社団法人 情報処理学会　研究報告
IPSJ SIG Technical Report

2003－FI－13　(3)
2003／11／13

相関係数を用いた実証的重みの分析と検索質問拡張

金 谷 敦 志[†]　梅 村 恭 司[†]

情報検索において代表的な情報検索モデルである関連性フィードバックでは，新しい用語を検索質問に付け加える検索質問拡張と用語の再重み付けにより検索質問の修正を行う．一般的にこの手法を用いる場合，元の検索質問のみを用いた場合と比較して再現率を向上させる反面，適合率が下がることが報告されている．原因としては，関連のある用語の選定方法と，その用語に対する重み付けの方法が明確でないことが考えられる．本稿では，検索質問と共起する用語に着目した関連のある用語の選定方法を提案し，選定された用語に対する実証的な重み付けの方法についてを述べる．

Empirical Term Weighting by Correlation Coefficient for Query Expansion

Atsushi Kanaya[†] and Kyoji Umemura[†]

At the Information Retrieval (IR), with the relevance feedback which is the typical IR model, queries are modified using Query Expansion (QE) and Term Re-weighting (TR), QE adds new collected term to query, and the TR re-weights to new (or original) query. In the case of using this method, as compared without using relevance feedback, it is reported that recall is improved but precision is decreased. We have considered that some methods for selecting relevant term and weighting these terms aren't clear. This paper proposes a method for selecting relevant term using coincided term with original query term, and shows empirical weighting methods for these terms.

1. はじめに

WWW を代表とするネットワーク上の膨大な電子データやデータベースを利用する上で，WWW における検索ではユーザの入力する検索語の数は一般に非常に少なく，少ない情報から関連のある文書集合を提供することは困難である．一般にユーザの入力する検索語は十分ではなく，漏れのない検索は難しい．これに対処するために関連のある用語を付与する検索質問拡張がある．これにより検索質問に関連のある用語を付与し再検索することにより，関連のある文書の漏れを減らすことができる．

関連のある用語の付与の方法にはいくつか種類があり，主には以下の 2 種類がある．

(1)　シソーラスやオントロジーなどの辞書や知識を利用

(2)　初期検索で得られた文書中の用語を利用

前者は関連のある語を辞書や知識を頼りに用語を付与する方法であるが，辞書は人手による維持コストが大きいことが問題である．後者は関連性フィードバックと呼ばれる手法であり，初期の検索結果より，上位にランク付けされた文書に対して適合性の判断を行い，適合文書に含まれる用語の重みを大きくし，不適合文書に含まれる用語の重みを小さくする．ここで適切な重みを関連ある語に用いることは，検索精度の向上には重要である．

用語を評価する実証的手法を用いた研究は，[7] にて報告されている．検索質問の用語について，関連の判断により用語の重みを評価する教師付きの実証的手法であり，訓練集合から機械学習により得られた重みを用いることにより，初見の検索質問に対しても適切な重みをかけることができ，検索の性能向上に寄与したことが報告されている．

本稿では[7] を基礎として，関連のある用語に対する重みを評価する実証的手法を検索質問に付与する用語に対して計算し，拡張された質問に対する重みを提唱する．

関連の研究としては，Roccio の式[14] を用いる方法[12] のように適合文書，不適合文書中の用語に重みをかけて再検索する方法や，高次元の空間にある文書ベクトルを低次元の空間へと射影し，関連のある語を同一に扱えるようにする潜在的意味インデキシング[8)10)] を用いる方法のように，ベクトル空間モデル

† 豊橋技術科学大学 情報工学系
Department of Information Engineering, Toyohashi University of Technology

表 1 記号の定義
Table 1 Definition of symbols

t	用語
d	文書
$tf(t,d)$	文書 d 中に現れる用語 t の数
$df(t)$	$tf(t,d) \geq 1$ である文書 d の数
N	文書数
$idf(t)$	文書頻度の逆数 $= -\log_2 \frac{df(t)}{N}$
$df(t\|rel, tf_0)$	$tf(t,d) = tf_0$ である適合文書数
$df(t\|\overline{rel}, tf_0)$	$tf(t,d) = tf_0$ である不適合文書数
$I(t_1;t_2)$	用語 t_1, t_2 の相互情報量

を用いた方法が多く報告されている．

関連性フィードバックで良く用いられる Roccio の式は元の検索質問 $\boldsymbol{q}$ に対して，適合文書集合 D_R および不適合文書集合 $D_{\overline{R}}$ による重みの調整分を，それぞれの文書集合に含まれる文書数で正規化し，各パラメータ α, β, γ による調整を加えて新たな検索質問とする式である．

$$\boldsymbol{q}' = \alpha \boldsymbol{q} + \frac{\beta}{|D_R|} \sum_{\boldsymbol{d}_i \in D_R} \boldsymbol{d}_i - \frac{\gamma}{|D_{\overline{R}}|} \sum_{\boldsymbol{d}_j \in D_{\overline{R}}} \boldsymbol{d}_j \quad (1)$$

