユーザに検索結果を提示する．このとき，1 回の検索によって所望する文書（適合文書）をたくさん集めることは難しい．これは，ユーザが正確かつ詳細にクエリを記述することが容易ではないので，提示文書の上位にユーザの所望しない文書（非適合文書）が多く含まれるためである．

このように，ユーザがクエリを正確に記述することは困難である一方，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難しくない．したがって，検索結果の文書について，適合文書か非適合文書かをユーザに評価してもらうこと

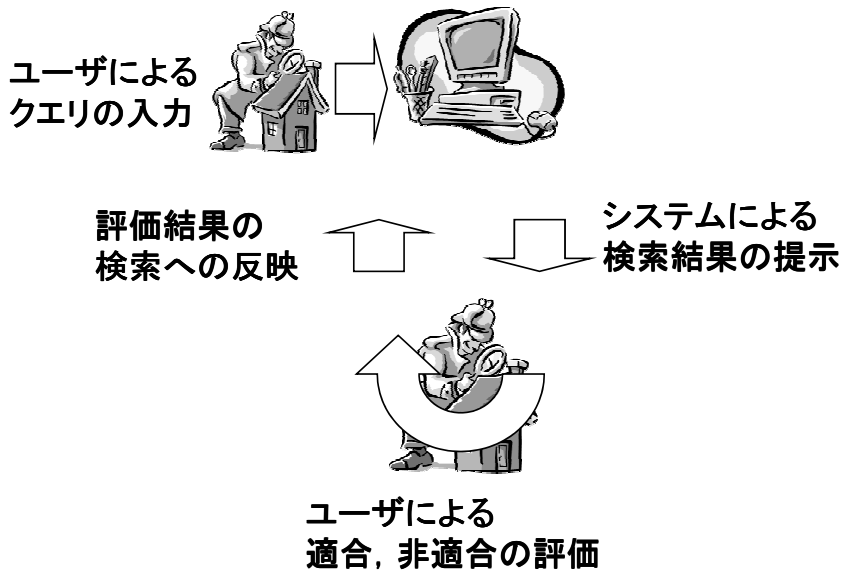


図 2.5: 適合フィードバックの概念図

ができれば，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となる．このような検索の枠組みを適合フィードバック（relevance feedback）と呼ぶ [Salton 71].

適合フィードバックのプロセスは次のようになる．

1. ユーザは初期検索のためのクエリをシステムに入力する
2. システムはクエリをもとに検索文書のリストを提示する
3. ユーザはリストの文書を評価し，適合するか否かをシステムにフィードバックする
4. システムはフィードバックされた文書を評価して，新たな検索文書のリストを提示する
5. 3 に戻って繰り返す.

図 2.5 に，適合フィードバックの概念図を示す.

適合フィードバックは，文書検索の他にも画像検索など様々な分野で利用されており，その有効性が示されている [Harman 92] [Rui 98] [Zhou 03] [Tao 06] [Deselaers 08] [Su 11].

また，一般的な適合フィードバックを，ユーザが適合文書と非適合文書を明示的に評価する Explicit Feedback（明示的フィードバック）と考えた場合，これに対する方法として，ブラウジングやスクロールなど，観察可能なユーザの行動に基づいて適合文書と非適合文書を推測する，Implicit feedback（潜在的フィードバック）という方法も提案されている [Kelly 01] [Kelly 03].

2.4.2 疑似適合フィードバック

適合フィードバックでは，ユーザが文書の適合性を判断する必要があるため，完全に自動化することはできない．疑似適合フィードバックは，システムが上位にランク付けした文書を適合文書とみなしてフィードバックを行うことで，自動的に適合フィードバックを行う方法である [Buckley 95b]．実験的には良い結果が得られると報告されているが，初期クエリによって性能が大きく変化する．

疑似適合フィードバックも，様々な分野での研究がおこなわれている [Yu 03] [Yan 03] [Cao 08].

2.4.3 Rocchio の手法

広く使用されている，ベクトル空間モデルに基づく適合フィードバックとして Rocchio の手法 [Rocchio 71] がある．この手法では，クエリベクトル $\mathbf{Q}_m$ を下式により更新する．

$$\mathbf{Q}_{m+1} = \mathbf{Q}_m + \beta \sum_{\mathbf{x} \in \mathcal{D}_r^m} \mathbf{x} - \gamma \sum_{\mathbf{x} \in \mathcal{D}_n^m} \mathbf{x} \tag{2.18}$$

ここで，$\mathbf{x}$ は判定済み文書ベクトル，そして $\mathcal{D}_r^m$，$\mathcal{D}_n^m$ は，m 回目のフィードバックで適合，非適合と判定された文書集合である．また，β，γ は適合／非適合文書をどの程度考慮するかを調整する定数である．Rocchio の手法では，この更新されたクエリベクトル $\mathbf{Q}_{m+1}$ と文書ベクトルのコサイン類似度で適合度を評価する．

Rocchio の手法に関しては，理論面についての検討 [Joachims 97] や，多くの応用研究が行われている [Schapire 98] [Muller 00] [Jordan 04] [Uğuz 10].

2.4.4 Rocchio の手法での係数決定

Rocchio の手法における係数決定については，後に提案されているさまざまな改良手法においても，係数決定は実験によって試行錯誤的に決定されている [Buckley 95a][Singhal 97]．本研究では，SVM に基づく適合フィードバックが Rocchio の手法と類似していることを示すことで，SVM に基づいて自動的な係数決定ができることを示す．

適合フィードバックではなく，文書分類に Rocchio の手法を適用した研究では，適合率と再現率が一致する breakeven point を最大化することで，係数の自動決定を行う研究がある [Moschitti 03]．この手法では，事前に学習用文書と評価用文書が必要であり，適合フィードバックへの適用は困難である．この意味でも，自動的な係数決定が可能である本研究の手法は優位性があるといえる．

また，適合フィードバックにおける係数決定として，Kullback-Leibler ダイバージェンスを用いた検索モデルにおいて，クエリとフィードバック文書のバランスを決めるフィードバック係数 a を機械学習で決定する研究がある [Lv 09]．

この研究では，クエリを q，判定済み文書を J としたとき，フィードバックを次のように表す．

$$(1-a)g_1(q) + ag_2(J)$$

ここで，$a \in [0,1]$ はフィードバック係数であり，g_1 と g_2 はそれぞれ，クエリと適合文書を比較可能な表現に写像する関数である．最終的には，このフィードバック係数 a の最適値を求める関数 B，つまり $a = B(q, J)$ を学習する．

検索モデルとしては，負の Kullback-Leibler ダイバージェンスを用いた言語モデルを使用する．クエリ q に対する文書 d のスコアは，クエリの言語モデルを θ_q，文書の言語モデルを θ_d とすると，

$$U(q,d) = -\sum_{t \in \mathcal{V}} p(t|\theta_q) \log \frac{p(t|\theta_q)}{p(t|\theta_d)}$$

となる．ここで，t は単語，$\mathcal{V}$ は語彙の集合である．

θ_q の推定精度を向上させるため，2 成分混合モデルを用いる．全文書集合 $\mathcal{C}$ の背景言語モデルを $p(t|\mathcal{C})$ に対して，未知のトピックの言語モデルを θ_T，フィードバックされる文書の集合を $F \subset \mathcal{C}$ とすると，混合モデル

の対数尤度関数は,

$$LLR(F|\theta_T) = \sum_{D \in F} \sum_{t \in \mathcal{V}} c(t, D) \log[(1-\lambda)p(t|\theta_T) + \lambda p(t|\mathcal{C})]$$

となる．ここで，$c(t,q)$ はクエリ q における単語 t の数，$\lambda \in [0,1)$ は，背景モデルの重みを制御する混合雑音パラメータである．λ が与えられたとき，$p(t|\theta_T)$ の推定に EM アルゴリズムが適用できる．元のクエリモデルを $p(t|q)$ とすると，クエリモデルは,

$$p(t|\theta_q) = (1-a)p(t|q) + ap(t|\theta_T)$$

となる．ここで，a は手動で決定するフィードバック係数である．この研究では，通常の適合フィードバックで固定されている係数 a を，上述の関数 B を学習することで最適化している．

フィードバック係数を予測するために，クエリの特徴，フィードバック文書の特徴，クエリとフィードバック文書間の相違の 3 つのヒューリスティックを提案している．具体的には，クエリの特徴として,
(1) クエリの長さ

$$|q| = \sum_{t \in q} c(t, q)$$

(2) クエリのエントロピー

$$qEnt_A = \sum_{t \in F'} -p(t|\theta_{F'}) \log_2 p(t|\theta_{F'})$$

ここで，F' はフィードバックの上位 N 文書；$p(t|\theta_{F'}) = \frac{c(t,F')}{\sum_t c(t,F')}$；
(3) クエリの明瞭度

$$\begin{aligned}
qEnt_R1 &= \sum_{t \in q} p(t|\theta_q) \log \frac{p(t|\theta_q)}{p(t|\mathcal{C})} \\
qEnt_R2 &= \sum_{t \in F'} p(t|\theta_{F'}) \log \frac{p(t|\theta_{F'})}{p(t|\mathcal{C})} \\
qEnt_R3 &= \log(qEnt_R1) \\
qEnt_R4 &= \exp(qEnt_R2)
\end{aligned}$$

フィードバック文書の特徴として,
(1) フィードバックの長さ

$$|F| = \sum_{d} \delta(d, F)$$

ここで，文書 $d \in F$ ならば $\delta(d, F) = 1$，それ以外は 0；
(2) フィードバック半径

$$FBRadius = \frac{1}{|F|} \sum_{d \in F} \sum_{t \in d} p(t|\theta_d) \log \frac{p(t|\theta_d)}{p(t|\theta_{centroid})}$$

ここで，$p(t|\theta_{centroid}) = \frac{1}{|F|} \sum_{d \in F} p(t|\theta_d)$ である．
(3) フィードバック文書のエントロピー

$$FBEnt_A = \sum_{t \in F} -p(t|\theta_F) \log_2 p(t|\theta_F)$$

ここで，$p(t|\theta_F) = \frac{c(t,F)}{\sum_t c(t,F)}$；
(4) フィードバック文書の明瞭度

$$\begin{aligned} FBEnt_R1 &= \sum_{t \in F} p(t|\theta_F) \log \frac{p(t|\theta_F)}{p(t|\mathcal{C})} \\ FBEnt_R2 &= \exp(FBEnt_R1) \\ FBEnt_R3 &= \sum_{t \in F} p(t|\theta_T) \log \frac{p(t|\theta_T)}{p(t|\mathcal{C})} \end{aligned}$$

クエリとフィードバック文書館の相違として，
(1) 絶対相違

$$qFBDiv_A = \sum_{t \in F} p(t|\theta_F) \log \frac{p(t|\theta_F)}{p(t|\theta_{F'})}$$

(2) 相対相違

$$qFBDiv_R = \sum d \in F \frac{prec(r_d)}{T}$$

ここで，r_d は文書 d の順位，つまり，最初の文書が 1，次が 2；$prec(r_d)$ は上位 r_d 文書の適合率；T は定数；を提案している．学習は，ヒューリスティックにより定義した特徴に対するロジスティック回帰により行っている．

トピッククエリを 2 つに分けて，一方を学習用のトピック，残りをテスト用のトピックとして TREC(Text Retrieval Conference) のデータセットで行った実験では，パラメータを固定した結果と比較して，検索性能が向上している．しかし，学習用のクエリを事前に用意する必要がある点や，事前に学習したクエリによって性能が変化する点は実用上問題となる．

2.4.5　適合フィードバックと分類学習

これまでも，適合フィードバックを対話的分類学習として捉え，機械学習の分類学習アルゴリズムを適用した研究がある．柘植ら [柘植 03] は，分類学習として本研究と同じ SVM を適用し，比較実験により検索性能が向上することを示している．彼らの研究では，判定文書を変化させた実験は行っているが，フィードバック回数は 1 回のみであり，一度得られた判別超平面を元にユーザへの次の提示文書をどのように決定するかという文書提示の問題には対応していない．本研究では，その問題に対処するヒューリスティクスを提案，実験による検証を行っている点が大きく異なる．

Drucker ら [Drucker 01] も，適合フィードバックに SVM を適用し，特に適合文書が文書データベース中に少ない場合に，有効であることを確認している．彼らの研究では，文書データベース中に適合文書が占める割合を変化させた実験は行っているものの，得られた判別超平面を元に，ユーザへ次に提示する文書の選択の違いによる検索/学習性能の差異を議論していない．

また，岡部，山田 [岡部 01] は，関係学習により適合文書の分類ルールを獲得する分類学習を対話的文書検索に応用している．単語の近接，存在などの述語を領域知識として，適合文書と非適合文書からそれらの分類ルールを学習する．その結果，トピックに依存するが，従来法の適合フィードバックより検索性能が向上することを実験的に示している．しかしながら，SVM などの連続値を扱える分類学習アルゴリズムの方が，より精度の高い分類性能を期待できる．

さらに，Tong ら [Tong 02] は，文書分類における，SVM に基づく能動学習の際のサンプル選択について研究している．そこでは，マージンに基づきバージョン空間を効率的にカットするためのサンプルの選択手法を提案し，ベンチマークによる実験で有効性を示している．しかし，あくまで文書分類への適用の議論しかされておらず，本研究で扱う対話的文書検索からの視点からの有効性検証は行われていない．

2.5　トランスダクティブ学習

トランスダクティブ (transductive) 学習 [Vapnik 98] は，帰納的 (inductive) 学習の代わりとして，少ない訓練サンプルにおける学習に用いられ

る．帰納的学習では，多くの訓練サンプルがある場合に，サンプル全体の分布から誤りの少ない判別関数を学習する．これに対して，トランスダクティブ学習は，訓練サンプルを利用して，ラベルなしデータも含めて，できるだけ小さな誤りで分類を行う．つまり，訓練サンプルにおけるデータの類似性に基づいて，ラベルなしデータにラベル付けを行っていく方法である．

トランスダクティブ学習のアルゴリズムはいくつか提案されているが [Joachims 03] [Zhu 03] [Blum 04]，ここではトランスダクティブサポートベクターマシン [Joachims 99] について述べる．

N 個の訓練サンプルを

$$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_N, y_N)$$

とし，ラベルなしの k 個のテストサンプルを

$$\mathbf{x}_1^*, \mathbf{x}_2^*, \ldots, \mathbf{x}_k^*$$

とする．これらは同じ分布から得られたとする．このとき，テストサンプルのラベル

$$y_1^*, y_2^*, \ldots, y_k^*$$

を推定するアルゴリズムは図 2.6 のように定義される．

図中の $solve\_SVM\_QP$ では，次の問題を解いている．

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i}^{N}\xi_i + C_-^*\sum_{j:y_j^*=-1}\xi_j^* + C_+^*\sum_{j:y_j^*=1}\xi_j^* \\ \text{subject to} \quad & y_i(\mathbf{w}\cdot\mathbf{x}_i + b) \geq 1-\xi_i,\ \xi_i \geq 0,\ \ i = 1, \ldots, N \\ & y_j^*(\mathbf{w}\cdot\mathbf{x}_j^* + b) \geq 1-\xi_j^*,\ \xi_j^* \geq 0,\ \ j = 1, \ldots, k \end{aligned}$$

ここで，C と C_-^*，C_+^* は次式で定義される緩和パラメータである．

このアルゴリズムの意味は次のようになる．まず，訓練サンプルを用いて判別関数を作成する．次に，その判別関数により，テストサンプルに仮のラベルを付け，ゆるい制約条件（小さな C_-^* と C_+^*）のもとで，テストサンプルも含めて新たな判別関数を作成する（一つ目の **while** ループ）．そして，ラベルが反対で，ずれの大きい（ξ^* の大きい）テストサンプルを選び，そのラベルを入れ替えて，もう一度新たな判別関数を作成する（二つ目の **while** ループ）．ラベルが反対で，ずれの大きいデータがなくなるまで，二つ目のループを繰り返し，テストサンプルの緩和パラ

$(\mathbf{w}, b, \xi, \_) := solve\_SVM\_QP(\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}, [], C, 0, 0);$

$\langle \mathbf{w}, b \rangle$ を使ってテストサンプルを分類する．
$\mathbf{w} \cdot \mathbf{x}_j^* + b$ の最大値から $num+$ 個のテストサンプルをクラス+$(y_j^* := 1)$ とし，
残りのテストサンプルをクラス $-(y_j^* := -1)$ とする．

$C_-^* := 10^{-5}$
$C_+^* := 10^{-5} \times \frac{num+}{k-num+}$

while$((C_-^* < C^*)||(C_+^* < C^*))$**{**
  $(\mathbf{w}, b, \xi, \xi^*) :=$
  $solve\_SVM\_QP(\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}, \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}, C, C_-^*, C_+^*);$
  while$(\exists m, l : (y_m^* \times y_l^* < 0) \& (\xi_m^* > 0) \& (\xi_l^* > 0) \& (\xi_m^* + \xi_l^* > 2))$**{**
    $y_m^* := -y_m^*;$
    $y_l^* := -y_l^*;$
    $(\mathbf{w}, b, \xi, \xi^*) :=$
    $solve\_SVM\_QP(\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}, \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)\}, C, C_-^*, C_+^*);$
  }
  $C_-^* := \min(C_-^* \times 2, C_-^*)$
  $C_+^* := \min(C_+^* \times 2, C_+^*)$
}
return$(y_1^*, \ldots, y_k^*)$

図 2.6: トランスダクティブサポートベクターマシンのアルゴリズム

メータ C_-^* と C_+^* の条件を厳しくして，一つ目のループに戻る．緩和パラメータが C^* 以上になったら終了となる．

このようなアルゴリズムにより，テストサンプルも含めたデータ全体で，類似のデータが極力同じクラスになるように分類を行うことになる．アルゴリズムからわかるように，サポートベクターマシンの繰り返し計算が多いため，計算量はかかるが，カーネルを用いて非線形に拡張が可能であるため，非線形性の高い問題には効果が高い．

