

認識誤りを含むテキストの検索手法に関する研究

太 田 学



学位請求論文

認識誤りを含むテキストの
検索手法に関する研究

A Study on Retrieval Methods for Text
with Misrecognized Characters

指導教官 安達 淳 教授

1998年 12月 18日 提出

東京大学大学院工学系研究科
電気工学専攻 67109

太田 学

目次

1 はじめに	1
1.1 研究の背景	2
1.1.1 文書を扱う電子図書館の現状	2
1.1.2 本研究で想定する電子図書館像	2
1.2 研究の目的と意義	3
1.3 研究の概要	4
1.4 本論文の構成	6
2 関連研究	8
2.1 文書認識	9
2.2 文字認識	10
2.2.1 OCR	10
2.2.2 活字OCRの認識誤り	10
2.2.3 認識誤りの分類実験	11
2.2.4 文字認識後処理	13
2.3 文書認識と情報検索	15
2.3.1 検索対象テキストの生成	15
2.3.2 OCR認識誤りの検索への影響	15
2.4 曖昧検索	16
2.4.1 類似文字テーブル (Confusion Matrix/Table)	17
2.4.2 複数認識候補型検索手法	17
2.4.3 部分文字列照合	18
3 提案する曖昧検索手法	22
3.1 提案手法の概要	23
3.2 認識誤りの抽出と分類	24
3.2.1 認識誤りの抽出	24
3.2.2 認識誤りの分類	25

3.3 得点付けのアルゴリズム	26
3.3.1 文字の確信度	26
3.3.2 類似文字テーブル	26
3.3.3 拡張類似文字テーブル	27
3.3.4 bigram 統計に基づいた文字の接続確率	27
3.3.5 BMR 法における文字の確信度	28
3.3.6 文字列の確信度	29
3.4 まとめ	31
4 曖昧検索手法の評価	33
4.1 実験の概要	34
4.1.1 検索対象テキスト	34
4.1.2 実験環境	34
4.2 実験結果と考察	36
4.2.1 確信度の閾値と検索効率	36
4.2.2 最適閾値における検索効率	36
4.3 まとめ	40
5 英文曖昧検索	41
5.1 英文曖昧検索における課題	42
5.2 英文曖昧検索における前処理	42
5.2.1 低適合率とその原因	42
5.2.2 前処理の導入	43
5.3 複合誤りの分割と曖昧検索への活用	44
5.3.1 複合誤りの分割方法	44
5.3.2 複合誤りの分割実験	45
5.3.3 検索効率への寄与	46
5.4 提案手法と拡張検索文字列	46
5.4.1 拡張検索文字列数	46
5.4.2 確信度に基づくランキング検索	47
5.5 拡張検索文字列削減のためのヒューリスティクス	48
5.5.1 提案する2つのヒューリスティクス	48
5.5.2 ヒューリスティクス1の適用結果	49
5.5.3 ヒューリスティクス2の適用結果	50
5.5.4 考察	51
5.6 逆確信度に基づくランキング検索	53

5.6.1	文字の逆確信度	53
5.6.2	文字列の逆確信度	54
5.6.3	本手法の検索性能	56
5.7	まとめ	57
6	HMM に基づいた英文曖昧検索手法	58
6.1	本手法の狙い	59
6.2	HMM の定式化	59
6.3	HMM に基づいた曖昧検索手法の概要	60
6.3.1	作成する HMM	60
6.3.2	得点付けのアルゴリズム	63
6.4	英文曖昧検索実験	67
6.5	考察	68
7	考察	71
7.1	和文曖昧検索	72
7.1.1	検索性能	72
7.1.2	複数認識候補型検索手法との比較	73
7.2	英文曖昧検索	74
7.2.1	課題と提案手法	74
7.2.2	検索性能	75
7.2.3	他の英文曖昧検索手法との比較	76
7.3	確信度及び逆確信度と確率	77
7.3.1	CMR 法の用いる確信度と確率	78
7.3.2	ECMR 法の用いる確信度と確率	78
7.3.3	ECMR 法の用いる逆確信度と確率	81
7.3.4	BMR 法の用いる確信度と確率	81
7.3.5	確信度及び逆確信度とは何か	82
7.4	確信度及び逆確信度と検索効率	82
7.4.1	文字列の出現確率	82
7.4.2	拡張検索文字列数と検索効率	83
7.4.3	ランキングの意味	85
8	おわりに	86
8.1	本論文のまとめ	87
8.1.1	曖昧検索手法の提案とその背景	87
8.1.2	本研究のまとめ	87

8.1.3 本研究の成果	88
8.2 今後の展望	88
謝辞	90
参考文献	91
発表文献	95
付録 A HMM に基づいた英文曖昧検索手法の生成する拡張検索文字列	97

目 次

1.1	ハイブリッド型電子図書館の構成	3
1.2	本研究の進行過程と本論文内での位置付け	7
2.1	文書のもつレイアウト構造の例	9
2.2	文書のもつ論理構造の例	10
2.3	OCRによる文字認識処理	11
2.4	複数認識候補型検索手法	18
2.5	エディット距離の計算例	20
3.1	3つの提案手法	24
3.2	拡張検索文字列の生成(英文ECMR法の例)	25
3.3	認識誤りの抽出方法	25
3.4	認識誤りの抽出例	26
3.5	類似文字テーブル	27
3.6	(a) 欠落文字テーブルと(b) 挿入文字テーブル	28
3.7	(a) 結合文字テーブルと(b) 分解文字テーブル	29
3.8	bigram テーブル	30
3.9	ECMR法における文字列の確信度の計算(和文の例)	31
4.1	P_E の閾値と検索効率(和文)	37
4.2	P_E の閾値と検索効率(英文)	38
5.1	英文における P_E の閾値と検索効率(前処理なし)	43
5.2	複合誤り情報の検索効率への寄与(英文トレーニングセット)	47
5.3	複合誤り情報の検索効率への寄与(英文テストセット)	48
5.4	英文における P_E の閾値と拡張検索文字列数(ECMR法)	49
5.5	英文ECMR法の検索効率と拡張検索文字列数(確信度に基づくランキング)	50
5.6	英文トレーニングセットにおけるヒューリスティクス1の効果(ECMR法)	51
5.7	英文テストセットにおけるヒューリスティクス1の効果(ECMR法)	52

5.8 英文トレーニングセットにおけるヒューリスティクス2の効果 (ECMR 法)	53
5.9 英文テストセットにおけるヒューリスティクス2の効果 (ECMR 法)	54
5.10 英文 ECMR 法の検索効率と拡張検索文字列数 (逆確信度に基づくランキング) . . .	56
6.1 簡単な HMM の例	60
6.2 作成する HMM の例	61
6.3 認識動作の HMM における表現	62
6.4 HMM 作成のためのテキスト	62
6.5 検索語 "task" のとりうる状態系列と出力シンボル	64
6.6 異なる状態系列が同じシンボル系列を出力する例	65
6.7 拡張検索文字列の一致する例 (状態系列及びシンボル系列が共に異なる)	66
6.8 拡張検索文字列の一致する例 (状態系列は等しくシンボル系列が異なる)	66
6.9 英文における検索効率と拡張検索文字列数	68
7.1 ECMR 法における誤りの種類の曖昧性 ($m = n = 4$ の場合)	79

目 次

2.1 英語活字 OCR の認識誤りの構成 (Times font)	11
2.2 日本語活字 OCR の認識誤り	13
2.3 英語活字 OCR の認識誤り	13
2.4 類似文字テーブルの例	17
2.5 様々な検索語長のメンバシップ関数の値	21
4.1 類似文字テーブル及び拡張類似文字テーブルのファイルサイズ (単位:Byte)	35
4.2 類似文字テーブル及び拡張類似文字テーブルの内容の具体例	36
4.3 和文トレーニングセットにおける検索効率	37
4.4 和文テストセットにおける検索効率	38
4.5 英文トレーニングセットにおける検索効率	39
4.6 英文テストセットにおける検索効率	39
5.1 英文トレーニングセットにおける検索効率 (前処理なし)	43
5.2 英文テストセットにおける検索効率 (前処理なし)	44
5.3 抽出された認識誤り 1	45
5.4 抽出された認識誤り 2	46
5.5 ヒューリスティクス 1 の検索性能	51
5.6 ヒューリスティクス 2 の検索性能	52
5.7 検索効率と拡張検索文字列数の最適値	57
6.1 検索効率と拡張検索文字列数の最適値候補 (英文トレーニングセット)	68
7.1 和文曖昧検索で用いた検索語に関するデータ	73
7.2 和文 CMR 法における拡張検索文字列数	73
7.3 検索効率と拡張検索文字列数 (英文トレーニングセット)	75
7.4 検索効率と拡張検索文字列数 (英文テストセット)	75
7.5 検索効率の比較 (英文)	77
7.6 文字の出現頻度	80

第1章

はじめに

本研究の背景、目的、概要について述べるとともに、本論文の構成を示す。

1.1 研究の背景

電子的に情報を蓄積し、利用者の閲覧や検索要求に応じて効率良く所望の情報を提供する電子図書館の研究は、90年代に入ってから加速度的に進み、世界各地で様々なコンセプトをもつプロジェクトが立ち上がっている。中でも文書を蓄積対象とする電子図書館 [1] では、主としてデータ入力コストの観点からの現実的なアプローチとして、大量の印刷文書を画像で入力し OCR (Optical Character Recognition) を使って全文データベースを構築する試みが最近行なわれるようになってきた。このとき OCR の出力するテキスト情報は、従来はその認識誤りを訂正してから電子図書館のもつデータベースに格納していたが、蓄えられるデータの多様化、大量化に伴って書誌情報など一部を除いては OCR 認識誤りの訂正は行なわずそのまま格納されるようになってきた。そのため、OCR 認識誤りのために検索時に見込まれる検索洩れなどに対処できる検索手法が強く求められるようになってきている。

1.1.1 文書を扱う電子図書館の現状

学術論文などの文書の提供を目的とした電子図書館の実現形態には大きく分けて、スキャナ等で読み込んだ文書イメージをそのまま蓄積して文書画像でみせるものと、全文を SGML (Standard Generalized Markup Language) や HTML (HyperText Markup Language) などマークアップ言語を用いて作成し WWW (World Wide Web) ブラウザなどでみせるものの2つがある。

- 文書画像を蓄積主体とするもの

閲覧用に誌面をスキャンしたページイメージ (文書画像) を蓄え、検索用に全文または抄録などを OCR 認識して得た誤りを含むテキストデータや人手で作成した2次情報などを用いている。

- 全文をテキストで蓄えるもの

ページイメージは雑誌の表紙程度にとどめ、蓄積主体の論文などは SGML 化や HTML 化した全文テキストとする。この場合1次/2次情報を問わず、誤りの極めて少ないテキストデータが得られるので、閲覧、検索双方にこの全文データを用いる。

これらの文書画像と全文テキストを用いる方法の優劣は常に話題となる。前者には、過去の膨大な資料の遡及入力が容易であること、印刷文書のもつレイアウト情報の保持、多言語への対応が容易であるなど文書の再現性が高いことなどの利点がある。一方後者には、閲覧と検索が1つの全文データで対応できること、蓄積・送信するデータ量が少なくすむなどの利点がある。

1.1.2 本研究で想定する電子図書館像

電子図書館における文書のデータ形式には大きく分けて、文書画像と SGML などのマークアップ言語による全文データの2つがあり、それぞれに利害得失があることは既に述べた。将来的に

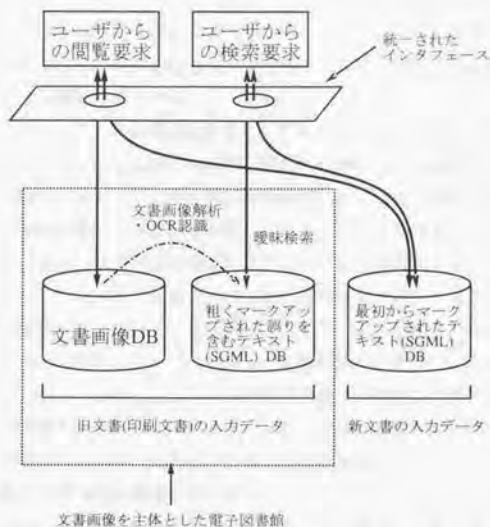


図 1.1: ハイブリッド型電子図書館の構成

は、執筆から流通、閲覧に至るまで一貫して電子的に文書データが作成されるようになるであろうが、現在は最終形態が印刷物である場合も少なくなく、過去の膨大な印刷文書も無視できない。よって理想的な電子図書館としてはどちらのタイプの文書も扱え、ユーザに蓄積された文書のデータ形式の違いを意識させないハイブリッド型のものが望まれる(図 1.1 参照)。

本研究では図 1.1 のような構成をもつ電子図書館の中で、特に閲覧に文書の再現性の高い文書画像を用い、検索に文書認識によって得た不完全なテキスト情報を活用するような、文書画像を蓄積主体とした電子図書館を想定する。

1.2 研究の目的と意義

前節で述べたような文書画像を主に扱う電子図書館を実現するには、

- 文書画像解析及び OCR 文字認識による検索データの自動作成・入力
- 誤りを含むテキストに対する検索手法

という 2 つの課題解決が必要で、前者はデータ入力時の後者はデータ検索時の課題である。このうち前者の課題を解決するための文書画像解析や OCR 文字認識などの文書認識の研究は、第 2

章で述べるように成熟期に入ってきており、現実的な誤り率を実現しつつある。そこで本研究では文書画像を主に扱う電子図書館における後者の課題、すなわち OCR 認識によって得た不完全なテキストに対する検索手法の提案を行なう。また本論文ではこのような認識誤りを含むテキストに対する検索のことを曖昧検索と呼ぶ。

近年の情報検索においては、書誌情報を対象とした検索から全文(フルテキスト)を対象とした検索に移行してきており、さらに文書の全文から得られる単語の出現頻度などを利用したより高度な検索手法が種々提案されている。また電子図書館においても全文情報を蓄えるものが増加しつつあるが、既存の印刷文書から誤りのないテキストデータを作成するコストは決して安くはない[2]。特に OCR 認識後の誤り訂正の精度を上げるためには良質かつ多大な人的資源を必要とし、このため文書 1 ページ当たりの認識コストは数ドルから数十ドルとなっており、これがデータ遡及入力最大の障壁となっている。そのため、人手で入力する誤りの極めて少ないテキスト情報は書誌情報にとどめ、全文については OCR の認識結果を訂正せずにそのまま蓄え認識誤りについては無視しているものもある。しかし本研究で提案する曖昧検索を採用すれば検索時に認識誤りを考慮した検索語拡張を行なうため、データ入力時のこの膨大な人的コストをほぼ無くすることができ、電子図書館のコンテンツ作成の大幅な促進が期待される。

曖昧検索は認識誤りを検索時に考慮するため OCR の出力する生のテキストをデータベースに蓄えることができるが、本提案手法は誤りのないテキスト情報が必要な場合は認識誤りの訂正に利用することもできる。これは提案手法が OCR の認識誤りに関する確率を定義しているためで、例えばこの確率を手掛かりに誤認識の訂正候補文字を求めることができる。また本研究では、曖昧検索する対象を OCR 認識によって得られたテキストとしているが、例えば音声認識の結果得られたテキストを検索対象とすることも原理的に可能である。これは提案手法が、認識装置が何であれそれをブラックボックスとしてとらえ、元のシンボル系列とそれに対応する認識結果のシンボル系列のみに着目して、認識装置の認識特性を獲得するからである。但し現時点では、OCR による文字認識ほどの性能は音声認識では期待できないため、音声認識結果に対する適用では文字認識に比べて低認識率、また文字認識とは異なる種類の認識誤りの定義など解決すべき課題も多いと考えられる。

1.3 研究の概要

本研究で提案する曖昧検索手法は、OCR の認識誤り特性に基づいて検索語を複数の妥当な検索文字列に拡張し、その拡張文字列を用いて認識誤りを含むテキストを検索することで検索ノイズを抑えながら検索洩れを救済する手法である。

具体的には予め以下のような 2 種類のテーブルを作成しておく。

- 認識誤りを置換・欠落・挿入・結合・分解誤りの 5 種類に分類し、トレーニングセットに現れるそれぞれの誤りの頻度統計を元に誤り易さを表す確率を求める。この確率を誤る可能性のある文字とともにテーブル(類似文字テーブル及び拡張類似文字テーブル)に保持する。

- 統計的言語モデルとして文字の bigram モデルを採用し、これに基づいて文字の接続確率を求め、その情報をテーブル (bigram テーブル) に保持する。

そして検索時にはこれらのテーブルを参照しながら検索語を拡張し、提案する曖昧検索手法のアルゴリズムに基づいて、個々の拡張検索文字列の妥当性を表す得点 (確信度) をこれらの確率から計算する。このとき高い検索効率を実現し、かつ拡張検索文字列数が不用意に増加するのを防ぐために、個々の拡張検索文字列の確信度に基づいて拡張検索文字列数を決定する。また、検索語拡張の際に参照する類似文字テーブルなどのテーブルの種類の違い、換言すれば妥当な検索文字列を選択するアルゴリズムの違いに基づいて、本研究ではまず合計 3 つの曖昧全文検索手法を提案している。

本研究ではこれらの提案手法を用いて和英文双方に対する検索実験を行ない、検索効率及び検索コストに関する定量的な評価を行なった。その結果認識誤り特性の学習に用いなかった和文テストセットに対する検索実験で、通常の 10 倍程度の検索コストで約 96% であった再現率を 99% 以上に改善し、かつ 99% 以上の適合率を実現できることを示した。一方英文テストセットに対する検索実験では、和文に比べて適合率が低く、検索コストがかかりすぎることが判明した。この両方に共通する原因は、日本語とは異なる英語という言語のもつ特質にあり、具体的には和文に比べて英文の文字種が圧倒的に少ないという点にあった。すなわち英文の文字種が少ないため、拡張検索文字列が検索語とは異なる文字列に偶然一致することが多くなり低適合率を招いていた。一方検索コストの問題は拡張検索文字列数が検索語長の指数のオーダーで増加するという提案手法の本質的な問題であったが、これが英文曖昧検索においてのみ特に問題となったのは、1 文字当りの認識誤り候補が英文では和文に比べてかなり多かったためである。

そこでこのような英語の言語的な特徴を考慮し、特に英文に適した曖昧検索手法の提案を行なった。まず低適合率の問題を解決するため、大文字の小文字への変換 (正規化)、デリミタによる単語の切り出しという前処理を行ない、これにより再現率は維持したまま適合率を和文と同程度まで改善できることを示した。また、検索コストの問題、すなわち拡張検索文字列数が多くなり過ぎるという問題に対しては、検索効率を維持したまま以下の 2 つの方法を用いて削減を図った。

- 拡張検索文字列中の認識誤りの数を制限するというヒューリスティクスの導入
- 新たな得点計算アルゴリズムの提案

これらにより既に実現していた 99% 以上という検索効率を殆んど下げることなく、ヒューリスティクスの導入では拡張検索文字列数を 50 程度に、新たな得点計算アルゴリズムでは 20 以下にまで下げることができた。

英文検索ではまた、認識誤り情報と文字の連接情報を統合的に扱える HMM (Hidden Markov Model) に基づいた曖昧検索手法の提案も行なった。この手法は、文字の切り出し誤りに起因する認識誤り、具体的には欠落・挿入・結合・分解誤りに関する確率を、HMM に当てはめることで文字の接続確率と統合的に扱えるようにしたもので、これは当初提案した 3 つの手法では扱いきれ