式 (1) に代表される従来の関連性フィードバックには次のような問題点がある．

(1) 関連のある用語の選定方法

(2) その用語に対する重み付けの方法

前者は適合文書，不適合文書中に含まれるすべての用語を追加するため，用語を厳選して選定することができない．後者は式 (1) における β,γ の適切な与え方が決まっておらず，試行錯誤することになりうる．

なお，初期検索結果からユーザが必要な文書を選択し，それらをフィードバック文書として関連性フィードバックのパラメータ調整などを行う手動型も研究されている[13]が，今回は手動型を考慮せず，自動でフィードバックを行う方法に絞っている．

記号の定義を表 1 に示す．

2. 実証的重みの分析

関連の判断により用語の重みを評価する実証的手法[6][7]の説明を簡単に行う．各文書のスコアは，以下のように用語 $t \in T$, 質問 q に対する重み関数 λ の総和で示す．

$$score(d,q) = \sum_{t \in T} \lambda(t,d,q) \quad (2)$$

課題は訓練データを用い，機械学習にて統計的に最良の λ である $\hat{\lambda}$ を計算することである．訓練データとしては文書集合にいくつかの検索質問とそれに対する正解判定を含めたテストコレクションを用いる．

より処理しやすくするために，全ての用語の空間を低次元の特徴空間に対応付けすることが共通の課題となる．$tf \cdot idf$ の例では同じ idf の全ての用語をグループ分けし，同じグループに対する重み λ を割り付ける ($tf \cdot idf$ やその変形など)．グループ分けにはビンを用いる[6]．用語は idf のような特徴を元にしたビンに割り当てられ，$\hat{\lambda}$ はそれぞれのビンに対して計算されるような，ヒストグラムベースの手法である．

計算方法はテストコレクションの質問 q からそれぞれの用語 t に対する適合文書と不適合文書の数を計算する．すなわち，それぞれの t, q に対し，$df(t|rel, tf_0)$ と $df(t|\overline{rel}, tf_0)$ を計算する．これらの訓練の観察結果より，ビンから初見の質問へ対応できるような $\hat{\lambda}$ を獲得する．λ を対数尤度比として推測すると式 (3) となる．

$$\hat{\lambda}(bin, tf) = \log_2 \frac{P(bin, tf|rel)}{P(bin, tf|\overline{rel})} \quad (3)$$

分母，分子は近似され，以下のようになる．

$$P(bin, tf|rel) \approx \frac{df(bin, rel, tf)}{\hat{N}_{rel}} \quad (4)$$

$$P(bin, tf|\overline{rel}) \approx \frac{df(bin, \overline{rel}, tf)}{\hat{N}_{\overline{rel}}} \quad (5)$$

$$df(bin, rel, tf) \equiv \frac{1}{|bin|} \sum_{t \in bin} df(t, rel, tf) \quad (6)$$

$$df(bin, \overline{rel}, tf) \equiv \frac{1}{|bin|} \sum_{t \in bin} df(t, \overline{rel}, tf) \quad (7)$$

$\hat{N}_{rel}$ は適合文書の総数の見積もりであり，これは平均で計算される．

$$\hat{N}_{rel} \equiv \frac{1}{|bin|} \sum_{t \in bin} N_{rel} \quad (8)$$

$\hat{N}_{rel} + \hat{N}_{\overline{rel}} = N$ を保証するために，$\hat{N}_{rel} \equiv N - \hat{N}_{rel}$ と定義する．用語のビンへの割り当ては $\lceil \log_2(df) \rceil$ ($df < 100$ の場合は $bin = 0$) としている．

以上の計算を行い，各 tf における $\hat{\lambda}$ は図 1 のようになる．結果としての重みは 0 と idf の間に存在し，その範囲を超える場合は 0 または idf に抑える．$tf \cdot idf$ はこの範囲を超えてしまうが，理由としては単純に tf の数だけ idf を倍加しているためである．図 1 より，各 tf 値による実証的重みは線形の関係が確認され，線形回帰した結果の重みを fit-G と呼ぶ．線形回帰したことにより idf の一次式は以下のようになる．$a(tf)$, $b(tf)$ は各 tf 値における初期値と傾きを示す係数となる．用語に対する実証的な重みは idf の一次式となり，初見の検索質問に対しても直接的に重みをかけることができる．

$$\hat{\lambda} \approx a(tf) + b(tf) \cdot idf \quad (9)$$

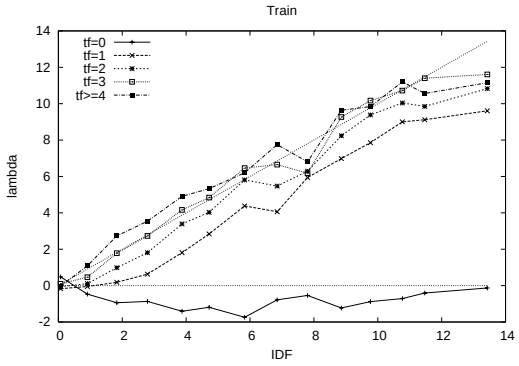

図 1 各 tf 値に対する $\hat{\lambda}$ の重み (fit-G)
Fig. 1 Term weight : $\hat{\lambda}$ for each tf value (fit-G)

