文書クラスタリングを用いた新情報アクセス手法

1　情報アクセス支援における文書クラスタリングの利用

1.1　はじめに

対話的な文書検索では，ユーザが検索システムに検索要求を与えると，検索システムは与えられた検索要求と検索対象文書との適合性 (relevance) を計算し，適合性が高いと判断した文書をユーザに提示する．ユーザは次に，それら幾つかの文書に対して実際に適合性を判定する．ユーザが適合/不適合と判定した文書群は検索システムにフィードバックされ，検索結果の改善に用いられる．この適合性フィードバック (relevance feedback) は，ユーザが検索結果に満足するまで繰り返される．

適合性フィードバックでは，ユーザから検索システムへのフィードバック量が多いほど，検索システムは検索結果を改善することができる．Buckley等は，検索精度 (recall/precision) はユーザが判定した適合文書数の対数にほぼ比例することを実験的に確かめた [6]．よって，できるだけ多くの適合性判定を行なうことがユーザに求められるが，これはインターネットを利用した検索など即時的な検索では過度の期待であることが多い．一方，文書フィルタリングでは，ユーザの検索要求が長く継続するため，各々の検索要求に対して比較的多くの適合性判定を集めることができる．従って，近年の適合性フィードバックの研究では，文書フィルタリングを対象にして，十分な数の適合性判定を仮定したものが多い [5, 20, 17]．

これに対し本研究では，対話的文書検索を対象にし，適合性判定数が 10 ないし 20 以下と非常に少ない状況を仮定する．インターネットにおける対話的文書検索では，ユーザに 100 以上もの適合性判定を強いるのは事実上不可能である．また，文書フィルタリングにおいても，10 ないし 30 の適合性判定をフィードバックするのみで，1000 以上の適合性判定をフィードバックした結果より高々10%劣る検索精度に達するという研究結果もある [3]．つまり，検索精度が本質的に改善するのは，フィードバックする適合性判定の数が非常に少ない領域においてである．本研究では，このような状況下でいかに効率良く検索結果を改善していくかを目的に，増進的適合性フィードバック (incremental relevance feedback) と検索結果の自動分類の二つの手法を比較検討する．

通常の適合性フィードバックが幾つかの適合性判定をまとめてフィードバックするのに対し，増進的適合性フィードバックは，ユーザの適合性判定を増進的にフィードバックする手法であり，ユーザがある文書の適合性を判定した時点で即座に検索結果を更新する．増進的適合性フィードバックは Aalbersberg により提案され [2]，後に，Allan により文書フィルタリングにおいてその効果が調べられた [3]．本研究では，対話的な文書検索に増進的適合性フィードバックを適用し，少ない適合性判定で効率良く検索精度が改善できるかどうかを実験により確かめる．

一方，検索結果の自動分類は，文書クラスタリングにより検索結果を自動分類してユーザに提示する方法で，近年では検索絞り込みの支援に用いられることが多い [7, 10]．通常の順位付けされた文書表示に比べ，検索結果を自動分類することで，ユーザの情報要求 (information need) に適合する文書を効率良く見つけることができる [9]．また，検索結果のクラスタリングは，自動適合性フィードバック (pseudo relevance feedback) の前処理としても有用である [4]．本研究では，フィードバックする適合性判定の数とそれによる検索精度改善率との関係に注目して，適合性フィードバックにおけるクラスタリングの効果を調べる．また，検索要求によるバイアスを考慮できるよう従来のクラスタリングアルゴリズムを改良し，その有効性を調べる．

1.2　比較基準モデル: ベースラインモデルと上限モデル

実験には TREC コレクションを用いた．ディスク 1 とディスク 2 に含まれる計 742,709 個の文書を検索対象とし，トピック 101 から 150 までの 50 のトピックそれぞれから”title” フィールド，“desc(description)“

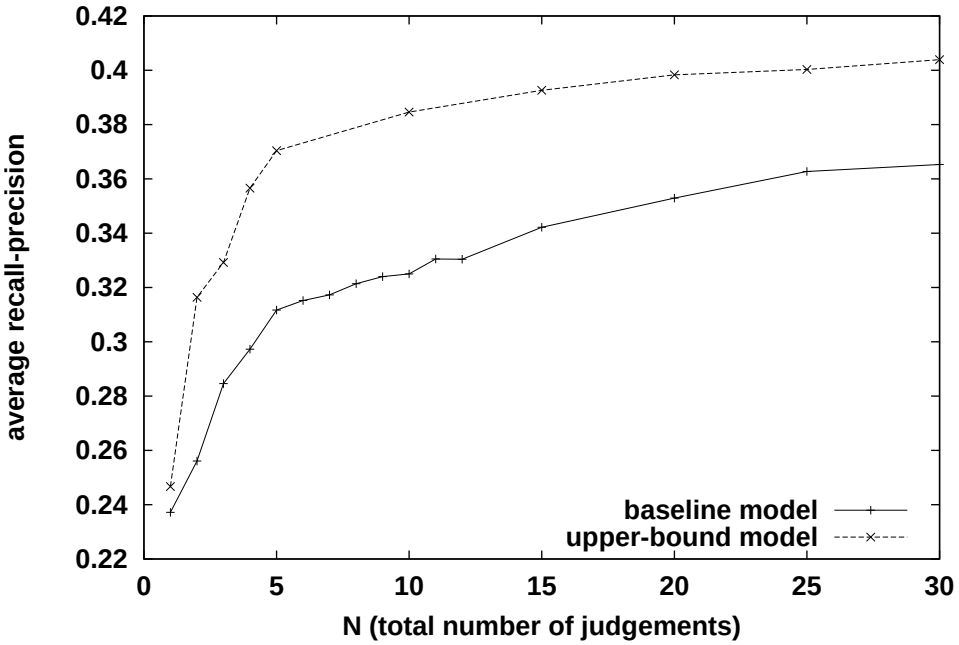


図 1: 比較基準モデルにおけるフィードバックの効果

フィールドを抜き出し50の検索要求を作成した．適合性判定はTRECコレクションで用意されているものをそのまま用いた．

検索モデルはベクトル空間モデルを用いた．タームの重み付け方法としては，Singhal等が提案したLt.Lnc方式を用いた [19]．

適合性フィードバックによるタームの重み修正法は，以下の改良Roccio法を用いた．

$$Q_i^{新} = \alpha Q_i^{旧} + \beta \frac{1}{|適合文書|} \sum_{適合文書} wt_i - \gamma \frac{1}{|不適合文書|} \sum_{不適合文書} wt_i \qquad (1)$$

ここで$Q_i^{旧}$はタームiの修正前の重み，$Q_i^{新}$は修正後の重みである．wt_iはそれぞれの判定文書におけるタームiの重みである．パラメータα, β, γはそれぞれ8, 16, 4に設定した．これらは，使用したTRECコレクションにおいて最も良く使われている値である．予備実験を行った結果，不適合文書をフィードバックしても検索精度がほとんど向上しなかったため [1]，本実験では適合文書のみフィードバックした．

比較基準モデルとして，ベースラインモデル，上限モデルを仮定した．それぞれのモデルではまず，初期検索要求から上述のベクトル空間モデルにより文書を検索し，順位付き文書リストを得る．ベースラインモデルでは，上位N位までの文書を調べ，適合する文書のみをフィードバックして，改良Rocchio法により新たな検索要求を作成する．上限モデルでは，同じ文書リストにおいて，最上位から下位方向に適合文書のみをN個抽出し，このN個の適合文書から新たな検索要求を作成する．つまり，上限モデルでは完璧な文書ランキングを仮定していることになる．ここでは，ユーザは適合文書のみに遭遇するを仮定している．一方，ベースラインモデルでは，実際の検索状況と同じく，初期検索精度に応じた数の不適合文書にも遭遇する．更新した検索要求からそれぞれ新たな順位付き文書リストを得て，この文書リストに対して平均recall-precisionを計算する [2]．

[1] 不適合文書のフィードバックも検索精度を改善したが，適合文書のフィードバックに比べるとその量は無視できるほど小さかった．我々の実験では，フィードバックに用いる適合性判定の数が少ないことに注意されたい．文書フィルタリングなど，十分な数の適合判定が利用できる環境では，不適合文書のフィードバックも効果的であることが知られている．