なかった問題の1つの解法でもある。本手法はまた英文トレーニングセットにおいて、他のどの手法よりも優れた検索効率と検索コストを実現しており、一定の有効性を示した。しかし HMM パラメータの再推定などを行なえなかったため、テストセットにおいては検索効率の改善が殆んどみられず、これは課題として残った。このように本研究では、和英両言語の活字 OCR の出力テキストに対して現実的な検索コストで十分な検索効率を得られる曖昧検索手法を複数提案している。

1.4 本論文の構成

本論文の構成を以下に示す。

- 第2章

関連研究として本研究と関わりの深い文書認識及び文字認識技術について説明するとともに、本研究と同様の目的をもつ曖昧検索手法を紹介する。

- 第3章

本研究で提案する曖昧検索手法のアルゴリズムを示す。

- 第4章

第3章において提案した曖昧検索手法について、和英両テキストに対する性能を検索効率の点から評価する。

- 第5章

第4章での評価を踏まえ、第3章で提案した曖昧検索手法をそのまま英文に適用した場合の問題点を指摘する。また、その改善策を盛り込んだ英文曖昧検索に特化した処理及び検索アルゴリズムを提案し、その評価を行なう。

- 第6章

第3章で提案した曖昧検索手法では扱いきれなかった、認識誤りの起こり易さを表す確率と bigram に基づいた文字の接続確率を統合的に扱うために、HMM に基づいた曖昧検索手法を提案し、英文における検索性能の評価を行なう。

- 第7章

提案した和英両文に対する曖昧検索手法についてその成果についてまとめ、他の手法と比較する。また、拡張検索文字列の妥当性を表す尺度として導入した得点とベイズ確率との関係について、またそのベイズ確率と検索効率の関係について考察する。

- 第8章

本論文を総括し、今後の展望について述べる。



図 1.2: 本研究の進行過程と本論文内での位置付け

また本研究は、まず和文を対象とした曖昧検索手法を提案し [3]、つづいてそれを英文に適用して問題点を抽出し、その結果に基づいて言語的な性質の異なる英文に適した曖昧検索手法を提案する [4] という形で進化した。本研究の進行過程と本論文の内容との関係を図 1.2 に示しておく。

第2章

関連研究

本章では文書認識及び文字認識技術について説明するとともに、本研究と同様の目的をもつ曖昧検索手法を紹介する。

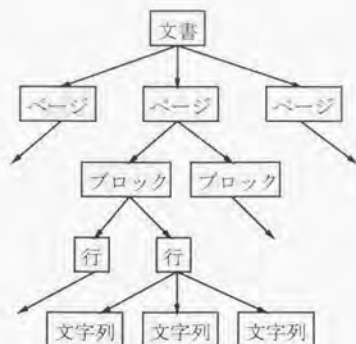


図 2.1: 文書のもつレイアウト構造の例

2.1 文書認識

文書認識 [5, 6, 7] とは印刷文書を電子化するための一連の処理を指し、通常以下のような手順で行なわれる。

1. スキャナを用いて印刷文書を文書画像に変換する。
2. 得られた文書画像を解析し、図表、テキストなどの文書の構成要素を抽出する。
3. 文書画像から抽出されたテキスト領域を OCR を用いて文字認識し、テキストコードを得る。

最初のスキャンニングでは、後の処理を考えて通常 300～400dpi(dots per inch) 程度の解像度で印刷文書を読み込む。またこのときの解像度や文書の傾きなどが、OCR による文字認識も含めた後の処理の精度に大きく影響することが知られている。

文書画像解析ではまず、ページのレイアウト解析によってページ画像をいくつかのブロックに分割し、そのブロックが文字を含むテキスト領域であるかそれとも図表を含む領域であるかなどを判別し、その結果を幾何学的な位置情報などとともに属性として各ブロックに付与して出力する。その結果得られるレイアウト構造は、例えば図 2.1 のようなものである [8]。図 2.1 は、文書が複数のページからなり、ページが 1 つ以上のブロックからなり、ブロックが行、行が文字列 (単語) からなることを示している。またさらに論理構造解析によって、文書の論理的な構成要素、例えばタイトル、著者名、節、段落など、とブロックとの対応関係を求める場合もある (第 2.3.1 節参照)。その結果得られる文書の論理構造は、例えば図 2.2 のようなものである [8]。

最終的には検索などに活用できるテキストコードを得るために OCR による文字認識を行なう。実用化されている OCR では、特にユーザがテキスト領域を文書画像上で指定しなくても、文書

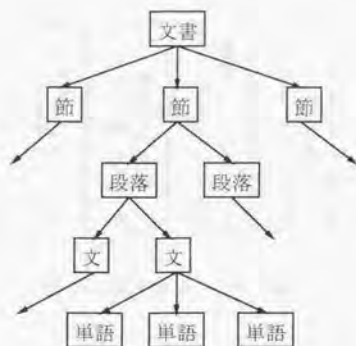


図 2.2: 文書のもつ論理構造の例

画像解析を行なって自動的にテキスト領域を抽出するものが多い。この文字認識については次節で詳説する。

2.2 文字認識

2.2.1 OCR

文書認識の中核をなす OCR 文字認識は、一般に活字文字認識、手書き文字認識、オンライン手書き文字認識に分類でき [9]、活字については和英文ともに実用化されており、最近では認識率が 99% のものも珍しくない。それぞれの認識アルゴリズムは、その認識対象、利用の目的などの違いから互いに多少異なるが、その概略はおおよ次のようにまとめられる。すなわち、まず入力された文書画像のテキスト領域に対して文字の切り出しを行ない、個々の文字認識 (ボタン認識) を行なう。その結果、複数の候補文字が類似度 (その候補文字の辞書ボタンと文書画像中に切り出された当該文字の間の類似性) とともに与えられる。そして不使用文字の除去や類似度による候補文字間の順位や各候補文字間の類似度の差や比などを利用して候補文字の絞り込みを行ない、さらに統計的な言語情報や文法規則、背景知識などを利用して候補文字を決定するという後処理を行なう (図 2.3)。

2.2.2 活字 OCR の認識誤り

本研究で利用する OCR は活字を対象としたものであり、第 2.2.1 節で述べたように 98~99% 程度の認識率が期待できる。しかしそれでもその認識誤りは検索時には無視できるものではないため、本節では活字 OCR の認識誤りについてとりあげる。一般に活字 OCR の認識率は、使用す

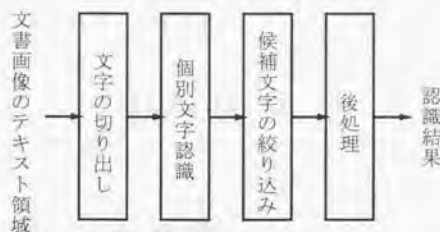


図 2.3: OCR による文字認識処理

表 2.1: 英語活字 OCR の認識誤りの構成 (Times font)

p:q	⇒0	⇒1	⇒2	⇒3	⇒4
0⇒	0	772	50	0	49
1⇒	258	634	59	3	0
2⇒	0	528	146	10	1
3⇒	0	3	4	6	0
4⇒	0	0	0	0	0

る OCR の認識性能だけでなく、文書画像の品質や文字のフォント、それを読み込むスキャナの性能、作業従事者の操作などにも大きく依存することが指摘されている。例えば Esakov ら [10] は英語活字 OCR の認識誤りを、元の文字列の長さ p とその認識結果の文字列の長さ q に着目して分類する実験を行なっている。この実験に用いた文書は Herman Melville の小説 “Moby-Dick” で 1,372,744 文字からなる。これを 400dpi のプリンタで Times font で出力し、紙面を自動給紙¹して文書画像に変換したものを OCR に認識させ、そのときの認識誤りの頻度についてまとめたのが表 2.1 である。

表 2.1 から英語活字 OCR の典型的な誤りは、頻度の高い順に挿入誤り ($0 \Rightarrow 1$)、置換誤り ($1 \Rightarrow 1$)、2 文字が 1 文字と認識される誤り ($2 \Rightarrow 1$) などであることが分かる。

2.2.3 認識誤りの分類実験

本研究では第 2.2.2 節の結果を踏まえ、日本語及び英語活字 OCR の認識誤りを以下に定める置換・欠落・挿入・結合・分解の 5 種類に限定した。そして実際に日本語及び英語活字 OCR の出

¹Esakov らの実験では、印刷文書を文書画像に変換するためにスキャナを使用する際に、人が印刷物を配置するよりも自動給紙した場合の方が良い結果が得られている。

力結果を用いて、認識誤りをこの5種類に分類するとともにその頻度分布を調べる予備実験を行った。その結果をそれぞれ表2.2、2.3に示す。

- 置換 (誤字)

1文字がOCRによって別の1文字に置換される誤り。

例: な→は ポ→ボ 間→間 i→i f→t

- 欠落 (脱字)

1文字以上がOCRによって抜け落ちる誤り。

例: E-R → ER 3-D → 3D

- 挿入

1文字以上がOCRによって挿入される誤り。

例: グラフィックス→グラ、フィックス

- 結合

2文字以上がOCRの文字の切り出し誤りによって1文字として認識される誤り。

例: し。→い (こ→に Pr → R ri → n li → h

- 分解

1文字がOCRの文字の切り出し誤りによって2文字以上として認識される誤り。

例: 利→夫 U 指→ま旨 ル→J し m → tn m → rn

この予備実験ではOCRに認識させるための元の誤りのないテキストと認識結果のテキストの比較を行なって認識誤りを抽出した。また実験における諸条件は以下の通りである。

- 使用した日本語活字OCRは(株)リコーのIMAZONEで認識率は98.3%、認識対象として情報処理学会論文誌1994年度のNo. 1~No. 5から抽出した約80KB(41,854文字)のテキストを使用した。
- 英文については、元のテキストとそれを認識したOCRのテキストの双方をElsevierから電子形態で出版されている*Artificial Intelligence*の1995年度8月号~1996年度5月号から抽出した約80KB(80,985文字)のテキストから得ており、そこで使用されているOCRの認識率は99.1%であった。

表2.2、2.3における“その他”は一意に分類できなかった誤りで、例えば“reconstruction”が“RConStMcDOn”と認識されるようないくつかの誤りが複合したような誤りである。表2.2から和文では置換誤りが約81%と誤りの大部分を占めるが、表2.3から英文では和文に比べて文字の切り出し誤りに起因する置換以外の誤りの割合が高いことが分かる。なお日本語活字OCRの認識誤りの原因の約4割は英数字(半角文字)の認識に絡むものであった。

表 2.2: 日本語活字 OCR の認識誤り

誤りの種類	出現頻度	相対頻度 (%)
置換 (誤字)	586	80.7
欠落 (脱字)	10	1.4
挿入	60	8.3
結合	49	6.8
分解	8	1.1
その他	13	1.8

表 2.3: 英語活字 OCR の認識誤り

誤りの種類	出現頻度	相対頻度 (%)
置換 (誤字)	232	40.1
欠落 (脱字)	68	11.7
挿入	60	10.4
結合	167	28.8
分解	38	6.6
その他	14	2.4

2.2.4 文字認識後処理

実際の文書では、低印字品質の文字 (かすれやにじみ、スキャナの解像度の影響など) があり得ることから、個々の文字レベルの認識では文字切り出しの誤りや1文字認識の誤りは避けられない。また日本語では文字の種類が多く、複雑な文字や、類似文字 (「り」と「リ」や「一」と「ー」など)、一文字が篇や旁など複数の部分から構成されるものがあることなども認識誤りの原因となっている。そこで文字列の言語的性質や背景知識といった様々な文脈情報を利用することで OCR 出力結果に含まれる認識誤りを自動検出あるいは訂正する、文字認識後処理の研究が行われてきた [9, 11, 12]。

英語のような言語では文が単語単位に区切られているため、スペースなどのデリミタを手掛かりとした単語単位の後処理が行われている。通例この後処理には、辞書、文字の連接に関する統計情報 (n-gram)、OCR の認識誤り特性などを活用するが、性能向上のためにこれらに加えて様々なヒューリスティクスを用いる場合も多い [11]。一方日本語の文章の後処理の場合、単語境界があいまいであることや日本語の単語の多くは1文字や2文字から成り立っていることなどから、文法的制約なども利用した後処理手法が研究されている [9]。

文字認識後処理の基本的なプロセスには、

- 認識されたテキストの正当性を文字の接続に関する統計情報や単語辞書との照合によって検査し、不適ならその位置の文字を別の候補文字に置き換える。
- 正解の可能性の高い文字候補のあらゆる組合せを単語辞書や文法規則などを参照しながら生成し (generate)、選択した文字の正当性を検証する (test) ことで正しい文章を構築していく。

という2つがある。また辞書等の具体的な活用方法をまとめると、以下のようになる。

- 辞書

辞書は主にスペルチェックに用いられ、辞書内の単語と照合しない文字列がまず認識誤りを含む単語として検出される。またこのような認識誤りを含む文字列と辞書内の単語との類似度を定義して、最も類似している単語が1つに定まればその単語に訂正するのが一般的である。類似度としてはエディット距離 (第2.4.3節) などが用いられ、同じ類似度をもつ訂正候補の取捨選択に他の種類の情報を活用する。また辞書としては通常用いる単語辞書だけでなく、認識文書がカバーする分野の専門用語辞書や、頭字語や固有名詞といったものを考慮する場合もある。

- 文字の接続に関する統計情報 (n-gram)

文字の n-gram (第2.4.3節) とは n 個の文字の組合せのことで、文書におけるその頻度情報から抽出する n-gram とめったに出現しない n-gram などの情報が得られる。この情報を利用して、出現頻度の低い n-gram ができたらその中に認識誤りを含んでいる可能性が高いと考え検出する。逆に出現頻度の低い n-gram を高い n-gram に置き換えることで訂正を行なう場合もあるが、訂正の性能はあまり良くない [12]。

- 言語的な制約 (文法など)

認識誤りと考えられる文字を訂正する際に、訂正後の文章が文法的に正しくなるように訂正候補となる文字を選別する。

- OCR の認識誤り特性

認識に使用した OCR の認識誤り特性を調べ、発生し易い認識誤りを考慮した訂正を行なう。具体的には、認識誤りを含む文字列と辞書内の単語との類似度の計算に際してこの情報を加味し、発生し易い認識誤りを含む組合せの類似度を大きくするなどの方法がとられる。

- ヒューリスティクス

例えば英語では、非テキスト領域を誤って認識したために出力された全く意味不明な文字列を除去するために、長過ぎる文字列や数字とアルファベットの混合文字列などを削除するなどのヒューリスティクスが用いられている。

また1種類のアルゴリズムによるOCRでは認識率に限界があるので、複数のOCRを同時に行い多数決をとることで誤りを訂正し認識率を上げるといった試みもある。

2.3 文書認識と情報検索

2.3.1 検索対象テキストの生成

文書画像解析は通常、ページ画像をいくつかのブロックに分割しその属性とともに出力する。このようなページのレイアウト構造に関する情報(図2.1参照)は有用ではあるが、文書認識の結果を検索目的に活用することを考えた場合、文書の論理構造に関する情報(図2.2参照)まで抽出できるとさらに望ましい。この文書の論理構造は、著者名やタイトル、節、段落などから構成されるが、通常のレイアウト構造の解析ではこれらは全てテキスト領域として扱われ区別されない。しかし文字認識の結果得られるテキストにこのような論理的な属性が付与されていれば、例えば著者やタイトルを指定した検索や文書内の節や段落を対象とした検索など柔軟な問い合わせができる。そのため最近では、自動的あるいは半自動的に学術論文の文書画像の論理構造を解析してSGML文書として出力する試み[7, 8, 13]が行なわれるようになってきた。アプローチとしては、レイアウト構造解析と同時に論理構成要素を特定しSGML文書として出力するものもあれば[13]、レイアウト構造解析の結果を一旦SGML文書として出力し、それを元に論理構造を表すSGML文書に変換するもの[7, 8]もある。ブロックへの文書の論理構成要素名の付与には、通常ページ画像のレイアウトに関する様々なヒューリスティクスが用いられる。またこのような文書画像からの検索用テキストデータの自動生成が試みられるようになった背景には、学術論文誌や名刺など比較的定型と考えられる文書に対するレイアウト構造解析技術がほぼ確立されたことや、活字OCRの認識率が高くなったことなどがある。

2.3.2 OCR認識誤りの検索への影響

検索対象となるテキストデータの文書画像からの自動生成は現実味を帯びてきたが、このテキストには認識誤りが含まれる。そこで本節では第2.2.2節や第2.2.3節で述べたOCRの認識誤りが情報検索に与える影響について考察する。

文字列検索への影響

認識誤りを含むテキストに対して何の対策も講じずに文字列検索を行なうと以下の2つの問題が発生することは想像に難くない。

- 検索洩れ(再現率の低下)

認識誤りによって検索語に照合する筈の文字列が変形してしまい検索できない。

- 検索ノイズ(適合率の低下)

認識誤りによって別の良く似た単語が検索語に変形してしまい、検索すべきでない文字列を検索してしまう²。

この2つのうち深刻なのは検索洩れ(再現率の低下)の問題で、例えば藤澤らの和文における実験[14]では、文字認識率98.2%のときに再現率は約96%に、認識率94.3%では約87%に低下するという結果が示されている。

文書検索への影響

認識誤りを含むテキストに対してその誤りを考慮せずに文書検索を行なった場合どのような影響がでるかについて、Taghvaらの英文における実験[15, 16, 17]から得られた知見を紹介する。文書検索には、問い合わせを任意の検索語の論理式で表す論理演算モデルや文書及び問い合わせの両方を索引語ベクトルで表現するベクトル空間モデルなど様々なものがあるが、各モデルに共通する知見として以下のものが得られている。

- 索引語の増加

認識誤りを含まない場合と比べて50%から400%索引語が増加している³。

- 固有名詞などの認識率が悪い

文書の内容を端的に表しており検索においても重要と考えられる、固有名詞や頭字語、数値などの誤りは、OCRのもつ辞書などを用いても訂正できないため特に認識率が悪く問題である。

- 短い文書ほど影響が大きい

短い文書(50語程度)では、認識誤りの検索に与える影響が大きい。これは長い文書では、文書のもつ冗長性のため検索効率の低下が緩和されるためである。

一方検索モデルによる違いとしては、単語の重み付けやランキングを行わないシステム(文字列の完全照合のみによる検索)では、認識誤りのないテキストを検索する場合と比較して有意な差がみられない。逆に、適合性フィードバックなど高度な検索を行なう場合には、認識誤りの考慮は不可欠であるという結果が得られている。

2.4 曖昧検索

手でOCR認識誤りを訂正することは、印刷文書の急速なデジタル化が進む現状を鑑みると、その労力・コストの観点から非現実的になりつつあり、また第2.2.4節で述べたようなOCR認識後に後処理を施しても完全に誤りのないテキストを得ることは不可能でもある。そこで、認識誤

²例えば、“ブランド”が“ブランド”と認識されると、“ブランド”で検索するとこの誤認識が検索ノイズとなる。

³この増加率の違いは、文書の平均語数やOCRの認識率による。

表 2.4: 類似文字テーブルの例

元の文字	語認識結果
l	l, i, i
i	l, i, t
t	l, c
c	C, e
e	o, a, c
a	o, s, e

りを検索時に考慮する曖昧検索手法が種々提案されており、本節では以下にその代表的な手法を説明する。

2.4.1 類似文字テーブル (Confusion Matrix/Table)

OCR の認識誤りに最も直接的に対処する方法は、OCR の認識誤りパターンを学習した類似文字テーブル (表 2.4) を作成する方法である。この類似文字テーブルは、認識結果のテキストに対して誤り訂正の後処理を行なうために利用されることも多い [12]。