トランスダクティブ学習では，ラベルなしデータを類似性に基づいて直接ラベル推定を行っている．これに対して本研究では，ラベルなしデー

タに順位をつけて，能動学習のための提示を行う点が異なっている．

2.6 ランキング学習

ランキング学習は，教師あり機械学習を用いて，検索ランキングを最適化する技術である．ランキングは，クエリ q が与えられたときに，そのクエリに対応する文書 d の順位として定義される．このときランキング学習の枠組みでは，クエリとランキングのペアを学習して，未知のクエリを与えたときのランキングを推定する．ランキング学習にはさまざまな方法が提案されているが [Crammer 01] [Nallapati 04] [Cao 06] [Sculley 09]，例として，Ranking SVM[Joachims 02] について説明する．

Ranking SVM は，2 つの文書間での順位関数の大小を規定することで，ランキングを二値分類問題として解いている．クエリ q と文書 d_i, d_j が与えられたとき，これらの文書のランキング関数を次式のように考える．

$$(d_i, d_j) \in f_{\mathbf{w}}(q) \Longleftrightarrow \mathbf{w}\phi(q, d_i) > \mathbf{w}\phi(q, d_j)$$

ここで，$\mathbf{w}$ は学習によって最適化された重みベクトル，$\phi(q, d)$ はクエリ q と文書 d との対応に関する特徴を表す写像関数である．この特徴として用いられるのは，クエリと文書の単語数や，HTML タグの単語数，ページランクなどである．

このとき，Ranking SVM は次式のように定義される

$$\begin{aligned}
\min_{\mathbf{w}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum \xi_{i,j,k} && (2.19)\\
\text{subject to} \quad & \forall (d_i, d_j) \in r_1^* : \mathbf{w}\phi(q_1, d_i) \geq \mathbf{w}\phi(q_1, d_j) + 1 - \xi_{i,j,1} && (2.20)\\
& \cdots \\
& \forall (d_i, d_j) \in r_N^* : \mathbf{w}\phi(q_N, d_i) \geq \mathbf{w}\phi(q_N, d_j) + 1 - \xi_{i,j,N} \\
& \forall i \forall j \forall k : \xi_{i,j,k} \geq 0 && (2.21)
\end{aligned}$$

ここで，$\xi_{i,j,k}$ はスラック変数，C は緩和パラメータである．

式 (2.20) の制約条件を

$$\mathbf{w}(\phi(q_k, d_i) - \phi(q_k, d_j)) \geq 1 - \xi_{i,j,k}$$

のように変形すると，差分ベクトル $\phi(q_k, d_i) - \phi(q_k, d_j)$ の SVM 分類問題と等価となる．

ランキング学習は，クエリに対する文書の順位が与えられている場合に行われる．これに対して本研究は，適合・非適合文書が順位付けなしに与えられたときに，それらの情報から，未知の文書に対する適合度の順位を求めるものである．

第3章　サポートベクターマシンに基づく能動学習による対話的文書検索

本章では，従来の対話的情報検索手法である適合フィードバックを対話的分類学習として考えたとき，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を応用した対話的文書検索の枠組みにおいて，ユーザが判定する文書の選択に有効なヒューリスティクスを提案する．このヒューリスティックスは，ユーザへの文書提示を能動的に行うものであり，文書検索タスクのような少数の正データと膨大な負データとからなるデータの分布において有効と考えられる．

提示提案手法の有効性を検証するために，従来の適合フィードバックシステム，そして能動学習なしのシステムを用いて，新聞記事検索のテストベッドを対象に比較実験を行い．提案システムの有効性を示す [Onoda 06] [Onoda 07b] [Onoda 08a] [Onoda 08b] [Murata 09].

3.1　能動的文書提示

ここでは，文書を選択的に提示する能動学習の考えに基づき，サポートベクターマシンによる適合フィードバック手法を用いた情報検索について述べる．このような選択的な文書提示を，本研究では能動的文書提示と呼ぶ.

分類学習におけるデータ集合を文書集合，分類すべき 2 つのクラスを適合／非適合，正例／負例を適合フィードバックにより得られる適合文書／非適合文書と考えると，分類学習により適合フィードバックが実現されることになる（図 3.1）[岡部 01].

検索モデルとして，文書とクエリを語の頻度分布からなる多次元ベクトルで表現するベクトル空間モデル（vector space model）[Salton 83] を

用いる．クエリベクトルと文書ベクトル間の類似度をベクトル間の内積により求め，その値の高い文書を検索結果として提示する．このモデルでは，それぞれの文書はベクトルで記述されるため，データとしてベクトルを想定するサポートベクターマシンを素直に適用できる．

一方，適合フィードバックでは，システムが提示した文書に対し，人間のユーザがその文書を読んで内容を理解した上で，適合／非適合の判定，つまり正／負データのラベル付けを行う．このラベル付けは，ユーザにとって非常にコストがかかる作業であるため，できるだけ少ない文書の判定で有効なフィードバックをかけることが必須となる．この要件を満たすように，ユーザに提示する文書を選択する問題は，機械学習における能動学習にあたる．能動学習は，学習者が環境から選択的に情報（訓練データ）を得ることにより学習を行う．

本研究では，サポートベクターマシンにより判定が曖昧なマージン（図 3.2 における破線と破線の間の領域）中にあり，かつ適合文書（図 3.2 における黒丸のデータ）の領域に最も近いところにある文書をユーザに提示し，ユーザがそれらに対し適合／非適合の判定を行って，その判定結果をシステムに再びフィードバックする能動的文書提示によるサポートベクターマシンに基づく対話的文書検索を提案する．この手法は，以下に示す手続きで適合フィードバック，検索を行う．

Step 1　初期検索：ベクトル空間モデルを用い，ユーザからのクエリに対して文書集合を検索し，クエリベクトルと文書ベクトルとの内積による類似度の高い上位 S 文書をユーザに提示する．

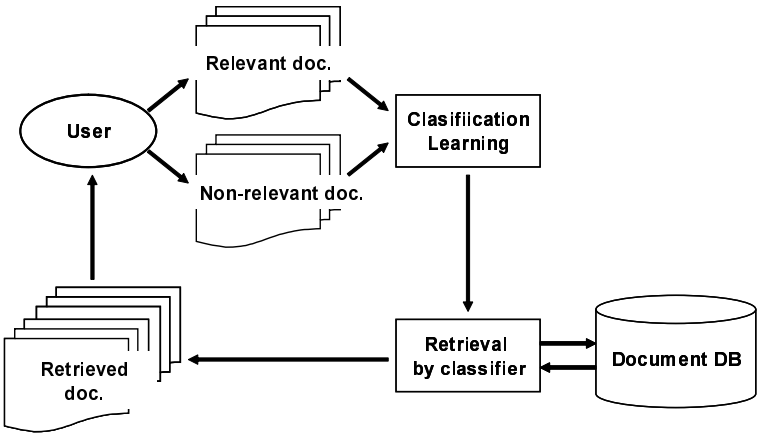


図 3.1: 対話的分類学習としての対話的文書検索

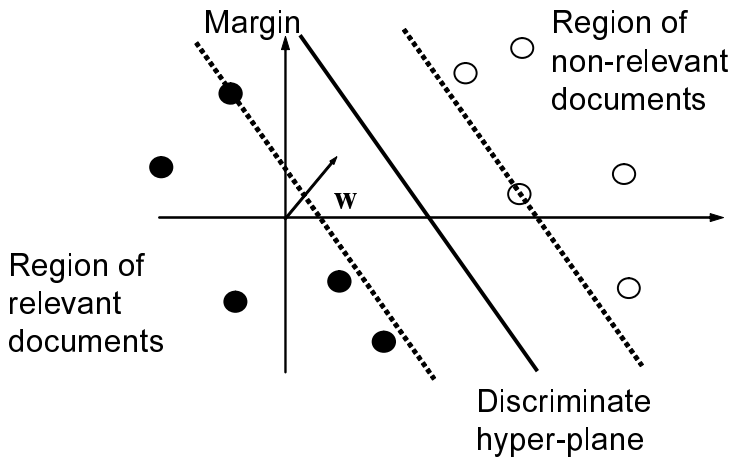


図 3.2: サポートベクターマシンを用いた，ユーザの文書評価結果に基づく最適判別超平面

Step 2　提示文書に対するユーザの判定：提示された文書に対し，ユーザは適合，非適合の判定を行う．適合と判定された文書には，ラベル “1”，非適合と判定された文書には，ラベル “–1” をつける．このとき，すべてのラベルが “1” か “–1” だけの場合，つまり判定されたすべての文書が適合か非適合の場合は，初期検索結果の次の S 文書をユーザに提示して，Step 2 へ．そうでなければ，Step 3 へ．

Step 3　サポートベクターマシンによる学習：ユーザが判定した文書すべてを用いてサポートベクターマシンにより学習を行い，検索文書全体を適合，非適合に分類する最適判別超平面を決定する（概念図は，図 3.2 を参照）．

Step 4　能動的文書提示：フィードバック回数が M 以上であれば，Step 5 へ．M 未満なら，サポートベクターマシンにより判定が曖昧なマージン中で，かつ適合文書（正データ）の領域に最も近い上位 S 文書をユーザへ提示し，Step 2 へ．

Step 5　検索結果出力：学習されたサポートベクターマシンにより適合と判断される未判定文書のうち，最適分離超平面からの距離が遠い順で H 文書を出力する．ただし，サポートベクターマシンにより適合と判断される文書の数が H 未満の場合は，マージン中のもの

を足して出力する．

Step 4 において，ユーザに文書を選択的に提示している点で，本手続きは能動的文書提示による対話的文書検索になっている．

上記の手続きで最初に設定しておくパラメータは，ユーザに一回に判定してもらう文書数 S，ユーザのフィードバック回数 M，および Step 5 でユーザに提示する文書の数 H である．次に，Step 4 で提示文書を選択するヒューリスティクスについて考察する．

3.2 文書提示のヒューリスティクス

対話的文書検索では，ユーザが文書判定をしながらフィードバックを複数回かけるため，最終的にできるだけ多くの適合文書を見つけることが重要である上に，文書提示の段階でも適合文書が多く含まれることが望ましい．よって，分類学習に基づく対話的文書検索に要求される条件は，以下の 2 つと考えられる．

(1) **学習性能の向上：**出来るだけ精度の高いクラシファイアを効率的に獲得するため，適合／非適合文書の判定の難しい文書を優先的にユーザに提示する．

(2) **検索性能の向上：**ユーザへ提示する文書が，できるだけ多くの適合文書を含んでいる．

条件 (1) は，分類学習一般に要求されるものであり，能動学習においては，条件 (1) を満たすような訓練データを選択することが望ましい．一方，条件 (2) は対話的文書検索において，ユーザはできるだけ早く適合文書を見つけたいという前提からの条件である．

図 3.3 に，**3.3** で使用するデータセットで，1 回のフィードバックの後に生成される判別超平面からの距離に対する，文書ベクトルの分布の一例を示す．ほぼすべてのトピックで，1 回のフィードバック後の適合／非適合である文書ベクトルの分布は，図 3.3 と同様となった．この図で，x 軸は判別超平面からの距離を表し，0 のところが判別超平面にあたる．そして，x 軸の値が $+1$ のところが適合文書の境界超平面を，-1 のころが非適合文書の境界超平面を表す．図 3.3 の y 軸は，判別超平面からの距離が幅 0.01 の窓の中に入っている非適合文書ベクトルの数を表す．また，y

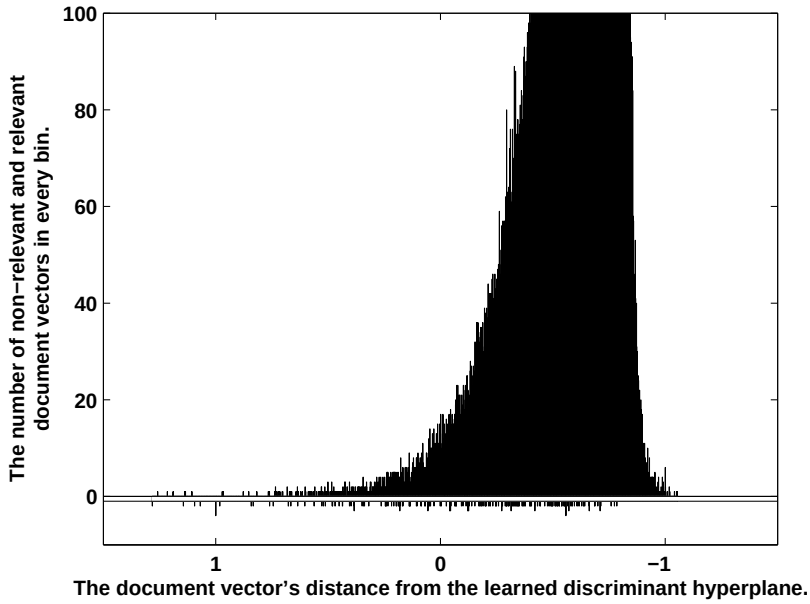


図 3.3: ユーザ評価 1 回後の判別関数値に対する文書ベクトルの分布

軸の負の部分にある下向きのヒストグラムは，判別超平面からの距離に対する適合文書ベクトルの数を表す．

一般に文書検索では，対象となる文書が膨大にもかかわらず，一回の判定文書数は非常に少ない．よって，その少ない判定文書から得られる最初の判別超平面，境界超平面は，真の判別超平面，境界超平面に対して非常に不正確なものである．そのため，図 3.3 からわかるように，判定されていない文書ベクトルのほとんどは，一回の判定文書からサポートベクターマシンで得られる境界超平面間（マージン内）に分布する．特に，大量にある非適合文書は，適合文書と判断される領域 $x > 0$ のみならず，正データの境界超平面を越えた $x > 1$ の領域にも存在している．このような状況で，学習性能の向上条件 (1)，つまり，サポートベクターマシンにとって適合，非適合文書の判別が難しいデータを優先的に提示するには，どのようなデータを選択すれば良いであろうか．それは，判別超平面に最も近い文書を提示すれば良いことになる．しかし，判別超平面に近い，つまり図 3.3 の x 軸の 0 に近い文書を提示してもその中に，適合文書が含まれる可能性は低く，仮りに含まれても非常に少数である．つまり，検索性能の向上条件 (2) がほとんど満たされなくなってしまう．

一方，検索性能の向上条件 (2) を満たす文書は，判別超平面から適合文書側に最も遠い文書であることがわかる．しかし，判別超平面から適合文書側に最も遠い文書をユーザが適合文書と判定しても，サポートベクターマシンはその文書を既に適合文書と判別する学習が済んでいるので，現状から学習は促進されず，学習性能の向上条件 (1) が満たされなくなってしまう．

この相反する二つの条件を満たすために本研究では，“サポートベクターマシンにより判定が曖昧なマージン中で，適合文書（正データ）の領域に最も近い上位の文書をユーザへの提示文書とする” というヒューリスティックスを用いる．図 3.3 より，正データである適合文書の境界超平面近傍の文書を選択すれば，非適合文書にもかかわらず適合文書と判別される文書と，その文書の近くに存在する適合文書を選択できる可能性が高いことがわかる．一方，適合文書の境界超平面の近傍をとることにより，適合文書を提示できる可能性も高くなるため，検索性能の向上条件 (2) を満たすことも期待できる．よって，“サポートベクターマシンにより判定が曖昧なマージン中で，適合文書（正例）の領域に最も近い上位文書をユーザへの提示文書とする” というヒューリスティクスは，学習性能の向上条件 (1)，検索性能の向上条件 (2) を適度に満たす妥当なものと考えられる．このヒューリスティクスの有効性は，次節で実験的に検証される．

3.3 比較実験

3.3.1 実験条件

3.1 と **3.2** で提案したサポートベクターマシンによる能動学習に基づく適合フィードバック手法の有効性を検討するための実験を行った．実験用データには，文書検索の評価実験で広く使用されている，国際会議 TREC[1] の第 6 回から第 8 回の ad hoc タスクで使用されたデータセットを用いた．このデータセットは約 53 万の新聞記事文書からなり，150 個の検索課題（トピック）とそれらの適合文書の情報が提供されている．

各トピックは，その内容を 2，3 語で表した title タグ，詳しく記述した description タグ，さらに詳しい情報を記した narrative タグからなっている．図 3.4 に実際のトピックの一例を示す．本研究では，クエリとして

[1] http://trec.nist.gov/

```
Number       301
Title        International Organized Crime
Description  Identify organizations that
             participate in international
             criminal activity, the
             activity, and, if possible,
             collaborating organizations
             and the countries involved.
Narrative    A relevant document must
             as a minimum identify the
             organization and the type
             of illegal activity (e.g.,
             Columbian cartel exporting
             cocaine).  Vague references
             to international drug trade
             without identification of
             the organization(s) involved
             would not be relevant.
```

図 3.4: トピックの一例

title タグの単語を使用した．また，各文書とクエリには，smart system のリストを使った stopword の除去[2]と Porter Stemming Algorithm[3]による stemming を施してある．

文書ベクトルは TFIDF を用い，その算出には，一般的に使われている次の計算式を使った．

$$w(t,d) = \frac{\log(\mathrm{tf}(t,d)+1)}{\log(\mathrm{uniq}(d))} \log \frac{N}{\mathrm{df}(t)} \tag{3.1}$$

- $w(t,d)$：文書 d における単語 t の重み．
- $\mathrm{tf}(t,d)$：文書 d における単語 t の出現頻度
- N：データ集合内の文書総数
- $\mathrm{df}(t)$：単語 t を含む文書数
- $\mathrm{uniq}(d)$：文書 d における単語の異なり数（種類）

[2]http://www.lextek.com/manuals/onix/stopwords2.html
[3]http://tartarus.org/~martin/PorterStemmer/