[2] つまり，単一のデータセットに対して検索，フィードバック，および評価を行う．文書フィルタリングなどのタスクでは試験用，

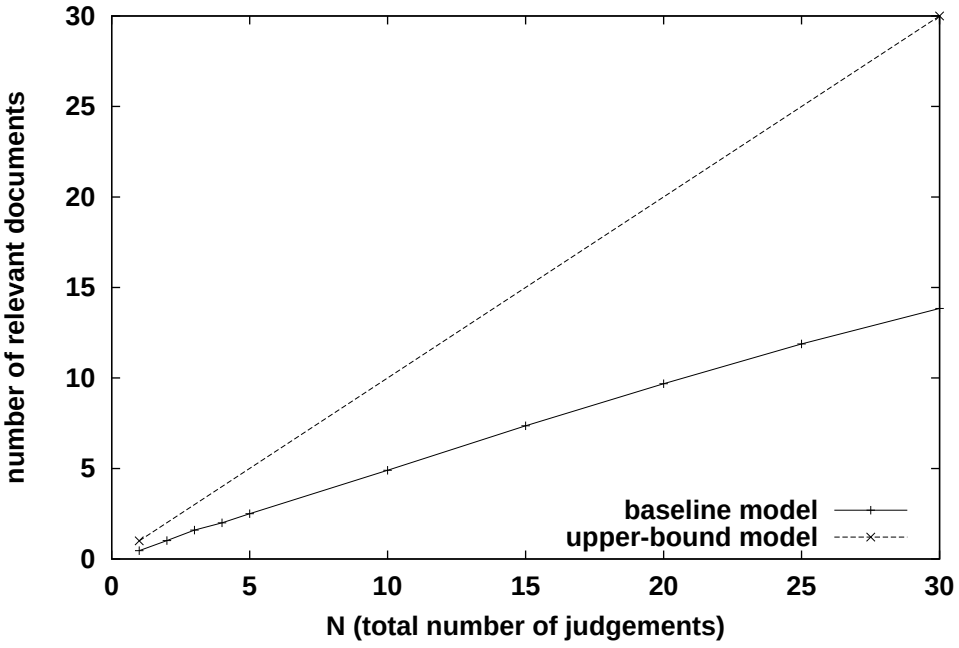


図 2: 比較基準モデルにおけるフィードバックの内容

比較モデルにおける適合性判定数Nと平均recall-precisionの関係を図 1に示す．本研究では，適合性判定の数が少ない状況での精度改善に注目するため，Nは30以下に限った．また，図 2に，N個の判定文書に含まれる適合文書数を示す．

1.3 増進的適合性フィードバック

1.3.1 手法

Aalbersbergが提案した増進的適合性フィードバック [2] では，検索システムは現在の検索要求に対し最も適合性が高いと思われる文書をユーザに提示し，ユーザはその文書に対する適合性を判定する．ユーザの適合性判定は即座に検索システムにフィードバックされ，検索システムは検索要求を更新し，それに応じて全文書のスコアも再計算する．従って，検索結果の文書ランキングは常に，ユーザがこれまで行なった適合性判定に基づいた最新のものに保たれている．この貪欲 (greedy) な戦略により，ユーザはより多くの適合文書を見つけることができる．

例えば，ユーザがある順位付き文書リストから一つの適合文書を見つけたとする．ここで，次の適合文書を見つけるために二つの選択肢がある．最初の選択肢は，同じ順位付き文書リストから次の適合文書を見つける方法 (ベースラインモデル) であり，もう一つは，たった今見つけた適合文書をシステムにフィードバックして順位付き文書リストを更新し，更新したリストから次の適合文書を見つける方法 (増進的適合性フィードバック) である．図 1より，フィードバックする適合性判定の数が増すにつれ検索精度も向上することがわかるため，更新した文書リストから探す後者の方が，より簡単に次の適合文書を見つけることができると期待できる．

Aalbersbergは，この仮説を実験的に確かめなかった [2] [3]．Allanは後に，増進的フィードバックに関す

統制用の二つのデータセットを用意し，評価は試験用データセットに対して行うのが一般的であり，これは実際の運用状況にも適合している．一方，実際の対話的文書検索では，単一のデータセットに対して検索，フィードバックが行われるため，今回は単一データベース内で評価を行った．

[3] Aalbersbergの実験では，適合性フィードバックの回数は一回に限られていた．

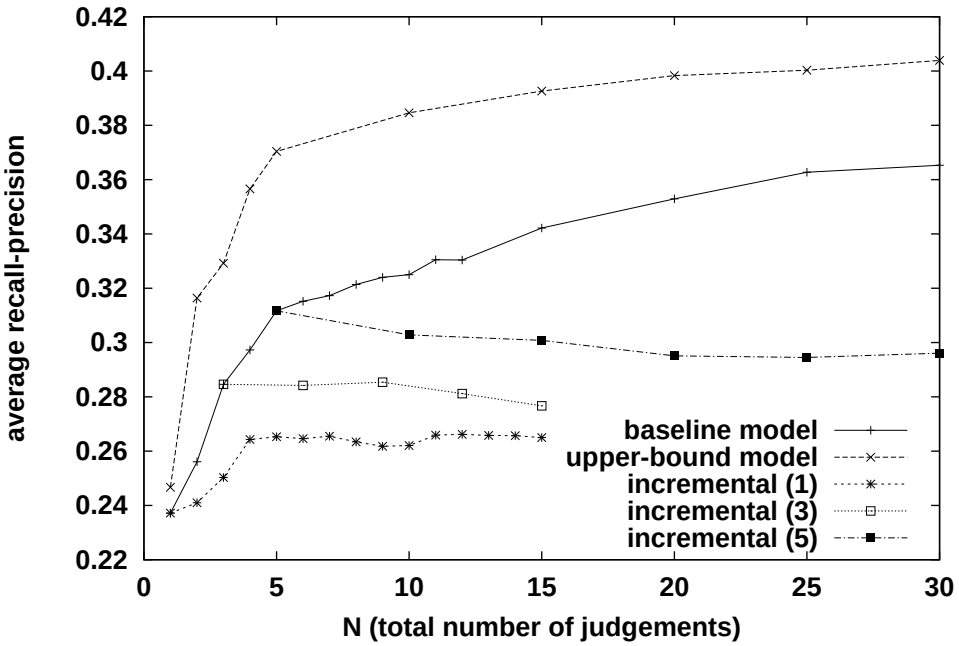


図 3: 増進的適合性フィードバックにおけるフィードバックの効果

る多くの実験を行ったが [3]，それらは対話的検索においてではなく，バッチ的な文書フィルタリングにおいてであった．Allan の実験では，適合性判定は増進的にフィードバックされるが，それらはあらかじめ用意した静的な判定集合の中からランダムに選ばれている．本研究で興味あるのは，フィードバックにより動的に更新される文書集合から，次のフィードバックに用いる文書を選ぶという状況である．

1.3.2 結果と考察

図 3に，実験結果を示す．増進的適合性フィードバックにおけるNとは，増進的に与えられた適合性判定の総数のことである．実験では，元々の増進的適合性フィードバックに加え，判定を一時的にためておき (バッファリング)，まとめてフィードバックする方法も比較した．一時的にためておく判定数 (バッファの大きさ) は3および5を試みた．元々の増進的適合性フィードバックはバッファの大きさが1の場合に相当する．

図からわかるように，残念ながら増進的適合性フィードバックはベースラインモデルを上回ることができなかった．バッファの大きさが1の場合，最初の数サイクルのフィードバックでしか精度が改善していない．バッファの大きさが3および5の場合，全てのフィードバックサイクルにおいて精度の改善が見られない．これは，フィードバックした判定に含まれる適合文書数が少ないためではない．図 4からわかるように，増進的適合性フィードバックは期待通りベースラインモデルより多くの適合文書を見つけている．

それではなぜ，より多くの適合文書をフィードバックするにも関わらず，増進的適合性フィードバックはベースラインモデルに劣るのだろうか．各フィードバックサイクルで新たに判定する文書は，以前に判定した文書とほとんど同じものであるか，現在の検索要求のサブトピックに関するものである可能性が高い．なぜなら，新たな判定は，現在の検索要求と深く関連するトップランクの文書に対してのみ行われるからである．よって，それらの文書を使って更新した検索要求は，ほとんど変化しないか，または特定化されるにすぎず，結果，検索精度も向上しないか，または総合的には悪くなってしまう．バッファの大きさが3および5の場合，フィードバックを続けても初期検索結果を改善できないのは，上記の影響だと考えられる．これに対しベースラインモデルでは，判定文書は常に初期検索結果から選ばれるため，Nが大きくなるにつ