3. 拡張質問に対する実証的重み

この節は前節を基礎として，検索質問に追加すべき単語の選定基準と拡張質問に対する適切な重み付けについてを述べる．本稿では，検索語を含む適合文書中によく出現する用語も検索語として有用であると仮定し，検索質問の用語と共起する用語に着目することとした．共起の度合いを示す尺度には idf と関連があると考えられる相互情報量を用いることを提案する．なお，この節は[9] を発展させたものである．

3.1 重みの学習データ

実証的重みの計算を行うにあたり，NTCIR1[4] 日本語論文アブストラクト 2 万件 (13MB) と検索質問，そして正解判定ファイルを対象データとした．日本語の用語切り分けには今回はバイグラム (2 文字) を用いた．

3.2 相互情報量

用語 x と y の出現確率がそれぞれ $P(x), P(y)$ とし，x, y が同時に出現する確率を $P(x, y)$ としたとき，2 語の持つ相互情報量 $I(x; y)$ は以下のように定義される[1]．

$$I(x; y) \equiv \log_2 \frac{P(x, y)}{P(x)P(y)} \tag{10}$$

ここで，x の出現確率は x を含む文書数と全文書数 N の比とし，$P(x) \approx df(x)/N$ とする．相互情報量は x, y が同時に出現する確率 $P(x, y)$ と x, y が独立して出現する確率 $P(x)P(y)$ の比であり，x, y が完全に独立して出現する場合は $I(x; y) = 0$，相互情報量が最大になる場合は $x = y$ のときであり，式 (11) のように idf となる．

$$I(x; x) = \log_2 \frac{1}{P(x)} = idf(x) \tag{11}$$

以上のように相互情報量は idf を基礎とした，[7] の $\hat{\lambda}$ の発展として 2 語間の重みにも線形の関係が予測される．

3.3 実証的重みの計算手順

実証的重みの計算手順を示す．はじめに，検索質問 q に対して初期検索を行い，検索質問と検索された文書集合 $D(d \in D)$ を得る．次に，検索質問中の用語 $q_i \in q$ と検索された文書中の用語 $t_d \in d$ に対して相互情報量 $I(q_i; t_d)$ を計算する．相互情報量の小さい組み合わせは膨大に存在するため，$I > 1$ を満たす組み合わせのみに対し，$\hat{\lambda}$ を計算する．

相互情報量は q_i, t_d に対する確率 $P(q_i)$，$P(t_d)$，$P(q_i, t_d)$ を計算することで求められる．用語 x の出現する確率は $P(x) \approx df(x)/N$, 2 語 x, y が同時に出現する確率は $P(x, y) = df(x, y)/N$ となるため，文書頻度 df を用いた相互情報量の計算は以下のようになる．

$$I(q_i; t_d) \approx log_2 \frac{N df(q_i, t_d)}{df(q_i)df(t_d)} \tag{12}$$

$I > 1$ を満たす t_d に対して，$\hat{\lambda}$ の計算を行う．$\hat{\lambda}$ は対数尤度比として解釈される．

$$\hat{\lambda}(t_d) = \log_2 \frac{P(t_d|rel)}{P(t_d|\overline{rel})} \tag{13}$$

分子はそれぞれ df の式に置き換えられ，以下のようになる：

$$P(t_d|rel) \approx \frac{df(t_d|rel)}{\hat{N}_{rel}} \tag{14}$$

$$P(t_d|\overline{rel}) \approx \frac{df(t_d|\overline{rel})}{\hat{N}_{\overline{rel}}} \tag{15}$$

3.4 実証的重みの分布

相互情報量 $I(q_i, t_d)$ を横軸，$\hat{\lambda}(t_d)$ を縦軸としてプロットした結果の一部を図 2 に示す．$\hat{\lambda}$ のプロット結果は，一部範囲を超える結果もあるが，ほとんどが 0 と $I(q_i, t_d)$ の間に収まり，かつプロットが $\hat{\lambda} = 0$ 付近と $\hat{\lambda} = I(q_i, t_d)$ 付近に集中していることが確認された．一部筋のようにプロットされている点があるが，これは t_d が同じ用語で共通している場合に出現することが多いようである．