なお，文書ベクトルの次元は約 76 万である．

サポートベクターマシンによる最適判別超平面は，線形分離により訓練データを 2 分割する手法により求めた．完全に訓練例集合を分離できない場合には，緩和変数の導入 [Cortes 95] で対応できるが，本研究で扱うベクトル空間モデルは非常に多次元（約 76 万次元）であり，訓練例集合を十分に分離可能と考えられるため，緩和変数を用いていない．また，サポートベクターマシンでは，カーネルトリックを用いる手法 [小野田 02][小野田 07a] が一般的に使われているが，ここでの文書のベクトル空間モデルは，すでに多次元表現されており，訓練例集合をさらに高次元空間で表現する必要がないため，カーネルは用いていない．なお，サポートベクターマシンは LibSVM[4] を用いて実装した．

提案手法の有効性を示すため，以下の 3 つのシステムにより，比較実験を行った．

- 提案ヒューリスティックスを用いた能動的文書提示によるサポートベクターマシンに基づく対話的文書検索システム：**3.1** の提案手法を実装したシステム．

- Simple 法によるサポートベクターマシンに基づく対話的文書検索システム：**3.1** の Step 4 において，その時点の最適判別超平面の近傍にある文書を提示するシステム．それ以外は，提案システムと同じ．この提示文書の選択方法は，“Simple 法” と呼ばれ，能動学習の最も単純な方法の一つである [Campbell 00]．例えば，文献 [Warmuth 03] の研究では，Simple 法により選択した未学習データを専門家に判定してもらうことにより，新たな製薬の生成の効率化を図ることに成功している．このシステムは，能動的文書提示のベースラインシステムとして採用した．

- Rocchio ベースの対話的文書検索システム：従来の適合フィードバックとして広く利用されている Rocchio の手法 [Rocchio 71] をパラメータチューニングしたシステム．従来の適合フィードバックのベースラインシステムとして採用した．

なお，Rocchio ベースの手法は，クエリベクトル $\mathbf{Q}_m$ を下式により更新する．

$$\mathbf{Q}_{m+1} = \mathbf{Q}_m + \beta \sum_{\mathbf{x} \in \mathcal{D}_r^m} \mathbf{x} - \gamma \sum_{\mathbf{x} \in \mathcal{D}_n^m} \mathbf{x}$$

[4] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

ここで，$\mathcal{D}_r^m$，$\mathcal{D}_n^m$ は，m 回目のフィードバックで適合，非適合と判定された文書集合であり，β，γ は適合／非適合文書をどの程度考慮するかを調整する定数である．この β，γ については，$(\beta,\ \gamma) = (1.0,\ 0.5)$ を使った．

各検索システムにおいて初期フィードバック以外では判定文書（分類学習の訓練データ）が異なるため，単純には性能を比較することができない．このことを考慮して本研究では，学習性能と検索性能の向上を評価する 2 つの評価指標を導入した．学習性能の指標は，**3.1** で示した検索手順の Step 5 において，全文書を適合文書に成りそうな順に並べ，文献 [Yates 99] で用いられている「上位 30 文書における精度 $P30$」を適用する．これにより，フィードバックにおける判定文書は，適用する手法ごとに異なるものの，最終的に生成されたクラシファイアの検索における学習性能が比較できる．

通常の機械学習の評価では，訓練データ（判定文書）とテストデータ（$P30$ の上位 30 文書）は分離すべきであるが，ここでの $P30$ における上位 30 文書には，判定文書が含まれている．本研究では 2 回以降のフィードバックで判定された適合文書の数がシステムによって異なる上に，もともと適合文書の数が少ないことから，判定文書をのぞいて $P30$ を計算するとその差異がパフォーマンスに大きく影響することから，このような $P30$ の評価方法をとっている．

また，検索性能の向上を評価する指標としては，検索が終了するまでにユーザへ提示された全文書中に含まれる適合文書の割合 P で，各検索システムの検索性能を比較した．具体的には，**3.1** の検索手順で，$H{=}S$ として，Step 5 に至るまでにユーザに提示された文書数 $S(M+1)$ とその文書中の適合文書の数 R から，$P = \frac{R}{S(M+1)}$ で計算される．この指標は，ユーザがフィードバック中に判定した適合文書と最終的な学習結果による適合文書の両方を含んでおり，その判定文書数も一定としている．最終的な学習結果から得られた適合文書のみを評価対象とせずに，フィードバック中の適合文書も評価に用いるのは，ユーザにとってフィードバック中に見つけた適合文書も最終的な学習結果によって得られた適合文書も一連の検索によって得られた文書に変わりはないこと，また，ユーザの文書判定にかかるコストが無視できないため，フィードバック中と最終判定時で差をつけないことによる．

広く知られているように，適合フィードバックのパフォーマンスは判定文書数により変化するため [Montgomery 04]，実験において 1 回の判定

文書数とフィードバック回数を変えて調べる必要がある．本研究の目的は，“ユーザからのフィードバックにより，より多くの適合文書を獲得する対話的文書検索システム” の開発である．この目的を考えると，精度-再現率曲線などのシステム指向の評価 [酒井 06] は適切ではなく，ユーザが目を通せる範囲でできるだけ多くの適合文書を見つけるユーザ指向の評価をすべきである．よって，ここでの評価は，経験的にユーザが目を通せる範囲と考えられる 100 を判定文書の上限として設定した．具体的には，フィードバック時の提示文書数 S を 10，20 とし，フィードバック回数 M を，それぞれ 1〜9，1〜4 へ変化させてパフォーマンスを調べた．また，最終的に提示数する文書数は $H = S$ とした．この設定下での $P30$，P は，“ユーザがフィードバックをかけた際に，実際にみることのできる適合文書をできるだけ多くする” という評価になっており，本研究の目指すシステムの評価に合致している．

3.3.2 実験結果

表 3.1，表 3.2 と図 3.5，図 3.6 に学習性能の評価結果を，表 3.3，表 3.4 と図 3.7，図 3.8 に検索性能の評価結果を示す．表中の Rocchio，SVM-S，SVM-A は，Rocchio ベースの対話的文書検索システム，Simple 法による対話的文書検索システム，そして，本研究で提案する能動的文書提示による対話的文書検索システムである．また，M はフィードバックの回数である．ここで，フィードバック回数 0 は，初期検索時の性能を表す．

学習性能の比較

表 3.1 と図 3.5 から提示文書数が 10 のときはフィードバック回数が 3 回以上になると，表 3.2 と図 3.6 から，提示文書数が 20 のときはフィードバック回数が 2 回以上になると，つまり，フィードバック文書数が 30 文書を越えると，SVM-A と SVM-S は，常に Rocchio よりも優れた性能を示している．これは，サポートベクターマシン に基づくフィードバック手法の優れた学習性能が実験的に示されたことを意味する．初期のフィードバックで，SVM-A，SVM-S が Rocchio ベースシステムより劣っているのは，Rocchio が 1 回目のフィードバックの結果がよくなるようにパラメータ調整されていること，そして 1 回目のフィードバックではシステムが能動的に学習データを選択していないため，能動学習の効果が表れていないことなどが原因と考えられる．

表 3.1: TFIDF による学習性能 (P30) の評価:提示文書数 10

M	Rocchio	SVM-S	SVM-A
0	0.176	0.176	0.176
1	0.244	0.202	0.202
2	0.297	0.304	0.290
3	0.324	0.373	0.377
4	0.356	0.440	0.436
5	0.375	0.489	0.497
6	0.390	0.537	0.546
7	0.394	0.568	0.581
8	0.402	0.606	0.616
9	0.399	0.635	0.648

表 3.2: TFIDF による学習性能 (P30) の評価:提示文書数 20

M	Rocchio	SVM-S	SVM-A
0	0.176	0.176	0.176
1	0.268	0.252	0.252
2	0.317	0.392	0.382
3	0.342	0.490	0.496
4	0.354	0.573	0.585

さらに注目すべきは，SVM-A がほぼすべての場合で SVM-S よりも優れた学習性能を示していることである．SVM-S では，その時点における最適判別超平面が真の判別超平面のよい近似であることを仮定しているが，正例が少なく，負例が大量にある文書検索において，この仮定が成り立ちにくいためと考えられる．

以上をまとめるに，学習性能において，提案法 SVM-A はある程度のフィードバック（30 文書以上の判定）があれば，従来のシステムおよび単純な能動的文書提示のシステムよりも高い性能を示すことがわかった．

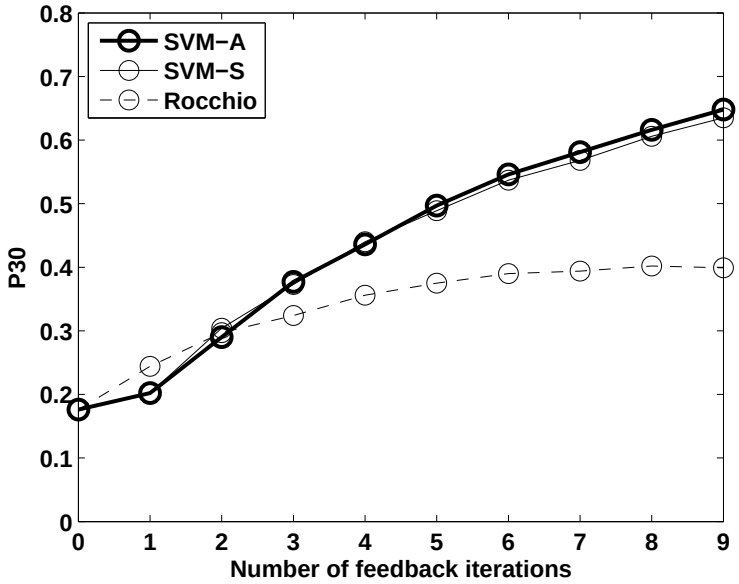


図 3.5: TFIDF による学習性能 (P30) の評価:提示文書数 10

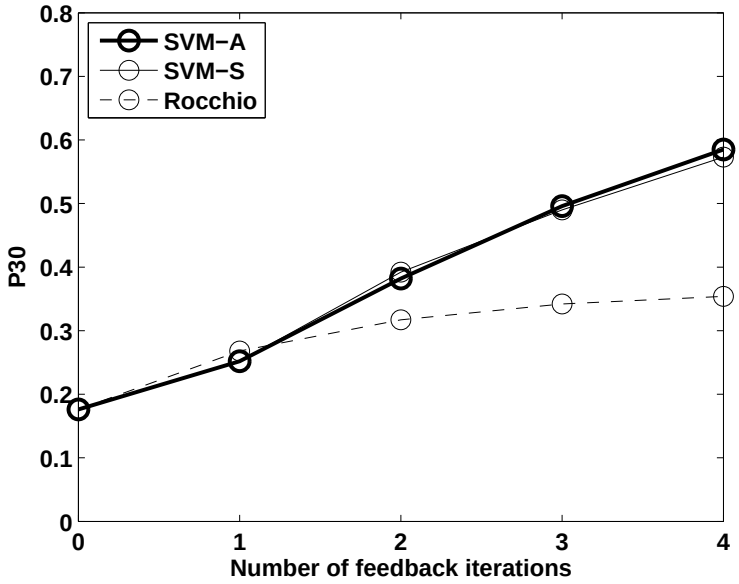


図 3.6: TFIDF による学習性能 (P30) の評価:提示文書数 20

表 3.3: TFIDF による検索性能 (P) の評価:提示文書数 10

M	Rocchio	SVM-S	SVM-A
0	0.227	0.227	0.227
1	0.243	0.158	0.208
2	0.251	0.171	0.231
3	0.247	0.178	0.251
4	0.242	0.188	0.254
5	0.233	0.197	0.260
6	0.227	0.200	0.263
7	0.217	0.204	0.263
8	0.208	0.203	0.262
9	0.200	0.206	0.261

表 3.4: TFIDF による検索性能 (P) の評価:提示文書数 20

M	Rocchio	SVM-S	SVM-A
0	0.194	0.194	0.194
1	0.202	0.135	0.188
2	0.201	0.157	0.206
3	0.190	0.172	0.220
4	0.180	0.180	0.229

検索性能の評価

表 3.3 と図 3.7 から提示文書数が 10 のときはフィードバック回数が 3 回以上で，表 3.4 と図 3.8 から，提示文書数が 20 のときはフィードバック回数が 2 回以上で，つまり，フィードバック文書数が 30 文書を越えると，SVM-A が Rocchio を凌駕して最も高い検索性能を示している．初期のフィードバックで，SVM-A の性能が Rocchio より劣っている原因として，1 回目のフィードバックではシステムが能動的に学習データを選択しておらず，能動学習の効果が表れていないこと，そして Rocchio ベースシステムでは，SVM-A，SVM-S とは異なり，フィードバックの際に常に適合文書の可能性の高い順に提示を行っていることが考えられる．また，注

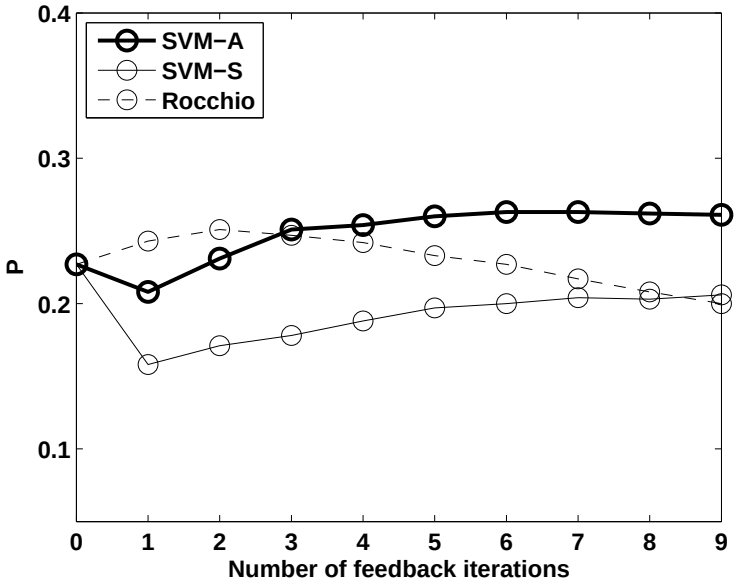


図 3.7: TFIDF による検索性能 (P) の評価:提示文書数 10

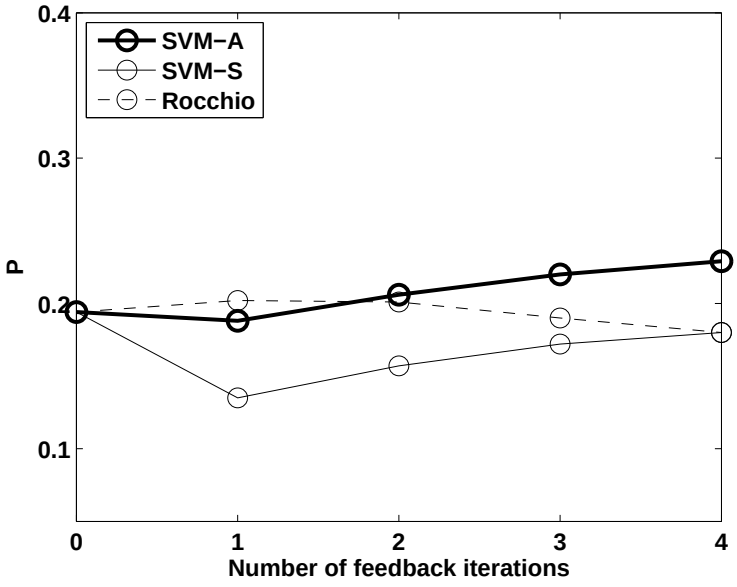


図 3.8: TFIDF による検索性能 (P) の評価:提示文書数 20

目すべきは，SVM-S が，ほとんどの場合で SVM-A のみならず，Rocchio よりもパフォーマンスが悪いことである．これにより，検索性能においては，サポートベクターマシンベースの文書検索であっても単純な能動的文書提示では Rocchio より劣ることを意味し，提案手法のようなより優れた能動的文書提示が必須なことがわかる．また，前節の結果と合わせると，SVM-A は，検索性能を大きく改善した上で，学習性能にも向上がみられることになる．

まとめると，検索性能においても，提案法 SVM-A はある程度のフィードバック（30 文書以上の判定）があれば，従来のシステム，あるいは単純な能動的文書提示のシステムよりも高い性能を示すことがわかった．

最後に，学習にかかる計算時間は，0.18〜4.05 sec (CPU: Xeon3.6GHz, RAM: 2GB, OS: linux 2.6.8, ライブラリ: libsvm-2.71) であった．サポートベクターマシンの計算コストは，データの次元数より訓練データ数に依存するため，訓練データ数が数十と少ない SVM-A, SVM-S では，学習に要する計算時間は短くなっている．

文書ベクトル分布の変化

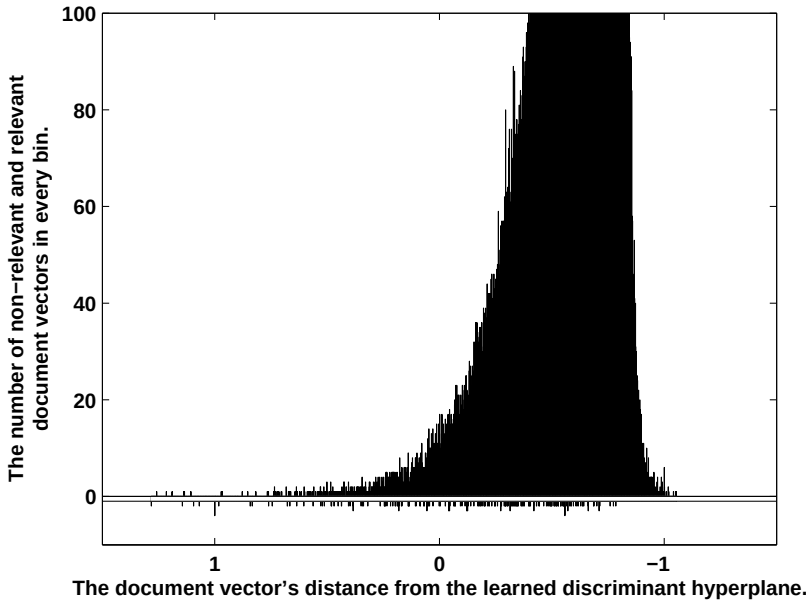
図 3.9 から図 3.12 にフィードバック 1 回目と 2 回目の判別超平面からの距離に対する文書ベクトルの分布の例を示す．図 3.9 の上図は，図 3.3 と同じものである．