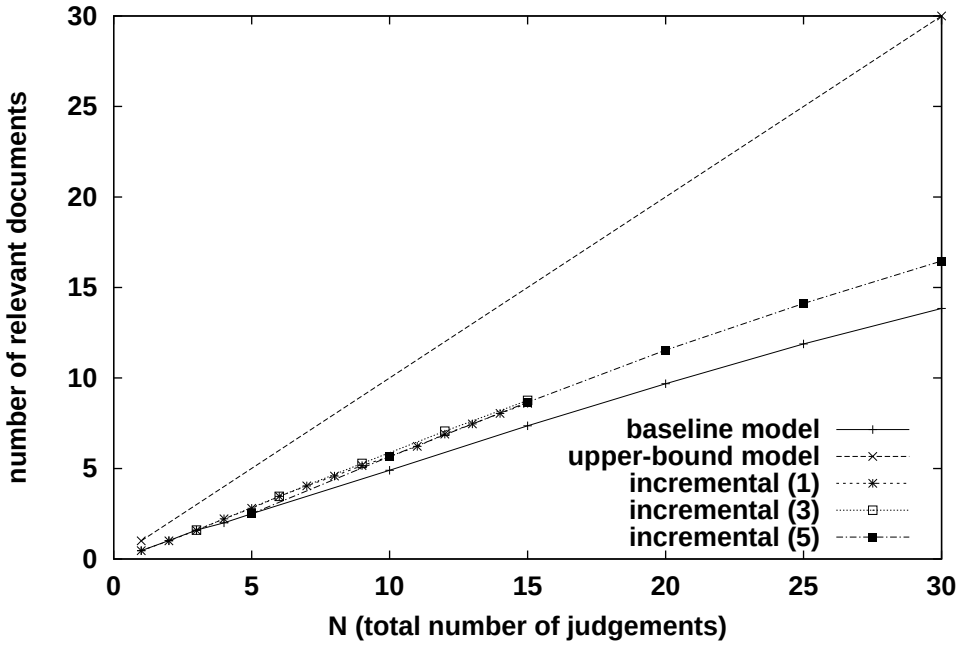


図 4: 増進的適合性フィードバックにおけるフィードバックの内容

れ，ユーザの情報要求に関連する様々なトピックを含む文書がフィードバックされやすくなる．

この現象を詳しく調べるために，50の検索要求を” 高品質/低品質” の2つのグループに分け，それぞれに対して実験結果を集計した．実際には，上位30位の検索結果に15個以上の適合文書を含むような検索要求を高品質とし，それ以外を低品質とした．高品質検索要求は，情報要求に関連する様々なトピックを含む文書群を上位に検索できる可能性が高く，逆に低品質な検索要求は，情報要求の表現としてあまり有用ではない．各々のグループに含まれる検索要求 (トピック番号) は以下のとおりである．

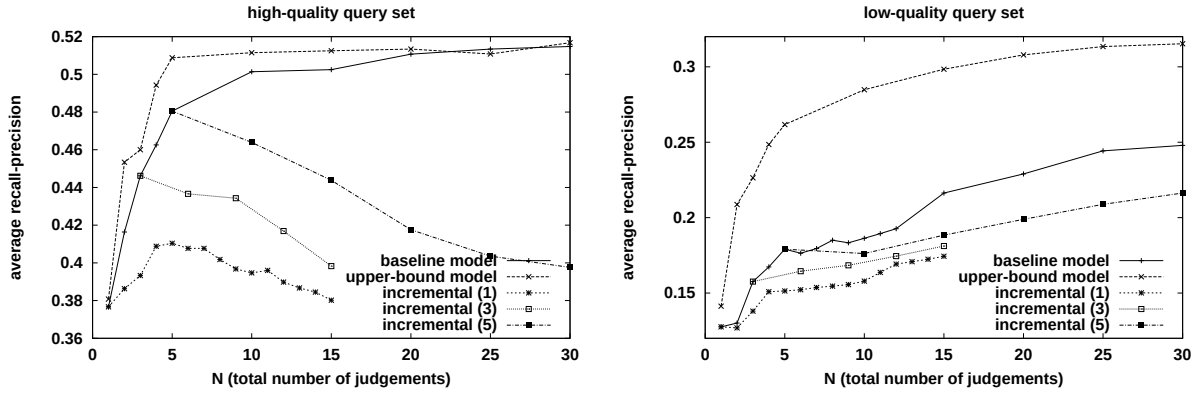


図 5: 増進的適合性フィードバックにおけるフィードバックの効果 (高品質/低品質検索要求)

	高品質検索要求	低品質検索要求
トピック番号	106, 107, 108, 109, 110, 111, 112, 115, 118, 123, 130, 132, 133, 134, 135, 136, 137, 142, 145, 146, 148, 150	101, 102, 103, 104, 105, 113, 114, 116, 117, 119, 120, 121, 122, 124, 125, 126, 127, 128, 129, 131, 138, 139, 140, 141, 143, 144, 147, 149

上位30位における平均適合率は，高品質セットで0.7500，低品質セットで0.2345であった．

図 5にそれぞれの結果を示す．低品質検索要求では，フィードバックが進むにつれ，検索結果も改善されていくが，それでも絶対的な精度はベースラインモデルに劣る．一方，高品質検索要求では，バッファの大きさが1の場合を除き，初期検索結果にフィードバックを加えるにつれ，検索精度は急激に落ちていく．これは，初期検索結果に含まれる様々なトピックの一面にのみ検索が集中していくことを示唆している．

以上の結果から，増進的適合性フィードバックは検索精度の総合的な向上には不向きであることがわかった．反面，検索要求が特定のトピックに集中していくという意味で，検索結果を絞り込んでいく過程で有用かもしれない．また，適合性フィードバックにおいては，フィードバックする文書数を単に増すことが総合的な精度向上にはつながらないこともわかった．総合的な精度を上げるには，むしろフィードバックする文書の多様性が重要となる．これは，文書分類で訓練例をいかに効率良くサンプリングするかという問題にも深く関係する [14].

1.4 検索結果の自動分類

1.4.1 手法

「ある情報要求に適合する文書はお互いに類似している」というクラスタ仮説 [21] が正しければ，検索結果をクラスタリングすることで，適合文書のみ含むクラスタと不適合文書のみ含むクラスタに分割することができる．更に，ユーザが適合クラスタを選ぶことができれば，その中に含まれる多くの適合文書を効率良くフィードバックすることができる．ここでは，不適合クラスタに集められた多くの不適合文書を調べる必要がない．

クラスタリングによる検索結果の自動分類は，多くの研究者により様々な検索環境で用いられている．Scatter/Gather [8] は，クラスタリングとクラスタ選択を繰り返すことで検索結果の絞り込みを支援する手法である．Buckley 等は，検索結果のクラスタリングは自動適合性フィードバックの前処理としても有用であることを示した [4]．Evans 等は，検索結果をクラスタリングして表示することで，ユーザが実際に効率良く適合性フィードバックを行えることを示した [9]．本研究では，フィードバックする適合性判断の数とそれによる検索精度改善率との関係に着目して，検索結果をクラスタリングすることの有用性を調べる．前節の実験結果からわかったように，単に多くの適合文書をフィードバックすれば検索精度もそれに応じて改善するわけではないため，クラスタ仮説により適合文書を効率良く集めることができても，それらがフィードバック情報として有用であるとは限らない．

実験では，各検索要求から150の文書を検索し，これら150の文書を5個のクラスタに分割した．クラスタリングアルゴリズムとしては確率的クラスタリング [12] を用いた．5個のクラスタから最良のクラスタを選ぶために，[18] で使われている”DENSITY” 法を用いた．“DENSITY” 法では，適合する文書 (正解) を含む割合が大きいクラスタから順にクラスタを選ぶ．各クラスタ内の文書は，検索要求との関連度によりソートする．実験では，まず1位のクラスタを選び，その中からN個の文書を選びフィードバックに用いる．1位のクラスタにN個以下の文書しか存在しない場合，2位のクラスタから残りの文書を選ぶ．

“DENSITY” 法を用いるということは，実際のユーザも適合文書を多く含むクラスタを比較的容易に選択できると仮定しているのだが，これは少々強すぎる仮定かもしれない．ユーザはほとんどの場合 “DENSITY”

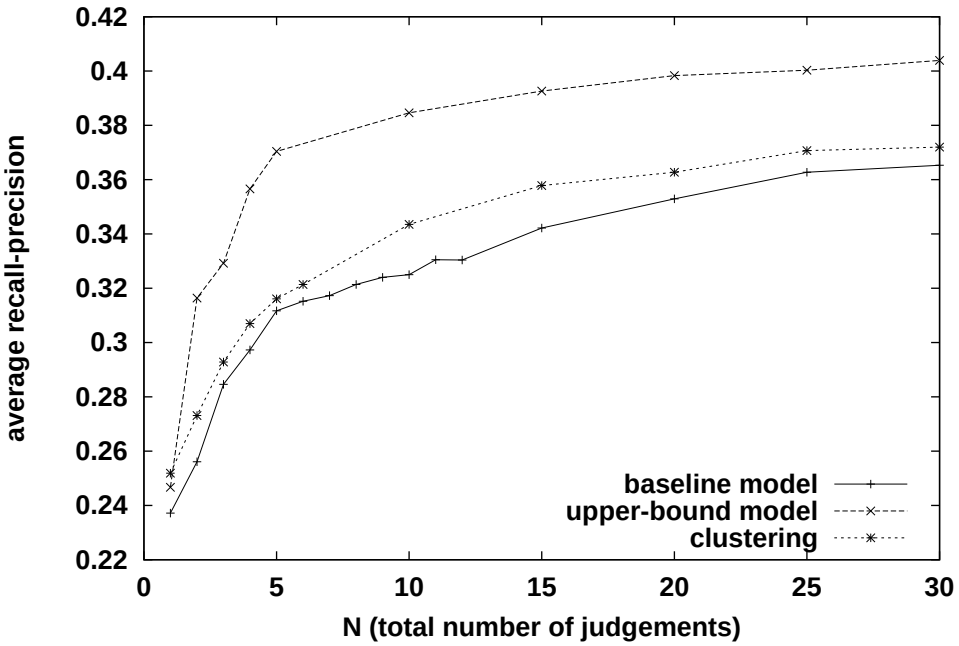


図 6: 検索結果自動分類法におけるフィードバックの効果

法による1位のクラスタを選ぶことができるという実験結果 [10] もあるが，そのコストも含め更に詳しく調べる必要がある．

1.4.2 結果と考察

図 6に実験結果を示す．増進的適合性フィードバックとは異なり，検索結果の自動分類法では，フィードバックにより常に検索精度が向上している．平均recall-precision の値も常にベースラインモデルを上回る．また，図 7よりわかるように，自動分類した検索結果から文書を選ぶほうが，オリジナルの検索結果から文書を選ぶ(ベースラインモデル)よりも多くの適合文書を見つけている．以上はクラスタ仮説の正当性，およびその有効性を実証している．