曖昧検索への活用方法としては、例えば類似文字の共通コード化がある [18]。これは予め何らかの方法で類似文字のクラスタリングを行ない類似文字をそのクラスターの標準文字である共通のテキストコードに変換することによって、検索時に検索語と照合するようにする方法である。この手法の場合、類似文字のクラスタリングをうまく行なうと検索誤れを減らすことができるが、類似文字を完全に同一のものとみなすため検索ノイズの問題が無視できない。

また類似文字テーブルを参照して検索語を拡張することで再現率を上げる試みもある [18, 19]。例えば、検索語 “cat” が与えられたら類似文字テーブル (表 2.4) を参照して “[cCe][aose][tle]” に拡張する。類似文字の共通コード化との違いは、認識誤りの方向性⁴を扱える点にある。このとき、元の文字と認識結果の文字の類似度 [19] や誤り易さに基づいた確率を用いて、拡張検索文字列の絞り込を行ない検索ノイズの低減を図ることもある。

2.4.2 複数認識候補型検索手法

丸川らの提案する複数認識候補型検索手法 [14, 19, 20] は、OCR が出力する複数の認識候補文字 (第 1 位候補文字以外) 中に正解文字が含まれている率が高いことを考慮して、これらの第 2 位以下の認識候補文字を検索に利用している。この手法は、まず不要な検索ノイズを低減するため類似度を用いて各候補文字の絞り込みを行ない、この精選した候補文字を認識結果のテキスト

⁴表 2.4 の例で説明すると、“l” は “c” と誤認識される可能性があるがその逆は起こらないということ。共通コードに変換するとこの区別はつけられない。

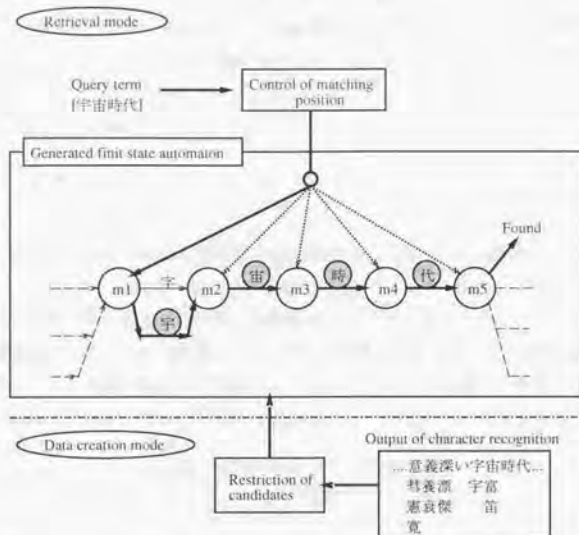


図 2.4: 複数認識候補型検索手法

にもたせる。そして検索時にこの複数の候補文字と照合を行なうことで検索洩れを防いでいる。

具体的には図 2.4 に示すように、ノード $m1$ とノード $m2$ 間のパスに $m2$ 文字目の候補文字を順次割り当てて有限オートマトンを生成し、検索時に各文書から生成されたオートマトンに検索語 (図 2.4 の例では「宇宙時代」) を入力して照合を行なう。この照合はまず照合位置制御により、検索語の最初の文字と照合を行なうテキスト中のノード番号を設定する。そして、遷移するパスが存在すれば遷移を行ない、検索語が受理されればその文字位置を検索結果とする。また遷移するパスが存在しなければ、設定したノードからの検索は不適切とし、照合位置制御により照合位置を右に 1 ノード分シフトし、同様の処理を繰り返す。

複数認識候補型検索手法により丸川らの実験では、文字列検索の再現率 96.2% を 99.7% に、87.3% を 99.5% に改善できたと報告されている [20]。

2.4.3 部分文字列照合

部分文字列照合はいわゆる類似検索で、検索語と似ている文字列を文書やレコード中から検索する手法である。よって適用対象は OCR の認識誤りを含むテキストに限らない。

部分文字列照合の最も単純なものは、検索語をワイルドカードを用いて拡張した検索であろう。

例えば、“cat”を“?at”、“c?t”、“ca?”に拡張するような場合である。しかし曖昧検索では、何らかの方法で元の検索語との類似性を規定して拡張検索文字列を絞り、不用意に検索ノイズをのせないようにするのが通常である。この類似性の規定方法としては以下の2つが良く用いられる。

- n-gram [18, 21, 22, 23]
- エディット距離 [18, 24, 25]

n-gram 文字の n-gram とはある文字列中に含まれる n 個の文字の組合せのことで、例えば“Markov”という文字列の trigram(3-gram) は通常“Mar”, “ark”, “rko”, “kov” とされる。検索語中の n-gram と同じものを多く含むものを曖昧検索し、例えばデータベース中のエントリ E と検索語 T の類似度 S を式 (2.1) で定義し、この類似度が閾値を超えるものを検索結果とする [18]。また場合によっては、単語の境界まで考慮した“Ma”, “ov” や単語中の任意の3文字によって構成される“Mak”, “Mkv”, “arv”なども trigram として利用される場合もある [23]。

$$S = \frac{|n - \text{grams}(T) \cap n - \text{grams}(E)|}{|n - \text{grams}(T)|} \quad (2.1)$$

エディット距離 エディット距離を用いて具体的に類似度を定義する方法では、検索語との類似度がある閾値を超えるような文字列のみを検索語と等価とみなす。エディット距離は基本的には、文字の (1) 削除 (2) 挿入 (3) 置換の3つの操作のみで、1つの文字列を別の文字列に変換するときの最小のコストと定義されている。ここでコストとは、3つの操作それぞれについて定義されており、全て等しくすることも操作によってコストを変えることもできる。またエディット距離は次の再帰関係式を動的計画法 (dynamic programming) を用いて求めることができる。すなわち検索語を $T = t_1 t_2 \dots t_m$ 、比較対象の文字列を $D = d_1 d_2 \dots d_n$ 、 T の最初の i 文字と D の最初の j 文字との距離を $dist_{i,j}$ 、3つの操作のコストをそれぞれ c_{del} 、 c_{ins} 、 c_{sub} とすると、次式を解くことで求める。

[初期条件]

$$dist_{0,0} = 0 \quad (2.2)$$

$$dist_{i,0} = dist_{i-1,0} + c_{del}(t_i) \quad 1 \leq i \leq m \quad (2.3)$$

$$dist_{0,j} = dist_{0,j-1} + c_{ins}(d_j) \quad 1 \leq j \leq n \quad (2.4)$$

[再帰関係式]

$$dist_{i,j} = \min \begin{cases} dist_{i-1,j} + c_{del}(t_i) \\ dist_{i,j-1} + c_{ins}(d_j) \\ dist_{i-1,j-1} + c_{sub}(t_i, d_j) \end{cases} \quad (2.5)$$



図 2.5: エディット距離の計算例

Lopresti ら [24, 25] は、初期条件の式 (2.4) を式 (2.6) に変更することで D が文書のように T と比較してはるかに長い場合にも適用できることを示した (図 2.5)⁵。

$$dist_{0,j} = 0 \quad 1 \leq j \leq n \quad (2.6)$$

すなわち、文書 D と検索語 T のエディット距離 $\varepsilon(T, D)$ を、 $\varepsilon(T, D, k)$ を図 2.5 の最下行の $j = k$ におけるエディット距離として、式 (2.7) のように定義した。

$$\varepsilon(T, D) = \min\{\varepsilon(T, D, k) | 0 \leq k \leq n\}. \quad (2.7)$$

これは検索語と文書中で最もよく照合した文字列との距離といえ、 $[0, m]$ の値をとる。これをもとにメンバシップ関数 $M(T, D)$ を $[0, 1]$ の値をとるように定めることで、文書検索への適用を可能にしている。

$$M(T, D) = \frac{1}{e^{\alpha \varepsilon(T, D) / (m - \varepsilon(T, D))}}. \quad (2.8)$$

表 2.5 に、いくつかの検索語長とエディット距離におけるこのメンバシップ関数の値を示す。この値は検索語と全く同じ文字列が文書中に存在すれば 1 となり、検索語中のどの文字も文書中に現れなければ 0 となる。また使用する論理演算をファジー理論を用いて以下のように定め、論理演算モデルによる検索も可能にしている。

$$F_{AND}(x, y) = \min(x, y). \quad (2.9)$$

$$F_{OR}(x, y) = \max(x, y). \quad (2.10)$$

$$F_{NOT}(x) = 1 - x. \quad (2.11)$$

⁵ $c_{del} = c_{ins} = c_{sub} = 1$ としている。

表 2.5: 様々な検索語長のメンバシップ関数の値

エディット距離 $\varepsilon(T, D)$	検索語長			
	2	3	4	5
0	1.000	1.000	1.000	1.000
1	0.368	0.607	0.717	0.779
2	0.000	0.135	0.368	0.513
3		0.000	0.050	0.223
4			0.000	0.018
5				0.000

よって例えば, “The quick brown fox jumps over the lazy dog” という認識誤りを含む文書 D と (fox AND dog) という問い合わせ Q が与えられると表 2.5 などから次のような計算が行なわれる。

$$\begin{aligned}
 Q(D) &= F_{AND}(M(\text{fox}, D), M(\text{dog}, D)) \\
 &= F_{AND}(0.607, 1.000) \\
 &= \min(0.607, 1.000) \\
 &= 0.607
 \end{aligned}
 \tag{2.12}$$

第3章

提案する曖昧検索手法

本研究で想定する電子図書館は文書画像を蓄積主体とするため、それをOCR処理して得られる不完全なテキストを対象とした検索システムが必要となる。本章ではその核となる、認識誤りを含むテキストに対する文字列検索手法、すなわち曖昧全文検索手法の提案を行なう。

まず第3.1節で提案する3つの手法の概要を説明し、つづく第3.2節で提案手法が参照する類似文字テーブルなどを構築するために必要な、認識誤りの抽出及び分類方法について述べる。第3.3節では、各手法が拡張検索文字列の得点計算の際に必要な文字及び文字列の確信度を定義するとともに、得点付けのアルゴリズムについて説明する。

3.1 提案手法の概要

本研究で提案する曖昧検索手法は基本的には、類似文字テーブルなどを用いて1つの検索語を複数の妥当な検索文字列に拡張し、その拡張文字列を用いて認識誤りを含むテキストを検索することで検索ノイズを抑えながら検索洩れを救済する手法である。また、検索語拡張の際に参照する類似文字テーブルなどのテーブルの種類の違い、換言すれば妥当な検索文字列を選択するアルゴリズムの違いに基づいて、本章では合計3つの曖昧全文検索手法を提案している(図3.1参照)。

1) Confusion Matrix Retrieval Method (CMR 法)

この手法は、OCRの認識誤りのうち置換誤りの可能性のある文字と、その誤り易さに基づく確率を格納した類似文字テーブル(Confusion Matrix)を利用した全文検索手法である。検索時にこの類似文字テーブルを参照して、認識誤りを含むテキストにおいて検索洩れを起こさないように、1つの入力検索語からOCRの置換誤りを考慮した複数の検索文字列を生成する。入力検索語の拡張過程では、検索文字列としての妥当性を評価して実際の検索文字列数を絞り込むべく、各生成文字列にOCRの誤る確率を元に計算した得点を与え、その得点が閾値を超えているものだけを拡張検索文字列として出力する。

2) Expanded Confusion Matrix Retrieval Method (ECMR 法)

この手法はCMR法と本質的に同じアルゴリズムを採用しているが、拡張検索文字列生成時に置換誤りを扱う類似文字テーブルだけでなく、欠落・挿入・結合・分解誤りを扱う拡張類似文字テーブル(Expanded Confusion Matrix)をも考慮する点が異なる(図3.2参照)。これによって、認識誤りを含むテキストと誤りのない元のテキストとが1対1に対応しない、文字切り出しの誤りを含むような文字列も検索できる。

3) Bigram Matrix Retrieval Method (BMR 法)

CMR法で生成された拡張検索文字列によって検索された文字列の得点を、類似文字テーブルと文字のbigram統計に基づいた接続確率を保持したbigramテーブル(Bigram Matrix)を用いて再計算し、閾値による再評価を行なって検索結果を出力する。bigramテーブルを参照することで、より信頼性の高い得点、ひいては検索結果を得ることを目的とする。

なお類似文字テーブル及び拡張類似文字テーブルは、次節で述べる方法でトレーニングセットとして与えられたOCRの出力した認識誤りを含むテキストと、それに対応する誤りのないテキストを比較することで検索前に予め作成しておく。すなわち、両テキストを比較することで正しい文字(列)とそれに対応する認識誤りを含む文字(列)を $m:n$ の形で抽出し、これが $m:m$ のとき m 個の置換誤りと解釈して類似文字テーブルに、 $m:0$ のとき m 個の欠落誤り、 $0:n$ のとき n 個の挿入誤り、 $2:1$ のとき結合誤り、 $1:2$ のとき分解誤りと解釈して拡張類似文字テーブルに格納する。一方bigramテーブルは、誤りを含まない(OCRの出力したものではない)比較的大量のコーパスを用いて推定された文字の接続確率を保持する。

3つの提案手法はそれぞれ、置換誤りを考慮した検索(CMR法)、ほぼ全ての種類の誤りを考慮した検索(ECMR法)、置換誤りと文字の接続確率を考慮した検索(BMR法)といえ、CMR法と

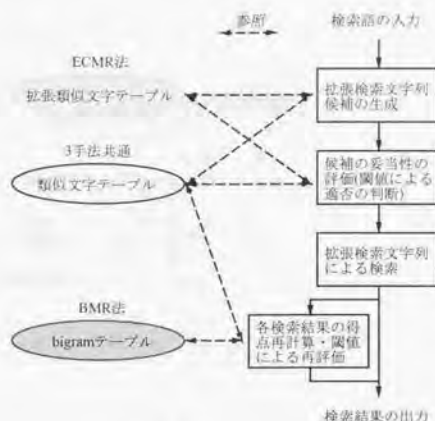


図 3.1: 3つの提案手法

ECMR 法のアルゴリズムには本質的な違いはない。また提案手法の特徴は、検索ノイズを抑えながら検索洩れを削減するために確率に基づいた得点計算により妥当性を考慮しながら入力検索語を拡張する点にある。

3.2 認識誤りの抽出と分類

本研究では、OCR 認識誤りを置換・欠落・挿入・結合・分解の 5 つの誤りのいずれかあるいはその複合誤りと解釈しており、本章で述べる提案手法はこの 5 種類の誤りそれぞれについて作成された類似文字テーブルあるいは拡張類似文字テーブルを参照する。それぞれのテーブルに蓄える情報は、正しい長さ m の文字 (列) α^m 、その OCR 認識結果である長さ n の文字 (列) β^n 、及び β^n が α^m とみなせる確率 $P(\alpha^m|\beta^n)$ で、定義した 5 種類の誤りと $\{m, n\}$ の関係は、置換誤り = $\{1, 1\}$ 、欠落誤り = $\{1, 0\}$ 、挿入誤り = $\{0, 1\}$ 、結合誤り = $\{2, 1\}$ 、分解誤り = $\{1, 2\}$ となる。

3.2.1 認識誤りの抽出

まず、トレーニングセットとして与えられた OCR の出力した認識誤りを含むテキストと、それに対応する誤りのないテキストを比較して、異なる部分、即ち認識誤りの部分を抽出する。具体的には図 3.3 に示すように、両テキストを先頭から 1 文字ずつ比較して不一致が起こる部分まで進み、そこから比較元を順に変えながら、連続した 2 文字 ($\Delta \blacktriangle$) が一致するまでこの操作を繰り返す。このような 2 文字が見つかった時点で、不一致が起きた時点からそこまでの間の文字列

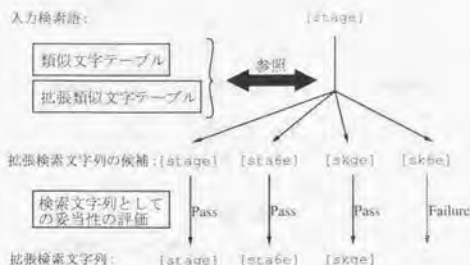
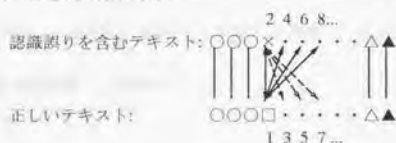


図 3.2: 拡張検索文字列の生成 (英文 ECMR 法の例)

2つのテキストの比較の順序



比較照合が失敗した文字位置から数文字先に一致する2文字の文字列(▲)が見つかるまで、比較元となる文字を上記の順に変えながら照合を行なう。

図 3.3: 認識誤りの抽出方法

を認識誤りとして抽出する。この比較を両テキストの最後まで行なうことで、トレーニングセットの認識誤りを全て抽出する。

3.2.2 認識誤りの分類

抽出された認識誤りは、一般に $\{\alpha^m, \beta^n\}$ の形をしており、任意の $\{\alpha^m, \beta^n\}$ の組合せについてそのままテーブルに蓄えることも考えられるが、 m, n が大きくなると学習量の観点から $P(\alpha^m | \beta^n)$ を正しく推定することが困難となる。そこで抽出された認識誤り $\{\alpha^m, \beta^n\}$ を、以下のヒューリスティクスに基づいて分類する。また、このヒューリスティクスに当てはまらない認識誤りについては無視し、認識誤りとして抽出されない部分については正しく認識されているものとしている。

認識誤りの分類ヒューリスティクス



(注)上側:認識誤りを含むテキスト 下側:正しいテキスト

図 3.4: 認識誤りの抽出例

1. $\{m, n\}$ が $\{1, 1\}$, $\{1, 0\}$, $\{0, 1\}$, $\{2, 1\}$, $\{1, 2\}$ のときは、それぞれ無条件に置換誤り、欠落誤り、挿入誤り、結合誤り、分解誤りと解釈する (図 3.4 参照)。
2. $\{m, n\}$ が $\{m, 0\}$ 及び $\{0, n\}$ の場合、それぞれ m 個の欠落誤り、 n 個の挿入誤りと解釈する。
3. $m = n$ の場合、 m 個の置換誤りと解釈する。

3.3 得点付けのアルゴリズム

3.3.1 文字の確信度

使用する全ての文字集合を $\{C_1, C_2, \dots, C_{all}\}$ とする。このとき文字の確信度とは、正しいテキストにおける文字が C_x である事象を A_x 、OCR のテキストにおける文字が C_y である事象を B_y とするとき、OCR のテキストにおける文字 C_y が正しいテキストにおいて文字 C_x である (と確信できる) 確率 $P(A_x|B_y)$ のことである。この $P(A_x|B_y)$ は、ベイズの定理から式 (3.1) で計算される。この文字の確信度は、得点付けの際に 3 つ全ての提案手法で利用する。

$$P(A_x|B_y) = \frac{P(A_x)P(B_y|A_x)}{\sum_{i=1}^n P(A_i)P(B_y|A_i)} \quad (3.1)$$

3.3.2 類似文字テーブル

類似文字テーブルには、トレーニングセットから得られた置換誤りの可能性のある文字が全てその確信度とともに格納されている (図 3.5 参照)。類似文字テーブルは、OCR の出力した認識誤りを含むテキストとそれに対応する元の誤りのないテキストを比較することで得られ、表 2.2、2.3 において最も頻度の高い置換誤りに対処するために構築する。また CMR 法が使用するのはこの類似文字テーブルのみである。

$B_y =$ 認識結果の文字 C_y

	B_1	B_2	B_3
$A_1 =$ 元の文字 C_x	$P(A_1/B_1)$	$P(A_1/B_2)$	0	
A_2	0	$P(A_2/B_2)$	0	
A_3	0	$P(A_3/B_2)$	$P(A_3/B_3)$	
.....				