これらの図から，非適合文書の分布が，マージン内から −1 で示される非適合領域にシフトしているのがわかる．この傾向は，ほぼすべてのトピックで見ることができた．

このように，非適合文書の分布が非適合文書領域にシフトすることで，非適合文書が提示されにくくなり，フィードバック時に適合文書が得られやすくなっていると考えられる．

3.3.3 トランスダクティブ学習の適用可能性

トランスダクティブ学習は，少数データの判定結果とデータ間の類似性を利用して，ラベルなしデータも含めて，小さな誤りで分類をおこなう方法である．判定文書は文書集合全体に対してごく少数であることから，トランスダクティブ学習の適用可能性について考察を行う．



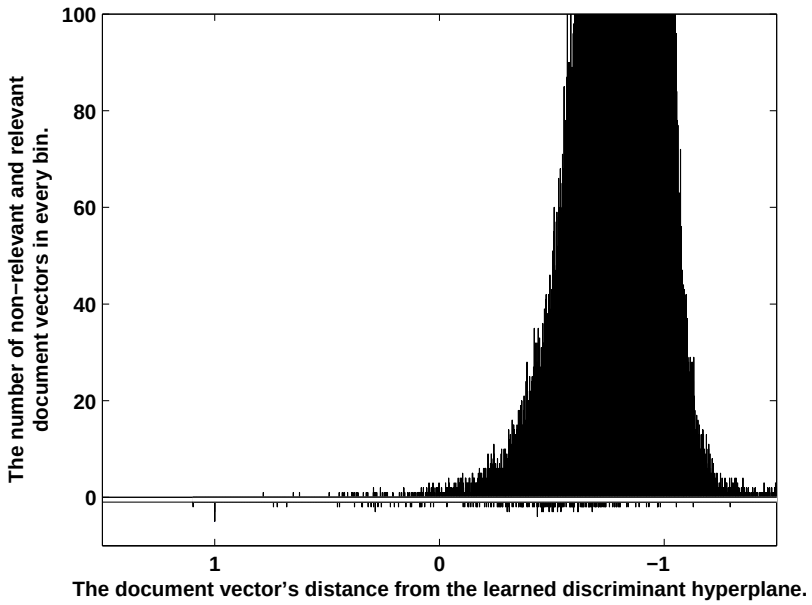
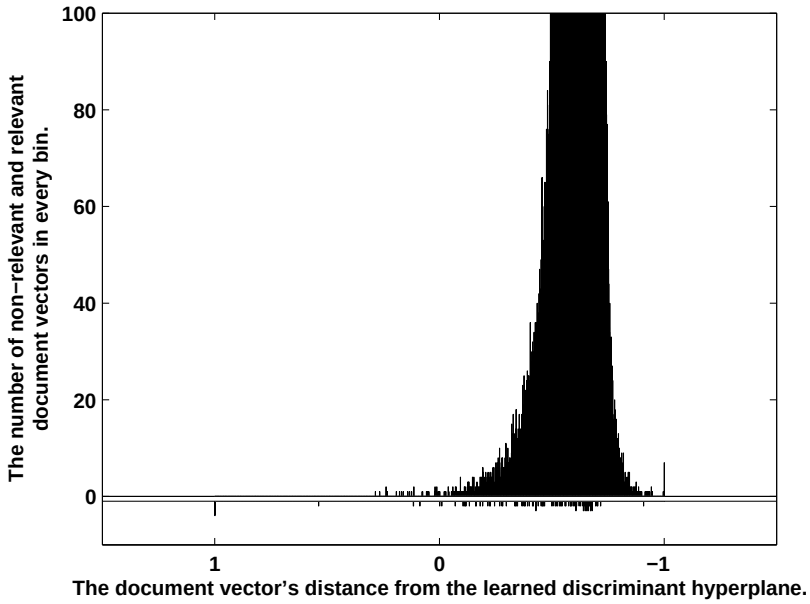


図 3.9: トピック 321 におけるフィードバック後の判別関数値に対する文書ベクトルの分布（上図：1 回目，下図：2 回目）



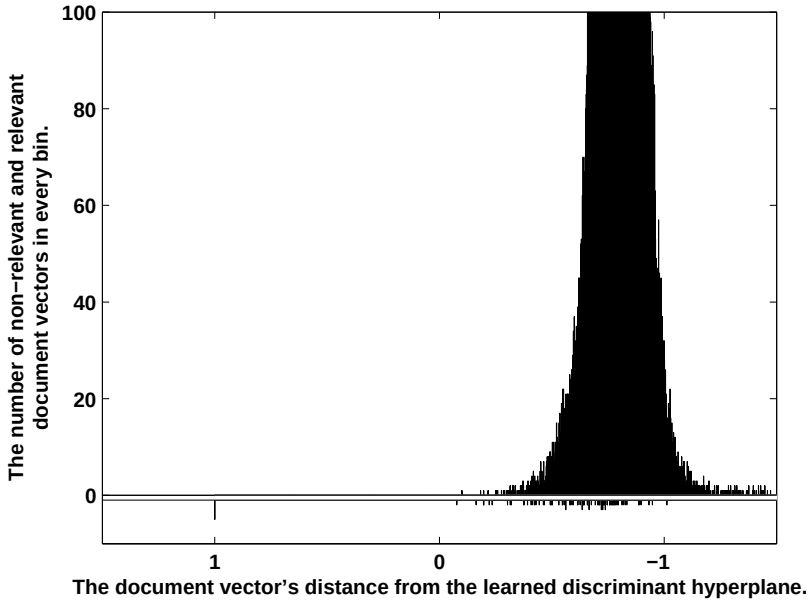
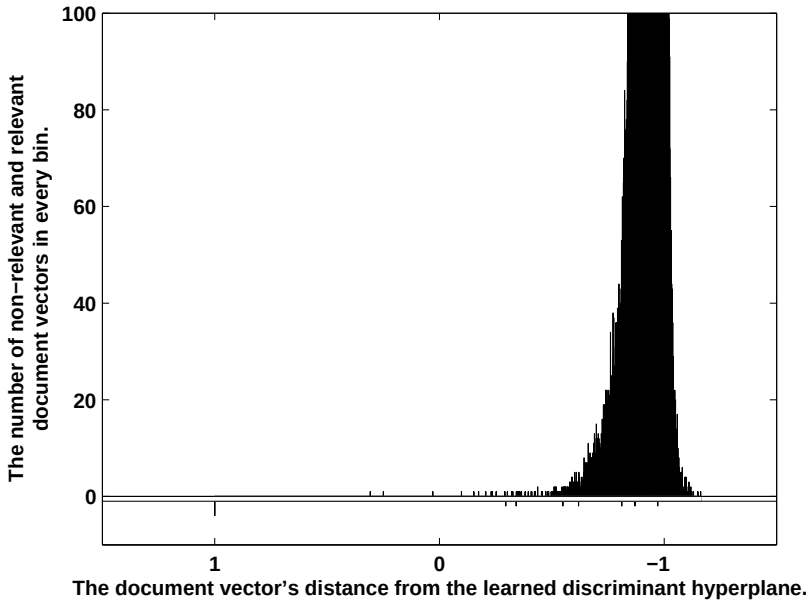


図 3.10: トピック 337 におけるフィードバック後の判別関数値に対する文書ベクトルの分布（上図：1 回目，下図：2 回目）



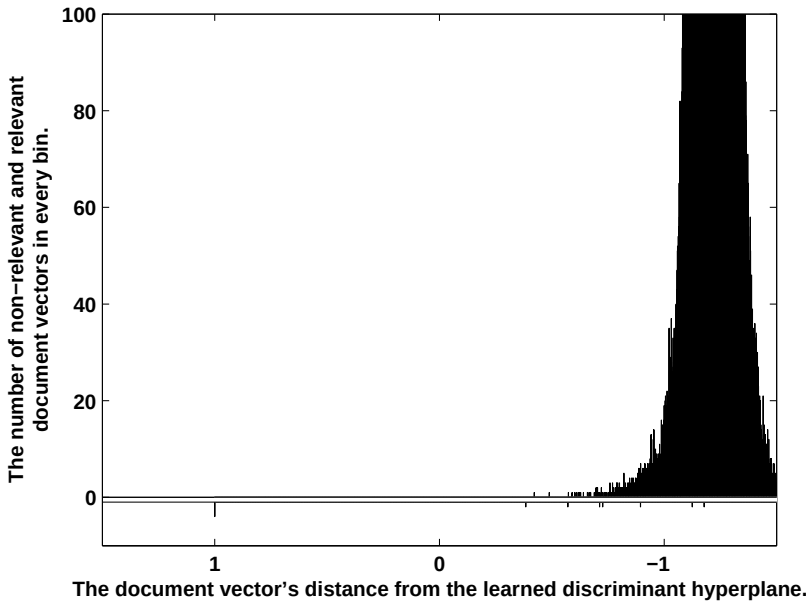
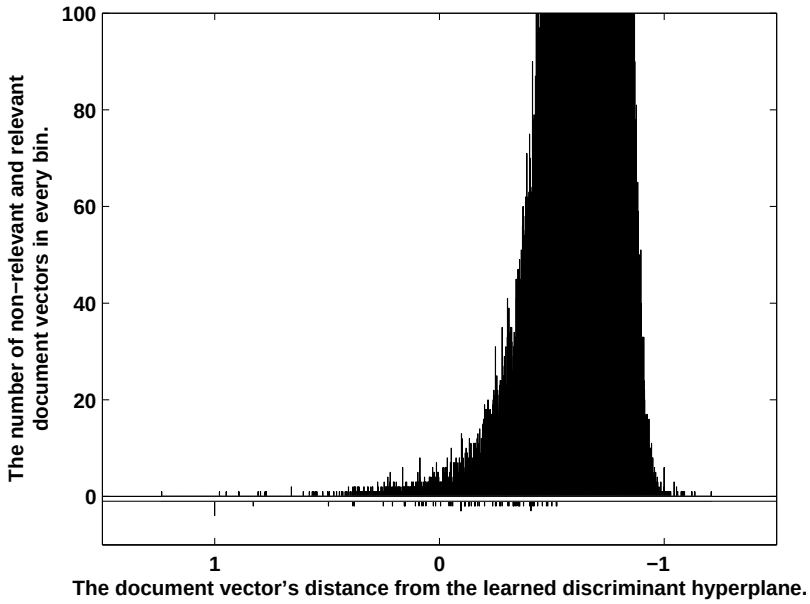


図 3.11: トピック 339 におけるフィードバック後の判別関数値に対する文書ベクトルの分布（上図：1 回目，下図：2 回目）



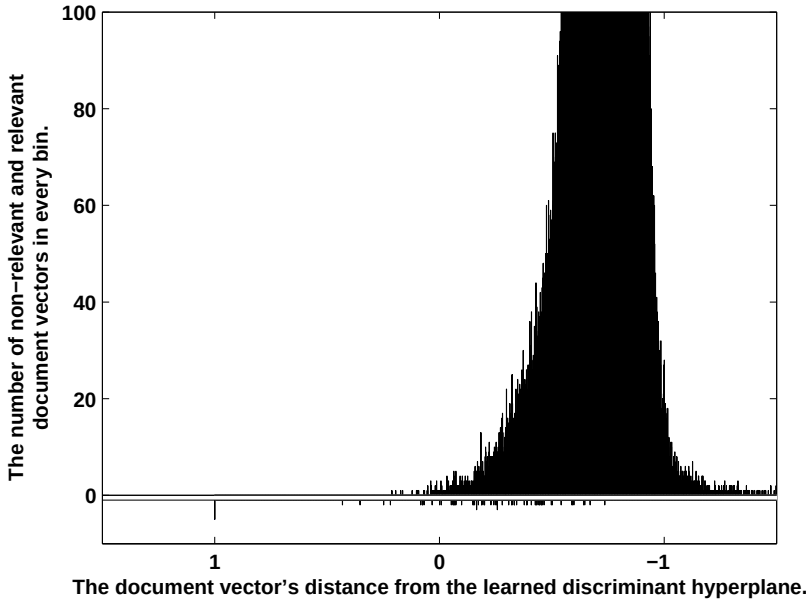


図 3.12: トピック 340 におけるフィードバック後の判別関数値に対する文書ベクトルの分布（上図：1 回目，下図：2 回目）

トランスダクティブ学習のアルゴリズムとしては，提案手法との親和性から，**2.5** で紹介したトランスダクティブ SVM[Joachims 99] について考える．トランスダクティブ SVM のアプローチは，**2.5** でも述べたように，ラベルなしデータに付けたラベルのうち，ラベルが反対でずれが大きいデータのラベルを入れ替えることにより，ラベルなしデータの誤りを小さくするものである．

トランスダクティブ SVM を適用した場合，ラベルなしデータ，つまり未判定の文書にも仮のラベルをつけて判別関数を生成する．その結果，すべての文書が適合，または非適合文書領域に分布し，ヒューリスティクスで提示するマージン領域内のデータが存在しないことになる．このため，トランスダクティブ学習に応じた，新たなヒューリスティクスを考える必要がある．

また，すべての未判定文書をトランスダクティブ学習に用いると，判別関数の生成に時間がかかりすぎるため，一部の未判定文書を用いるアプローチが必要になると考えられる．この場合，どの未判定文書をトランスダクティブ学習に用いるかを選択する方法の検討が必要となる．

3.3.4 他のデータへの適用可能性

提案するヒューリスティクスは，正例と負例の極端な偏りに基づく方法である．対話的な検索を行う場合には，ユーザが判定したデータは全体のごく一部であるため正例と負例の偏りが生じる．このような性質はデータによるものではないので，文書に限らず画像など他のドメインのデータについても適用可能であると考えられる．

また，正例と負例に偏りがないデータの場合には，初期検索で正例が大量に得られるため，対話的な検索を行う必要がないと考えられる．

3.4 文書ベクトル表現の違いによる性能比較

前節では，ベクトル空間モデルにおいて最もよく用いられる文書ベクトル表現である，TFIDF を用いて，提案するヒューリスティクスが有効に機能することを実験的に示した．しかし，サポートベクターマシンを文書処理に適用する場合，文書ベクトル表現の違いによって分類性能が変わってくることが報告されている [Drucker 02]. そこで本節では，文書ベクトル表現を変えて，従来の適合フィードバック手法である Rocchio の

手法と，サポートベクターマシンを用いた従来手法と提案手法を適用することで，文書ベクトル表現の違いによる学習性能と検索性能の比較を行う．

3.4.1 実験条件

基本的な実験条件は，**3.3.1** と同じである．

追加実験するベクトル空間モデルにおける文書ベクトル表現として，ブーリアン (Boolean) と term frequency(TF) を用いた．Boolean は，文書 d_j 内の語 k_i の重みとして，d_j 中にその語が存在するか否かの $\{0, 1\}$ の 2 値を使うものである．また，TF は文書 d_j 内の語 k_i の頻度によって重みを表す．

3.4.2 実験結果

学習性能の比較

表 3.5 から表 3.8 と図 3.13 から図 3.16 に学習性能の評価結果を示す．

Boolean については，表 3.5 と図 3.13 から提示文書数が 10 のときはフィードバック回数が 2 回以上になると，表 3.6 と図 3.14 から，提示文書数が 20 のときについてもフィードバック回数が 2 回以上になると，つまり，フィードバック文書数が 20 文書を越えるあたりから，SVM-A と SVM-S は，常に Rocchio よりも優れた性能を示している．これは，TFIDF の場合と同様に，サポートベクターマシンに基づくフィードバック手法の優れた学習性能が実験的に示されたことを意味する．

また，SVM-A がほぼすべての場合で SVM-S よりも優れた学習性能を示しているのは，TFIDF の場合と同じである．

これに対して，TF では，表 3.7 と図 3.15 から提示文書数が 10 のときはフィードバック回数が 6 回以上になると，表 3.8 と図 3.16 から，提示文書数が 20 のときについてもフィードバック回数が 3 回以上になると，つまり，フィードバック文書数が 60 文書を越えると，SVM-A と SVM-S は，常に Rocchio よりも優れた性能を示している．学習性能のグラフが，Rocchio の場合はほぼ一定に近いのに対して，サポートベクターマシンを用いる場合には上がっていくのは，TFIDF や Boolean と同様であるが，1 回目のフィードバックでの性能の落ち方が大きい．これは，前節で考察した理由の他にも原因があると推測される．

表 3.5: Boolean による学習性能 (P30) の評価:提示文書数 10

M	Rocchio	SVM-S	SVM-A
0	0.101	0.101	0.101
1	0.123	0.082	0.082
2	0.148	0.168	0.162
3	0.156	0.223	0.221
4	0.175	0.276	0.287
5	0.181	0.315	0.327
6	0.189	0.349	0.358
7	0.192	0.380	0.387
8	0.199	0.404	0.412
9	0.200	0.429	0.436

表 3.6: Boolean による学習性能 (P30) の評価:提示文書数 20

M	Rocchio	SVM-S	SVM-A
0	0.101	0.101	0.101
1	0.141	0.117	0.117
2	0.143	0.241	0.240
3	0.148	0.317	0.331
4	0.152	0.375	0.387

また，SVM-A と SVM-S を比較した場合には，SVM-A がほぼすべての場合で優れた学習性能を示しており，提案手法がサポートベクターマシンにおける文書選択において有効であることがわかる．

検索性能の評価

表 3.9 から 表 3.12 と図 3.17 から図 3.20 に検索性能の評価結果を示す．

Boolean については，表 3.9 と図 3.17 から提示文書数が 10 のときはフィードバック回数が 3 回以上で，表 3.10 と図 3.18 から，提示文書数が 20 のときはフィードバック回数が 2 回以上で，つまり，フィードバック文書数が 30 文書を越えると，SVM-A が Rocchio を凌駕して最も高い検