検索結果の自動分類法，増進的フィードバック両者は，ほぼ同数の適合文書をフィードバックするにも関わらず，フィードバックの効果が全く異なることから，両者は全く異なる種類の適合文書を見つけていることもわかる．

ここで，増進的適合性フィードバックの場合と同様に，50 の検索要求を高品質/低品質の二つのグループに分け，それぞれに対して結果を集計し図 8に示す．図より，高品質検索要求においてクラスタリングの問題点が見てとれる．高品質検索要求は多くの適合文書を検索するため，クラスタリングの対象文書もほとんどが適合文書である．今回用いたクラスタリングアルゴリズムは重複するクラスタを許さないため[4]，これら適合文書をクラスタリングし，その中の1つか2つを選ぶことは，適合文書がカバーする様々な適合トピックの一部分しか考慮しないことに相当し，これは総合的な検索精度の低下を招く．実際，クラスタリングによる検索結果自動分類法はベースラインモデルを上回ることができない．一方，低品質検索要求は比較的小数の適合文書しか検索しないため，適合文書はクラスタリングにより1つないしは2つのクラスタに集まりやすく，これらのクラスタを選ぶことで必要かつ十分なフィードバックが可能になる．実際，検索結果の自動分類法はベースラインモデルを上回る．高品質検索要求におけるクラスタリングの問題点を解決するためには，例えば，複数のクラスタから上位の文書のみを集めてフィードバックするなどの方法もある

[4] ほとんどのクラスタリングアルゴリズムも同じである．

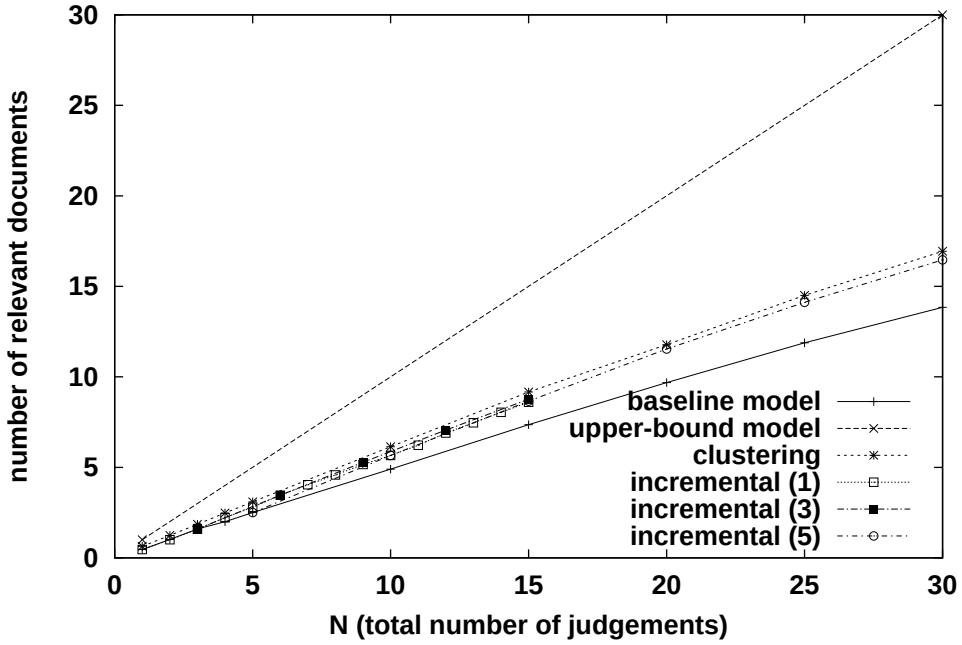


図 7: 検索結果自動分類法におけるフィードバックの内容

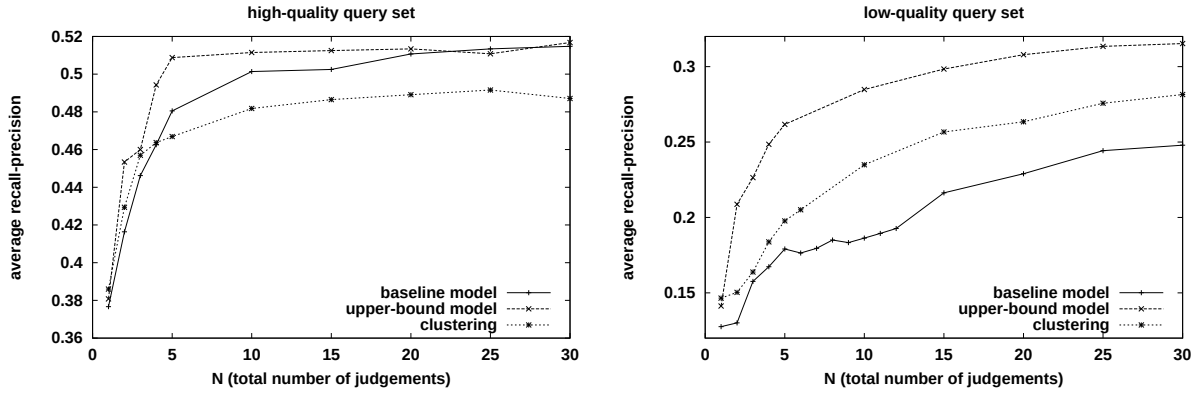


図 8: 検索結果自動分類法におけるフィードバックの効果 (高品質/低品質検索要求)

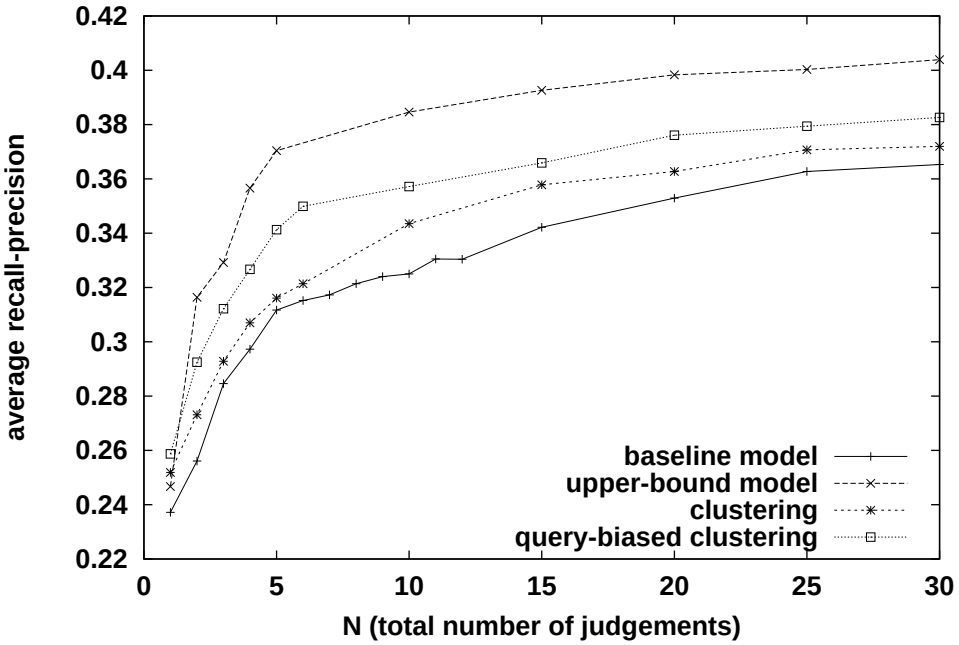


図 9: 検索要求でバイアスされたクラスタリングを用いた検索結果自動分類法におけるフィードバックの効果

が，次節では別の本質的な解決法を試みる．

1.5 検索要求によるバイアスを考慮したクラスタリング

前節の結果は，適合性フィードバックにおける検索要求の重要性を示唆している．特に，検索要求が多くの適合文書を検索できるほど，当然ながらその重要性も高い．実際，そのような高品質検索要求に関しては，検索要求との関連度が強い順に検索結果をフィードバックすること (ベースラインモデル) で総合的に良い結果が得られ，検索結果のクラスタリングは必ずしも有効ではない．本節では，検索要求によるバイアスを考慮したクラスタリングアルゴリズムを提案し，その有効性を調べる．