図 2 より，$I(q_i, t_d)$ と $\hat{\lambda}(t_d)$ の間には正の相関があることが確認された．この結果より，検索質問の用語と共起する確率の高い用語に対して，単語の重みを直接的にかけられる見込みが立った．しかしこのままでは分布に幅があるため，各 tf 値についての線形回帰を行う．

3.5 ビン化による重みの線形回帰

各 tf 値についての線形回帰を行うときに，ビンを用いる．ビン化により，相互情報量のある範囲におけるデータをグループ化し，各ビンに対して，用語の出現数を考慮した $\hat{\lambda}$ の計算を行う．同じビンには相互情報量 $I(q_i; t_d)$ の小数点切り捨てを行った結果が集まる．

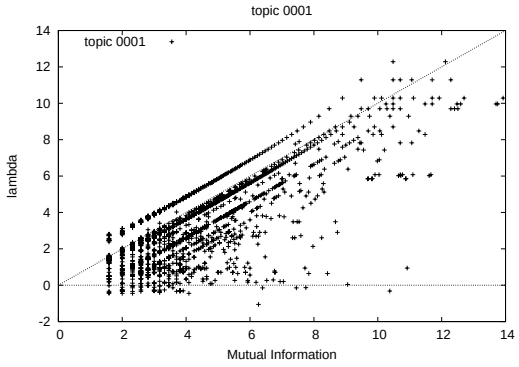

図 2 検索質問 1 問に対する相互情報量-λ のプロット結果
Fig. 2 of Mutual Information vs λ for one query

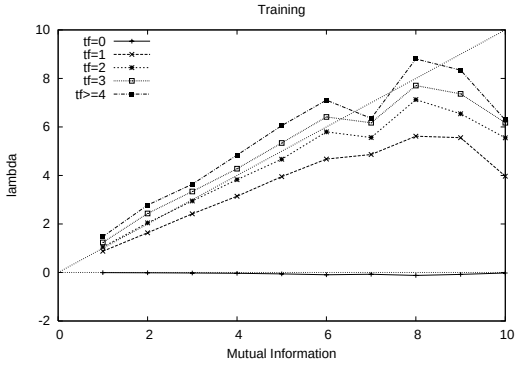

図 3 各 tf 値に対する $\hat{\lambda}$ の重み (fit-MI)
Fig. 3 Term weight : $\hat{\lambda}$ for each tf value (fit-MI)

各ビンには，2 節と同様の計算を行うように拡張する．すなわち，各ビンに対して tf 値が $0, 1, 2, 3, \geq 4$ における文書頻度 $df(t_d|rel, tf)$, $df(t_d|\overline{rel}, tf)$ とビン内の適合文書数，不適合文書数の情報が使用される．

各ビンに対する $\hat{\lambda}$ は以下のように変更される．

$$\hat{\lambda}(bin, tf) = \log_2 \frac{P(bin, tf|rel)}{P(bin, tf|\overline{rel})} \tag{16}$$

$$P(bin, tf|rel) \approx \frac{df(bin|rel, tf)}{N_{rel}} \tag{17}$$

$$P(bin, tf|\overline{rel}) \approx \frac{df(bin|rel, tf)}{N_{\overline{rel}}} \tag{18}$$

以上のように各ビンに対して tf 値を考慮した $\hat{\lambda}$ を計算した結果を図 3 に示す．ビン化することにより，$tf = 0$ における重みは 0 付近に留まることになり，$tf > 0$ においては各 tf 値の順で傾きが大きくなっていることが確認された．よって，すべての tf 値における重み関数は線形の関係を得ることができた．$I = 10$ における $\hat{\lambda}$ に対してはある程度の下降が確認されたが，現在は機械学習用データの不足が原因と考えられ，今後の課題となる．

$I = 10$ は不安定な部分があるため，$l \leq 9$ に対してのみ 3 を用い，各 tf 値に対して線形回帰を行った結果，各 tf 値における重み関数 $\hat{\lambda}$ は一次式 (19) となり，初期値 $a(tf)$ と傾き $b(tf)$ は表 2 に示す通りとなる．以降，この重みを fit-MI と呼ぶこととする．

$$\hat{\lambda}(t_d) \approx a(tf) + b(tf) \cdot I(q_i; t_d) \tag{19}$$

この式で相互情報量と重みに線形の関係を定めたのはデータの観測結果に従ったものであり，仮定ではない．

表 2 各 tf 値における初期値 $a(tf)$ と傾き $b(tf)$
Table 2 Intercept $a(tf)$ and slope $b(tf)$ for each tf value

tf	$a(tf)$	$b(tf)$
0	0.009	-0.013
1	0.548	0.618
2	0.698	0.740
3	0.911	0.802
4+	1.052	0.887

4. 検索実験

本節では，拡張質問に対する重みである fit-MI を用い，比較的小規模である英語のテストコレクションに対する情報検索システムを構築し，検索性能評価を行う．性能評価には trec_eval[5] を用い，11 点平均適合率 (11pt) と R-適合率を元に評価している．