表 3.7: TF による学習性能 (P30) の評価:提示文書数 10

M	Rocchio	SVM-S	SVM-A
0	0.137	0.137	0.137
1	0.136	0.024	0.024
2	0.152	0.035	0.043
3	0.149	0.071	0.090
4	0.154	0.110	0.138
5	0.160	0.144	0.170
6	0.164	0.184	0.212
7	0.174	0.207	0.243
8	0.170	0.230	0.283
9	0.170	0.256	0.305

表 3.8: TF による学習性能 (P30) の評価:提示文書数 20

M	Rocchio	SVM-S	SVM-A
0	0.137	0.137	0.137
1	0.110	0.018	0.018
2	0.115	0.057	0.065
3	0.117	0.117	0.145
4	0.120	0.190	0.218

索性能を示している．これは，TFIDF の場合と同様に，提案手法である SVM-A が，検索性能を大きく改善した上で，学習性能にも向上がみられることを表している．

一方，TF については，表 3.11，表 3.12 と図 3.19，図 3.20 からほぼすべての場合で，Rocchio の性能を上回ることができないことがわかる．しかしながら，SVM-A と SVM-S を比較した場合には，SVM-A の性能が常に上回っているため，サポートベクターマシンを用いた適合フィードバックにおいて，提案手法である能動学習が有効に機能していることがわかる．この Rocchio との性能差は，TF の文書ベクトル表現が，サポートベクターマシンにおいてなんらかの問題点を含んでいることを示唆している．

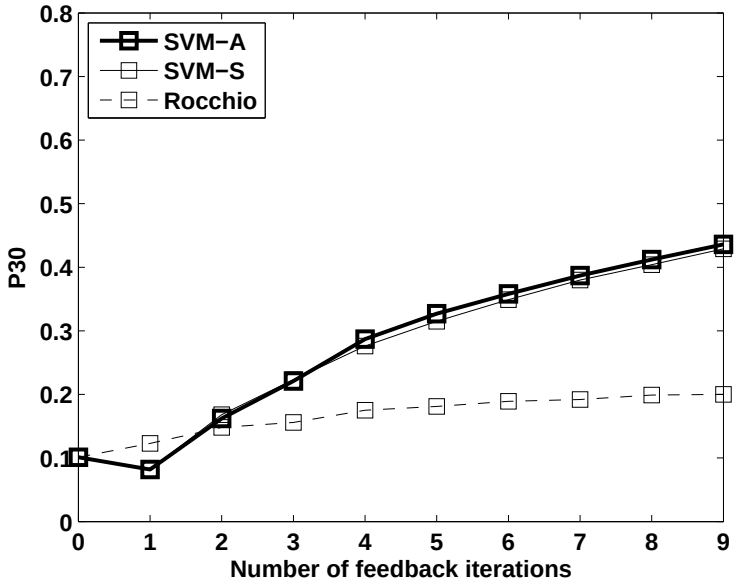


図 3.13: Boolean による学習性能 (P30) の評価:提示文書数 10

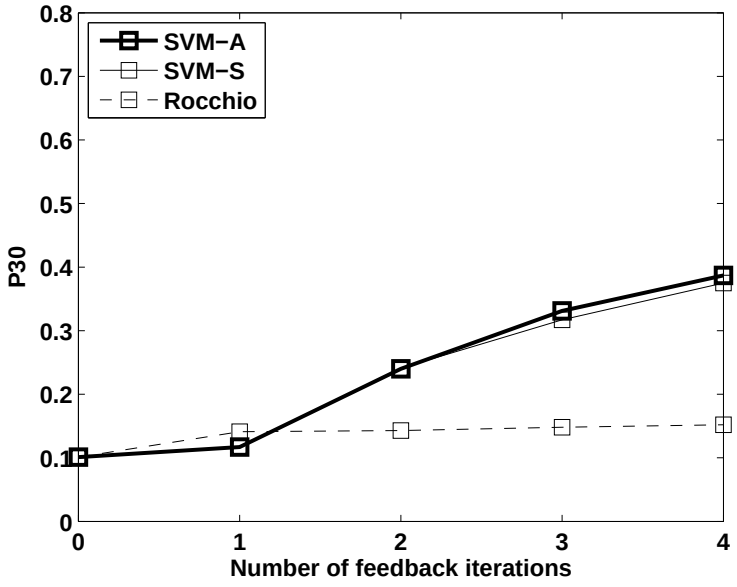


図 3.14: Boolean による学習性能 (P30) の評価:提示文書数 20

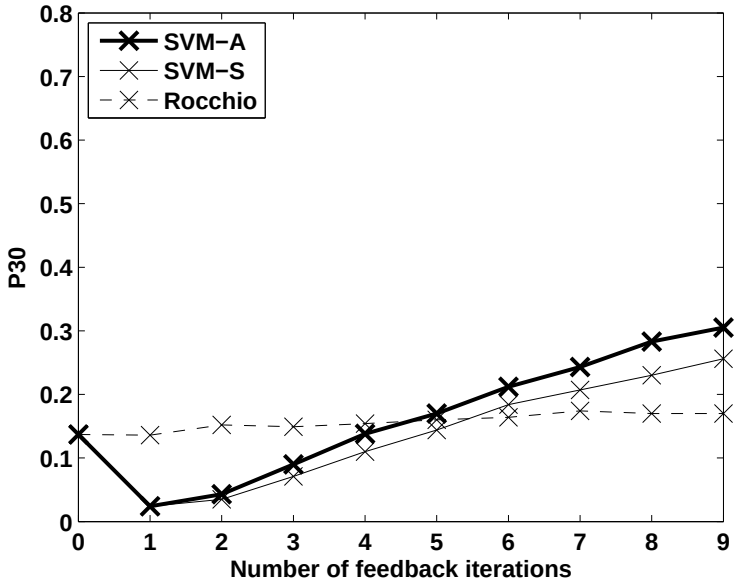


図 3.15: TF による学習性能 (P30) の評価:提示文書数 10

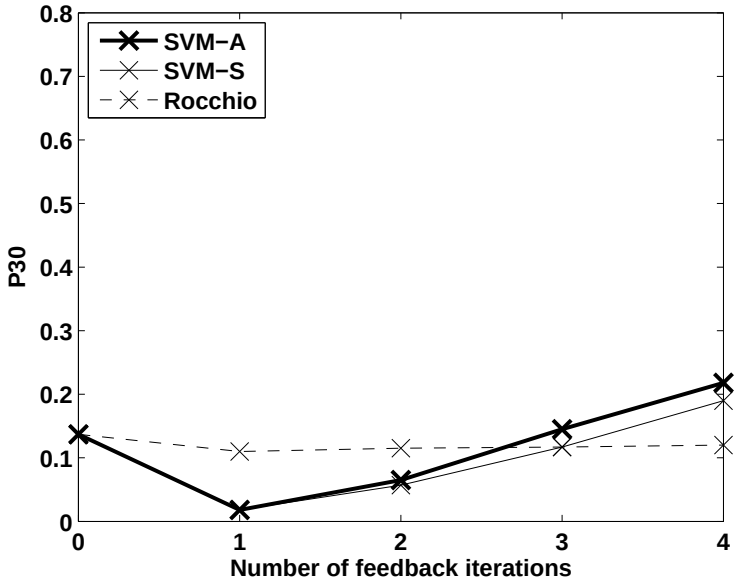


図 3.16: TF による学習性能 (P30) の評価:提示文書数 20

表 3.9: Boolean による検索性能 (P) の評価:提示文書数 10

M	Rocchio	SVM-S	SVM-A
0	0.142	0.142	0.142
1	0.136	0.087	0.107
2	0.136	0.093	0.122
3	0.128	0.107	0.137
4	0.125	0.118	0.155
5	0.120	0.126	0.163
6	0.116	0.127	0.168
7	0.111	0.130	0.172
8	0.106	0.134	0.173
9	0.102	0.137	0.172

表 3.10: Boolean による検索性能 (P) の評価:提示文書数 20

M	Rocchio	SVM-S	SVM-A
0	0.117	0.117	0.117
1	0.108	0.073	0.100
2	0.101	0.089	0.123
3	0.092	0.103	0.138
4	0.085	0.113	0.146

表 3.11: TF による検索性能 (P) の評価:提示文書数 10

M	Rocchio	SVM-S	SVM-A
0	0.197	0.197	0.197
1	0.171	0.110	0.114
2	0.158	0.078	0.086
3	0.142	0.068	0.075
4	0.131	0.066	0.074
5	0.122	0.064	0.078
6	0.116	0.069	0.081
7	0.111	0.070	0.084
8	0.105	0.071	0.087
9	0.101	0.071	0.088

表 3.12: TF による検索性能 (P) の評価:提示文書数 20

M	Rocchio	SVM-S	SVM-A
0	0.154	0.154	0.154
1	0.118	0.082	0.086
2	0.100	0.063	0.068
3	0.089	0.060	0.068
4	0.078	0.061	0.071

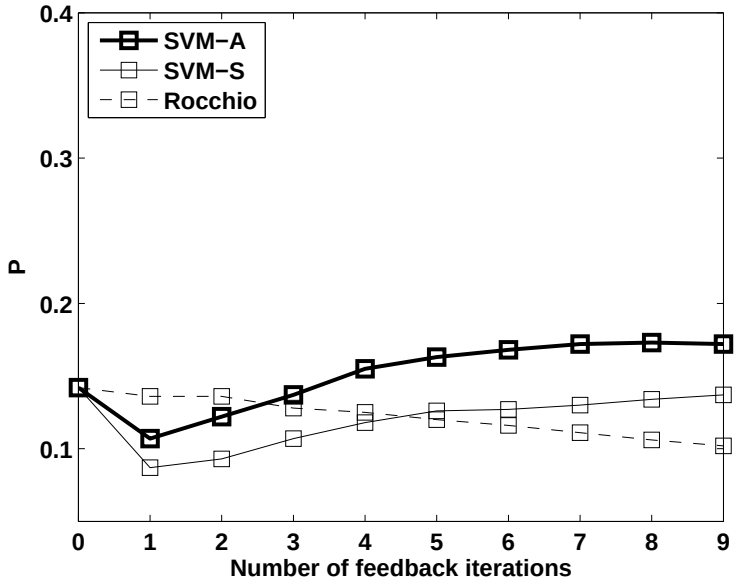


図 3.17: Boolean による検索性能 (P) の評価:提示文書数 10

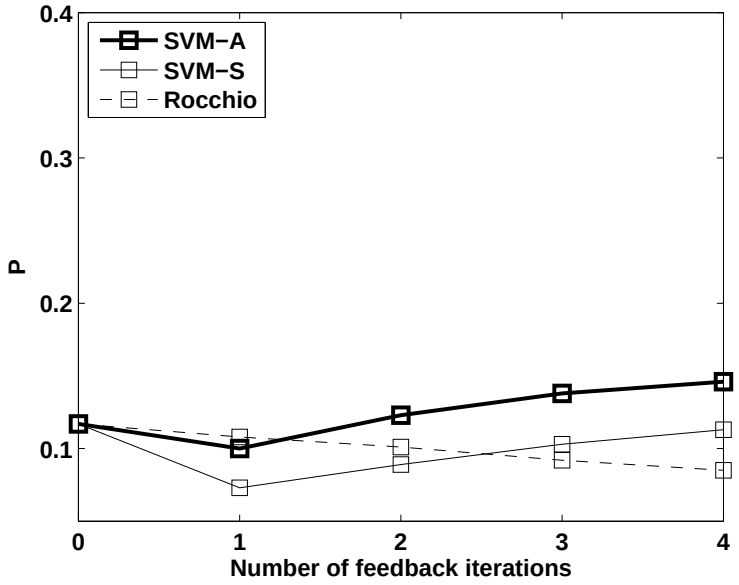


図 3.18: Boolean による検索性能 (P) の評価:提示文書数 20

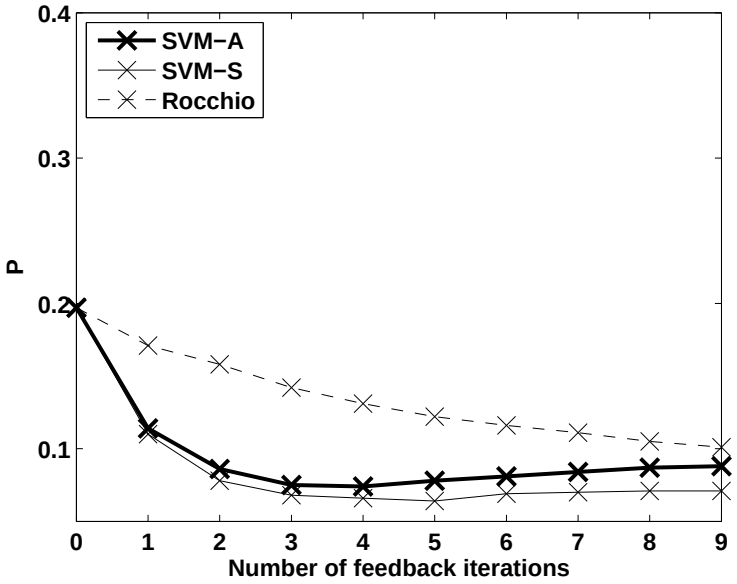


図 3.19: TF による検索性能 (P) の評価:提示文書数 10

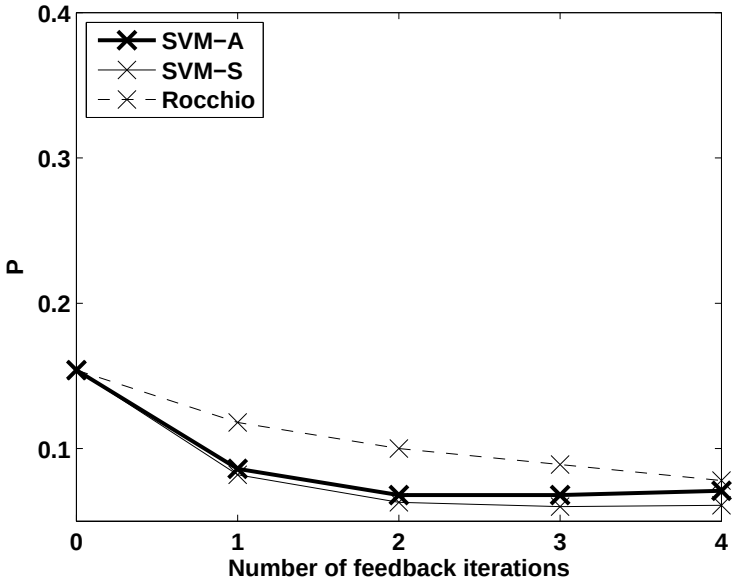


図 3.20: TF による検索性能 (P) の評価:提示文書数 20

第4章　サポートベクターマシンに基づく対話的文書検索の比較分析

前章では，サポートベクターマシンを用いた能動的文書検索において，ユーザへの文書提示のヒューリスティクスについて有効性を実験的に示した．しかし，文書ベクトル表現によっては，従来法である Rocchio の手法のパフォーマンスを下回る結果しか得られない場合が存在した．提案したヒューリスティクスは，どの文書を提示すればよいかの指標は示しているものの，サポートベクターマシンを用いた際の文書の順位付け方法を規定しているものではない．

文書の順位付けは，その文書が適合文書にどれだけ似通っているかを判定する適合度を計算することで行う．サポートベクターマシンを用いた対話的文書検索における文書の適合度は，ベクトル空間モデル上の符号付距離であるが，これが，ベクトル空間モデル上でどのような特性を持つのかは，明らかになっていない．

そこで本章では，サポートベクターマシンにおける距離を用いた適合度を定式化し，対話的文書検索における従来手法である Rocchio の手法との比較分析を行った．また，そこから得られた知見より，サポートベクターマシンに基づく手法に適したカーネルを提案し，本手法の有効性を検証するために検証実験を行う [Murata 10][村田 11].