前節の検索結果自動分類法では，検索要求はクラスタリングの対象文書を決めるのみで，クラスタリングアルゴリズム自体は検索要求とは無関係である．本節では，検索要求など任意のバイアスを考慮できるようクラスタリングアルゴリズムを改良する．前節で用いた確率的クラスタリングは，文書 d が与えられたという条件でのクラスタ C の確率 $P(C|d)$ を計算し，クラスタ集合の確率 $\prod_C \prod_{d \in C} P(C|d)$ が最大になるようボトムアップにクラスタを併合していく [12]．ここで，確率 $P(C|d)$ の条件部にバイアス (検索要求) q の情報を加た確率 $P(C|d,q)$ で，全ての $P(C|d)$ を置き換えることにより，バイアスの影響を考慮したクラスタリングアルゴリズムを得る．これは，検索要求 q に含まれる各ターム (単語) の重みを，文字通りバイアスとして，文書 d におけるそれらタームの重みに加えることに相当する．

前節の実験法において，クラスタリングアルゴリズムのみを上述のアルゴリズムで置き換えて実験した結果を図 9に示す．図から，検索要求でバイアスされたクラスタリングは検索結果の分類法として効果的であることがわかる．オリジナルのクラスタリングを用いた検索結果分類法およびベースラインモデルを上回り，かつその差は有為である．これらの差は，高品質検索要求での差によるところが大きい．図 11に高品質/低品質検索要求での結果を示す．高品質検索要求における結果を見ると，検索要求でバイアスされたクラスタリングは，ベースラインモデルと同じかやや上回る結果を出す．つまり，ベースラインモデル

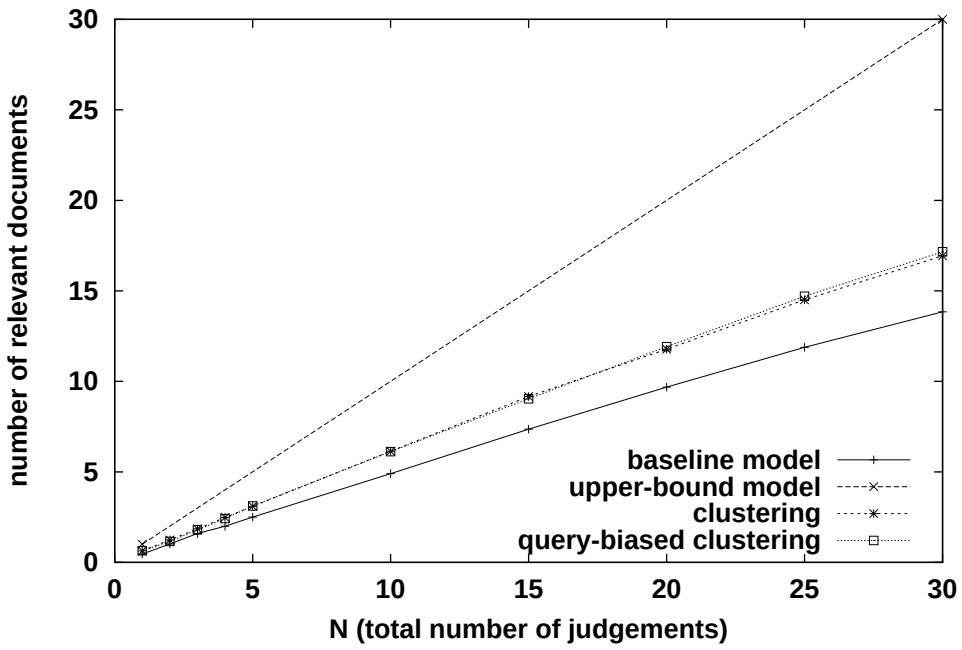


図 10: 検索要求でバイアスされたクラスタリングを用いた検索結果自動分類法におけるフィードバックの内容

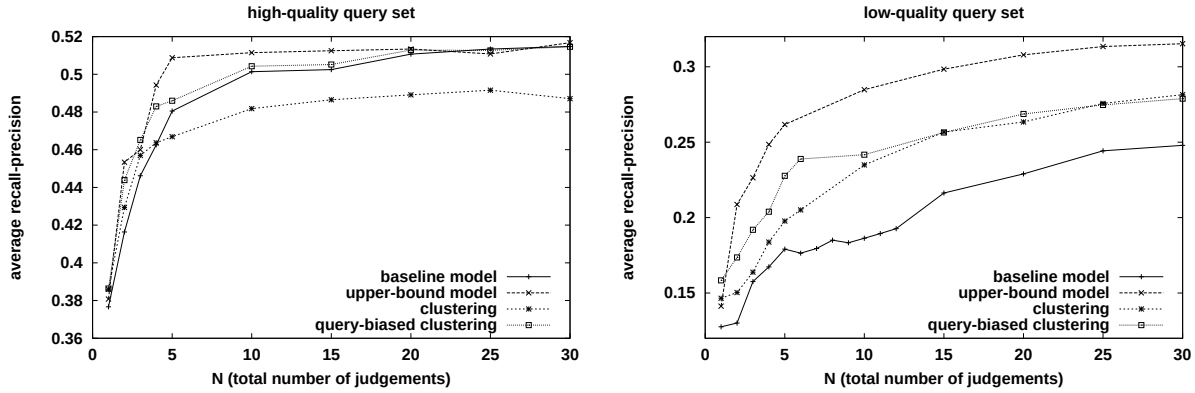


図 11: 検索要求でバイアスされたクラスタリングを用いた検索結果自動分類法におけるフィードバックの効果 (高品質/低品質検索要求)

と同じく，検索要求の有効性を十分反映した手法になっていることがわかる．一方，低品質検索要求では，オリジナルのクラスタリングと同じかやや上回る結果を出す．つまり，検索結果をクラスタリングすることのそもそもの動機となったクラスタ仮説の有効性も十分反映した手法になっていることがわかる．また，図 10から，二つのクラスタリング法はほぼ同じ数の適合文書をフィードバックしていることがわかる．即ち，検索要求でバイアスされたクラスタリングが優れている理由は，フィードバックする適合文書の数が多いからではなく，検索要求に強く関連した文書をより多くフィードバックするからである．加えて，クラスタ仮説による優位性も保たれている．

1.6 おわりに

フィードバックする適合性判定数が少ないという状況での適合性フィードバックにおいて，増進的適合性フィードバックと検索結果自動分類の効果を比較した．TREC コレクションを用いた実験結果から，増進的適合性フィードバックは総合的な精度向上には寄与しないことがわかった．クラスタリングによる検索結果の自動分類は有効であることが実証されたが，検索要求によっては自動分類の弊害も見られた．この問題を解決するために，検索要求のバイアスを考慮したクラスタリングアルゴリズムを提案し，その有効性を示した．

増進的適合性フィードバックの結果は否定的であったが，情報要求のサブトピックに効率良くフォーカスしていくためには有用かもしれない．本論文ではこの点に関して詳しく検討しなかったが，今後の課題として必要な事項である．なぜなら，適合性フィードバックは繰り返して適用するものという常識があるが，少なくとも，今回のようにフィードバックを小刻みに適用し，かつトップランクの文書しか見ないという方法では，総合的に良い結果を出すことができないからである．

検索結果の自動分類に関しては，実際のユーザが適合クラスタを容易に選ぶことができるかどうかを確かめる必要がある．Hearst と Pedersen による実験結果 [10] はこの仮説を部分的に支持しているが，本論文で提案したバイアス付きクラスタリングに関しては定かでない．著者の個人的な観察によれば，検索要求でバイアスされたクラスタリングは，通常のクラスタリングに比べ，適合文書と非適合文書をより明確に分けることができ，ユーザによるクラスタ選択も容易になると思われるが，この点についても実験を加える必要がある．

最後に，バイアス付きのクラスタリングは様々な検索環境に適用できる．例えば，Scatter/Gather のような検索絞り込みや，検索結果の自動要約において有用である．通常のクラスタリングとは異なり，同じ文書集合からでもバイアスに応じて異なる分類が可能になるため，ユーザの情報要求をバイアスとすることで，よりユーザに適応した自動分類法が実現できる．

2 検索結果のクラスタリングを用いた直観的インタラクションモデル

3 はじめに

対話的な文書検索の多くは，関連度フィードバック (relevance feedback) という手法を用いて，ユーザとシステムが対話的に情報を交換しながら検索結果の向上を試みる．具体的には，検索結果の幾つかの文書に対してユーザが適合/不適合性の判定を行い，これらの判定を使ってシステムは検索要求を更新し新たな検索を行う．