4.1 データ

情報検索の性能評価に用いた英語のテストコレクションは以下の通りである．これらに対しては，あらかじめ不要語の除去と接辞処理を行った．fit-MI の重みは NTCIR1 テストコレクションにて得られた重みをそのまま使用する．

- Medlars 医学抄録 1033 件, 質問数 30 件
- Cranfield 航空学抄録 1400 件, 質問数 225 件

4.2 情報検索システム

情報検索システムの処理の流れを示す．まず，検索質問を元に初期検索を行う．このときの検索質問に対する重み関数は $tf \cdot idf$ や fit-G を用いる．次に，検索結果の上位に含まれる用語について，検索質問の用語と検索結果中の用語の組み合わせを作り，それぞれについて相互情報量を計算する．最後に相互情報量の高い検索結果中の用語を取得し，検索質問に用語を追加して再建策を行う．このときの拡張質問に対する重みは fit-MI を用いる．

今回は実証的重み付けが妥当であることを確認する目的であるために，速度を求めるわけではないことと，実験中にも試行錯誤を行いたい理由から，Scheme の処理系である Gauche[3] を用いて実装した．

表 3 Medlars に対して検索質問に $tf \cdot idf$, 拡張質問に fit-MI を用いたときの検索性能 (上位 5 件, 下位 5 件を抜粋)
Table 3 IR performance using $tf \cdot idf$ and fit-MI for Medlars

使用文書数	拡張質問数	11pt	R-適合率
5	100	0.5510	0.5328
5	80	0.5424	0.5312
10	100	0.5281	0.5164
10	80	0.5131	0.5153
5	50	0.5050	0.4977
5	1	0.4521	0.4573
ベースライン		0.4516	0.4573
20	1	0.4513	0.4555
20	5	0.4512	0.4570
5	10	0.4511	0.4579

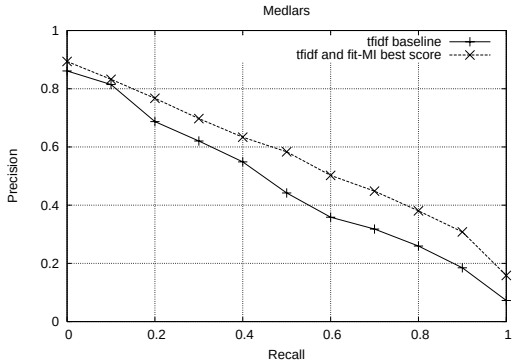

図 4 ベースラインと比較したときの再現率-適合率曲線 (Medlars)
Fig. 4 Recall - Precision curve (Medlars)

4.3 初期検索に $tf \cdot idf$ を用いた実験

実験を行うにあたり，ベースラインとして $tf \cdot idf$ 重みを用いて初期検索だけを行ったときの検索性能を示す．次に，初期検索として $tf \cdot idf$ 重みによる初期検索を行い，検索質問拡張を用いる際の 2 つのパラメータを設定しながら検索を行う．ひとつは初期検索で得られた文書中の上位何件を用いるかの設定であり，もうひとつは検索質問中の用語と相互情報量の高い用語を何語用いるかの設定を行う．

4.3.1 Medlars を用いた実験結果

初期検索で得られた文書中の上位何件を用いるかの設定は，1, 5, 10, 20 件の設定を行い，相互情報量の高い用語を何語用いるかの設定は，1, 5, 10, 20, 30, 50, 80, 100 語の設定を行った．

検索実験を行った結果のうち，上位 5 件と下位 5 件を表 3 に示す．下位 5 件の中にベースラインの結果が含まれ，上位には拡張質問数を多くしたときの結果が含まれている．上位に拡張質問数の多い結果が並んでいることから，初見の拡張質問に対する重み付けが適切であり，不要な語が検索結果に及ぼす影響も少なくなっているものと考えられる．拡張質問を収集するために使用する文書数に関しては，多ければ性能の向上に寄与しないことも確認された．これは，相互情報量の高い用語だけを多く収集することができるが，相互情報量が若干低めであるが新規性の高い用語を採用しなくなる傾向にあり，結果として再現率が向上しないことが考察される．検索結果ランキングに対する再現率の推移は 4.5 に示す．

Medlars テストコレクションの検索性能が最高であった条件は，使用文書数が上位 5 件，拡張質問に使用した用語数 100 語のときである．この結果をベースラインと比較し，再現率-適合率曲線に表した結果を図 4 に示す．検索質問拡張を用いないベースラインと比較して，すべての再現率レベルにおいての適合率が上回っていることが確認された．

4.3.2 Cranfield を用いた実験結果

初期検索で得られた文書中から使用する文書数は上位から 1, 5, 10 件の設定で行い，相互情報量の高い用語を拡張質問に使用する数は 1, 5, 10, 20, 30, 50, 80, 100 語で検索実験を行った．