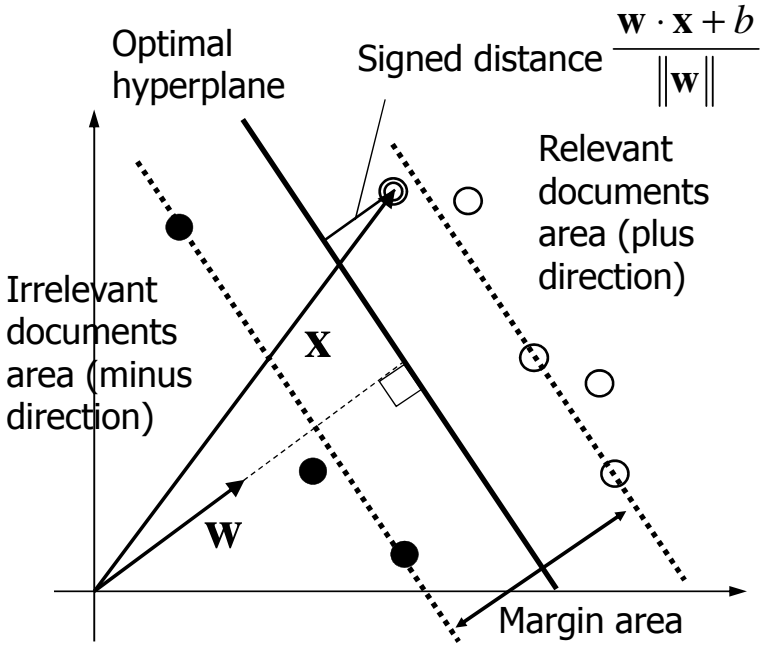


図 4.1: サポートベクターマシンに基づく適合フィードバック

4.1 サポートベクターマシンに基づく適合フィードバックとRocchioアルゴリズムの比較分析

4.1.1 適合フィードバック文書検索の比較分析

図 4.1 にサポートベクターマシンに基づく適合フィードバックの概念図を示す．図中の○と●は，それぞれ判定済みの適合文書と非適合文書であり，判定済み文書ベクトルを $\mathbf{x}_i$，クラスラベルをそれぞれ $y_i = 1$, $y_i = -1$ とする．このとき未判定の文書ベクトル $\mathbf{x}$ に対する，判別超平面の関数は次のように表される．

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b \tag{4.1}$$

ここで，$\mathbf{w}$ は判別超平面の法線ベクトル，b はバイアス項と呼ばれる定数である．

図 4.1 中の◎で表される未判定文書ベクトル $\mathbf{x}$ の判別超平面からの符号

付距離は,

$$\begin{aligned}\frac{\mathbf{w}\cdot\mathbf{x}+b}{||\mathbf{w}||} &= \frac{||\mathbf{w}||||\mathbf{x}||\cos\theta_w + b}{||\mathbf{w}||} \\ &= ||\mathbf{x}||\cos\theta_w + \frac{b}{||\mathbf{w}||} \end{aligned} \tag{4.2}$$

となる．ここで，θ_w は $\mathbf{w}$ と $\mathbf{x}$ のなす角である．

一方，**2.2** の式 (2.3) から $\mathbf{w}$ は，適合文書のラベルが $y_i = 1$，非適合文書のラベルが $y_i = -1$ なので,

$$\mathbf{w} = \sum_j \alpha_j \mathbf{x}_j - \sum_k \alpha_k \mathbf{x}_k \tag{4.3}$$

となる．ここで，j は適合文書，k は非適合文書を示す添字である．

これに対し，式 (2.18) で示した Roochio の方法でのクエリベクトル更新式は，次のように変形できる．

$$\mathbf{Q}_{m+1} = \mathbf{Q}_0 + \sum_j \beta \mathbf{x}_j - \sum_k \gamma \mathbf{x}_k \tag{4.4}$$

ここで，$\mathbf{Q}_0$ は初期クエリベクトル（ユーザが最初に与えたクエリのベクトル）である．

式 (4.3) と式 (4.4) を比較すると，サポートベクターマシンに基づく適合フィードバックのベクトル $\mathbf{w}$ の式は，Rocchio のクエリベクトル更新式において初期クエリベクトルがゼロベクトルの場合と同等であることがわかる．

適合フィードバックでは，サポートベクターマシンと Rocchio のいずれにおいても，初期クエリを用いてコサイン類似度による適合度評価を行い，それに基づいて初期提示文書を決定する．そのため，初期提示文書は初期クエリに含まれる単語のいずれかを必ず含んでいる．一方，Rocchio のクエリベクトル更新式では，初期クエリベクトルに，提示文書から選ばれたフィードバック文書のベクトルを重みつきで加算していくので，フィードバックを繰り返すごとに初期クエリベクトルの影響は小さくなると考えられる．

また，式 (4.4) では，すべての文書が更新に関与しているのに対して，式 (4.3) では，サポートベクターにならない文書，つまり $\alpha_i = 0$ の文書は関与しないことになる．これは，Rocchio ベースの手法が，適合文書と非適合文書の重み係数として定数を用いているのに対して，サポートベ

クターマシンベースの手法では，サポートベクターマシンでの重み係数 α_i の値が 0 となってサポートベクターにならないことも含めて，重みづけを行っていると考えることができる．

これらのことから，サポートベクターマシンに基づく適合フィードバックのベクトル $\mathbf{w}$ の式は，Rocchio のクエリベクトル更新式の近似となっており，式 (4.3) の $\mathbf{w}$ を Rocchio ベース適合フィードバックのクエリベクトルと捉えることができる．

また，Rocchio ベースの手法で適合／非適合の影響を考慮するパラメータ β，γ は，全部の適合／非適合文書について共通であり，これらの値は試行錯誤的に決定している．一方，サポートベクターマシンによる手法では，適合文書と非適合文書が完全分離でき，$||\mathbf{w}||$ が最小となる，つまり経験的リスクが 0 となり，構造的リスクが最小になる判別超平面を選択するように [Vapnik 98][小野田 07a]，個々の文書ベクトルに対して α_i を決定していることになる．

4.1.2 比較分析に基づく文書表現の改善

前節で，サポートベクターマシンに基づく適合フィードバックのベクトル $\mathbf{w}$ の式は，Rocchio のクエリベクトル更新式と同等であることを示した．また，サポートベクターマシンによる手法での適合度の評価は，$\mathbf{w}$ と b が判定済み文書ベクトルにより一意に決まることから，式 (4.2) より，$||\mathbf{x}|| \cos\theta_w$ を評価していることになる．一方，Rocchio の手法では文書ベクトルとクエリベクトルとの角度を θ_q としたときのコサイン類似度，つまり $\cos\theta_q$ を評価している．

以上の比較分析から，サポートベクターマシンによる手法では，対象となる文書ベクトルが大きい，つまり $||\mathbf{x}||$ が大きいと，適合度が高くなることになる．これは，$\mathbf{w}$ との θ_w が小さい未判定文書より，$||\mathbf{x}||$ が極端に大きい未判定文書のほうが適合度が高くなることを意味する．つまり，単語を多く含む文書は適合度が高くなり，反対に単語の少ない文書は適合度が低くなってしまう．このようなことを避けるために，適合度をベクトル空間モデルで一般的である純粋なコサイン類似度，つまり $\cos\theta_w$ のみにしたほうがよいと考えられる．そのためには，$||\mathbf{x}||$ を定数にする必要がある．

これを実現する方法としては，文書ベクトルを単位ベクトルに正規化することが考えられるが，本論文では，サポートベクターマシンのカー

ネル関数を用いることで，文書ベクトルを処理することなく同様の効果を得ることを目指す．

次章では，文書の単位ベクトル化と文書ベクトル間のコサイン類似度の関係性から，コサイン類似度をサポートベクターマシンのカーネル関数とするコサインカーネルの提案を行う．

4.2 コサインカーネルの提案

サポートベクターマシンでは，**2.2.1** で述べたように，非線形判別を行うため，カーネルトリックと呼ばれる方法を用いる．

このカーネル $K(\mathbf{x}, \mathbf{x}')$ として，二つのベクトル $\mathbf{x}$ と $\mathbf{x}'$ のなす角を θ としたときのコサイン類似度を用いることを考える．このとき，

$$K(\mathbf{x}, \mathbf{x}') = \cos\theta = \frac{\mathbf{x} \cdot \mathbf{x}'}{||\mathbf{x}||||\mathbf{x}'||} \tag{4.5}$$

となる．これは，正定値カーネルであることを表す Mercer の定理を満たしている．この式を見ると，ベクトルのコサイン類似度をカーネルに用いることは，前章で提案した文書ベクトルの単位ベクトル化と同じであることがわかる．このコサイン類似度を用いたカーネルを，コサインカーネルと呼ぶこととする．

本研究で提案したコサインカーネルは，線形カーネルを正規化したものととらえることもできる．正規化線形カーネルについては，すでに提案されており有効性が示されている [Hotta 10]．しかし，その導出は非線形カーネルにおける正規化の効果から直感的に用いているものである．我々は，対話的文書検索における適合性評価の比較分析からカーネルを導出しており，その有効性の原因を示している点が大きく異なる．

コサインカーネルを用いることにより，サポートベクターマシンにおける距離を用いた適合度の評価においても，ベクトル空間モデルと同様の評価を行うことになる．この改善方法の有効性は，次章で実験的に検証される．

4.3 比較実験

4.2 では，一般に用いられている適合度であるベクトルのコサイン類似度とサポートベクターマシンにおける適合度である判別超平面からの距

離との関係から，サポートベクターマシンに基づく適合フィードバックにコサイン類似度を導入する，コサインカーネルを提案した．この方法により，理論的には適合フィードバックの性能向上が期待できるが，実際に改善されるか，またどの程度の改善になるのかについては，実験的に検証する必要がある．そこで，実際の文書データセットを用いた比較実験を行った．

4.3.1 実験条件

実験条件は，**3.3.1** と同じである．ただし，前章でも述べたように，サポートベクターマシンの分類性能は文書ベクトル表現のベクトル空間に依存するため，提案するコサインカーネルによる性能の比較評価を行うのと同時に，文書ベクトル表現の違いによる比較評価を行った．なお，本研究の目的が，ベクトル空間モデルにおけるサポートベクターマシンベースと Rocchio ベースの適合フィードバックの比較分析と，それによりサポートベクターマシンベースの適合フィードバックを改善することから，Okapi BM25 [Robertson 96] をはじめとする確率モデルとの実験的比較は行っていない．

各文書ベクトル表現は，Boolean，TF と TFIDF の 3 種類を比較した．

4.3.2 実験結果

4.2 で提案したコサインカーネルの効果を見るため，カーネル使用の有無による検索および学習性能を比較した．提示文書数 S が 10 と 20 のときの学習性能 $P30$ を表 4.1，表 4.2 と図 4.2，図 4.3 に示す．検索性能 P を表 4.3，表 4.4 と図 4.4，図 4.5 に，ここで，フィードバック回数 0 は，初期検索時の性能を表す．また，太線がコサインカーネルを，細線が線形分離した場合を表す．

これらの図から，提案するコサインカーネルによって，TF の性能が大きく向上することがわかる (図中の×のグラフの比較)．

さらに，提案手法と Rocchio ベースのシステムとの性能比較を行った．Rocchio の係数については，前章と同じ値 ($\beta = 1.0, \gamma = 0.5$) を用いた．最も一般的な文書ベクトル表現である TFIDF における，学習性能 $P30$ の比較結果を表 4.5，表 4.6 と図 4.6，図 4.7 に検索性能 P の比較結果を表 4.7，表 4.8 と図 4.8，図 4.9 に示す．

表 4.1: 提示文書数 $S = 10$ のときの学習性能 $P30$

M	Boolean	Boolean cos	TF	TF cos	TFIDF	TFIDF cos
0	0.101	0.101	0.137	0.137	0.176	0.176
1	0.082	0.108	0.024	0.155	0.202	0.208
2	0.162	0.179	0.043	0.262	0.290	0.322
3	0.221	0.249	0.090	0.323	0.377	0.382
4	0.287	0.295	0.138	0.381	0.436	0.459
5	0.327	0.326	0.170	0.433	0.497	0.499
6	0.358	0.358	0.212	0.475	0.546	0.542
7	0.387	0.381	0.243	0.505	0.581	0.583
8	0.412	0.413	0.283	0.538	0.616	0.609
9	0.436	0.429	0.305	0.547	0.648	0.632

表 4.2: 提示文書数 $S = 20$ のときの学習性能 $P30$

M	Boolean	Boolean cos	TF	TF cos	TFIDF	TFIDF cos
0	0.101	0.101	0.137	0.137	0.176	0.176
1	0.117	0.169	0.018	0.209	0.252	0.271
2	0.240	0.257	0.065	0.323	0.382	0.430
3	0.331	0.332	0.145	0.426	0.496	0.509
4	0.387	0.384	0.218	0.498	0.585	0.582

学習性能 $P30$ については，表 4.5，表 4.6 と図 4.6，図 4.7 より，$S = 10$ のとき $M = 2$ 以上で，$S = 20$ のとき $M = 2$ 以上でサポートベクターマシンの性能が Rocchio を上回っている．つまり，フィードバック文書数が 20 文書を越えるあたりから，提案手法が Rocchio より高い学習性能を示している．同様の傾向は検索性能 P にも見られ，表 4.7，表 4.8 と図 4.8，図 4.9 より，$S = 10$ のとき $M = 3$ 以上で，$S = 20$ のとき $M = 2$ 以上でサポートベクターマシンの性能が Rocchio を上回っている．つまり，フィードバック文書数が 30 文書を越えると，提案手法が Rocchio より高い検索性能を示している．

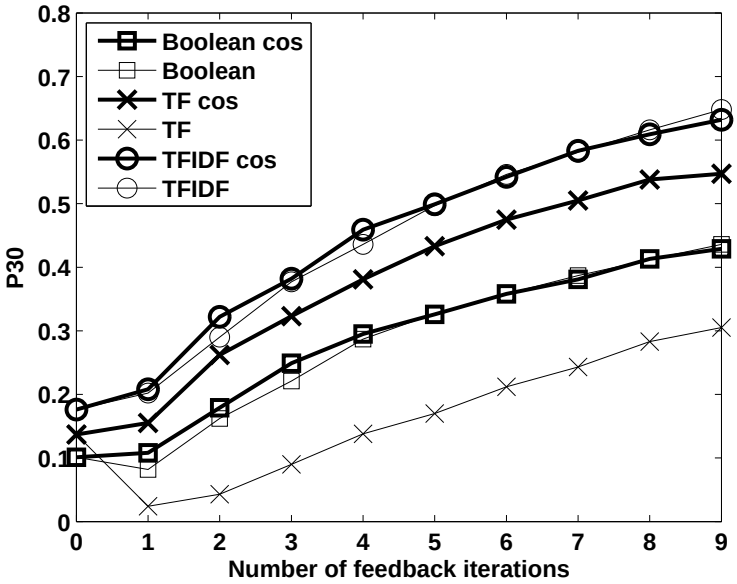


図 4.2: 提示文書数 $S = 10$ のときの学習性能 $P30$

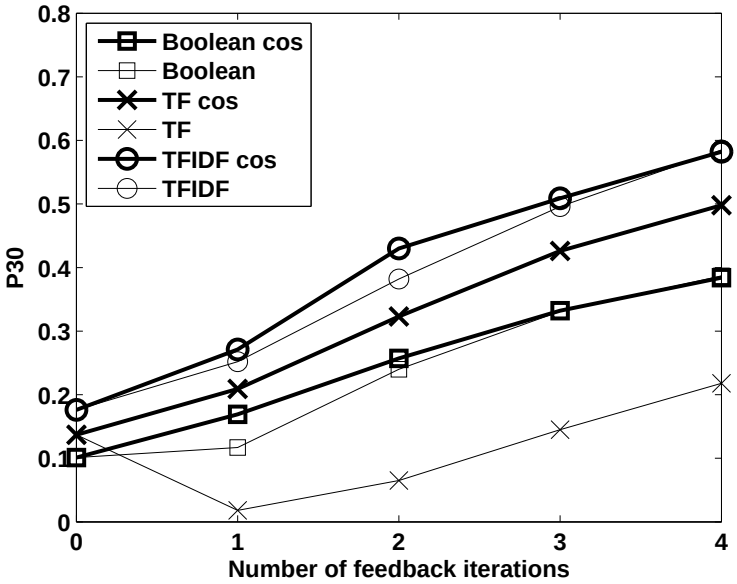


図 4.3: 提示文書数 $S = 20$ のときの学習性能 $P30$

表 4.3: 提示文書数 $S = 10$ のときの検索性能 P

M	Boolean	Boolean cos	TF	TF cos	TFIDF	TFIDF cos
0	0.142	0.142	0.197	0.197	0.227	0.227
1	0.107	0.121	0.114	0.173	0.208	0.215
2	0.122	0.136	0.086	0.201	0.231	0.245
3	0.137	0.156	0.075	0.209	0.251	0.256
4	0.155	0.168	0.074	0.217	0.254	0.264
5	0.163	0.172	0.078	0.220	0.260	0.266
6	0.168	0.172	0.081	0.225	0.263	0.265
7	0.172	0.171	0.084	0.228	0.263	0.264
8	0.173	0.173	0.087	0.226	0.262	0.261
9	0.172	0.171	0.088	0.223	0.261	0.258

表 4.4: 提示文書数 $S = 20$ のときの検索性能 P

M	Boolean	Boolean cos	TF	TF cos	TFIDF	TFIDF cos
0	0.117	0.117	0.154	0.154	0.194	0.194
1	0.100	0.119	0.086	0.152	0.188	0.195
2	0.123	0.134	0.068	0.170	0.206	0.232
3	0.138	0.146	0.068	0.179	0.220	0.236
4	0.146	0.149	0.071	0.183	0.229	0.236

4.4 考察

4.4.1 コサインカーネルの効果

ここでは，サポートベクターマシンに基づく適合フィードバック文書検索において本研究で導入したコサインカーネルの効果について考察する．

コサインカーネルは，表 4.1 から表 4.4 と図 4.2 から図 4.5 において，特に TF で改善方法の効果が大きいことがわかる．この理由として，TF ベクトルは，単語の頻度を重みとしているので，単語数が多い文書が存在すると，ベクトルの大きさが大きくなる．これに対して，コサインカーネルでは，コサイン類似度によりベクトルの大きさではなく角度のみを評価