関連度フィードバックによる検索精度向上の割合は，ユーザがシステムに与えた判定の数に大きく依存する [6]．我々は，ユーザが効率良くできるだけ多くの適合文書を選べるようなインターフェイスについて研究してきた [15, 11]．本論文では，文書クラスタリングを利用して検索結果を自動分類表示することの有効性を調べる．実際には「カテゴリーバー表示」と「デンドログラム表示」の二つの表示法について評価した．

カテゴリーバー表示では，クラスタリングアルゴリズムを用いて検索結果から3個の主カテゴリを見つけ，各文書とそれら主カテゴリとの距離をカラーバー表示する．ユーザは各々のカテゴリに注目して検索結果を並べかえることもできる．この表示法は，Scatter/Gather [7] で提案されている表示法や，Evans等によって用いられた表示法 [9] と似たものでり，適切な分類に注目することで多くの適合文書を効率良く集めることができる．

デンドログラム表示では，階層的クラスタリングの結果をそのまま表示する．類似している文書対はなるべく近い場所に表示されるため，ある適合文書を種にして，その文書に類似する文書群も芋づる式に見つけることができる．

本研究では，NTCIR-2 [1] での実験を介して，上記二つの表示法を評価した．実験では実際にユーザとシステム間の対話を記録しているため，再現率/精度だけではなくユーザの行動も考慮した評価を行った．

3.1 検索結果の表示法

本研究で評価した検索結果の表示法を説明する．

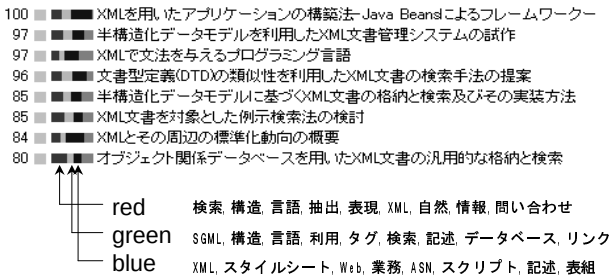
3.1.1 ランキング表示

検索要求との適合度が高い順に検索結果をならべて表示する．多くの検索システムで用いられているため，本研究でも提案手法のベースラインとしてランキング表示を用いる．ユーザは以下の操作を行うことができる．

AVISIT	指定した文書のフルテキストを表示する
ASEL	指定した文書 (複数可) に適合マークを付ける
AUNSEL	適合マークをはずす

3.1.2 カテゴリーバー表示

検索結果の各文書には，カテゴリーへの所属の度合を示すカテゴーバーが付いている．文書の初期並びはランキング表 れは．検索トピック「XMLを用いた自



タイトルの横にあるRGBスペクトルがカテゴリーバーである．それぞれの色 (R:赤，G:緑，B:青) は，検索結果 (実験では上位150文書) を要約する3個の主カテゴリに相当する．システムは階層的クラスタリングアルゴリズム [13] を使って，検索結果の文書群を3個のクラスタに分割する．これら3個のクラスタを主カテゴリとみなす．次に，各文書と3個のクラスタ間の距離を計算し，正規化の後にRGBスペクトルに変換する．よって，各色が占める割合は，対応するカテゴリーへの所属度の強さの割合に対応している．ここで，ユーザは各カテゴリの代表語を見ることができる．

VCAT	選択したカテゴリの代表語を見る

あるカテゴリに興味を持った場合，ユーザはそのカテゴリに注目して現在の検索結果を並べかえることができる．

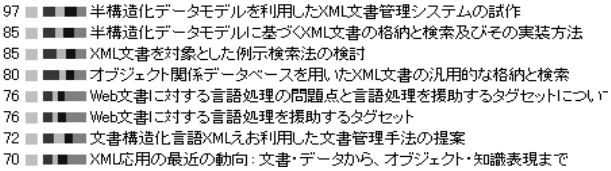
GCAT　選択したカテゴリに注目して 検索結果を並べかえる

この並べかえにより，注目しているカテゴリを代表する文書が上位に集まる．現在並べかえの方法として，

1. 注目カテゴリへの所属度が他カテゴリへの所属度よりも大きい文書のみを集める．ただし順位は検索要求との適合度の順である．

2. 単純に注目カテゴリの色の長さでソートする．

の2種類が選択可能である．以下は，上記の例に対して赤カテゴリに注目して並べかえを行った結果である．並べかえの方法は

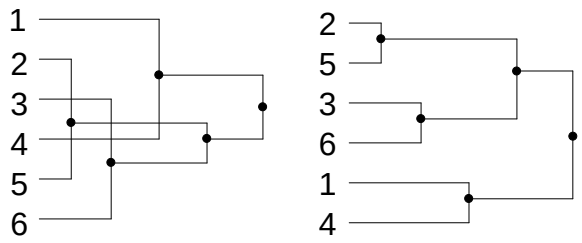


このように，カテゴリを介したインタラクションにより，ユーザは効果的に検索結果を絞りこむことができる．

なお，カテゴリーバー表示では，この他にも前述のAVISIT, ASEL, AUNSELコマンドが利用できる．

3.1.3　デンドログラム表示

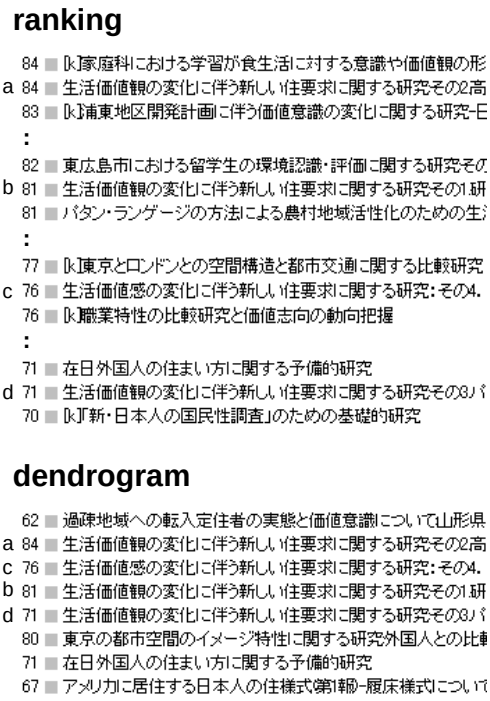
デンドログラム表示では，階層的クラスタリングアルゴリズムの適用結果をそのまま表示する．階層的クラスタリングアルゴリズムでは，まずクラスタリングの対象文書それぞれを別々のクラスタとして設定する．次に一番近いクラスタ対をマージする．このマージを繰りかえすと最終的には以下のような木ができあがる．この木(



上図で左側のデンドログラムは文書の順序を整列しないで描いた木で，ここでは多数の枝が交差していて文書間の類似性が見にくい．右側の図は交差をほぐして文書を並べかえたデンドログラムである．類似する文書はできるだけ近くに配置されている．

本研究では，デンドログラムの木構造は表示せず，並べかえ後のデンドログラム(上図右)における文書の順序のみをユーザに提示する．とはいっても，この順序そのものは有用であり多くの情報を含んでいる．例えば，あるユーザが文書3を適合文書として選んだ場合，すぐ隣りの文書6もおそらく適合文書である．なぜなら文書3と文書6は類似しているからである．このようにして，種文書の近くにある文書を見ることで，種文書に類似する多くの文書を見つけることができる．

似ている文書が近くに配置されることでタイトル間の類似/差異といった一覧性も良くなり，ユーザはタイトルを見るだけでこれらの文書の関係をとらえやすくなる．例えば以下の例を見てみる．これは検索トピック「日本人の生活価値観の変化」に対する検索結果の一部分である．



今，文書 a, b, c, d に注目する．タイトルを見ればわかるように，これらは同じ著者による一連の論文である．ランキング表示では，適合度の値がばらついているため，これらの文書群が離れた位置に表示されているのに対し，デンドログラム表示では，まとまって近くに表示されている点に注目してほしい．よって，デンドログラム表示では文書 a, b, c, d がシリーズを成していることは一目瞭然である．

デンドログラム表示では，AVISIT, ASEL, AUNSEL コマンドが使用可能である．

3.2 実験環境

NTCIR-2 [1] の日本語検索に参加して，クラスタリングに基づく分類表示の評価を行った．以下は，NTCIR-2 に提出した結果の作成手順である．

1. システムは各トピックから初期検索要求を作る．具体的には，各トピックの <DESCRIPTION> と <NARRATIVE> フィールドからストップワードを除く全ての単語を抽出して検索タームとした．今回は <CONCEPT> フィールドは使わなかった．形態素解析プログラムには ANIMA [16] を，単語の重み付け法には Lt.Lnc 法 [19] を用いた．よって，検索システムはベクトル空間モデルに基づいていることになる．

2. システムは初期検索要求から 150 文書を検索し，以下のいずれかの表示法で各被験者に表示する．

   - ランキング表示 (3.1.1 節参照)
   - カテゴリーバー表示 (3.1.2 節参照)
   - デンドログラム表示 (3.1.3 節参照)