検索実験を行った結果のうち，上位 5 件と下位 5 件を表 4 に示す．こちらも下位 5 件のなかにベースラインが含まれており，残る多くはベースラインよりは性能が良くなったと思われるが，上位 5 件の結果を見てもそれほど向上したとは言えない．

Cranfield テストコレクションの検索性能が最高であった条件は，使用文書数が上位 1 件，拡張質問に使用した用語数が 30 語のときである．この結果をベースラインと比較したときの再現率-適合率曲線を図 5 に示す．各再現率レベルにおける適合率はベースラインよりも上回っているが，それほど性能向上に寄与しているとは言いがたい．性能をより向上させるためには，現在は NTCIR1 テストコレクションで計算された重みを用いているため，Cranfield に対しての重みを再学習させる方法と，fit-MI と同じ単位である $\hat{\lambda}$ の重み fit-G を初期検索として採用することが考えられる．後者は 4.6 にて行った．

4.4 $tf \cdot idf$ 重みによる検索質問拡張との比較

本稿において検索質問拡張に必要な用語の選定方法とそれに対する実証的重み付けが妥当であるか，すなわち，相互情報量と fit-MI が検索結果から妥当であるかを比較実験より考察する．比較対照としては，$tf \cdot idf$ による初期検索ののち，上位の文書集合から得る用語の選定方法に idf か相互情報量，得られた単語に対しての重み付けとしては $tf \cdot idf$, idf の代わりに相互情報量を使用した $tf \cdot MI$, fit-MI のそれぞれ

表 4 Cranfield に対して検索質問に $tf \cdot idf$, 拡張質問に fit-MI を用いたときの検索性能 (上位 5 件, 下位 5 件を抜粋)
Table 4 IR performance using $tf \cdot idf$ and fit-MI for Cranfield

使用文書数	拡張質問数	11pt	R-適合率
1	30	0.3376	0.3070
1	50	0.3360	0.2959
1	80	0.3331	0.2902
1	10	0.3295	0.3041
5	30	0.3282	0.3132
10	100	0.3151	0.3014
1	1	0.3149	0.2957
10	1	0.3146	0.2970
ベースライン		0.3128	0.2953
5	1	0.3115	0.2963

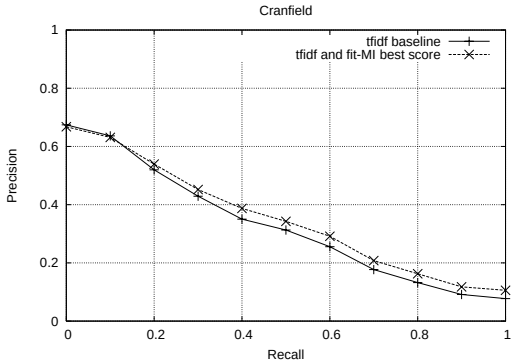

図 5 ベースラインと比較したときの再現率-適合率曲線 (Cranfield)
Fig. 5 Recall - Precision curve (Cranfield)

に対して実験を行う．なお，今回は Roccio の式[14] は元の検索モデルがベクトルモデルであり，本稿で扱っている確率モデルには該当しないため，比較の対象とはしなかった．

実験は本稿で提案した相互情報量と fit-MI の組み合わせと，用語の選定に idf と重みに $tf \cdot idf$ の組み合わせ，そして用語の選定に相互情報量と重みに $tf \cdot MI$ の 3 種で行った．それぞれの結果に対して最良である条件のときの 11 点平均適合率を表 5 に示す．Medlars, Cranfield それぞれに対して計算を行ったが，拡張質問のための用語選定に idf を用いたときの検索性能は相互情報量を用いた他の 2 種と比較して性能が優れないことがわかる．拡張質問の数もそれほど多くないことから，単純に 1 語のみの情報を利用した検索質問拡張はそれほど有用でないことがわかる．$tf \cdot MI$ は Medlars に対しての性能は fit-MI よりは良いが，Cranfield に対しての性能は落ちている．拡張質問の数も少なくなっていることから，再現率の向上には寄与しがたいと考えられる．

表 5 重みによる性能変化
Table 5 Precisions for different weighting methods

Medlars

拡張質問への重み	使用文書数	拡張質問数	11pt
fit-MI	5	100	0.5510
$tf \cdot idf$	20	1	0.4520
$tf \cdot MI$	5	100	0.5562

Cranfield

拡張質問への重み	使用文書数	拡張質問数	11pt
fit-MI	1	30	0.3376
$tf \cdot idf$	1	10	0.3201
$tf \cdot MI$	1	5	0.3269

以上より，拡張質問として有用な用語の選択には検索文書中の用語だけを着目するのではなく，検索質問中の用語との共起情報，本稿では相互情報量を用いることが妥当であることと，再現率を向上させるためには fit-MI の方が妥当であろうということが考察される．