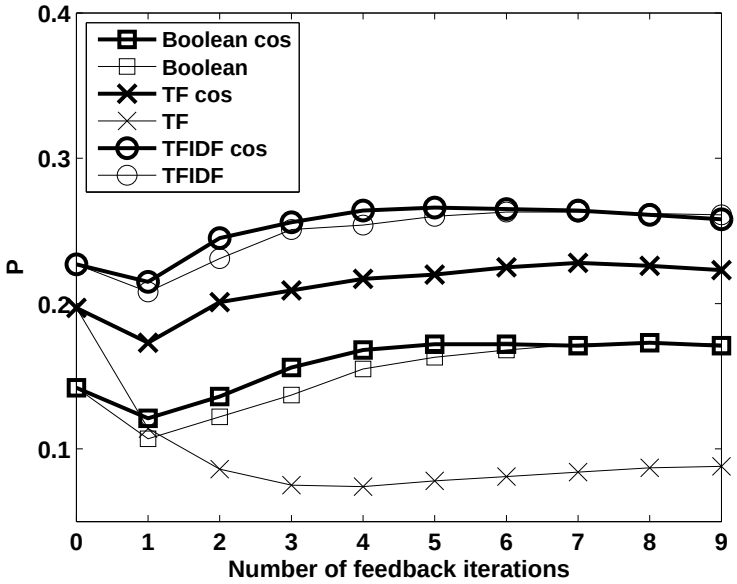


図 4.4: 提示文書数 $S = 10$ のときの検索性能 P

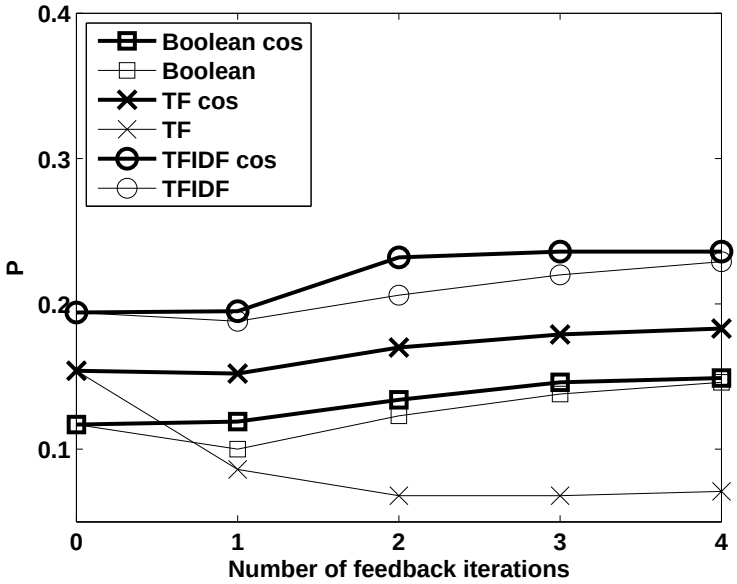


図 4.5: 提示文書数 $S = 20$ のときの検索性能 P

表 4.5: TFIDF における学習性能 $P30$ (提示文書数 $S = 10$)

M	Rocchio	TFIDF cos
0	0.176	0.176
1	0.244	0.208
2	0.297	0.322
3	0.324	0.382
4	0.356	0.459
5	0.375	0.499
6	0.390	0.542
7	0.394	0.583
8	0.402	0.609
9	0.399	0.632

表 4.6: TFIDF における学習性能 $P30$ (提示文書数 $S = 20$)

M	Rocchio	TFIDF cos
0	0.176	0.176
1	0.268	0.271
2	0.317	0.430
3	0.342	0.509
4	0.354	0.582

するため，ベクトルの大きさの影響が強く現れるためと考えられる．表 4.9 は，サポートベクターマシンにおいて $S = 10$ のとき，1 回目のフィードバックで提示されるベクトルの大きさ $||\mathbf{x}||$ を比較したものである．この表から，TF ベクトルは他のベクトル表現と比べて，非常に大きなベクトルが存在しており，この大きなベクトルの存在が，検索精度に悪影響を与えていると考えられる．コサインカーネルにより，非常に大きなベクトルの影響が解消され，1 回目のフィードバック性能の改善効果が大きくなっていると考えられる．

ベクトル空間モデルを用いた文書検索で最も一般的な文書ベクトルの一つは TFIDF である．TFIDF の算出のためには，TF（1 文書中における，ある語の頻度）と DF（文書頻度：全文書中で，ある語が含まれる文

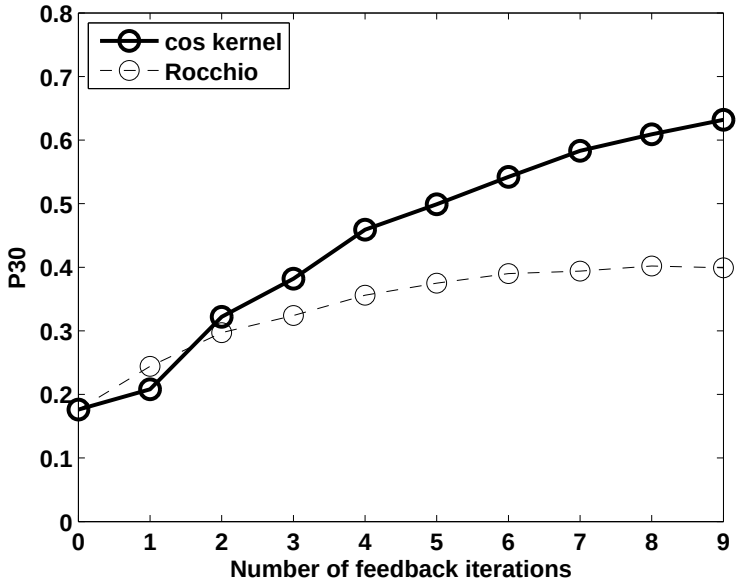


図 4.6: TFIDF における学習性能 $P30$ (提示文書数 $S = 10$)

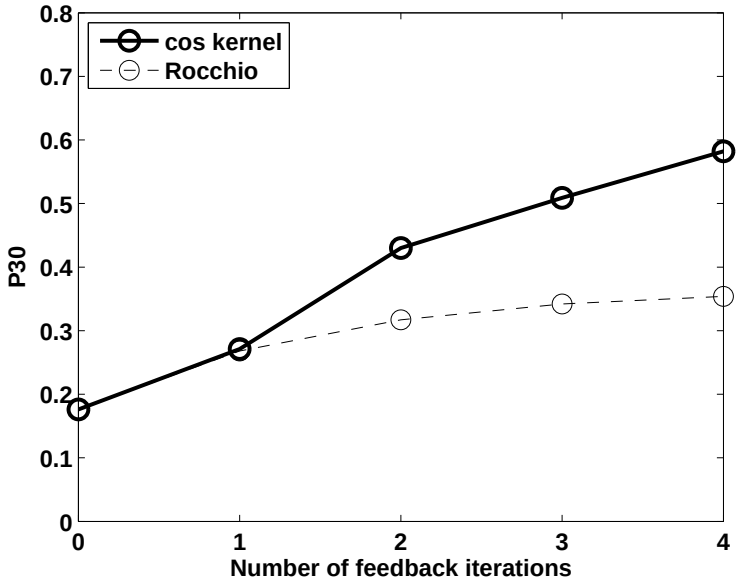


図 4.7: TFIDF における学習性能 $P30$ (提示文書数 $S = 20$)

表 4.7: TFIDF における検索性能 P (提示文書数 $S = 10$)

M	Rocchio	TFIDF cos
0	0.227	0.227
1	0.243	0.215
2	0.251	0.245
3	0.247	0.256
4	0.242	0.264
5	0.233	0.266
6	0.227	0.265
7	0.217	0.264
8	0.208	0.261
9	0.200	0.258

表 4.8: TFIDF における検索性能 P (提示文書数 $S = 20$)

M	Rocchio	TFIDF cos
0	0.194	0.194
1	0.202	0.195
2	0.201	0.232
3	0.190	0.236
4	0.180	0.236

書の数）の計算が必要となる．ここで，TF の算出は文書単位で可能なため 1 つの文書が与えられれば計算可能であるが，DF の計算は固定された全文書集合が既与である必要がある．しかし，現在情報検索研究における主流の 1 つである Web ページの検索やブログまたは Twitter に代表されるマイクロブログの検索など，全文書集合が膨大でかつ極めて動的に変化する文書検索においては，DF の計算に必要なこの前提は成り立たたないことが指摘されている [Gamon 08][Klein 09]．このように，本研究でパフォーマンス向上に特に効果のあった TF は，Web，マイクロブログ検索などの情報検索においても重要な文書表現であり，その性能向上を達成した本研究は有用性を持つと考えられる．

一方，Boolean と TFIDF ではフィードバックの初期段階で改善方法の

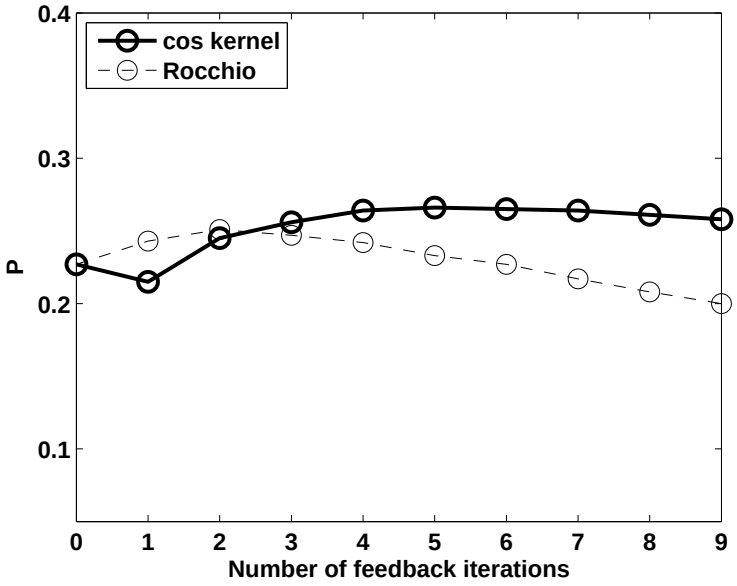


図 4.8: TFIDF における検索性能 P (提示文書数 $S = 10$)

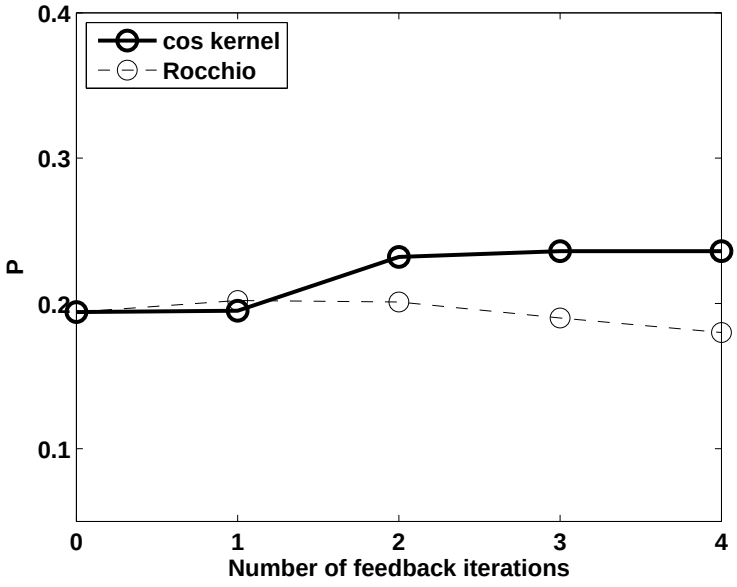


図 4.9: TFIDF における検索性能 P (提示文書数 $S = 20$)

表 4.9: $M = 1, S = 10$ のときにサポートベクターマシンが提示するベクトルの大きさ $||\mathbf{x}||$ の比較

	mean	median	min	max	SD
Boolean	23.0	16.1	2.6	118.7	20.1
TF	248.4	77.7	3.5	12258.6	724.2
TFIDF	29.1	16.2	3.2	228.4	32.9

SD:Standard Deviation

効果が見られるが，フィードバックを繰り返すごとに効果が見られなくなる．これは，フィードバックを繰り返すことでサポートベクターマシンによる判別超平面の分類精度が徐々に上がっていくことと，表 4.9 から $||\mathbf{x}||$ に大きな差がないことから，$\cos\theta$ のみでの適合度による提示順が $||\mathbf{x}||$ をかけても大きく変更されることがないためと考えられる．

このようにフィードバックの初期段階で効果があるということは，少ない文書判定で効果があることを意味する．一般に，文書検索の適合フィードバックにおいてユーザが文書の適合／非適合を判定するためには，実際に文書の内容を読んで理解する必要があり，それはユーザにとって多くの認知的負荷のかかるタスクとなる．よって，ユーザに利用してもらう実用性の観点から，ユーザにかかる負荷は小さい方がよい．そのためには，できるだけ少ない文書判定で高い精度の検索が実現されることが重要となる．この意味で提案手法には利点があると考えられる．

ここで，文書間の順位付けを行うランキング学習の一つであるランキング SVM[Joachims 02] におけるコサインカーネルの効果について考察する．ランキング SVM は **2.6** で述べたように，クエリに対する文書のランキングが与えられたときに，2 つの文書間での順位関数の大小を規定することで，ランキングを二値分類問題として解くものである．一方，コサインカーネルは Rocchio の手法との比較分析から導出された，SVM における適合フィードバックにおける提示文書の順位付けの改善手法である．ただし，前述のように正規化線形カーネルととらえることもでき，画像認識の分野での有効性が示されている [Hotta 10]．したがって，ランキング SVM においても，画像データに関して有効である可能性がある．

4.4.2 提案手法と Rocchio の性能比較

ここでは，提案手法であるコサインカーネルを用いたサポートベクターマシンベースのシステムと，文書ベクトルの正規化を行った Rocchio ベースのシステムとの性能比較結果について考察する．

表 4.5，表 4.6 と図 4.6，図 4.7 より，提案手法の学習性能は，初期段階では劣るもののフィードバックを繰り返すと Rocchio を上回っている．また，表 4.7，表 4.8 と図 4.8，図 4.9 より，検索性能についても同様の傾向が見られる．フィードバックの初期段階で，提案手法の性能が劣る原因としては，初期検索結果を用いた 1 回目のフィードバックではシステムが能動的に訓練データを選択しておらず，能動学習の効果が表れていないことが考えられる．

さらに，図 4.8 と図 4.9 より，サポートベクターマシンに比べて，Rocchio の方が検索性能 P の減衰率が大きい．Drucker らの研究 [Drucker 02] でも，Rocchio の手法は，フィードバックを繰り返すと適合率が大きく下がることが報告されており，今回の実験でも同様の現象が現れている．

第5章　結論

大量の文書集合からユーザが望む文書をできるだけ多く見つけることは，情報検索の重要な課題である．本論文では，従来の対話的情報検索である適合フィードバックを分類学習としてとらえ，それに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を応用した対話的文書検索の枠組みにおいて，性能向上に寄与する二つの方法を提案した．

一つは，ユーザが判定する文書の選択に有効なヒューリスティックスである．このヒューリスティックスは，ユーザへの文書提示を能動的に行うものであり，少数の正データと膨大な負データとからなるデータの分布において有効と考えられる．そして，この能動的文書提示を組み込んだ分類学習に基づく適合フィードバックの手続きを開発し，システムを実装した．さらに，提示提案手法の有効性を検証するために，従来の適合フィードバックシステム，そして能動学習なしのシステムを用いて，新聞記事検索のテストベッドを対象に比較実験を行った．その結果，提案システムが従来方法と同等以上のパフォーマンスであることを示した．

もう一つは，対話的文書検索の比較分析に基づくカーネルの提案である．サポートベクターマシンを用いた対話的文書検索における文書の適合度は，従来手法におけるベクトル空間モデル上でどのような特性を持つのかが明らかになっていない．そこで本論文では，サポートベクターマシンにおける距離を用いた適合度を定式化し，対話的文書検索における従来手法である Rocchio の手法との比較分析を行った．また，そこから得られた知見より，サポートベクターマシンに基づく手法に適したカーネルを提案し，本手法の有効性を検証するために検証実験を行った．Boolean，TF，TFIDF の文書ベクトル表現について比較した結果，すべてのベクトル表現で性能が向上し，特に TF ベクトル表現において，性能が大きく向上することを示した．

以上により，サポートベクターマシンを用いた対話的文書検索において，二つの提案手法が有効であり，性能向上に寄与することを示した．

謝辞

本論文は，様々な方々のお力添えのもとに完成いたしました．

はじめに，本論文審査委員会の主査であり，主任指導教官でもありました山田誠二教授におかれましては，本研究全般に関して多大な御指導と御鞭撻を賜りました．私の勤務先においでいただき，度々研究打合せをしていただくなど様々な配慮をしていただきました．また，本研究をまとめるにあたり幾度となく激励していただくとともに我慢強くご支援くださったことなど，先生の御指導なくしては本研究は成し得ないものでありました．心より厚く御礼申し上げます．

職場の上司であり，論文審査委員でもありました小野田崇連携教授にも，本研究全般に関して多大な御指導と御鞭撻を賜りました．また，職場の業務面でも様々な御配慮をいただきました．心より御礼申し上げます．

本論文をまとめるにあたり，論文審査委員の先生方からも多大な御指導と御鞭撻を賜りました．相澤彰子教授には，情報検索に関する深い見識に基づいた御指摘を賜りました．市瀬龍太郎准教授には，機械学習に関する深い見識に基づいた御指摘を賜りました．佐藤健教授には，非常に幅広く深い見識に基づいて本論文の主張すべき点について重要な御助言を賜りました．皆様の御指導，御鞭撻により本論文をまとめ，より質の高いものとすることができました．重ねて御礼申し上げます．

山田誠二研究室のみなさまにも大変お世話になりました．研究室ミーティングで有益なコメントを頂きましたことに感謝いたします．

総合研究大学院大学での研究にご理解をいただきました (財) 電力中央研究所システム技術研究所の方々に感謝をいたします．特に，谷口治人前所長（現首席研究員），栗原郁夫所長，松井正一前領域リーダー，堤富士夫上席研究員，篠原靖志上席研究員の皆様には，上司として御配慮いただいたことに感謝いたします．

最後に，本研究の遂行と本論文の作成に対して，いつも心の支えとなり元気づけてくれた両親と妻に感謝します．

参考文献

[Blum 04] Blum, A., Lafferty, J., Rwebangira, M. R., and Reddy, R.: Semi-supervised learning using randomized mincuts, in *Proceedings of the 21st international conference on Machine learning*, ICML '04, pp. 13– (2004)

[Buckley 95a] Buckley, C. and Salton, G.: Optimization of relevance feedback weights, in *SIGIR '95: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 351–357 (1995)

[Buckley 95b] Buckley, C., Salton, G., Allan, J., and Singhal, A.: Automatic query expansion using SMART: TREC 3, in *Proceedings of the 3th Text Retrieval Conference (TREC-3)*, pp. 69–80 (1995)

[Campbell 00] Campbell, C., Cristianini, N., and Smola, A. J.: Query Learning with Large Margin Classifiers, in *ICML '00: Proceedings of the 17th International Conference on Machine Learning*, pp. 111–118 (2000)