3. 各被験者は提示された検索結果から 15 分以内にできるだけ多くの適合文書をマークする．この間になされた操作は全て時刻付きで記録する．

4. 被験者がマークした適合文書をシステムにフィードバックして，システムは初期検索要求を更新する．具体的には，適合文書から上位 300 タームをフィードバックして改良 Rocchio 法により検索要求を更新した．改良 Rocchio 法のパラメータは $\alpha = 8, \beta = 16, \gamma = 0$ とした．つまり，負のフィードバックは行わなかった．

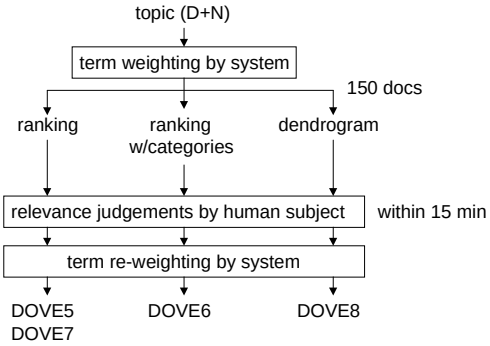


図 12: 対話的run の概要

ID	other info	平均精度	
		S+A	S+A+B
DOVE5	ランキング表示(DOVE6 のベースライン)	0.4095	0.4020
DOVE6	カテゴリーバー表示	0.3996	0.3943
DOVE7	ランキング表示(DIVE8 のベースライン)	0.4052	0.3976
DOVE8	デンドログラム表示	0.4069	0.3891

表 1: 平均精度(average precision)

5. 更新した検索要求に基づいてシステムは再検索を行う．検索結果の 1,000 文書を評価対象として提出した．

図 12に概略を示す．

7 人の被験者 (著者ら) が実験に参加した．

実験目的は，ベースラインのランキング表示とクラスタリングに基づく二つの表示法とを比較することである．よって，各検索トピックでの一貫性を保つために，同じ被験者には二つの表示法について実験してもらった．一つはベースラインのランキング表示で，もう一つはカテゴリーバー表示，デンドログラム表示のいずれかである．ただし，二つの試行を行う順番が問題であるため，どちらを先に行なうかはランダムに決めた．更に，念のため，二つの試行の間には最短でも一週間の間隔をおいてもらった．まとめると，NTCIR-2 には以下の 4 つの対話的run を提出した．

DOVE5	ランキング表示 (DOVE6 のベースライン)
DOVE6	カテゴリーバー表示
DOVE7	ランキング表示 (DOVE8 のベースライン)
DOVE8	デンドログラム表示

3.3 実験結果と考察

3.3.1 総合結果

表 1に平均精度(average precision) を示す．ここで，S, A, B は適合性のレベルで，S は「特に適合」，A は「適合」，B は「部分的に適合」を意味する．

図からもわかるように，残念ながら，カテゴリーバー表示(DOVE6)，デンンドログラム表示(DOVE8) 共に，ベースラインのランキング表示を有為に上回ることができなかった．デンドログラム表示(DOVE8) は，かろうじてベースラインを上回ったがその差は小さい．

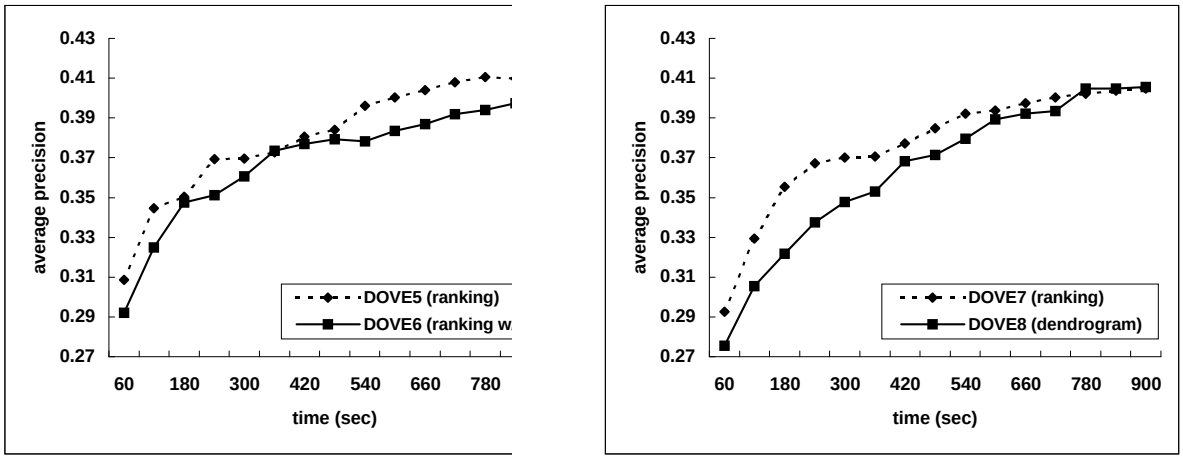


図 13: 平均精度の時間推移

ID	ASEL	S	A	B	C	(S+A)/ASEL query-averaged	(S+A+B)/ASEL query-averaged
DOVE5 (ランキング)	688	120	383	62	122	0.7264	0.8371
DOVE6 (カテゴリーバー)	598	94	325	50	129	0.7336	0.8202
DOVE7 (ランキング)	671	99	387	66	116	0.7201	0.8372
DOVE8 (デンドログラム)	543	87	319	40	94	0.7496	0.8368

表 2: 被験者が行った適合性判定の精度

図 13は，平均精度の時間推移である．各時刻までにマークされた適合文書をフィードバックして得た平均精度をプロットしてある．ここでも，カテゴリーバー表示 (DOVE6) とデンドログラム表示 (DOVE8) の優位性は見てとれない．デンドログラム表示 (DOVE8) は，15 分近くになってやっとベースラインに追い付いているが，ほとんどの時刻においてベースラインを下回っている．

3.3.2 適合判定について

表 2に，被験者が行った適合性判定のどれだけが正規の判定と一致したかを示す．つまり被験者が行った判定の精度である．

全 run において，精度は 70%を上回っている．よって，被験者の判定はそれほど間違っていなかったことがわかる．また，S ランクと A ランクの正解文書に関しては，ランキング表示を用いるよりも，カテゴリーバー表示やデンドログラム表示を用いるほうが精度が高いこともわかる．ただし，差は大きくない．B ランクの正解文書も含めるとカテゴリーバ表示，デンドログラム表示共にベースラインに劣る．

一方，表 3は，被験者による判定の再現率である．つまり，提示された 150 文書に含まれている正解文書のうち，どれだけを実際に見つけることができたかである．

図 3からもわかるように，再現率は押しなべて低い．全てにおいて 45%を割り込んでいる．よって被験者は多くの正解文書を判定し逃していることになる．現在原因を調査中であるが，多くの場合は検索トピックの解釈の相違のようである．もし判定の精度/再現率が 100%ならば，検索の平均精度は S+A 判定で 0.5166，S+A+B 判定で 0.4775 に達っし，これらの値は本実験での上限に相当する．

また，いずれの run においても B 判定の再現率が低いのは，実験のインストラクションで「S および A 判定に相当する適合文書を探す」ことを指示したためであろう．

ID	S	A	B	S+A	S+A+B
DOVE5 (ランキング)	0.2915	0.4150	0.1287	0.4483	0.4048
DOVE6 (カテゴリーバー)	0.2768	0.3591	0.1687	0.4092	0.3807
DOVE7 (ランキング)	0.2617	0.4142	0.1420	0.4429	0.3912
DOVE8 (デンドログラム)	0.2798	0.3761	0.1254	0.4090	0.3627

表 3: 被験者が行った適合性判定の再現率

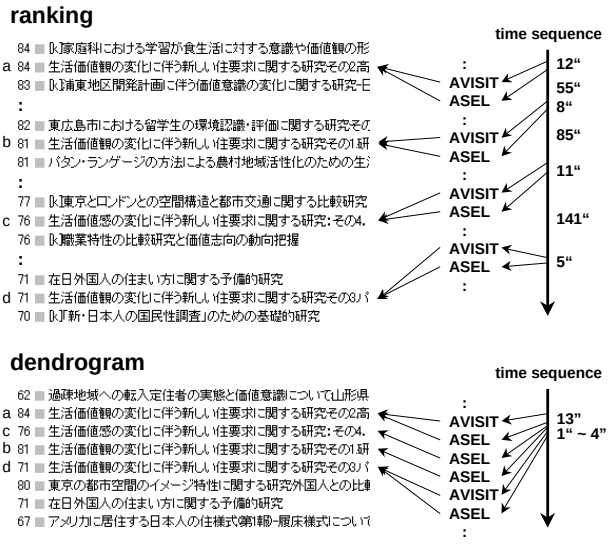


図 14: 被験者とシステムとの対話例

3.3.3 対話ログの解析

クラスタリングを用いた自動分類表示の利点は，お互いに類似している文書を簡単に見つけることができる点である．例えばカテゴリーバー表示では，あるカテゴリーに着目して検索結果を並べかえることで，そのカテゴリーという観点で検索結果が整理できる．デンドログラム表示では，近い文書対は近く配置されるため類似する文書が一覧できる．本節では，自動分類表示の利点を調べるために，実際に被験者が行った操作のログを解析する．