図 3.5: 類似文字テーブル

3.3.3 拡張類似文字テーブル

ECMR 法は、図 3.5に示す類似文字テーブルと同様な構成をもつ欠落文字テーブル、挿入文字テーブル、結合文字テーブル、分解文字テーブルを利用して置換以外の誤りにも対処できるようにした手法である (図 3.6、3.7参照)。拡張類似文字テーブルとは、これら 4 つのテーブルの総称である。

図 3.6(a)、(b) はそれぞれ、欠落文字テーブルと挿入文字テーブルを表しており、ここで仮想文字 V_m と V_i はそれぞれ認識結果の文字と元の文字を表す。このような仮想文字を考える事で、これらのテーブルを類似文字テーブルと同様に扱うことができ、ECMR 法でも CMR 法と同じ得点付けのアルゴリズムを用いる事が可能になる。また図 3.7(a)、(b) はそれぞれ結合文字テーブルと分解文字テーブルを表し、同様の理由からここでも $a_x = (A^{i-1}A^i)_x$ 、 $b_y = (B^{i-1}B^i)_y$ という元のテキストと認識結果のテキストにおける 2 つの連続する文字を定義する。さらに認識結果の文字 B_y が元のテキストにおいて a_x という文字列とみなせる確率 (確信度) $P(a_x|B_y)$ が結合文字テーブルには蓄えられており、同様に認識結果の文字列 b_y が元のテキストにおいて A_x という文字とみなせる確信度 $P(A_x|b_y)$ が分解文字テーブルに蓄えられている。

3.3.4 bigram 統計に基づいた文字の接続確率

BMR 法では、類似文字テーブルに保持する OCR の誤り確率とは別の観点からの確信度を導くために、bigram 統計から得られる文字の接続情報 (接続確率) を利用している。通常我々が使用

(a) 欠落文字テーブル

	A_1	A_2	A_3	...
V_m	$P(A_1/V_m)$	$P(A_2/V_m)$	$P(A_3/V_m)$	

V_m = 欠落文字 A_i に対応する仮想的な文字

(b) 挿入文字テーブル

	B_1	B_2	B_3	...
V_i	$P(V_i/B_1)$	$P(V_i/B_2)$	$P(V_i/B_3)$	

V_i = 挿入文字 B_j に対応する仮想的な文字

図 3.6: (a) 欠落文字テーブルと (b) 挿入文字テーブル

している日本語の文字種は約 3000 種類¹と英語に比べて桁違いに多いため、十分な学習量のテキストデータに対して n -gram 統計 ($n \geq 3$) を求めるのは困難である²。そこで文字の接続情報として BMR 法では、比較的頻度統計がとり易く、自然言語の統計モデルとして様々な研究で利用されている bigram[26] を用いている。

本稿でいう文字の接続確率とは、元のテキスト中の文字 A^i が A^{i-1} の次に続く確率の推定値で式 (3.2) で求められる。

$$P(A^i|A^{i-1}) = \frac{C(A^{i-1}A^i)}{C(A^{i-1})}. \quad (3.2)$$

式 (3.2) において、 $C(A^{i-1}A^i)$ と $C(A^{i-1})$ はそれぞれ文字列 $A^{i-1}A^i$ (bigram) の出現頻度と文字 A^{i-1} (unigram) の出現頻度を表す。そしてこのようにして求めた文字の接続確率を図 3.8 に示す bigram テーブルに保持し検索された文字列の得点付けの際に参照する。

3.3.5 BMR 法における文字の確信度

認識結果の文字 B_y^i が元のテキストにおいて A_x^i とみなせる確率は、類似文字テーブル (図 3.5) によると $P(A_x^i|B_y^i)$ であった。しかし BMR 法では認識結果のテキストにおける B_y^i の直前の文字 B^{i-1} に着目して式 (3.3) の確率を計算する。

$$P(A_x^i|B^{i-1}) = \sum_x P(A_x^i|A_x^{i-1})P(A_x^{i-1}|B^{i-1}), \quad (3.3)$$

つまりこれは認識結果のテキストにおいてその前の文字が B^{i-1} であるとき、次の文字が元のテキストにおいて A_x^i とみなせる確率である。式 (3.3) の右辺の $P(A_x^i|A_x^{i-1})$ が bigram テーブル (図

¹例えば JIS 第 1 水準の漢字は 2965 文字で、これに仮名、数字、アルファベットなどを加えれば一般的な文書に含まれる文字は大体まかなえる。

²最近では大容量テキストから n -gram 統計 ($n \geq 3$) をとることも可能になりつつある [21]。

(a) 結合文字テーブル					(b) 分解文字テーブル				
	B_1	B_2	B_3	...		$b_1 = (B^{i-1} B^i)_1$	$b_2 = (B^{i-1} B^i)_2$	$b_3 = (B^{i-1} B^i)_3$...
$a_1 = (A^{i-1} A^i)_1$	$P(a_1 B_1)$	$P(a_1 B_2)$	$P(a_1 B_3)$		A_1	$P(A_1 b_1)$	$P(A_1 b_2)$	$P(A_1 b_3)$	
$a_2 = (A^{i-1} A^i)_2$	$P(a_2 B_1)$	$P(a_2 B_2)$	$P(a_2 B_3)$		A_2	$P(A_2 b_1)$	$P(A_2 b_2)$	$P(A_2 b_3)$	
$a_3 = (A^{i-1} A^i)_3$	$P(a_3 B_1)$	$P(a_3 B_2)$	$P(a_3 B_3)$		A_3	$P(A_3 b_1)$	$P(A_3 b_2)$	$P(A_3 b_3)$	
...					...				

b_y = 認識結果の文字列 $B^{i-1}B^i$ の y 番目の組合せ

a_x = 元の文字列 $A^{i-1}A^i$ の x 番目の組合せ

図 3.7: (a) 結合文字テーブルと (b) 分解文字テーブル

3.8) を参照することで得られる文字の接続確率で、一方 $P(A_x^{i-1}|B^{i-1})$ が式 (3.1) で定義した、認識結果のテキストにおいて B^{i-1} である文字が元のテキストにおいて A_x^{i-1} とみなせる確率、つまり確信度である。このように類似文字テーブルと bigram テーブルを参照することで、認識結果のテキストにおける直前の文字 B^{i-1} から、正しいテキストにおける次の文字が A_x^i である確率を求めることができる。その結果、類似文字テーブルのみから得られる確信度 $P(A_x^i|B_y^i)$ とは別に、認識結果のテキストにおけるその直前の文字から、 $P(A_x^i|B^{i-1})$ という異なる観点からの確率が得られる。

さてここで類似文字テーブルのみによって定まる $P(A_x^i|B_y^i)$ を $m_1(A_x^i)$ 、式 (3.3) で求めた $P(A_x^i|B^{i-1})$ を $m_2(A_x^i)$ とおく。するとこの2つの確率を統合した新たな確率 $m(A_x^i)$ は、Dempster の結合規則 [27, 28] を用いると式 (3.4) のように計算される。

$$m(A_x^i) = \frac{m_1(A_x^i)m_2(A_x^i)}{1 - \sum_x m_1(A_x^i)(1 - m_2(A_x^i))}. \quad (3.4)$$

BMR 法では、式 (3.1) で定義した元の確信度 $m_1(A_x^i) = P(A_x^i|B_y^i)$ の代わりに、文字の接続確率も考慮したこの $m(A_x^i)$ を新たな文字の確信度として使用する。

3.3.6 文字列の確信度

CMR 法では、認識結果のテキストを検索して得られる文字列が正しい確率を、文字列中の各文字が正しい確率の積と仮定する。つまり、認識結果の文字列 $B^{12...n}$ が元の文字列 $A^{12...n}$ に対応する確率を、

$$P(A^{12...n}|B^{12...n}) = P(A^1|B^1)P(A^2|B^2)...P(A^n|B^n). \quad (3.5)$$

で表す。ここで $P(A^{12...n}|B^{12...n})$ のことを文字列の確信度と定義する。

$A^i =$ 後に続く文字

	A_1^i	A_2^i	A_3^i
A_1^{i-1}	$P(A_1^i A_1^{i-1})$	$P(A_2^i A_1^{i-1})$	$P(A_3^i A_1^{i-1})$	
A_2^{i-1}	$P(A_1^i A_2^{i-1})$	$P(A_2^i A_2^{i-1})$	$P(A_3^i A_2^{i-1})$	
A_3^{i-1}	$P(A_1^i A_3^{i-1})$	$P(A_2^i A_3^{i-1})$	$P(A_3^i A_3^{i-1})$	
...				

$A^{i-1} A^i =$ 元のテキストにおける bigram

図 3.8: bigram テーブル

BMR 法でも文字列の確信度は式 (3.5) で定義するが、式 (3.4) で定義した $m(A^i)$ を式 (3.1) で定義した $P(A^i | B^i)$ の代わりに用いて計算する。

しかし ECMR 法では、文字列の確信度の計算方法が若干異なる。これは拡張類似文字テーブルを用いて入力検索語から複数の検索文字列を生成すると、入力文字列と生成文字列との間に 1 対 1 の文字の対応関係が成り立たなくなるためである。よって ECMR 法では文字列の確信度は以下のようにして算出する (図 3.9 参照)。

- 欠落誤りを含む場合

欠落文字に対応する仮想的な文字 V_m を認識結果のテキストにおいて考え、欠落文字の部分の確信度 P_m を次式で与える。

$$P_m = P(A_z | V_m) \cdot P_{m_0}. \quad (3.6)$$

式 (3.6) において $P(A_z | V_m)$ は図 3.6 に示した欠落文字テーブルより得られる。また P_{m_0} は、欠落の起こる確率で、和文では $P_{m_0} = 0.00024$ (表 2.2 参照)、英文では $P_{m_0} = 0.00084$ (表 2.3 参照) と推定した。実際には認識結果のテキストにおける文字 V_m は仮想的なもので観測できないため、 P_{m_0} は必ず考慮しなければならない。文字列の確信度の算出は、この欠落文字の確信度を考慮した上で式 (3.5) と同様に算出する。

- 挿入誤りを含む場合

認識結果のテキストに挿入された文字に対応する仮想的な文字 V_i を元のテキストにおいて



図 3.9: ECMR 法における文字列の確信度の計算 (和文の例)

考え、図 3.6 に示した挿入文字テーブルから挿入文字の確信度を得る。式 (3.5) による文字列の確信度の算出の際にその挿入文字の確信度も余分に掛け合わせる。

● 結合、分解誤りを含む場合

それぞれ図 3.7 に示した結合文字テーブルと分解文字テーブルを参照して結合文字の確信度、分解文字の確信度を得る。文字列の確信度は、結合文字あるいは分解文字の確信度も置換文字のそれと同様に扱い式 (3.5) によって算出する。

3.4 まとめ

本章では検索語拡張によって認識誤りを含むテキストを曖昧検索する手法として、置換誤りのみを考慮する CMR 法、置換・欠落・挿入・結合・分解誤りを考慮する ECMR 法、置換誤りと文字の接続確率を考慮する BMR 法の 3 つを提案した。提案手法を利用するには、全ての手法が参照する類似文字テーブルと ECMR 法が参照する拡張類似文字テーブルをまず作成する必要がある。そこでそのために必要な認識誤りの抽出と分類方法を第 3.2 節で説明した。またこれら 3 つの提案手法は、扱う認識誤りの種類及び文字の接続確率を考慮するかどうかという点で異なっているが、拡張検索文字列の確信度を定義しこの確信度を閾値で評価することで妥当な拡張検索文

字列を選択するというアルゴリズムは共通である。よって、これらの提案手法の検索性能を比較することで、扱う認識誤りの種類や文字の接続確率が検索性能に与える影響を明らかにすることができる。本論文では、この3つの提案手法の検索性能の評価及び比較については第4章で行ない、また第3.3節で定義した文字列の確信度と確率との関係については第7章において考察する。

第4章

曖昧検索手法の評価

第3章において提案した3つの曖昧検索手法を用いて、日英それぞれの活字OCRが実際に出力したテキストデータに対する検索効率を調べる実験を行なった。本章ではその実験内容を説明し、和英両テキストにおける提案手法の検索効率を示すとともに、3つの提案手法の実現する検索効率の比較を和英両テキストそれぞれにおいて行なう。具体的には、第4.1節で検索実験の概要について説明し、第4.2節で実験結果として確信度の閾値と検索効率の関係、各手法の最適閾値における検索効率を示し、その結果について考察する。

4.1 実験の概要

4.1.1 検索対象テキスト

曖昧検索手法を評価する場合、誤りを一定の割合で挿入したテキストを人工的に生成してこれを対象として検索実験を行なう場合がある [29]。この方法は認識対象テキストの誤り率を任意に設定できるため、OCR の認識率に対する曖昧検索手法の性能変化を詳細に調べることができるという点で有効である。しかし人工的に誤りデータを生成する方法はまだ確立されておらず、最も単純には1文字の置換・欠落・挿入誤りを誤り率に基づいて無作為に発生させることもあれば、まず類似文字テーブルを実際の OCR 出力に対して作成し、それを参照しながら誤り率に基づいて誤りを挿入することもあるなど様々である。疑似誤りテキスト生成時に課すいくつかの条件により、結果として疑似生成されたテキストは実際の OCR が出力するテキストよりも比較的扱い易いテキストになっている場合が多い。そこで本研究ではこのような疑似誤りテキストではなく、和英活字 OCR が実際に出力したテキストデータを用いて検索効率の評価実験を行ない、提案手法の有効性を示すことにした。

4.1.2 実験環境

和英文ともに検索対象テキストにはトレーニングセット及びテストセットの2種類を用意し、検索語としては検索対象テキストに含まれる文字列で、名詞のものを無作為に50抽出して用いた¹。ここでトレーニングセットとは、類似文字テーブル及び拡張類似文字テーブルの学習に用いたテキストで、テストセットとはそれに含まれないテキストである。拡張検索文字列の確信度は、手法に応じて $P(\text{CMR 法})$ 、 $P_E(\text{ECMR 法})$ 、 $P_B(\text{BMR 法})$ で表し、(文字列の確信度 \geq 閾値) の条件を満たすもののみを用いて検索が行なわれる。また本稿でいう検索効率とは全文文字列検索における再現率・適合率のことでそれぞれ検索洩れの少なさ及び検索ノイズの少なさを表し、式(4.1)及び式(4.2)を用いて計算される。その他の諸条件は以下に示す通りである。

1. OCR

和文:(株)リコーのIMAZONE(認識率98.3%)

英文:EES²論文誌にて使用のもの(認識率99.1%)

2. トレーニングセット

和文:情報処理学会論文誌の1994年No.1~No.5から抽出した約80KBのテキスト

¹検索語の具体例は、和文では“幾何図形”、“効率”、“アルゴリズム”など、英文では“estimation”、“flow”、“image”などである。

²ELSEVIER Electronic Subscriptions の略。ELSEVIER は電子出版の一環として、論文誌のページ画像、各ページ画像を OCR 認識した結果の未修正のテキストデータ、書誌情報の SGML テキストの3つを CDROM にまとめて購読者に配布している [30]。

表 4.1: 類似文字テーブル及び拡張類似文字テーブルのファイルサイズ (単位:Byte)

	置換	欠落	挿入	結合	分解
和文	7939	20	233	615	257
英文	1210	175	94	747	353

英文: *Artificial Intelligence* の 1995 年 8 月号～1996 年 5 月号から抽出した約 80KB のテキスト

3. テストセット

和文: 電子情報通信学会論文誌の 1994 年 Vol. J77-A No. 1～No. 3 から抽出した約 55KB のテキスト

英文: *Cognition* の 1995 年 9 月号～1996 年 6 月号から抽出した約 50KB のテキスト

4. bigram テーブル構築のためのデータ

和文: 情報分野の学術論文雑誌から得たテキストデータ約 500 万文字

英文: 情報分野の学術論文雑誌から得たテキストデータ約 250 万文字

また文字の接続確率が 0 になったものについては、 1.0×10^{-5} に底上げすることで補間 [31] を行なった。

$$\text{再現率} = \frac{\text{提案手法で検索された正解文字列数}}{\text{元のテキストで検索された文字列数}} \quad (4.1)$$

$$\text{適合率} = \frac{\text{提案手法で検索された正解文字列数}}{\text{提案手法で検索された全文字列数}} \quad (4.2)$$

また上記のトレーニングセットを基に作成した類似文字テーブル及び拡張類似文字テーブルの大きさは表 4.1 に示す通りである。これらのテーブルのファイルへの格納形式は基本的に、

格納形式 1 B^a (認識結果の 1 文字または 2 文字): A^m (元の 1 文字または 2 文字): $P(A^m|B^a)$

の形をしており、トレーニングセットにおいて認識誤りと無関係であった文字 C_x はファイルには記録されず、 $P(A_x|B_x) = 1$ として扱っている。よってある種類の誤りが多ければその種類の誤りのテーブルを格納するファイルも大きくなるため、これらのテーブルのファイルの大きさが OCR の認識性能の目安となる。日本語は 2Byte 文字で英語は 1Byte 文字であるから単純に両者のファイルサイズを比較することはできないが、表 4.1 が表 2.2、表 2.3 に示した日英活字 OCR の認識誤り特性をよく反映していることが分かる。また、類似 (置換)・欠落・挿入・結合・分解文字テーブルの内容の具体例を上記の格納形式 1 に従って表 4.2 にまとめる。表 4.2 から、“夫 U” や “未 U” といった文字列は“利”を認識 (分解誤り) したときにしかトレーニングセットには現れなかったことが分かる。

表 4.2: 類似文字テーブル及び拡張類似文字テーブルの内容の具体例

誤りの種類	テーブルの内容の具体例		
置換	間:間:0.146341	は:な:0.056112	(和文)
欠落	$V_{m,i}:0.073171$	$V_{m,i}:0.024390$	(英文)
挿入	$\cdot:V_i:0.103093$	$\cdot:V_i:0.500000$	(和文)
結合	n:ri:0.010029	n:rt:0.000393	(英文)
分解	夫 U:利:1.000000	未 U:利:1.000000	(和文)

一方文字の接続確率を求めるのに必要な bigram テーブルのファイルへの格納形式は基本的に、

格納形式 2 bigram(正しいテキストに出現する連続した 2 文字):頻度

の形をしているので、このファイルの大きさは異なり bigram 数に比例する。上記のデータから構築された bigram テーブルのファイルサイズは、和文で約 845KB、英文で約 20KB となっており、日本語の文字種の多さが端的に示されている。ただし和文はアルファベットなども含めて扱っているので、文字集合としては英文は和文の部分集合となっている。

4.2 実験結果と考察

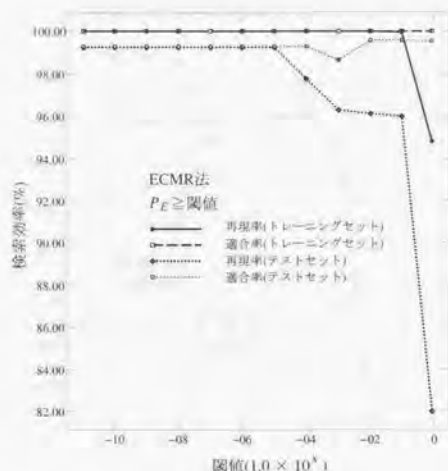
4.2.1 確信度の閾値と検索効率

まず文字列の確信度 (P, P_E, P_B) の閾値と検索効率の関係を求める実験を行なった。この両者の関係は基本的には、閾値が大きければ低再現率、高適合率となり、逆に閾値が小さければ高再現率、低適合率となるというもので、再現率と適合率の間にはトレードオフの関係が成り立つ。またこの傾向は提案手法の種類に依らないため、図 4.1、4.2 にそれぞれ和英文における ECMR 法の場合の確信度 P_E の閾値と検索効率の関係を示す。なおこれらの検索効率の値は和英文とも、50 の検索語で検索した結果の平均値である。また、英文においてはピリオドなどのデリミタを用いた単語の切り出しと大文字から小文字への変換(正規化)を前処理として行なっている。