[Cao 06] Cao, Y., Xu, J., Liu, T.-Y., Li, H., Huang, Y., and Hon, H.-W.: Adapting ranking SVM to document retrieval, in *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pp. 186–193 (2006)

[Cao 08] Cao, G., Nie, J.-Y., Gao, J., and Robertson, S.: Selecting good expansion terms for pseudo-relevance feedback, in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pp. 243–250 (2008)

[Cohn 96] Cohn, D. A., Ghahramani, Z., and Jordan, M. I.: Active Learning with Statistical Models, *Journal of Artificial Intelligence Research*, Vol. 4, pp. 129–145 (1996)

[Cortes 95] Cortes, C. and Vapnik, V.: Support vector networks, *Machine Learning*, Vol. 20, pp. 273–297 (1995)

[Crammer 01] Crammer, K. and Singer, Y.: Pranking with Ranking, in *Proceedings of the 2001 Neural Information Processing Systems*, pp. 641–647 (2001)

[Deselaers 08] Deselaers, T., Paredes, R., Vidal, E., and Ney, H.: Learning weighted distances for relevance feedback in image retrieval, in *Proceedings of the 19th International Conference on Pattern Recognition*, pp. 1–4 (2008)

[Drucker 01] Drucker, H., Shahrary, B., and Gibbon, D. C.: Relevance feedback using support vector machines, in *Proceedings of 18th International Conference on Machine Learning*, pp. 122–129 (2001)

[Drucker 02] Drucker, H., Shahrary, B., and Gibbon, D. C.: Support vector machines: relevance feedback and information retrieval, *Information Processing & Management*, Vol. 38, pp. 305–323 (2002)

[Gamon 08] Gamon, M., Basu, S., Belenko, D., Fisher, D., Hurst, M., and König, A. C.: BLEWS: Using Blogs to Provide Context for News Articles, in *In Proceedings of International Conference on Weblogs and Social Media* (2008)

[Harman 92] Harman, D.: Relevance feedback revisited, in *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '92, pp. 1–10 (1992)

[Hotta 10] Hotta, K.: Local normalized linear summation kernel for fast and robust recognition, *Pattern Recognition*, Vol. 43, pp. 906–913 (2010)

[Ingwersen 92] Ingwersen, P.: *Information Retrieval Interaction*, Taylor Graham (1992)

[Joachims 97] Joachims, T.: A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization, in *Proceedings of the Fourteenth International Conference on Machine Learning*, ICML '97, pp. 143–151 (1997)

[Joachims 99] Joachims, T.: Transductive inference for text classification using support vector machines, in *Proceedings of International Conference on Machine Learning*, pp. 200–209 (1999)

[Joachims 02] Joachims, T.: Optimizing search engines using clickthrough data, in *Proceedings of the 8th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pp. 133–142 (2002)

[Joachims 03] Joachims, T.: Transductive learning via spectral graph partitioning, in *Proceedings of the 20th international conference on Machine learning*, pp. 290–297 (2003)

[Jordan 04] Jordan, C. and Watters, C.: C.: Extending the Rocchio Relevance Feedback Algorithm to Provide Contextual Retrieval, in *Proceedings of AWIC04*, pp. 135–144 (2004)

[Kelly 01] Kelly, D. and Belkin, N. J.: Reading ti scrolling and interaction: exploring implicit sources of user preferences for relevance feedback, in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '01, pp. 408–409 (2001)

[Kelly 03] Kelly, D. and Teevan, J.: Implicit feedback for inferring user preference: a bibliography, *SIGIR Forum*, Vol. 37, pp. 18–28 (2003)

[Klein 09] Klein, M. and Nelson, M. L.: Correlation of Term Count and Document Frequency for Google N-Grams, in *Proceedings of the 31th European Conference on IR Research on Advances in Information Retrieval*, ECIR '09, pp. 620–627 (2009)

[Koenemann 96] Koenemann, J. and Belkin, N. J.: A case for interaction: a study of interactive information retrieval behavior and effectiveness,

in *Proceedings of 27th Annual SIGCHI Conference on Human factors in Computing Systems*, pp. 205–212 (1996)

[Lewis 94] Lewis, D. D. and Gale, W. A.: A sequential algorithm for training text classifiers, in *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 3–12 (1994)

[Lv 09] Lv, Y. and Zhai, C.: Adaptive Relevance Feedback in Information Retrieval, in *Proceedings of the 18th ACM Conference on International Knowledge Management*, pp. 255–264 (2009)

[Montgomery 04] Montgomery, J., Si, L., Callan, J., and Evans, D. A.: Effect of varying number of documents in blind feedback: analysis of the 2003 NRRC RIA workshop "bf_numdocs" experiment suite, in *Proceedings of 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 476–477 (2004)

[Moschitti 03] Moschitti, A.: A Study on Optimal Parameter Tuning for Rocchio Text Classifier, In Proceedings of the 25th European Conference on Information Retrieval Research (ECIR '03), pp. 420–435 (2003)

[Muller 00] Muller, H., Muller, W., Marchand-Maillet, S., Pun, T., and Squire, D.: Strategies for positive and negative relevance feedback in image retrieval, in *Proceedings of 15th International Conference on Pattern Recognition*, Vol. 1, pp. 1043–1046 (2000)

[Murata 09] Murata, H., Onoda, T., and Yamada, S.: SVM-based Relevance Feedback Document Retrieval in Different Representations of Document Vectors, in *Proceedings of Artificial Intelligence and Applications 2009*, pp. 100–105 (2009)

[Murata 10] Murata, H., Onoda, T., and Yamada, S.: A Kernel for Interactive Document Retrieval Based on Support Vector Machines, in *Proceedings of International Symposium on Advanced Intelligent Systems 2010*, pp. 1316–1321 (2010)

[村田 11] 村田 博士, 小野田 崇, 山田 誠二：SVM を用いた対話的文書検索における適合性評価の比較分析, 日本知能情報ファジィ学会誌, Vol. 23, No. 6, pp. 853–862 (2011)

[Nallapati 04] Nallapati, R.: Discriminative models for information retrieval, in *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pp. 64–71 (2004)

[岡部 01] 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. 6, p. 880 (2001)

[小野田 02] 小野田 崇：Large margin classifiers, 人工知能学会誌, Vol. 17, No. 1, pp. 21–30 (2002)

[Onoda 06] Onoda, T., Murata, H., and Yamada, S.: Support Vector Machines Based Active Learning for the Relevance Feedback Document Retrieval, in *Proceedings of the 2006 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology*, WI-IATW '06, pp. 389–392 (2006)

[小野田 07a] 小野田 崇：サポートベクターマシン, オーム社 (2007)

[Onoda 07b] Onoda, T., Murata, H., and Yamada, S.: Comparison of Learning Performance and Retrieval Performance for Support Vector Machines Based Relevance Feedback Document Retrieval, in *Proceedings of the 2007 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology*, pp. 249–252 (2007)

[Onoda 08a] Onoda, T., Murata, H., and Yamada, S.: Comparison of Performance for SVM Based Relevance Feedback Document Retrieval in Several Vector Space Models, in *Proceedings of the 2007 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology*, pp. 169–172 (2008)

[Onoda 08b] Onoda, T., Murata, H., and Yamada, S.: SVM-based interactive document retrieval with active learning, *New Generation Computing*, Vol. 26, pp. 49–61 (2008)

[Robertson 96] Robertson, S., Walker, S., Beaulieu, M., Gatford, M., and Payne, A.: Okapi at TREC-4, in *Proceedings of the 4th Text Retrieval Conference (TREC-4)*, pp. 73–96 (1996)

[Rocchio 71] Rocchio, J.: *Relevance feedback in information retrieval*, pp. 313–323, Prentice Hall, Englewood, Cliffs, New Jersey (1971)

[Rui 98] Rui, Y., Huang, T., Ortega, M., and Mehrotra, S.: Relevance feedback: a power tool for interactive content-based image retrieval, Vol. 8, No. 5, pp. 644–655 (1998)

[酒井 06] 酒井 哲也：よりよい検索システム実現のために：正解の良し悪しを考慮した情報検索評価の動向, 情報処理学会誌, Vol. 47, No. 2, pp. 147–158 (2006)

[Salton 71] Salton, G. (ed.): *The SMART Retrieval System - Experiments in Automatic Document Processing*, Prentice Hall, Englewood, Cliffs, New Jersey (1971)

[Salton 83] Salton, G. and McGill, M. J.: *Introduction to Modern Information Retrieval*, McGraw-Hill (1983)

[Schapire 98] Schapire, R., Singer, Y., and Singhal, A.: Boosting and rocchio applied to text filtering, in *Proceedings of 21st Annual International ACM SIGIR Conference on Research and development in information retrieval*, pp. 215–223 (1998)

[Sculley 09] Sculley, D.: Large Scale Learning to Rank, in *Proceedings of NIPS 2009 Workshop on Advances in Ranking*, pp. 1–6 (2009)

[Seung 92] Seung, H. S., Opper, M., and Sompolinsky, H.: Query by committee, in *COLT '92: Proceedings of the 5th annual workshop on Computational learning theory*, pp. 287–294 (1992)

[Singhal 97] Singhal, A., Mitra, M., and Buckley, C.: Learning routing queries in a query zone, *SIGIR Forum*, Vol. 31, pp. 25–32 (1997)

[Su 11] Su, J.-H., Huang, W.-J., Yu, P. S., and Tseng, V. S.: Efficient Relevance Feedback for Content-Based Image Retrieval by Mining User

Navigation Patterns, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 23, pp. 360–372 (2011)

[Tao 06] Tao, D., Tang, X., Li, X., and Wu, X.: Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval, *IEEE Transactions Pattern Analysis and Machine Intelligence*, Vol. 28, pp. 1088–1099 (2006)

[Tong 02] Tong, S. and Koller, D.: Support vector machine active learning with applications to text classification, *Journal of Machine Learning Research*, Vol. 2, pp. 45–66 (2002)

[柘植 03] 柘植 覚, 獅々堀 正幹, 黒岩 眞吾, 北 研二：サポートベクターマシンによる適合性フィードバックを用いた情報検索, 情報処理学会論文誌, Vol. 44, No. 1, pp. 59–67 (2003)

[Uğuz 10] Uğuz, H. and Arslan, A.: A new approach based on discrete hidden Markov model using Rocchio algorithm for the diagnosis of the brain diseases, *Digital Signal Processing*, Vol. 20, pp. 923–934 (2010)

[Vapnik 98] Vapnik, V.: *Statistical Learning Theory*, John Wiley and Sons Inc. (1998)

[Warmuth 03] Warmuth, M. K., Liao, J., Rätsch, G., Mathieson, M., Putta, S., and Lemmen, C.: Active Learning with Support Vector Machines in the Drug Discovery Process, *Chemical Information and Computer Sciences*, Vol. 43, No. 2, pp. 667–673 (2003)

[Yan 03] Yan, R., Hauptmann, A., and Jin, R.: Multimedia search with pseudo-relevance feedback, in *Proceedings of the 2nd international conference on Image and video retrieval*, CIVR'03, pp. 238–247 (2003)

[Yates 99] Yates, R. B. and Neto, B. R.: *Modern Information Retrieval*, Addison Wesley (1999)

[Yu 03] Yu, S., Cai, D., Wen, J.-R., and Ma, W.-Y.: Improving pseudo-relevance feedback in web information retrieval using web page segmentation, in *Proceedings of the 12th international conference on World Wide Web*, WWW '03, pp. 11–18 (2003)

[Zhou 03] Zhou, X. S. and Huang, T. S.: Relevance feedback in image retrieval: A comprehensive review, Vol. 8, No. 6, pp. 526–544 (2003)

[Zhu 03] Zhu, X., Ghahramani, Z., and Lafferty, J.: Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions, in *Proceedings of the 20th international conference on Machine learning*, pp. 912–919 (2003)

付 録A　実験に使用したクエリ

表 A.1 から表 A.6 に，3.3 と 4.3 での実験に使用したクエリを示す．これらは，TREC の第 6 回から第 8 回の ad hoc タスクで使用された 150 の検索課題（トピック）のうち，title タグと呼ばれる，検索課題の内容を数語で表現したものである．

表 A.1: 実験に使用したクエリの一覧

課題番号	クエリ
301	International Organized Crime
302	Poliomyelitis and Post-Polio
303	Hubble Telescope Achievements
304	Endangered Species (Mammals)
305	Most Dangerous Vehicles
306	African Civilian Deaths
307	New Hydroelectric Projects
308	Implant Dentistry
309	Rap and Crime
310	Radio Waves and Brain Cancer
311	Industrial Espionage
312	Hydroponics
313	Magnetic Levitation-Maglev
314	Marine Vegetation
315	Unexplained Highway Accidents

表 A.2: 実験に使用したクエリの一覧

課題番号	クエリ
316	Polygamy Polyandry Polygyny
317	Unsolicited Faxes
318	Best Retirement Country
319	New Fuel Sources
320	Undersea Fiber Optic Cable
321	Women in Parliaments
322	International Art Crime
323	Literary/Journalistic Plagiarism
324	Argentine/British Relations
325	Cult Lifestyles
326	Ferry Sinkings
327	Modern Slavery
328	Pope Beatifications
329	Mexican Air Pollution
330	Iran-Iraq Cooperation
331	World Bank Criticism
332	Income Tax Evasion
333	Antibiotics Bacteria Disease
334	Export Controls Cryptography
335	Adoptive Biological Parents
336	Black Bear Attacks
337	Viral Hepatitis
338	Risk of Aspirin
339	Alzheimer's Drug Treatment
340	Land Mine Ban

表 A.3: 実験に使用したクエリの一覧

課題番号	クエリ
341	Airport Security
342	Diplomatic Expulsion
343	Police Deaths
344	Abuses of E-Mail
345	Overseas Tobacco Sales
346	Educational Standards
347	Wildlife Extinction
348	Agoraphobia
349	Metabolism
350	Health and Computer Terminals
351	Falkland petroleum exploration
352	British Chunnel impact
353	Antarctica exploration
354	journalist risks
355	ocean remote sensing
356	postmenopausal estrogen Britain
357	territorial waters dispute
358	blood-alcohol fatalities
359	mutual fund predictors
360	drug legalization benefits
361	clothing sweatshops
362	human smuggling
363	transportation tunnel disasters
364	rabies
365	El Nino

表 A.4: 実験に使用したクエリの一覧

課題番号	クエリ
366	commercial cyanide uses
367	piracy
368	in vitro fertilization
369	anorexia nervosa bulimia
370	food/drug laws
371	health insurance holistic
372	Native American casino
373	encryption equipment export
374	Nobel prize winners
375	hydrogen energy
376	World Court
377	cigar smoking
378	euro opposition
379	mainstreaming
380	obesity medical treatment
381	alternative medicine
382	hydrogen fuel automobiles
383	mental illness drugs
384	space station moon
385	hybrid fuel cars
386	teaching disabled children
387	radioactive waste
388	organic soil enhancement
389	illegal technology transfer
390	orphan drugs

表 A.5: 実験に使用したクエリの一覧

課題番号	クエリ
391	R&D drug prices
392	robotics
393	mercy killing
394	home schooling
395	tourism
396	sick building syndrome
397	automobile recalls
398	dismantling Europe's arsenal
399	oceanographic vessels
400	Amazon rain forest
401	foreign minorities, Germany
402	behavioral genetics
403	osteoporosis
404	Ireland, peace talks
405	cosmic events
406	Parkinson's disease
407	poaching, wildlife preserves
408	tropical storms
409	legal, Pan Am, 103
410	Schengen agreement
411	salvaging, shipwreck, treasure
412	airport security
413	steel production
414	Cuba, sugar, exports
415	drugs, Golden Triangle
416	Three Gorges Project
417	creativity
418	quilts, income
419	recycle, automobile tires
420	carbon monoxide poisoning

表 A.6: 実験に使用したクエリの一覧

課題番号	クエリ
421	industrial waste disposal
422	art, stolen, forged
423	Milosevic, Mirjana Markovic
424	suicides
425	counterfeiting money
426	law enforcement, dogs
427	UV damage, eyes
428	declining birth rates
429	Legionnaires' disease
430	killer bee attacks
431	robotic technology
432	profiling, motorists, police
433	Greek, philosophy, stoicism
434	Estonia, economy
435	curbing population growth
436	railway accidents
437	deregulation, gas, electric
438	tourism, increase
439	inventions, scientific discoveries
440	child labor
441	Lyme disease
442	heroic acts
443	U.S., investment, Africa
444	supercritical fluids
445	women clergy
446	tourists, violence
447	Stirling engine
448	ship losses
449	antibiotics ineffectiveness
450	King Hussein, peace

研究業績

学術論文

- 村田 博士，小野田 崇，山田 誠二: SVM を用いた対話的文書検索における適合性評価の比較分析, 日本知能情報ファジィ学会誌，Vol. 23，No. 6, pp. 853–862 (2011)

国際会議 (全文査読あり)

- Murata, H., Onoda, T., Yamada, S.: SVM-based Relevance Feedback Document Retrieval in Different Representations of Document Vectors, in *Proceedings of Artificial Intelligence and Applications 2009*, pp. 100–105 (2009)
- Murata, H., Onoda, T., Yamada, S.: A Kernel for Interactive Document Retrieval Based on Support Vector Machines, in *Proceedings of International Symposium on Advanced Intelligent Systems 2010*, pp. 1316–1321 (2010)
- Murata, H., Onoda, T., Yamada, S.: Non-relevance Feedback Document Retrieval using Large Data Set, in *Proceedings of 8th International Symposium on Advanced Intelligent Systems*, pp. 141–146 (2007)

国内会議

- 村田 博士，小野田 崇，山田 誠二: 文書検索における非適合性フィードバック手法の一検討, 人工知能学会全国大会, 1E1-02 (2008)

- 村田 博士，小野田 崇，山田 誠二: SVM に基づく対話的文書検索におけるカーネルの提案, 人工知能学会全国大会, 3D1-04 (2010)