図 14は，検索トピック「日本人の生活価値観の変化」に関する対話ログの一部である．ここでも，同じ著者による一連の論文 a, b, c, d に注目する．これらの論文は様々な適合性のスコアを持つため，ランキング表示では離れて表示されてしまう．かつ，ユーザは通常，上位から下位に向って文書を調べていくため，これら一連の論文を見る間に多くの無関係な論文も目にはいってしまい，a, b, c, d 間のまとまりを識別することが困難である．対話ログを見てみると，被験者は a, b, c, d の順に見ているのだが，その間隔は，55 秒，85 秒，141 秒と比較的長い．よって，前に見た文書の内容を忘れ，a, b, c, d 全てにおいてフルテキストを参照している．参照時間も比較的長い．

一方，デンドログラム表示では，a, b, c, d がまとまって表示されるため，被験者もまとまりとしてこれらの文書群が認識できる．対話ログを見ると，最上位の a についてはフルテキストの参照を行っているが，次の c, b については，タイトルを見ただけで適合性の判定を行っている．かつその間隔は非常に短い．最後の文書 d については，フルテキストを参照しているが，参照時間は 4 秒と短いため確認のための参照と言える．

表 4に ASEL コマンドの連続数を示す．ASEL コマンドは適合性マークを付けるコマンドであるため，こ

ID	
DOVE5 (ランキング)	65
DOVE6 (カテゴリーバー)	57
DOVE7 (ランキング)	50
DOVE8 (デンドログラム)	135

表 4: ASEL コマンドの連続数

ID	ASEL	ASEL without AVISIT		
		total	S+A	S+A+B
DOVE5 (ランキング)	688	76 (0.1105)	62	66
DOVE6 (カテゴリーバー)	598	65 (0.1087)	56	58
DOVE7 (ランキング)	671	51 (0.0760)	38	40
DOVE8 (デンドログラム)	543	107 (0.1971)	70	79

表 5: フルテキストの参照なしに適合と判断できた文書数

の連続が意味するところは，後ろの文書に対してはフルテキストの参照なしに適合と判断した可能性が高いということである．表 5には，実際にフルテキストの参照なしに適合と判定できた文書数を示す．いずれの表でも，デンドログラム表示では，これらの値が大きい．ベースラインのランキング表示と比べると2倍程度である．一方，カテゴリーバー表示はここでもベースラインを上回ることができなかった．

最後に図 15に，ASEL/AVISIT の割合を時間経緯で示す．ASEL/AVISIT の割合は，被験者が行った操作の効率性 (どれだけ無駄なフルテキスト参照がなかったか) を計っている．カテゴリーバー表示は，ランキング表示とほとんど同じ曲線を描いているのに対し，デンドログラム表示は，ランキング表示とは傾向が異っている．約100秒を過ぎると，ベースラインが次第に下っていくのに対し，デンドログラム表示では依然として約50%の割合を保っている．つまり，ランキング表示においては時間が増すにつれ無関係な文書を読む割合が増えるのに対し，デンドログラム表示では，セッションの後半になってもあまり多くの不適合文書を読むことがない．

3.4 おわりに

適合性フィードバックを効果的に行うための検索結果表示法として，二つの自動分類表示「カテゴリーバー表示」「デンドログラム表示」を提案し評価した．平均精度の観点からは両者の効果は認められなかったが，ユーザとの対話ログを調べた結果，特にデンドログラム表示の有効性が確認できた．検索結果をデンドログラム表示することにより，ユーザは類似する適合文書を効果的に集めることができた．カテゴリーバー表示に関しては，いずれの評価においても通常のランキング表示を上回ることがなかった．一つの原因として，インターフェイスが未熟でユーザが操作にとまどっていることが挙げられる．例えば，あるカテゴリーに注目した並べかえについて幾つかの手法をユーザに選ばせているが，明らかに複雑でわかりにくいインターフェイスである．今後は，カテゴリーバー表示のインターフェースを洗練化する予定である．

参考文献

[1] *NTCIR Workshop 2: Proceedings of the Second NTCIR Workshop Meeting on Evaluation of Chinese & Japanese Text Retrieval and Text Summarization*, 2001.

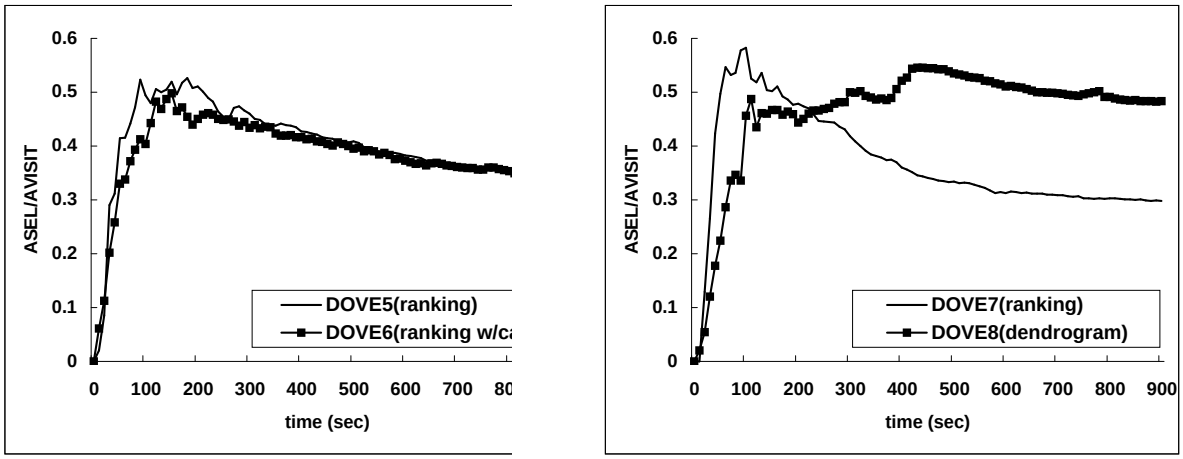


図 15: ASEL/AVISITの時間推移

[2] I. J Aalbersberg. Incremental relevance feedback. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 11–22, 1992.

[3] J. Allan. Incremental relevance feedback for information filtering. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 270–278, 1996.

[4] C. Buckley, M. Mitra, J. Walz, and C. Cardie. Using clustering and SuperConcepts within SMART: TREC 6. In *Proceedings of the Sixth Text REtrieval Conference (TREC-6)*, 1998.

[5] C. Buckley and G. Salton. Optimization of relevance feedback weights. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 351–357, 1995.

[6] C. Buckley, G. Salton, and J. Allan. The effect of adding relevance information in a relevance feedback environment. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 292–300, 1994.

[7] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey. Scatter/Gather: A cluster-based approach to browsing large document collections. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 318–329, 1992.

[8] D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun. A practical part-of-speech tagger. In *Proc. of the Third Conference on Applied Natural Language Processing*, 1992.

[9] D. A. Evans, A. Huettner, Tong X., P. Jansen, and J. Bennett. Effectiveness of clustering in ad-hoc retrieval. In *Proceedings of the Seventh Text REtrieval Conference (TREC-7)*, 1999.

[10] M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1996.

[11] M. Iwayama. Relevance feedback with a small number of relevance judgements: Incremental relevance feedvack vs. document clustering. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 10–16, 2000.

[12] M. Iwayama and T. Tokunaga. Cluster-based text categorization: A comparison of category search strategies. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 273–280, 1995.

[13] M. Iwayama and T. Tokunaga. Hierarchical bayesian clustering for automatic text classification. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 1322–1327, 1995.

[14] D. D. Lewis and W. A. Gale. A sequential algorithm for training text classifiers. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 3–12, 1994.

[15] Y. Niwa, M. Iwayama, T. Hisamitsu, S. Nishioka, A. Takano, H. Sakurai, and O. Imaichi. Interactive document search with DualNAVI. In *Proceedings of the First NTCIR Workshop on Research in Japanese Text Retrieval and Term Recognition*, pp. 123–130, 1999.

[16] H. Sakurai and T. Hisamitsu. A data structure for fast lookup of grammatically connectable word pairs in japanese morphological analysis. In *International Conference on Computer Processing of Oriental Languages (ICCPOL'99)*, pp. 467–471, 1999.

[17] R. E. Schapire, Y. Singer, and A. Singhal. Boosting and rocchio applied to text filtering. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 215–223, 1998.

[18] H. Schütze and C. Silverstein. Projections for efficient document clustering. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1997.

[19] A. Singhal, C. Buckley, and M. Mitra. Pivoted document length normalization. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 21–29, 1996.

[20] A. Singhal, M. Mitra, and C. Buckley. Learning routing queries in a query zone. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 25–32, 1997.

[21] C. J. van Rijsbergen. *Information Retrieval*. Butterworths, London, 1979.