これらの図からは、拡張検索文字列候補の確信度 P_E の閾値として適当なものを選ぶと、和英文を問わず適合率を殆ど下げずに高い再現率が得られることが分かる。また、和文曖昧検索では閾値の値に依らず全般的に適合率が高くなっており、検索語を拡張することにより検索ノイズが増える心配が殆どないことが分かる。これは、英文に比べて格段に文字種が多いという日本語の特徴を反映したものといえる。

4.2.2 最適閾値における検索効率

和文トレーニングセットとテストセットそれぞれの最適閾値における検索効率を表 4.3 及び表 4.4 に示す。同様に英文トレーニングセットとテストセットそれぞれの最適閾値における検索効率

図 4.1: P_E の閾値と検索効率 (和文)

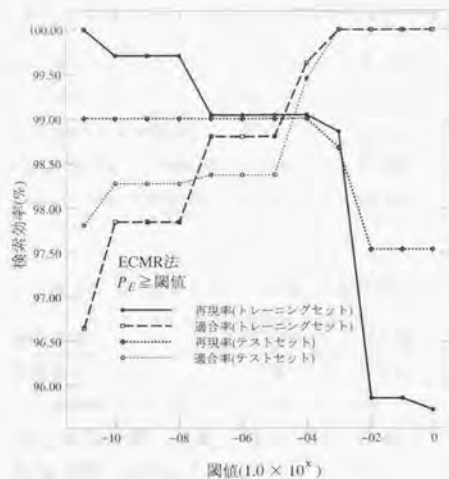
を表 4.5 及び表 4.6 に示す。

OCR の認識誤りを考慮せずに完全照合で検索すると、再現率は和文では 96.62%(表 4.3)、96.01%(表 4.4)、英文では 95.73%(表 4.5)、97.54%(表 4.6) であった。これに対して CMR 法を用いると、再現率が和文では 99.69%(表 4.3)、99.26%(表 4.4) と約 3% 改善され、英文においても 97.67%(表 4.5)、98.61%(表 4.6) と約 1~2% 改善される。また CMR 法を用いることで適合率が若干下がる場合があるが、和英文とも適合率は 99% 以上を実現している (表 4.4、4.5、4.6 参照)。

ECMR 法は、表 4.4 を除けば全て CMR 法よりも高い再現率を実現しており、適合率も CMR 法に劣らない。これは、拡張類似文字テーブル (ECMR 法) によって類似文字テーブルだけ (CMR 法) ではカバーできなかった認識誤りにも対処できたことを示している。特に英文では、ECMR 法

表 4.3: 和文トレーニングセットにおける検索効率

検索条件	再現率 (%)	適合率 (%)
完全照合	96.62	100.0
$P \geq 0.01$ (CMR 法)	99.69	100.0
$P_E \geq 0.01$ (ECMR 法)	100.0	100.0
$P_B \geq 0.01$ (BMR 法)	99.69	100.0

図 4.2: P_E の閾値と検索効率 (英文)

と CMR 法による再現率の差が和文に比べて大きく、表 2.2、2.3に示した日英両言語の活字 OCR の認識誤りの種類の違いを反映する結果となった。また表 4.4において CMR 法と比較して検索効率の差がみられないのは、再現率が 100%とならない原因が拡張類似文字テーブルにも含まれない認識誤りにあったからだ。具体的には次のような認識誤りが存在したためである。

1. 比較→上と藪

“比→上ヒ”という分解誤りが拡張類似文字テーブルに存在せず、“較→藪”という置換誤りも類似文字テーブルに存在しなかった。

2. 非線形→フ E 線形

表 4.4: 和文テストセットにおける検索効率

検索条件	再現率 (%)	適合率 (%)
完全照合	96.01	99.60
$P \geq 0.00001$ (CMR 法)	99.26	99.28
$P_E \geq 0.00001$ (ECMR 法)	99.26	99.28
$P_B \geq 0.01$ (BMR 法)	99.26	99.28

表 4.5: 英文トレーニングセットにおける検索効率

検索条件	再現率 (%)	適合率 (%)
完全照合	95.73	100.0
$P \geq 0.0001$ (CMR 法)	97.67	99.63
$P_E \geq 0.0001$ (ECMR 法)	99.05	99.63
$P_B \geq 0.0001$ (BMR 法)	97.67	99.63

表 4.6: 英文テストセットにおける検索効率

検索条件	再現率 (%)	適合率 (%)
完全照合	97.54	100.0
$P \geq 0.0001$ (CMR 法)	98.61	99.47
$P_E \geq 0.0001$ (ECMR 法)	99.01	99.47
$P_B \geq 0.001$ (BMR 法)	98.61	100.0

“非→ヲ E” という分解誤りが拡張類似文字テーブルに存在しなかった。

これらはどちらもテーブルを構築する際に用いるトレーニングセットに存在しなかった誤りであり、このような問題は一般にゼロ頻度問題 [32] として知られる。そしてこの問題は、トレーニングセットの量はどのくらいが適当かという学習量の問題や、ゼロ頻度となるトレーニングセットにおける未観測事象の確率をどのように推定するかという問題を示唆している。

BMR 法を CMR 法と比較すると、表 4.3、4.5 では差が見られないが、表 4.4、4.6 では最適閾値が BMR 法の方が大きく、表 4.6 では適合率が CMR 法よりも高くなっている。検索ノイズを削減したり検索コストの目安となる拡張検索文字列数を少なくしたりするには閾値はできるだけ高く設定できる方がよい。また表 4.6 における BMR 法と CMR 法の閾値及び適合率の差は、BMR 法が文字の接続確率を考慮することで検索されるべき文字列の確信度を引き上げることに成功したことを表している。その結果最適閾値を高く設定できるようになり、CMR 法よりも検索ノイズを抑えることができた。よって CMR 法の与える P よりも BMR 法の与える P_B の方がより得点として好ましいといえる。但し、統計という性質上常に BMR 法が CMR 法よりもすぐれた得点付けを行なっている訳ではなく、中には文字の接続確率を考慮することでかえって確信度が改悪され、検索効率が下がる場合もあることに注意する必要がある。またそのためにも検索対象となるテキストの分野にふさわしく、かつ充分な量の bigram 統計用の学習データを用いることが重要となる。

また BMR 法と ECMR 法の比較では、以下に示す利害得失が挙げられる。

- 達成できる再現率の点では ECMR 法の方が優れている (表 4.3、4.5、4.6 参照)。

これが 4 種類の文字の切り出し誤り (欠落・挿入・結合・分解誤り) それぞれに対応して検索文字列を生成する ECMR 法の長所である。

- BMR 法の方が最適閾値が高い (表 4.4、4.6 参照)。

閾値が高い方が検索ノイズを削減し易く、また拡張検索文字列も少なく済むので、 P_B の方が P_E よりも得点として好ましい場合もある。

4.3 まとめ

本章では、第 3 章で提案した 3 つの曖昧検索手法を検索効率の観点から評価する実験について説明し、実験結果とその考察を示した。第 4.2.2 節に示した実験結果は、ECMR 法によって和英文を問わず再現率・適合率ともに 99% 以上という高い値を実現できることを示している。しかし BMR 法と ECMR 法の比較において述べたような BMR 法の有効性を考慮すると、BMR 法で用いた文字の接続確率を ECMR 法でも利用できればさらに良い結果 (高い適合率) が期待できる。但しそのためには文字の接続確率を文字切り出し誤りも考慮に入れたものに拡張する必要があり、本論文ではその 1 つの統合方法を第 6 章において提案する。

第5章

英文曖昧検索

第3章で提案した曖昧検索手法はもともと和文を対象としたものであり、そのまま英文に適用すると和文ほどの検索性能が得られなかった。そこで本章では英文曖昧検索の課題について考察し、英文に適した曖昧検索手法の提案及び評価を行なう。

まず第5.1節で、第3章の提案手法をそのまま英文に適用した場合の課題について述べる。課題の1つは低適合率の問題で、第5.2節ではその改善策として英語の言語特性を考慮した前処理を導入する。また第5.3節では複合誤りの分割を試み、その情報を類似文字テーブル及び拡張類似文字テーブルに反映させることで、適合率が改善されることを示す。第5.4節ではもう1つの課題である検索コストについて、生成される拡張検索文字列数に基づいて説明する。この検索コストの問題については、第5.5節で拡張検索文字列中に含まれる認識誤りの数を制限するヒューリスティクスを、第5.6節では新たな拡張検索文字列の得点付けのアルゴリズムを提案してその削減効果を評価する。

5.1 英文曖昧検索における課題

英文曖昧検索においてデリミタによる単語切り出しや大文字の小文字への正規化といった前処理を行わずに、第3章で提案した曖昧検索のアルゴリズムをそのまま適用すると、和文曖昧検索ほどの検索効率には得られず、また和文曖昧検索とは比較にならないほど膨大な検索コストがかかることが判明した。このような英文曖昧検索特有の課題を具体的に以下にまとめる。

- 和文に比べて適合率がかなり低い

第4章において示した英文曖昧検索における検索効率の値は、和文のそれと比べて殆んど遜色ないが、これは次節で述べる前処理に負うところが大きい。

- 和文に比べて生成される拡張検索文字列数が圧倒的に多い

拡張検索文字列数は検索コストに比例すると考えることができるため、実用性を考えると無視できない問題である。

これらの問題の根幹には、日本語と比べて英語は文字種が少ないなどといった両言語の特質の差があり、本章では以下第5.2節で適合率の問題について、第5.4節で拡張検索文字列数の問題についてより詳しく検討する。

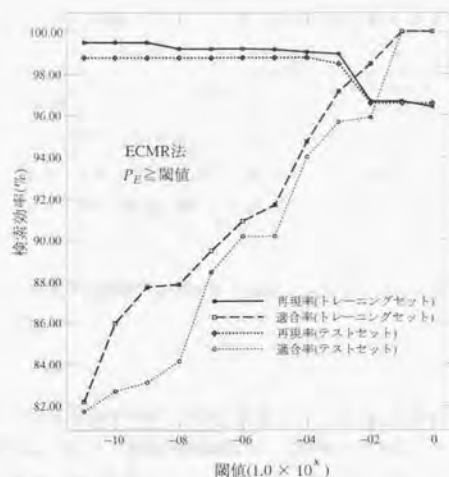
5.2 英文曖昧検索における前処理

5.2.1 低適合率とその原因

まず英文曖昧検索において前処理を行わずに ECMR 法のアルゴリズムを適用した場合、検索効率と確信度の閾値との関係は図 5.1 のようになった。和文曖昧検索の場合を示した図 4.1 と比較すると、閾値の値に関わりなく全般的に適合率が悪くなっており、最低の閾値 (1.0×10^{-11}) を適用した場合英文曖昧検索では約 82% にまで下がっている。これは最適閾値における検索効率についても当てはまり、前処理を行わないとこれらの値は英文トレーニングセットとテストセットそれぞれについて表 5.1、5.2 のようになり、和文のそれ (表 4.3、4.4 参照) との差は歴然としている。

そこで英文曖昧検索においてこのような適合率低下をもたらした原因を細かく調べると、以下に示す 3 種類の検索ノイズの影響が大きいことが分かった。

1. 英語のアルファベットの大文字と小文字を別の文字として扱うと、これらは形の似ているものが多いため OCR が間違い易い。(例: "system" という検索語に対して "System" が検索される。)
2. 英文では文字種が和文に比べて極端に少ないことなどから、検索語と被検索語の部分文字列が似ている場合が度々出現しそれが検索ノイズとなる。(例: "flow" という検索語に対して "allow" の部分文字列である "llow" が検索される。)

図 5.1: 英文における P_E の閾値と検索効率 (前処理なし)

3. 英文曖昧検索ではスペースを文字として扱っているため、2つの単語にまたがった文字列を1つの単語として検索してしまう。(例: “test” という検索語に対して “...incorporates these...” や “...estimates to...” などの部分文字列 “tes t” がスペースを挿入誤りと考えて検索される。)

5.2.2 前処理の導入

本研究では、前節で指摘した検索ノイズに対処するために、英文曖昧検索において次のような前処理を行なって適合率の改善を図った。

1. 大文字の小文字への正規化

表 5.1: 英文トレーニングセットにおける検索効率 (前処理なし)

検索条件	再現率 (%)	適合率 (%)
完全照合	96.23	100.0
$P \geq 0.00001$ (CMR 法)	97.94	95.50
$P_E \geq 0.00001$ (ECMR 法)	99.16	91.67
$P_B \geq 0.0001$ (BMR 法)	97.78	96.80

表 5.2: 英文テストセットにおける検索効率 (前処理なし)

検索条件	再現率 (%)	適合率 (%)
完全照合	96.57	100.0
$P \geq 0.0001$ (CMR 法)	98.55	94.53
$P_E \geq 0.0001$ (ECMR 法)	98.79	93.99
$P_B \geq 0.001$ (BMR 法)	98.55	95.53

形の似た大文字と小文字の問題に対処するために、大文字と小文字の区別をなくし使用する文字を小文字に限定する。

2. 単語の切り出し

誤って検索語と異なる単語の部分文字列を検索したり2つの単語にまたがって検索したりしてしまう問題については、単語の切り出しを行ない単語を対象とする検索を行なうことで対処する。そのためにまず単語をスペースやピリオド、カンマなど17種類のデリミタで区切られた文字列と定義し、このように定義した単語のみを検索対象とする。

この結果図5.1の通りであった閾値と検索効率の対応関係は、図4.2に示したように適合率が格段に改善された。また最適閾値における検索効率についてみても、表5.1、5.2と表4.5、4.6との比較から前処理による適合率の改善効果は明らかである。その結果、英文においても和文に劣らない適合率(99%以上)を達成した。

5.3 複合誤りの分割と曖昧検索への活用

本節では、第3.2.2節の認識誤りの分類ヒューリスティクスに当てはまらない誤りのうち、本研究で定義した5種類の誤りの複合誤りと考えるのが妥当なものについて考える。第3章で示した類似文字テーブル及び拡張類似文字テーブルではこのような複合誤りは無視されているが、複合誤りをその構成要素となる誤り、すなわち置換・欠落・挿入・結合・分解誤りのいずれかに分割することができれば、これらのテーブルに分割後の誤りの情報を反映させることができる。本節では、このような複合誤りの分割と複合誤り情報の検索効率への寄与について述べる。

5.3.1 複合誤りの分割方法

第3.2.2節の認識誤りの分類ヒューリスティクスによって分類できない認識誤り $\{\alpha^m, \beta^n\}$ では、 $(m, n \geq 2) \cap (m \neq n)$ が成り立つ。この場合考えられる全ての誤りの組合せを求めると、 m, n が大きくなると組合せが爆発的に増加する。しかも正しい組合せはただ1つでその他は全て間違っていると考え、全ての解釈を類似文字テーブル及び拡張類似文字テーブルに反映させればか

表 5.3: 抽出された認識誤り 1

$\{m, n\}$	$\{1, 1\}$	$\{1, 0\}$	$\{0, 1\}$	$\{2, 1\}$	$\{1, 2\}$
頻度	138	41	19	176	34

なりのノイズがのることになる。このようなテーブルのノイズを防ぐためには、考えられる組合せをなるべく減らせばよく、本節では以下のヒューリスティクスを 1 から順に適用することで複合誤りをなるべく一意に解釈することを提案する。

複合誤りの分割ヒューリスティクス

1. α^m 及び β^n 中に同じ文字が存在し、かつその文字の文字列中の相対位置が近い場合は、その文字は正しく認識されていると考えてその文字で α^m 及び β^n をそれぞれ分割する。分割の結果 $\{\alpha^{m_i}, \beta^{n_i}\}_{i=1}^k$ が得られた場合¹、各 $\{\alpha^{m_i}, \beta^{n_i}\}$ に第 3.2.2 節の認識誤りの分類ヒューリスティクスを適用して一意な解釈を試み、それが不可能な場合は次の 2 以降の処理に移る。
2. α^m 及び β^n 中に認識誤りの分類ヒューリスティクスによって得られた既知の置換・結合・分解誤りが存在し、かつその誤りを構成する双方の文字 (列) の文字列中の相対位置が近い場合は、その誤りが存在すると考えてそれらの文字 (列) で α^m 及び β^n をそれぞれ分割する。分割後は 1 と同様に、各 $\{\alpha^{m_i}, \beta^{n_i}\}$ に認識誤りの分類ヒューリスティクスを適用して一意な解釈を試み、それが不可能な場合は次の 3 または 4 の処理に移る。
3. $m > n$ の場合、 $\alpha^m \rightarrow \beta^n$ は置換、結合の 2 種類の誤りのいずれかによって構成される複合誤りと解釈する。さらに複合誤りを構成する要素誤りは、互いの文字列中の相対位置が近いもの同士で誤りを構成するものとする。 $m = 2n$ の場合は、 n 個の結合誤りと一意に解釈する。
4. $m < n$ の場合、 $\alpha^m \rightarrow \beta^n$ は置換、分解の 2 種類の誤りのいずれかによって構成される複合誤りと解釈する。さらに複合誤りを構成する要素誤りは、互いの文字列中の相対位置が近いもの同士で誤りを構成するものとする。 $n = 2m$ の場合は、 m 個の分解誤りと一意に解釈する。

このヒューリスティクスの 1 及び 2 において、複数の分割方法が考えられる場合は分割を行わず、次の 3 または 4 の処理に移る。このヒューリスティクスの 3 または 4 を適用する場合は、複合誤りの解釈が 1 つに定まらない場合がある。

5.3.2 複合誤りの分割実験

¹ $(k-1) + \sum_{i=1}^k m_i = m, (k-1) + \sum_{i=1}^k n_i = n$.