4.5 fit-MI による再現率の推移

拡張質問に対して fit-MI を使用したときの再現率の推移を示す．比較対象は $tf \cdot idf$ ベースラインと 11 点平均適合率が最良，最悪である条件の結果を用いる．再現率の計算は検索結果のランキングを元に，あるランキングの時点における平均再現率をプロットする．ランキングは上位 100 件までをプロットした．

Medlars, Cranfield テストコレクションに対しての再現率の推移を示した結果を図 6 に示す．Medlars の結果では，最良の結果とベースラインとでは，上位 10 件までの再現率は最大 0.03, 以降は再現率が最大 0.1 程向上した．逆に最悪時の結果とベースラインとでは，上位 10 件までの再現率は逆に最大 0.003 程低下し，以降は再現率が少々向上するものの，上位の結果の影響が大きく出ているものと考察される．

Cranfield では最良の結果とベースラインとでは，再現率が最大 0.02 程しか向上しなかった．Cranfield に関しては相互情報量による再現率の向上がそれほど見込まれなかったため，相互情報量の他に，補完類似度[2)11)] などの評価尺度に関する検討も今後の課題となる．

4.6 初期検索に fit-G を用いた実験

検索質問，拡張質問ともに $\hat{\lambda}$ を用いたときの実験を行う．今までは初期検索に対して $tf \cdot idf$ を用いていたが，$tf \cdot idf$ と $\hat{\lambda}$ とでは尺度が違うため，検索質問に対する各文書のスコアと拡張質問に対する各文書のスコアを加算することはどをちらかの性能に悪影響を及ぼしかねない．そこで，検索質問に対しての重みを同じ $\hat{\lambda}$ 尺度である fit-G を用いた．

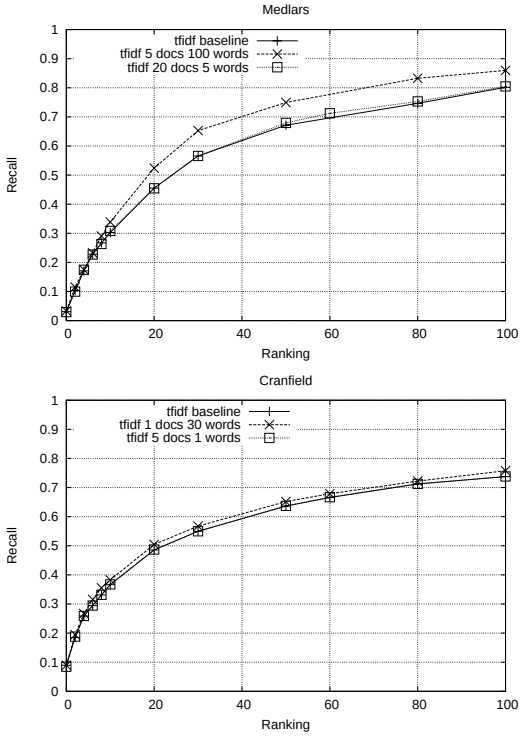

図 6 拡張質問に使用する文書数と用語数による再現率曲線の変化
Fig. 6 Effect of # of documents and # of terms for query expansion

$tf \cdot idf$ ベースライン，$tf \cdot idf$ に fit-MI による拡張質問を使用，fit-G のみ，fit-G に fit-MI による拡張質問を使用，の 4 パターンについて検索実験を行ったときの最良の結果をそれぞれ表 6，図 7 に示す．fit-G は非常に有効な重みであることが[7] にて報告されているが，fit-MI も使用することにより，さらに 11 点平均適合率を上げることができた．しかし，fit-G 単体と比較して fit-MI を併用したときは再現率レベル 0, 0.1 における適合率が若干下がっていることが確認された．再現率レベルが低いときの適合率の向上が今後の課題である．

5. まとめと今後の課題

本稿では検索質問拡張における，検索質問に関連する用語の選定方法に相互情報量を用い，選定された用語に対する重み関数 $\hat{\lambda}$ に関しての実証的重みの計算方法を示し，実証的重み fit-MI を得た．fit-MI を用いることにより，従来の $tf \cdot idf$ ベースラインと比較して検索性能が向上することを示した．機械学習を行うためのテストコレクションには NTCIR1 日本語論文アブストラクトを用い，得られた重みを用いた検索実験には英語のテストコレクションを用いたが，全体的に再現率の向上から来る検索性能の向上が確認された．これは重み関数 $\hat{\lambda}$ は 0 付近と相互情報量の大きさ付近に現れることからどのテストコレクションに対しても有用である汎用の重みであることがわかる．[7] の fit-G と共に fit-MI を用いることにより fit-G 単体と比較して検索性能が向上することを示した．