表 5.4: 抽出された認識誤り 2

$\{m, n\}$	$\{m, 0\}$	$\{0, n\}$	$\{m, m\}$	$\{m, n\} m \neq n$
頻度	8	13	16	33

第 4.1.2 節に示した英文トレーニングセット²から抽出された認識誤り $\{\alpha^m, \beta^n\}$ を、 $\{m, n\}$ に基づいて分類すると表 5.3、5.4 のようになる。但し、表 5.4 における m と n については、 $m, n \geq 2$ である。

表 5.3、5.4 から、合計 478 の認識誤りのうち、33 の複合誤りを除いた 445 の誤りは第 3.2.2 節の認識誤りの分類ヒューリスティクスによって直ちに一意な解釈が可能であることが分かる。また、33 の複合誤りのうち 27 は、複合誤りの分割ヒューリスティクスによって要素誤りを一意に定めることができ、複合誤りからも確度の高い誤りパターンを抽出できることが示された。

5.3.3 検索効率への寄与

一意に分割された複合誤りの情報を類似文字テーブル及び拡張類似文字テーブルに反映させ、第 4.1.2 節に示した英文トレーニングセット及びテストセットに対して検索実験を行なった。この実験ではトレーニングセットにおける最適閾値での検索効率には殆んど変化がみられなかったが、テストセットにおける実験では ECMR 法の実現する適合率が表 4.6 に示した 99.47% から 99.80% に改善された。また ECMR 法の示す検索効率は英文トレーニングセット及びテストセットに対してそれぞれ図 5.2、5.3 のように変化した。

これらの図から、複合誤りを考慮したことによって主に適合率が影響を受けていることが分かる。すなわち閾値の値が小さい場合 (再現率が高い場合) は、複合誤りによって得られた認識誤り情報によって、1 つの検索語から拡張される検索文字列候補が増えたことによって適合率が下がっている。しかし再現率・適合率ともに 99% を超える最適閾値付近では逆に適合率は上がっており、実際に検索される文字列についてみると複合誤りを考慮する価値があることが分かる。

5.4 提案手法と拡張検索文字列

5.4.1 拡張検索文字列数

提案手法では拡張検索文字列数が検索語長の指数のオーダーで増加するため、拡張検索文字列数の問題は提案手法の本質的な問題でもある。しかしこれが英文曖昧検索においてのみ問題となるのは両言語の特徴が異なるためで、具体的には和文では類似文字テーブルが非常にスパースになるのに対して、英文ではそうはならず結果として文字当りの認識誤り候補が多くなるためである。

²正規化 (全ての大文字を小文字に変換) したものを利用したため、表 2.3 の頻度統計と若干異なる。

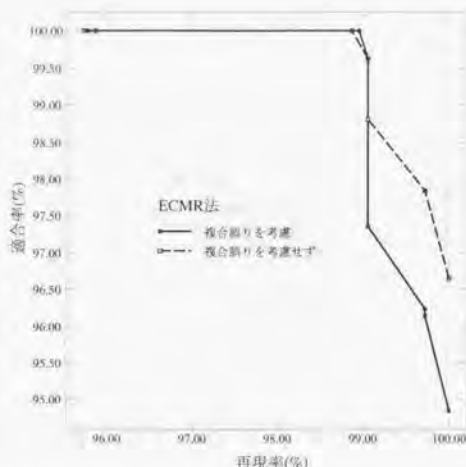


図 5.2: 複合誤り情報の検索効率への寄与 (英文トレーニングセット)

例えば和文トレーニングセットにおける曖昧検索では、類似文字テーブルを参照すると置換候補は自身も含めて平均 1.26 文字あり、置換誤りのみを想定する CMR 法では長さ n の検索語から平均 1.26^n の拡張検索文字列の候補が生成される。ところが英文トレーニングセットにおける曖昧検索では、置換候補は平均 5.50 文字存在するため、置換誤りのみを想定しても平均 5.50^n の拡張検索文字列の候補が生成される。仮に $n = 10$ とすると、和文では $1.26^{10} \approx 10$ しか文字列候補は生成されないが、英文では $5.50^{10} \approx 25,000,000$ の候補が生成されることになる。実際には拡張検索文字列の妥当性を確信度に基づいて評価するため、拡張検索文字列数はこれより減少するが、置換誤りに加えて欠落、挿入、結合、分解誤りを考慮する ECMR 法を用いて英文において検索語拡張を行なうと、生成文字列数は図 5.4 のようになる。

図 5.4 から、英文 ECMR 法において最適閾値である 0.0001 を用いて拡張検索文字列を生成すると、その数は英文トレーニングセットにおいて約 170,000、テストセットにおいて約 93,000 となることが分かる。

5.4.2 確信度に基づくランキング検索

確信度を閾値によって評価して拡張検索文字列を決定すると、拡張検索文字列数と検索効率の因果関係を細かく把握できない。そこで本節では拡張検索文字列を ECMR 法の定める確信度 P_E に基づいてランキングを行ない、その上位 n 個の拡張検索文字列を用いて英文トレーニングセット及びテストセットに対する検索を行なって、検索効率の変化をみた (図 5.5)。

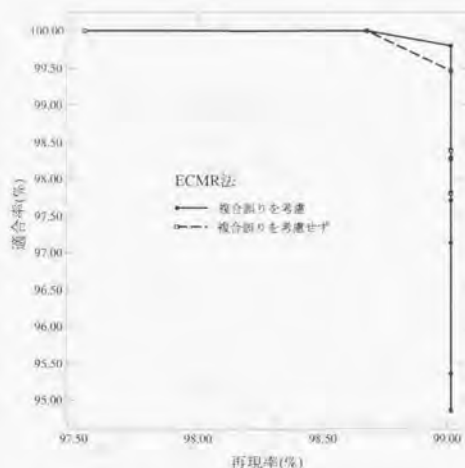


図 5.3: 複合誤り情報の検索効率への寄与 (英文テストセット)

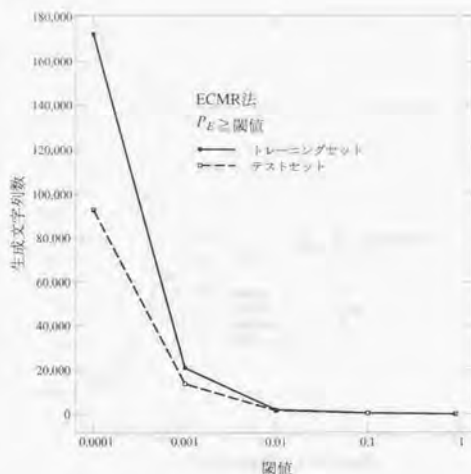
例えば確信度の上位 10,000 個の拡張検索文字列を用いて検索を行なっても、トレーニングセットで 97.67%、テストセットで 98.68% の再現率しか達成できず、これは表 4.5 及び表 4.6 に比べてかなり低い値であることが分かる。またトレーニングセットにおいては、10,000 個の拡張検索文字列を用いた検索でも 99.14% という高い適合率を示しているのに対し、テストセットにおいては拡張検索文字列数が増えるとともに適合率が急激に下がっている。例えばテストセットにおいて 1,000 個の拡張検索文字列を用いて曖昧検索すると 97.97% まで適合率が下がる。

5.5 拡張検索文字列削減のためのヒューリスティクス

5.5.1 提案する 2 つのヒューリスティクス

第 5.4.1 節で示した約 170,000 という拡張検索文字列数は、検索コストの観点からは非現実的である。そこで本節では確信度とは全く独立にヒューリスティクスを導入し、最も高い再現率を実現している ECMR 法の確信度による評価と組み合わせて拡張検索文字列数の削減を図る。但し拡張検索文字列を不用意に削減すると、現在 ECMR 法が実現している検索効率、特に再現率を劣化させる恐れがある。そこで具体的には以下の 2 つのヒューリスティクスを確信度による評価の後に適用して、検索効率及び拡張検索文字列数の変化をみる。この 2 つのヒューリスティクスはどちらも、拡張検索文字列中の誤りの数に着目し、その数がある値以下に抑えるというものである。

- ヒューリスティクス 1

図 5.4: 英文における P_E の閾値と拡張検索文字列数 (ECMR 法)

文字列長に関わりなく 1 検索語につき x 以下の誤り³を含むような文字列に絞り込む。

● ヒューリスティクス 2

1 検索語につき検索語長に比例した数以下の誤りを含むような文字列に絞り込む。つまり文字列長 m の検索語から生成される拡張検索文字列に含まれる誤りの数の最大値 Max を、

$$Max = \left\lceil \frac{m}{y} \right\rceil. \quad (5.1)$$

とする。ここで y は含まれる誤りの割合を決めるパラメータで、 $(n-1) \cdot y < m \leq n \cdot y$ ならば n 個以下の誤りを含む文字列に絞り込む。

5.5.2 ヒューリスティクス 1 の適用結果

ヒューリスティクス 1 において $x=1, 2, 3$ と変化させながら英文トレーニングセット及びテストセットを検索した結果をそれぞれ図 5.6、5.7 に示す。

これらの図では、閾値が最適値である 0.0001 のとき再現率は最高に、適合率は最低になっている。また図中に再現率、適合率とのみ記してあるものは、ヒューリスティクスを用いなかった場合の結果である。図 5.6 は、トレーニングセットにおいては、 $x=1$ のとき再現率が 99.05% から

³置換、欠落、挿入、分解誤りはそれぞれ 1 文字の誤りを 1 つと数えるが、結合誤りは 2 文字が誤って 1 文字と認識されるのでその 2 文字で 1 つと数える。

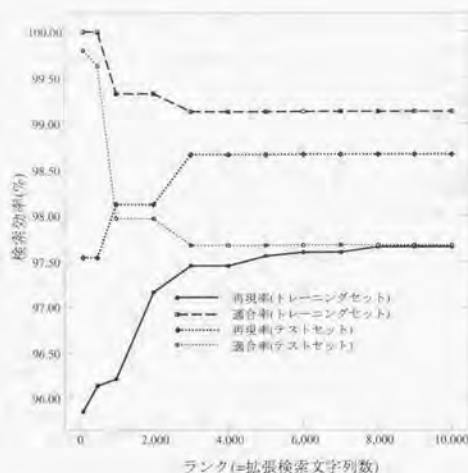


図 5.5: 英文 ECMR 法の検索効率と拡張検索文字列数 (確信度に基づくランキング)

99.01%へと約 0.04%下がるものの、生成文字列数を 56.24 にまで絞り込めることを示している。この再現率低下の原因は以下の認識誤りがトレーニングセットに存在したためである。

- “problem” → “pmblen”

拡張検索文字列の候補である “pmblen” は、“ro” → “m” という結合誤りと “m” → “n” という分解誤りの 2 つの認識誤りを含む。そのため確信度は 0.002517 と閾値 0.0001 を超えていたが、 $x=1$ であるため検索文字列として生成されず、結果として検索できなかった。

一方図 5.7 は、テストセットにおいて検索効率を下げることなしに $x=1$ のとき生成文字列数を 52.66 まで絞り込めることを示している。

5.5.3 ヒューリスティクス 2 の適用結果

ヒューリスティクス 2 において $y=1, 2, \dots, 15$ と変化させながら英文トレーニングセット及びテストセットを検索した結果をそれぞれ図 5.8、5.9 に示す。

これらの図では、 $y=5, 10, 15$ のときの結果とヒューリスティクスを用いなかった場合の結果を示しており、やはり閾値が最適値である 0.0001 のとき再現率は最高に、適合率は最低になっている。トレーニングセットにおける実験では、 $y \geq 14$ のとき再現率を 99.05% から 99.01% へと約 0.04% 下げるものの、生成文字列数を 56.24 にまで絞り込めた。一方テストセットにおいては、検索効率を下げることなしに $y \geq 13$ のとき生成文字列数を 52.66 まで絞り込めた。またトレーニン

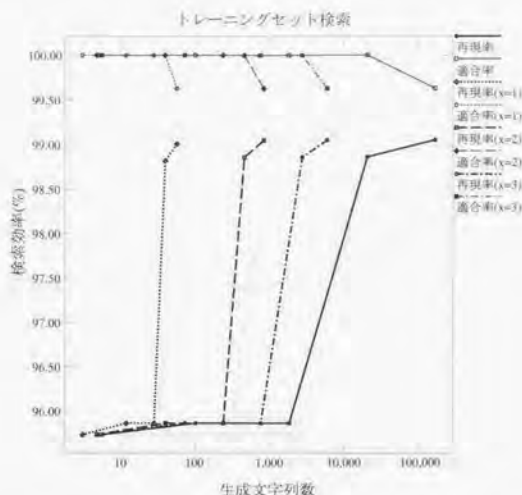


図 5.6: 英文トレーニングセットにおけるヒューリスティクス 1 の効果 (ECMR 法)

グセットにおける $y \geq 14$ 、テストセットにおける $y \geq 13$ という条件は、ヒューリスティクス 1 において $x=1$ とする場合と等価である。

5.5.4 考察

2つのヒューリスティクスによって英文トレーニングセット及びテストセットに対する検索において実現されている検索効率及び拡張検索文字列数を、ヒューリスティクスごとにまとめると表 5.5、5.6となる。

ヒューリスティクス 2 は、検索語が長いほど誤りの混入する確率が高く結果的に誤りの数が増えることを予想して導入したが、本節の実験結果からは検索語長に関わりなく高々1つの誤りを含む

表 5.5: ヒューリスティクス 1 の検索性能

検索対象テキスト	x	再現率 (%)	適合率 (%)	生成文字列数
トレーニングセット	1	99.01	99.63	56.24
トレーニングセット	2	99.05	99.63	845.1
テストセット	1	99.01	99.47	52.66

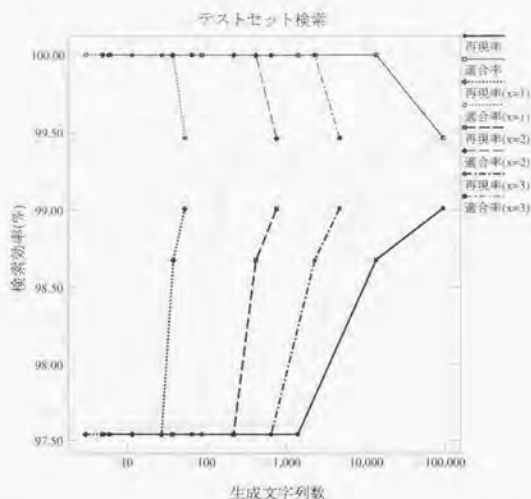


図 5.7: 英文テストセットにおけるヒューリスティクス 1 の効果 (ECMR 法)

ような検索文字列に制限しても、再現率は従来の ECMR 法と比べてわずかしき低下せず 99% 以上を保つことが示された (表 4.5, 4.6 参照)。これはつまりヒューリスティクス 1 を用いて $x=1$ とすれば、英文曖昧検索において検索効率については十分な結果が得られることを示す。一方このヒューリスティクスの適用による拡張検索文字列数の削減効果は、トレーニングセットにおいて約 170,000 から約 56 に、テストセットにおいて約 93,000 から約 53 に削減されていることから高いといえる。

またトレーニングセットにおいて再現率を劣化させないように検索を行なうには、ヒューリスティクス 1 及び 2 においてそれぞれ $x \geq 2$, $y \leq 6$ という条件が必要となる。また $x=2$ 及び $y=6$ のときの生成文字列数はそれぞれ約 845、約 2,009 である。よって一般的には x 及び y の値

表 5.6: ヒューリスティクス 2 の検索性能

検索対象テキスト	y	再現率 (%)	適合率 (%)	生成文字列数
トレーニングセット	≥ 14	99.01	99.63	56.24
トレーニングセット	6	99.05	99.63	2008.7
テストセット	≥ 13	99.01	99.47	52.66

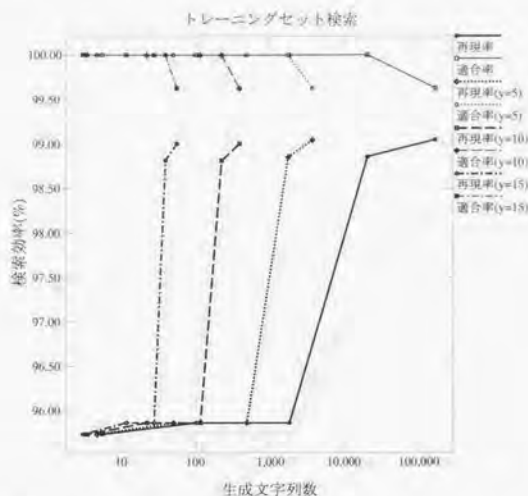


図 5.8: 英文トレーニングセットにおけるヒューリスティクス 2 の効果 (ECMR 法)

はトレーニングセットにおいて本節で示したような実験結果から求める必要があるが、長い検索語において 3 つ以上の誤りを含む拡張検索文字列を生成するのは検索コストの観点から現実的ではなく、英文曖昧検索ではヒューリスティクス 1 において $x=1$ または $x=2$ を用いるのが適当といえる。

5.6 逆確信度に基づくランキング検索

本節では ECMR 法において確信度とは異なる拡張検索文字列の得点を提案し、その得点に基づいて各検索文字列をランキングすることで、少ない拡張検索文字列で高い検索効率が得られることを示す。

5.6.1 文字の逆確信度

第 3.3.1 節で定義した記号とベイズの定理を用いて、文字の逆確信度を以下のように定義する。

$$P(B_y|A_x) = \frac{P(B_y)P(A_x|B_y)}{\sum_{z=1}^n P(B_z)P(A_x|B_z)} \quad (5.2)$$

つまり逆確信度 $P(B_y|A_x)$ とは、任意の元の正しい文字 A_x が OCR によって B_y と認識される確率である。そして逆確信度もまた確信度と同様に類似文字テーブルに蓄える。

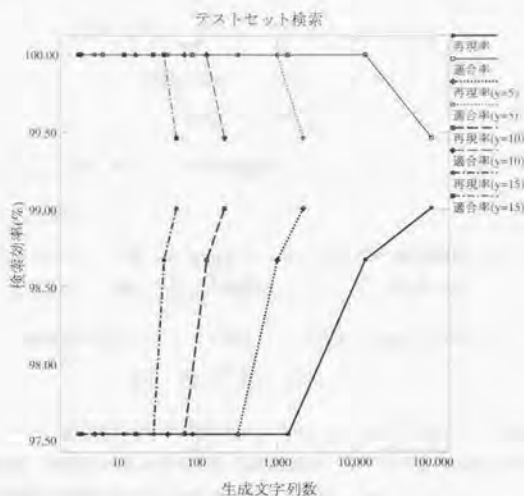


図 5.9: 英文テストセットにおけるヒューリスティクス 2 の効果 (ECMR 法)

一方、図 3.6、3.7 に示した拡張類似文字テーブルには、欠落・挿入・結合・分解誤りの逆確信度である $P(V_m|A_x)$ 、 $P(B_y|V_i)$ 、 $P(B_y|a_x)$ 、 $P(b_y|A_x)$ を確信度と同様にそれぞれ蓄える⁴。

5.6.2 文字列の逆確信度

文字列の逆確信度 $P(B^{012...n}|A^{012...n})$ を式 (3.5) と同様に以下のように定義する。

$$P(B^{012...n}|A^{012...n}) = P(B^0|A^0)P(B^1|A^1)...P(B^n|A^n). \quad (5.3)$$

これは、置換誤りのみを想定した場合に、任意の文字列 $A^{012...n}$ が $B^{012...n}$ と認識される確率である。

置換以外の欠落・挿入・結合・分解誤りを含む場合も、文字列の逆確信度の計算は確信度の場合 (図 3.9 参照) と同様の方法で行なう。すなわち欠落・挿入・結合・分解文字の部分の逆確信度 RP_m 、 RP_i 、 RP_c 、 RP_d をそれぞれ求めて、後はこれらを置換誤りの逆確信度と同様に扱い、式 (5.3) により文字列の逆確信度を求める。以下に欠落・挿入・結合・分解誤りを含む場合の逆確信度の計算方法について、それぞれ具体例を挙げながら説明する。

• 欠落誤りを含む場合

⁴ 記号の定義については第 3.3.3 節を参照のこと。

例えば検索語 “3-D” に対して “3D” という文字列が認識結果として考えられる場合、“3-D” が “3D” に対応する逆確信度を以下のように計算する。

$$\begin{aligned} P(3D|3-D) &= P(3|3)RP_mP(D|D), \\ RP_m &= P(V_m|-). \end{aligned} \quad (5.4)$$

このとき RP_m は欠落文字テーブルから得る。

- 挿入誤りを含む場合

例えば検索語 “object” に対して “object” という文字列が認識結果として考えられる場合、“object” が “object” に対応する逆確信度を以下のように計算する。

$$\begin{aligned} P(\text{object}|\text{object}) &= P(o|o)P(b|b)RP_iP(j|j)P(e|e)P(c|c)P(t|t), \\ RP_i &= P(,|V_i) \cdot P_{ins}. \end{aligned} \quad (5.5)$$

式 (5.5) において $P(,|V_i)$ は挿入文字テーブルから得られる。また P_{ins} は挿入誤りの起こる確率で、英文ではトレーニングセットの認識結果のテキストが 80,884 文字であり、また表 2.2 から挿入誤りの数が 60 であるので、次式で推定する。

$$P_{ins} = \frac{60}{80,884} \doteq 0.00074. \quad (5.6)$$

実際には元のテキストにおける文字 V_i は仮想的なもので観測できないため、 P_{ins} を考慮する。

- 結合誤りを含む場合

例えば検索語 “ring” に対して “nng” という文字列が認識結果として考えられる場合、“ring” が “nng” に対応する逆確信度を以下のように計算する。

$$\begin{aligned} P(\text{nng}|\text{ring}) &= RP_cP(n|n)P(g|g), \\ RP_c &= P(n|ri). \end{aligned} \quad (5.7)$$

RP_c は結合文字テーブルから直接得られる。

- 分解誤りを含む場合

例えば検索語 “image” に対して “irnage” という文字列が認識結果として考えられる場合、“image” が “irnage” に対応する逆確信度を以下のように計算する。

$$\begin{aligned} P(\text{irnage}|\text{image}) &= P(i|i)RP_dP(a|a)P(g|g)P(e|e), \\ RP_d &= P(rn|m). \end{aligned} \quad (5.8)$$

RP_d もまた分解文字テーブルから直接得られる。

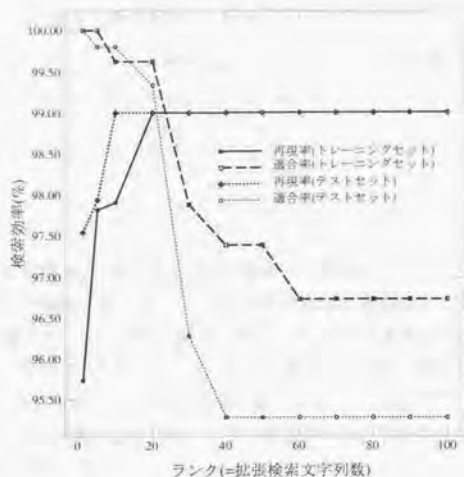


図 5.10: 英文 ECMR 法の検索効率と拡張検索文字列数 (逆確信度に基づくランキング)

5.6.3 本手法の検索性能

第 5.6.2 節で定義した逆確信度に基づいて拡張検索文字列のランキングを行ない、上位 n 個の拡張検索文字列を用いて英文トレーニングセット及びテストセットを検索した際の検索効率を図 5.10 に示す。また、この検索実験において検索効率と拡張検索文字列数の双方を総合的にみて最適と判断したそれぞれの値を表 5.7 にまとめる。

図 5.10 を ECMR 法において確信度に基づいてランキングを行なった場合 (図 5.5) と比較すると、本手法が極めて少ない拡張検索文字列で高い検索効率を実現していることが分かる。逆確信度に基づくランキングを行なった場合の特徴は、拡張検索文字列数が比較的少ないときに、拡張検索文字列数の増加に伴う再現率の上昇と適合率の低下の度合いが大きいことで、結果的には適合率の深刻な低下が起こる前に、少ない拡張検索文字列で高い再現率を得ることに成功している。また表 5.7 を表 5.5、5.6 と比べると、本手法がトレーニングセットにおいて上位 20 個、テストセットにおいて上位 9 個の拡張検索文字列によって実現する検索効率の値が、第 5.5 節で提案したヒューリスティクスを用いた手法とトレーニングセットにおいて同等、テストセットにおいては適合率の点でより優れていることが分かる。そしてこの拡張検索文字列数は、ヒューリスティクスを用いた手法と比較すると、トレーニングセットにおいて約 $1/3$ 、テストセットにおいては約 $1/5$ である。よって完全照合の場合の 20 倍以下の検索コストで 99% 以上の検索効率を実現している本手法は、ヒューリスティクスを用いた手法よりもさらに英文曖昧検索に適した有効な手法といえる。

表 5.7: 検索効率と拡張検索文字列数の最適値

検索対象テキスト	再現率 (%)	適合率 (%)	拡張検索文字列数
トレーニングセット	99.01	99.63	20
テストセット	99.01	99.80	9

5.7 まとめ

本章では、第4章の提案手法を英文にそのまま適用した場合に起こる低適合率、膨大な拡張検索文字列数という2つの課題を取り上げ、それぞれの改善策を複数提案した。

まず低適合率の問題に対しては、第5.2節で大文字の小文字への変換(正規化)、デリミタによる単語の切り出しという前処理を行なうことで、和文と同様に高い検索効率が得られることを示した。また第5.3節では当初考慮していなかった複合誤りをヒューリスティクスに基づいて分割し、その要素となる誤りの情報を類似文字テーブル及び拡張類似文字テーブルに反映させることで適合率が改善されることを示した。

一方数十万生成されていた拡張検索文字列数の問題に対しては、英文曖昧検索において最も良い検索効率を示したECMR法を改良する形で2つの曖昧検索手法を新たに提案し、検索効率及び拡張検索文字列数の両方の観点から評価した。1つは第5.5節で提案した拡張検索文字列中の認識誤りの数を制限するというヒューリスティクスを導入する手法で、ECMR法の実現していた検索効率を殆んど下げることなく拡張検索文字列数を50程度にまで削減できることを示した。またもう1つは第5.6節で提案した、拡張検索文字列の得点を逆確信度によって求め、逆確信度の値に基づいてランキングを行ない上位 n 個を拡張検索文字列として検索を行なう手法である。この手法は第5.5節のヒューリスティクスを用いる手法に勝るとも劣らない検索効率を、20以下の拡張検索文字列によって実現できることを示した。

第 6 章

HMM に基づいた英文曖昧検索手法

本章では、認識前の元の文字及び 2 文字を状態とし、状態に遷移する際にその認識結果を出力するような HMM (Hidden Markov Model) に基づいた曖昧検索手法を提案し、英文における検索性能を評価する。

6.1 本手法の狙い

第3章で提案したCMR法、ECMR法、BMR法の英文曖昧検索における特徴の1つは、第4章で述べたように拡張検索文字列が多くなることである。これは英語では日本語と比べて1文字当たりの認識誤り候補が多いためであるが、この拡張検索文字列の増加は検索コストの増加を意味するだけでなく、第7.4節で述べるように適合率の低下とも密接に関連している。よって英文曖昧検索では有効な拡張検索文字列をなるべく少なく生成する手法が重要となる。

そこで本章では、このような英文曖昧検索の特徴を考慮しても十分有効と考えられる手法として、認識前の元の文字及び2文字を状態とし、状態に遷移する際にその認識結果を出力するようなHMMに基づいた手法を提案する。本手法は文字の接続確率を状態遷移確率で、文字の誤り確率をシンボル出力確率で表しており、既に提案したECMR法とBMR法の1つの統合方法ともいえる。すなわち、文字切り出しの誤りを含む欠落・挿入・結合・分解誤りと文字bigramに基づく文字の接続確率を同時に扱っている。そのため、ECMR法の高い再現率とBMR法の高い適合率及び少ない拡張検索文字列という特徴を得られることが期待できる。

6.2 HMMの定式化

HMMはシンボル(ラベル)系列を出力するマルコフモデルであり、 N 個の状態 s_1, s_2, \dots, s_N をもち、一定周期ごとに状態を次々に遷移すると共に、その遷移の際に、シンボルを1つずつ出力する。次にどの状態に遷移するか、またその際にどのシンボルを出力するかは、それぞれ「遷移確率」、「出力確率」によって確率的に決められている。またシンボル出力を、状態遷移にではなく遷移前の状態や遷移後の状態に対応づける定式化もある。通常のマルコフモデルと異なる点は、出力シンボル系列は観測できるが、状態そのものは直接観測できないという点にあり、その意味で“Hidden” Markov Model(HMM)と呼ばれる[33, 34, 35, 36, 37, 38]。

HMMの簡単な例を図6.1に示す。このHMMは、3つの状態で構成され、2種類のシンボル a と b のみからなるシンボル系列を出力する。例えば s_1 からは、 $p_{s_1 s_1}$ の確率で s_1 自体に遷移し、その際にシンボル a, b をそれぞれ $q_{s_1 s_1}(a), q_{s_1 s_1}(b)$ の確率で出力する。

この例に示しているようにHMMは、次のパラメータで定義される。

- 状態数 N : 各状態はそれぞれ s_1, s_2, \dots, s_N と表す。
- シンボル数 K : 各シンボルはそれぞれ $sym_1, sym_2, \dots, sym_K$ と表す。
- 遷移確率 $p_{s_i s_j}$: 状態 s_i から s_j に遷移する確率。

$$\sum_i p_{s_i s_j} = 1, \text{ or} \quad (6.1)$$

$$\sum_j p_{s_i s_j} = 1. \quad (6.2)$$

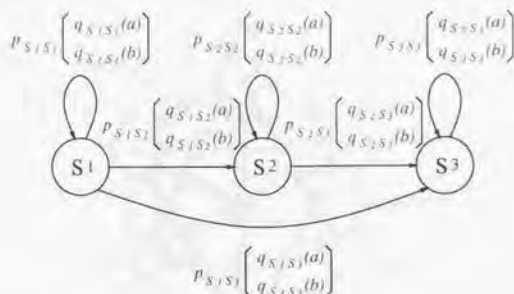


図 6.1: 簡単な HMM の例

- 出力確率 $q_{s_i s_j}(sym_k)$: 状態 s_i から s_j への遷移の際にシンボル k を出力する確率。

$$\sum_k q_{s_i s_j}(sym_k) = 1. \quad (6.3)$$

- 初期状態確率 π_{s_i} : 初期状態が s_i である確率。

$$\sum_i \pi_{s_i} = 1. \quad (6.4)$$

状態系列の最初の状態を s_1 と決めてしまうことも多い。その場合は、

$$\pi_{s_i} = \begin{cases} 1 & (i=1). \\ 0 & (i \neq 1). \end{cases} \quad (6.5)$$

6.3 HMM に基づいた曖昧検索手法の概要

本提案手法は、トレーニングセットに基づいて作成された HMM を用いて 1 つの検索語 α を複数の検索文字列 β に拡張する。このとき、拡張検索文字列の妥当性を表す尺度として作成した HMM を元に確率 $P(\beta|\alpha)$ を求め、この確率に基づいてランキングを行なう。検索は、この確率の大きい順に上位 n 個を拡張検索文字列として生成して行なう。

6.3.1 作成する HMM

まず図 6.2 に示すような HMM を第 4.1.2 節に示した英文トレーニングセットをもとに作成する。図 6.2 は、説明のため文字として “a”、“b”、“c”、“SP” (スペース) の 4 つのみを扱った例である。作成する HMM はこのように、状態として認識前の誤りのない 1 文字、2 文字、及び挿入誤りに対応する仮想的な文字 V_i という状態をもつ。そして、文字が正しく認識されている場合及び置換・欠落・分解誤りの場合は 1 文字の状態に遷移し、挿入誤りの場合は V_i という状態に遷移し、結合

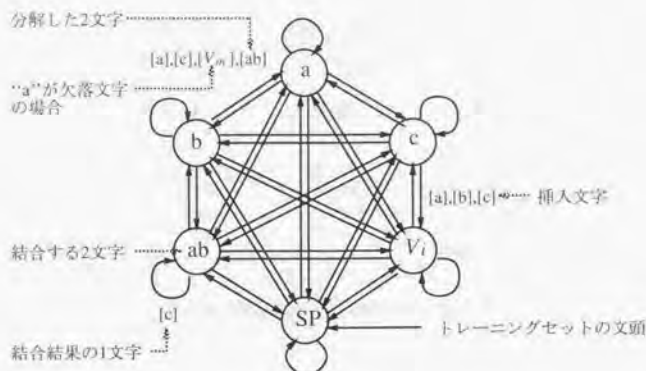


図 6.2: 作成する HMM の例

誤りの場合はその結合する 2 文字の状態へと遷移する。同時に遷移先の状態を表す文字を OCR が認識した結果の文字をシンボルとして出力確率に基づいて出力する。このとき、文字が正しく認識されている場合及び置換・挿入・結合誤りの場合は認識結果としての 1 文字を、欠落誤りの場合は欠落誤りに対応する仮想的な文字 V_m を、分解誤りの場合はその分解した結果の 2 文字をシンボルとして出力する。よって本手法は、文字切り出しの誤りに対応するとともに bigram に基づいた文字の接続確率も考慮したものとなっている。

本研究では、OCR は文字を正しく認識するか先に定義した 5 種類の認識誤りを起こすかのいずれかの動作をすると定めている。よってこの OCR の認識動作を本節の HMM で表現すると図 6.3 にまとめられる。この図の例はそれぞれ、“a” が正しく“a”と認識される、“a”が“e”と認識される(置換誤り)、“a”が欠落する(欠落誤り)、“a”が挿入される(挿入誤り)、“ab”が“c”と認識される(結合誤り)、“a”が“bc”と認識される(分解誤り)場合を示している。

このような HMM を作成するためにまず英文トレーニングセットに対して、第 3.2.2 節の認識誤りの分類ヒューリスティクス及び第 5.3.1 節の複合誤りの分割ヒューリスティクスを用いて認識誤りを抽出・分類し、図 6.4 のような認識される元の文字とその認識結果の対応関係を示したテキストを作成した。

英文トレーニングセットの文頭はスペースという文字から始まることとして、図 6.4 のテキストを入力として状態間の遷移及び遷移の際に出力するシンボルの出力頻度を数え上げることでその確率を求める。このとき一意に解釈できない複合誤りは、複合誤りの分割ヒューリスティクスに従って複数の解釈を許した。そしてこの解釈が x 通りのとき、それぞれの解釈に現れる要素誤りが $1/x$ の頻度で現れることとした。具体例を挙げると、“th” → “fh” というような分割できない

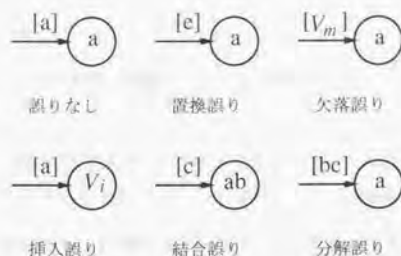


図 6.3: 認識動作の HMM における表現

[正しい文字]認識結果]

[tɪ][hɪ][li][sls][l][il1][sls][l][alo][l][sls][ale][mɪrɪ][pɪp][li][elo][.l]

図 6.4: HMM 作成のためのテキスト

い複合誤りに、第 5.3.1 節に示した複合誤りの分割ヒューリスティクスの 4 を適用して、[tɪ][hɪ][fɪ]、[tɪ][fɪ][hɪ] の 2 通りに解釈する。そしてこの解釈の中に現れる [tɪ][fɪ]、[hɪ][fɪ] という認識がそれぞれ 1/2 の頻度で現れるものとしてその頻度を考慮した。

このようにして求めた状態 s_i から状態 s_j への遷移回数を $C(s_i \rightarrow s_j)$ として、状態 s_i から状態 s_j への遷移確率 $p_{s_i s_j}$ を次式で定義する。

$$p_{s_i s_j} = \frac{C(s_i \rightarrow s_j)}{\sum_j C(s_i \rightarrow s_j)}. \quad (6.6)$$

よって $\sum_j p_{s_i s_j} = 1$ 、すなわち状態 s_j に入ってくる遷移確率の和が 1 となるように定めている。一方状態 s_i から状態 s_j への遷移の際にシンボル sym_k を出力する確率 $q_{s_i s_j}(sym_k)$ は、その回数を $C(s_i \xrightarrow{sym_k} s_j)$ として次式で定義する。

$$q_{s_i s_j}(sym_k) = \frac{C(s_i \xrightarrow{sym_k} s_j)}{\sum_k C(s_i \xrightarrow{sym_k} s_j)} = \frac{C(s_i \xrightarrow{sym_k} s_j)}{C(s_i \rightarrow s_j)}. \quad (6.7)$$

よって $\sum_k q_{s_i s_j}(sym_k) = 1$ とする。

このようにして英文トレーニングセットに基づいて、状態数 98、出力シンボル数 84 の HMM を作成した。

6.3.2 得点付けのアルゴリズム

検索語の状態系列への展開

検索語を a とすると、まず検索語と同値とみなせる状態系列 S^a を以下の規則に従って全て求める。

- 最初の状態は第 5.2.2 節で定めたデリミタのいずれかとする。これはスペースやピリオドなどデリミタと考える文字を全て同じものとして扱って、1つの状態として扱うことに等しい。
- 2 番目の状態は、検索語の 1 文字目の状態であり、1 文字目と 2 文字目が結合誤りを起こす可能性がある場合、すなわち 1 文字目と 2 文字目からなる状態が存在する場合は、この状態も 2 番目の状態となる。また、検索語の前に挿入誤りが来る可能性は、検索語の後ろに挿入誤りがくる可能性とともに無視する。
- 3 番目以降の状態は、前の状態の次の文字の状態、結合誤りの可能性があれば次の 2 文字からなる状態、また挿入誤りの可能性があれば V_i の状態となる。
- 最後の状態は検索語の最後の文字の状態、あるいは検索語の最後の 2 文字が結合誤りを起こす可能性があればその 2 文字からなる状態のいずれかとなる。
- 挿入誤りについては、2 文字以上続けて挿入誤りが起こる可能性は無視する。これは挿入誤りに何らかの制限を加えないと、挿入誤りの発生する可能性のある箇所で無限個の挿入誤りが発生する可能性があり、結果として状態系列も無限に考えられることになるからである¹。

例えば検索語 “task” については、作成した HMM から図 6.5 のような状態遷移系列が求められる。図 6.5 では遷移確率が 0 より大きく、“task” を構成する状態間にのみ矢印を記している。また状態 “D” は、デリミタを全て同一文字として扱った場合の状態を表している。よって図 6.5 から “task” を構成する状態遷移系列は、

- 状態系列 1: “D” → “t” → “a” → “s” → “k”
- 状態系列 2: “D” → “ta” → “s” → “k”
- 状態系列 3: “D” → “t” → “ V_i ” → “a” → “s” → “k”

の 3 つであることが分かる。

このような状態系列の 1 つを $S_2^a = s^1 s^2 \dots s^n$ とすると、そのとりうる確率 $P(S_2^a)$ が次式で求められる。ここで 1 番目の状態を “D” としているので、 $s^1 = s_{DD}$ であり、 $\pi_{s^1} = 1$ である。

$$P(S_2^a) = \pi_{s^1} \prod_{i=1}^{n-1} p_{s^{i+1}s^i} = \prod_{i=1}^{n-1} p_{s^{i+1}s^i}. \quad (6.8)$$

¹ 勿論、挿入誤りの個数が増えれば増えるほどそのような状態系列をとる確率は低くなる。

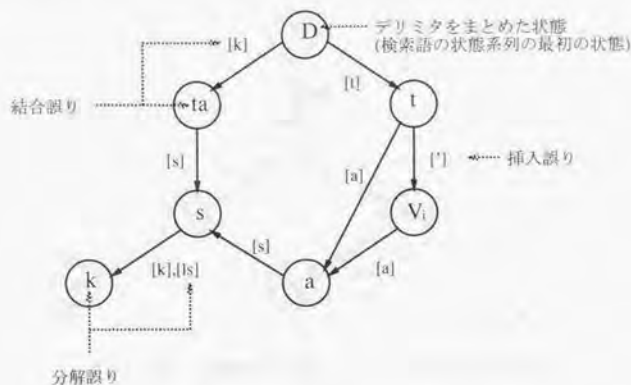


図 6.5: 検索語“task”のとりうる状態系列と出力シンボル

このようにして状態系列とその起こる確率が全て求められると、この HMM に基づいた $P(S_z^a|\alpha)$ を次式によって求めることができる。

$$\begin{aligned} P(S_z^a|\alpha) &= \frac{P(S_z^a)}{P(\alpha)} \\ &= \frac{P(S_z^a)}{\sum_{\alpha} P(S_z^a)}. \end{aligned} \quad (6.9)$$