今後の課題の課題としては，[7] の拡張質問に対する重み fit-E との比較と，日本語テストコレクションに対する検索実験，文書数を増加させたときの重みの調

表 6 初期検索に $tf \cdot idf$ と fit-G を用いたときの性能比較
Table 6 IR performance using $tf \cdot idf$ and fit-G for initial query

Medlars

重み関数	使用文書数	拡張質問数	11pt
$tf \cdot idf$	0	0	0.4516
$tf \cdot idf$ + fit-MI	5	100	0.5510
fit-G	0	0	0.5011
fit-G + fit-MI	5	80	0.5571

Cranfield

重み関数	使用文書数	拡張質問数	11pt
$tf \cdot idf$	0	0	0.3128
$tf \cdot idf$ + fit-MI	1	30	0.3376
fit-G	0	0	0.3739
fit-G + fit-MI	1	10	0.3929

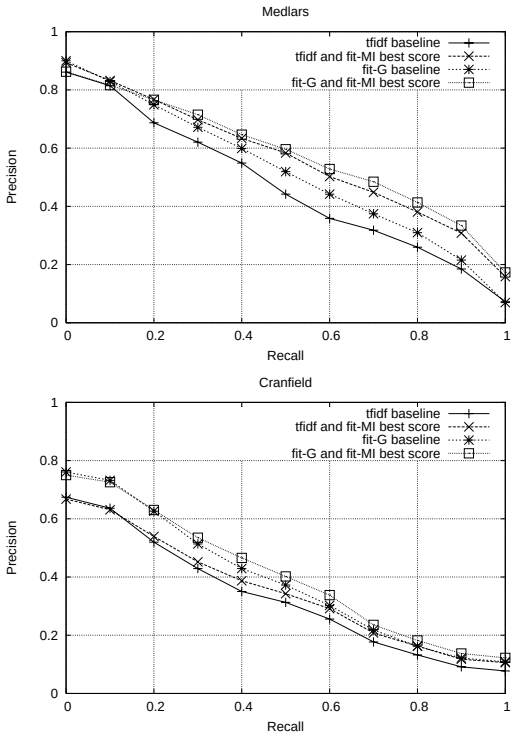

図 7 fit-G を考慮したときの再現率-適合率曲線
Fig. 7 Recall - Precision curve using fit-G

査が挙げられる．本稿では計算量の問題から機械学習用データを削ることになったため，2 語を同時に含む文書数の計算を高速に行うアルゴリズムの検討が求められる．

参 考 文 献

1) Kenneth W. Church and Patrick Hanks. Word association norms, mutual information, and lexicography. *Computational Linguistics*, Vol. 16, No. 1, pp. 22–29, March 1990.
2) N. Hagita and M. Sawaki. Robust recognition of degraded machine-printed characters using complimentary similarity measure and error-correction learning. *Proceedings of the SPIE - The International Society for Optical Engineering, 2442*, pp. 236–244, 1995.
3) Shiro Kawai. *Gauche - A Scheme Interpreter*. http://shiro.dreamhost.com/scheme/gauche/.
4) National Institute of Informatics. *NTCIR Project*. http://research.nii.ac.jp/ntcir/.
5) National Institute of Standards and Technology. *Text REtrieval Conference (TREC)*. http://trec.nist.gov/.
6) Carl Sable and Kenneth W. Church. Using bins to empirically estimate term weights for text categorization. *Conference on EMNLP2001*, 2001.
7) Kyoji Umemura and Kenneth W. Church. Empirical term weighting and expansion frequency. *Workshop SIGDAT, EMLNP2000*, pp. 117–123, 2000.
8) Yinghui Xu and Kyoji Umemura. Very low dimensional latent semantic indexing for local query regions. In Jun Adachi and Kam-Fai Wong, editors, *Proceedings of the Sixth International Workshop on Information Retrieval with Asian Languages*, pp. 84–91, 2003.
9) 金谷敦志, 梅村恭司. 相関係数と情報検索のための実証的重みの分析. 情報処理学会全国大会, 第 65 回, pp. (3)145–146, 3 月 2003 年.
10) 佐々木稔, 新納浩幸. 潜在的文脈関連度を用いた検索質問拡張. 情報処理学会研究報告 2002-FI-68 2002-NL-151, pp. 65–72, 9 月 2002 年.
11) 山本英子, 梅村恭司. コーパス中の一対多関係を推定する問題における類似尺度. 自然言語処理, Vol. Vol.9, No. 2, pp. 45–75, 4 月 2002 年.
12) 中島浩之. シソーラスを用いた語の共起関係推定による roccio フィードバックの精度向上. 情報処理学会論文誌, Vol. Vol.43, No.5, pp. 1457–1469, 5 月 2002 年.
13) 柘植覚, 獅子堀正幹, 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. Vol.44, No. 1, pp. 59–67, 1 月 2003 年.
14) 北研二, 津田和彦, 獅々堀正幹. 情報検索アルゴリズム. 共立出版, 2002 年.