式 (6.9) では、連続した挿入誤りを許さないという制限を設けることで、有限個の状態系列 S_z^a の起こり得る確率の和で $P(\alpha)$ を近似している。

状態系列から出力されるシンボル系列

HMM における状態系列 S^a が与えられると、その状態遷移に沿ってシンボル系列 SYM^a が次式の $P(SYM^a|S^a)$ という確率で出力される。

$$\begin{aligned} P(SYM^a|S^a) &= \prod_{i=1}^{n-1} q_{s^i s^{i+1}}(sym^i). \\ SYM^a &= sym^1 sym^2 \dots sym^{n-1}. \end{aligned} \quad (6.10)$$

例えば検索語“task”の場合図 6.5 から、先に述べた 3 つの状態系列に従って、それぞれ以下に示すシンボル列 SYM^a を出力する。

- 状態系列 1 の場合の出力シンボル系列: [t][a][s][k].[t][a][s][l]

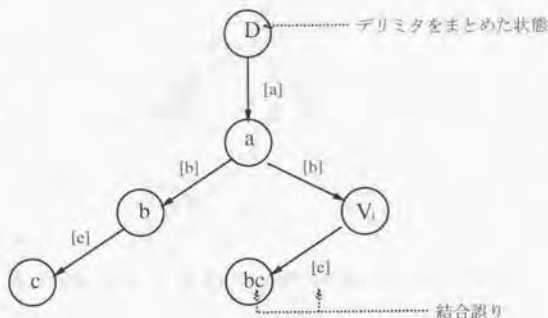


図 6.6: 異なる状態系列が同じシンボル系列を出力する例

- 状態系列 2 の場合の出力シンボル系列: $[k][s][k][k][s][ls]$
- 状態系列 3 の場合の出力シンボル系列: $[t][\text{'}][a][s][k],[t][\text{'}][a][s][ls]$

この検索語 “task” の例では、出力シンボルに仮想文字は含まれておらず、またシンボル列を連結して文字列を構成したものに重複がないため、検索語 “task” の拡張検索文字列候補は、上記の 6 つとなる。

拡張検索文字列の得点

シンボル系列が求められると、検索語 α がある状態系列 S_x^α をとりシンボル列 SYM^β を出力する確率 $P(SYM^\beta, S_x^\alpha | \alpha)$ は、式 (6.9) と式 (6.10) の積で求められる。また検索語 α から SYM^β が出力される確率 $P(SYM^\beta | \alpha)$ は、そのような状態系列 S_x^α について和をとることで求められるので、結局次式が成り立つ。

$$\begin{aligned}
 P(SYM^\beta | \alpha) &= \sum_x P(SYM^\beta, S_x^\alpha | \alpha) \\
 &= \sum_x P(SYM^\beta | S_x^\alpha) P(S_x^\alpha | \alpha).
 \end{aligned} \tag{6.11}$$

和をとるのは状態系列が異なり、出力シンボル系列が一致する例として、図 6.6 のような場合が考えられるからである。図 6.6 の HMM において検索語 “abc” を考えると、

- “D” \rightarrow “a” \rightarrow “b” \rightarrow “c”
- “D” \rightarrow “a” \rightarrow “V_i” \rightarrow “bc”

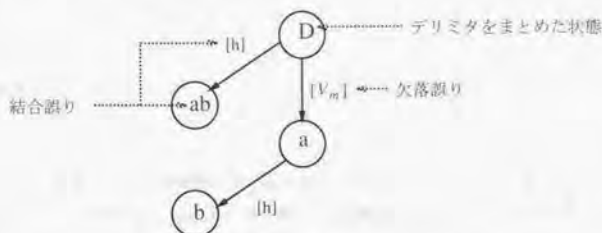


図 6.7: 拡張検索文字列の一致する例 (状態系列及びシンボル系列が共に異なる)

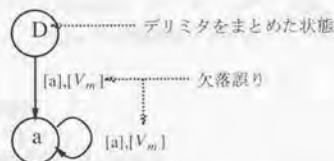


図 6.8: 拡張検索文字列の一致する例 (状態系列は等しくシンボル系列が異なる)

の 2 つの状態系列が存在し、それぞれ 1 つのシンボル系列を出力するが、このシンボル系列はどちらも $[a][b][e]$ となる。図 6.6 のような例は非常に稀ではあるが、可能性としては無視できないため、式 (6.11) のように各 S_y^2 について和をとる必要がある。

次に出力されたシンボル列の 1 つ $SY M_y^2$ をそれに対応する文字列 β_2 に変換する。このとき、 $P(\beta_2 | SY M_y^2) = 1$ は成り立つが、一般的には $P(SY M_y^2 | \beta_2) = 1$ は成り立たない。よって最終的な得点として扱う確率 $P(\beta_2 | \alpha)$ は次式によって得られる。

$$\begin{aligned} P(\beta_2 | \alpha) &= \sum_y P(\beta_2 | SY M_y^2) P(SY M_y^2 | \alpha) \\ &= \sum_y P(SY M_y^2 | \alpha). \end{aligned} \quad (6.12)$$

式 (6.12) に示すように和をとる必要があるのは、具体的には図 6.7 あるいは図 6.8 のような場合が存在するためである。図 6.7 は状態系列及びシンボル系列が共に異なる場合を、図 6.8 は状態系列は等しくシンボル系列が異なる場合を示しているが、いずれの場合も等しい拡張検索文字列が生成される。

図 6.7 の HMM では検索語 “ab” に対して、以下の 2 つの状態系列が存在し、それぞれ 1 つのシンボル系列を出力する。シンボル V_m は仮想的な文字で欠落することを表しているので、この 2 つのシンボル系列はシンボル列としては異なるが、同一の拡張検索文字列 “h” を表す。

- 状態系列 1: "D" → "a" → "b"

出力シンボル系列: $[V_m][h]$

- 状態系列 2: "D" → "ab"

出力シンボル系列: $[h]$

一方図 6.8 の HMM において検索語 "aa" を入力すると、とりうる状態系列は "D" → "a" → "a" のみであるが、そのとき出力されるシンボル系列は $[a][V_m]$ と $[V_m][a]$ の 2 つが存在する。しかし文字列としてみるとこの 2 つのシンボル系列はどちらも "a" となるので等しい。

拡張検索文字列の生成

本手法では HMM に基づいて式 (6.12) によって求めた確率 $P(\beta_z|\alpha)$ を得点と考え、この得点に基づいてランキングを行ない、上位 n 個を拡張検索文字列として出力する。

6.4 英文曖昧検索実験

第 4.1.2 節に示した実験環境のもとで、英文トレーニングセット及びテストセットを検索対象として本手法の検索効率と拡張検索文字列数の関係を調べる実験を行なった。本手法は、検索語 α に対して生成された拡張検索文字列 β_z を、HMM パラメータを用いて計算された確率 $P(\beta_z|\alpha)$ に基づいてランキングしており、その上位 n 個による検索を行なう。よってこの n をパラメータとして変化した場合の検索効率を図 6.9 に示す。

図 6.9 からトレーニングセットにおける実験では、本手法が少ない拡張検索文字列で高い検索効率を示していることが分かる。またこのとき最適と考えられる検索効率と拡張検索文字列数の組合せを示したのが表 6.1 である。表 6.1 から、拡張検索文字列数を 7 としたときの検索効率の値は、第 5.6 節で提案した逆確信度に基づくランキング手法のそれよりも高くなっており、拡張検索文字列数も半分以下であることが分かる。また拡張検索文字列数を 24 としたときには 100% の再現率と 99.83% の適合率を実現しているが、逆確信度に基づくランキング手法では拡張検索文字列数を 5000 としても再現率は 99.71% にすぎず、このとき適合率は 92.77% まで低下してしまう。ランキングの際に同じ $P(\beta_z|\alpha)$ の推定値を用いているにも関わらずこのような差がみられるのは、文字の unigram モデルを採用する ECMR 法²の限界を示していると考えられる。一方 HMM に基づく本手法は文字の bigram モデルを採用することで文字の接続を考慮しながら認識誤り候補文字を選択し検索語を拡張するため、生成される合計の拡張検索文字列数自体が少なくなっている。これにより、定義した全種類の誤りと文字の接続確率の両方を考慮した本手法の有効性が示されたと考えられる。

²逆確信度に基づいてランキングを行なう手法は、第 3 章で提案した ECMR 法の用いる得点を確信度から逆確信度に変えたもので、HMM に基づく本手法を除けば英文曖昧検索において最も検索性能が高いことが示されている (第 7.2.2 節参照)。

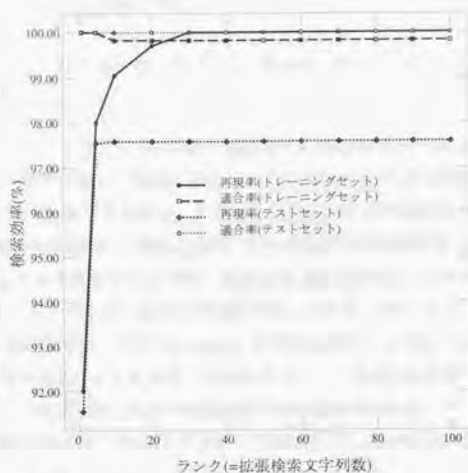


図 6.9: 英文における検索効率と拡張検索文字列数

表 6.1: 検索効率と拡張検索文字列数の最適値候補 (英文トレーニングセット)

再現率 (%)	適合率 (%)	拡張検索文字列数
99.05	100.0	7
99.71	99.83	17
100.0	99.83	24

しかしテストセットにおいては、完全照合の場合と比べても本手法が殆んど再現率を改善できていないことが分かる。すなわち、完全照合で英文テストセットを検索した際の再現率が表 4.6 から 97.54%であるのに対して、本手法では拡張検索文字列数を 100 とした場合でも 97.60%にしか改善できない。この原因について次節で考察する。

6.5 考察

前節で述べた通り本手法は、トレーニングセットにおいては極めて良好な結果が得られるが、テストセットにおいては殆んど再現率の改善がみられないという結果となった。これはすなわちテストセットにおける検索では、適切な拡張検索文字列が生成されていないことを意味しており、

この原因としては以下のものが考えられる。

- トレーニングセットが HMM のパラメータ (遷移確率やシンボル出力確率など) を推定するのに十分な量でない。

どの程度のトレーニングセットが必要であるかという学習量の問題は HMM に基づく本手法に限らず、本研究において提案した他の手法についても共通の課題である。しかし文字の unigram モデルを採用する場合、文字種と同数の文字の出現確率と、置換誤りだけを考えれば (文字種)² の置換誤りに関する確率 (文字の確信度や逆確信度) を求めればよいが、文字の bigram モデルを採用する本手法で推定しなければならない文字の接続確率は (状態数)² \approx (文字種)²、シンボル出力確率は置換誤りだけを考えた場合で (文字種)³ 存在する。よって少なくとも本手法では、文字の unigram モデルを採用する手法よりも多くのトレーニングセットが必要であることがいえる。またテストセットにおける検索ではトレーニングセットの場合と比べて生成される合計の拡張検索文字列数自体が少なくなっており、このことから第 4.1.2 節に示したトレーニングセットでは不十分であった可能性が高い。

- 認識誤りの分類が間違っている。

本手法の HMM は、認識誤りの分類ヒューリスティクス及び復合誤りの分割ヒューリスティクスを用いて切り分けられた図 6.4 のようなテキストを元に作成している。しかしこれらのヒューリスティクスの精度は高いと考えられるものの、認識誤りの種類を誤って解釈する場合もあれば、人間がみても解釈できないような複合誤りを 1 通りの解釈に確定して分割する場合もある。

結局、学習量の問題はトレーニングセットの適切な量について検討しなければならないことを示唆し、認識誤りの誤分類の問題はヒューリスティクスに依存しないパラメータ推定を行なう必要があることを示唆している。またテストセットにおいてトレーニングセット同様に高い検索効率を実現するためには、その他にも以下に示す課題が挙げられる。これらは全て、HMM のパラメータあるいは構造を変更することで HMM を改善する代表的な手法である。

- HMM パラメータの補間

トレーニングセットに出現しない文字の遷移や認識誤りについては、第 6.3.1 節のように出現頻度に基づいて遷移確率なり出力確率を推定するとその値は 0 となる。トレーニングセットが少ない場合にはこのような未観測事象の確率を 0 として扱うと、本章のテストセットに対する実験結果のような問題が発生する (ゼロ頻度問題)。よって一般的には、出現頻度から得られた結果については未観測事象の確率を補間することが多い。補間はスムージングとも呼ばれ、例えば代表的な補間法の 1 つであるバックオフスムージング [39] では、bigram

モデルにおける文字 C_1 の後に C_2 が接続する確率を次のようにして求める。

$$P(C_2|C_1) = \begin{cases} (1 - \frac{r}{N_{\text{bigram}}}) \cdot \frac{N(C_1, C_2)}{N(C_1)} & : N(C_1, C_2) \geq 1, \\ \frac{r}{N_{\text{bigram}}} \cdot \frac{N(C_2)}{K(C_1)} & : N(C_1, C_2) = 0. \end{cases} \quad (6.13)$$

ここで、 $N(C_i)$ は文字 C_i の、 $N(C_i, C_j)$ は文字列 $C_i C_j$ のトレーニングセット中の出現頻度を表し、また N_{bigram} はトレーニングセット中の文字 bigram の総数、 r は $N(C_i, C_j) = 1$ となる bigram の個数、 $K(C_1)$ は正規化定数を表す。補間を行わなければ $P(C_2|C_1)$ は $\frac{N(C_1, C_2)}{N(C_1)}$ で推定される。このようにしてバックオフスムージングでは未観測事象の全体に対して観測事象全体から割いた $\frac{r}{N_{\text{bigram}}}$ の確率を割り振っている。

- HMM パラメータの再推定 [33, 34]

HMM を用いた音声認識などでは通常、HMM パラメータの初期値を適当に与え、トレーニングセットを用いた学習によって HMM パラメータを再推定するといった方法をとる [36]。よってこれを本章で提案した HMM に当てはめて考えると、第 6.3.1 節で説明した方法で得られたパラメータの値を初期値として別のトレーニングセットを用い学習を行なうことが考えられる。しかしこの場合も初期値の与え方が再推定の結果に大きな影響を与えることが知られており [38]、この最初のパラメータ推定が重要であることに変わりはない。

- 状態の統合 [40]

本手法においても、デリミタとなるスペースやピリオドなどの状態は実質まとめて 1 つの状態として扱っているが、デリミタ以外についても類似状態と考えられる状態をまとめて 1 つの状態にすることで推定するパラメータの数を減らす方法が考えられる。類似状態としては、文字の形の似ているものや接続確率や出力確率が似ているものなどが考えられ、この場合問題としては状態のクラスタリングに帰着する。またこれは、未観測事象の確率を類似した文字 (状態) の確率から補う方法と考えることができる。

第7章

考察

本章ではまず、本論文で提案した曖昧検索手法の和英両文における検索性能についてまとめる。第7.1節では、第3章の提案手法の和文を対象とした曖昧検索における検索性能についてまとめ、他の和文曖昧検索手法との比較を行なう。第7.2節では、第5章及び第6章で提案した英語の言語特性に適した曖昧検索手法の検索性能についてまとめ、他の代表的な英文曖昧検索手法との比較を行なう。つづいて第7.3節では、本研究において拡張検索文字列の妥当性を表す尺度として各手法が導入した得点(確信度及び逆確信度)とベイズ確率との関係について考察する。また第7.4節では、確信度及び逆確信度に基づいてランキングされた拡張検索文字列の数と検索効率の関係について定性的に説明し、これらのランキングのもつ意味について考察する。

7.1 和文曖昧検索

本論文では第3章においてCMR法、ECMR法、BMR法の3つの曖昧検索手法を提案し、第4章において和文並びに英文曖昧検索における検索効率を評価した。本節では、これらの提案手法の和文曖昧検索における検索性能として、検索効率については第4章の結果に基づいてまとめ、また検索コストについてはCMR法の平均の拡張検索文字列数に関する考察に基づいてまとめる。また他の和文曖昧検索手法として第2.4.2節で紹介した複数認識候補型検索手法との比較を行ない、提案手法の有効性を示す。

7.1.1 検索性能

第4章の和文を対象とした検索実験の結果から、3つの提案手法の和文における有効性と、ECMR法及びBMR法のCMR法に対する優位性、すなわち置換誤りのみを考慮するCMR法よりもそれに加えて欠落・挿入・結合・分解誤りを考慮するECMR法や、文字の接続確率を併せて考慮するBMR法の方が優れていることを検索効率の点から示すことができた(表4.3、4.4参照)。和文テストセットにおける検索実験の結果(表4.4)は、認識誤りを含むテキストに対して提案した3つの手法全てが99%以上の再現率と適合率を同時に実現できることを示している。これは、実用的な日本語活字OCRの出力テキストを膨大なコストと労力をかけて訂正しなくても、提案手法を用いれば検索効率の観点から十分検索目的に活用できることを示すものである。また和文では、認識誤りに占める置換誤りの割合が約81%(表2.2)と他の種類に比べて圧倒的に多く、置換誤りしか考慮しないCMR法やBMR法でも高い再現率が得られている。

次に完全照合と比較した検索コストについて拡張検索文字列数に基づいて考える(表7.1、7.2参照)。まず第4章の和文曖昧検索の実験で使用した検索語は、トレーニングセット及びテストセットでそれぞれ50用意したが、各検索語の平均長はそれぞれ4.12文字及び4.02文字で、約4文字といえる。一方、これらの検索語の各文字について確信度を考慮せずに類似文字テーブルを参照して得られる置換候補は自身も含めてトレーニングセットで平均1.26文字、テストセットで平均1.36文字であった。よってCMR法において確信度を考慮しなければ、トレーニングセットにおいて平均 $1.26^4 \approx 2.5$ 、テストセットにおいて平均 $1.36^4 \approx 3.4$ の拡張検索文字列が生成されることになる。また最も長かった検索語の長さは、トレーニングセットにおいて11文字、テストセットにおいて8文字であったので、この場合同条件で考えるとそれぞれ約12.7、約11.7の拡張検索文字列が生成されることになる。実際の検索では確信度を考慮して検索文字列を絞り込むため、CMR法及びBMR法ではこれらの値が拡張検索文字列数の上限となる。またECMR法においては、欠落・挿入・結合・分解誤りを考慮するため拡張検索文字列数はCMR法と比較して増えるが、和文では置換誤りの割合が他の種類の誤りに比べて圧倒的に多いため、挿入誤り以外の影響はそれほど大きくないと考えられる。実際和文では、表4.3、4.4に示したようにCMR法やBMR法が99%以上の高い検索効率を実現しており、そのときの検索コストが完全照合の場合のせいぜい10倍程度であるといえる。

表 7.1: 和文曖昧検索で用いた検索語に関するデータ

トレーニングセット		テストセット	
平均検索語長 (文字)	最大検索語長 (文字)	平均検索語長 (文字)	最大検索語長 (文字)
4.12	11	4.02	8

表 7.2: 和文 CMR 法における拡張検索文字列数

トレーニングセット		テストセット	
平均検索文字列数	最大検索文字列数	平均検索文字列数	最大検索文字列数
2.5	12.7	3.4	11.7

7.1.2 複数認識候補型検索手法との比較

本研究で提案した CMR 法、ECMR 法、BMR 法を、同じく和文を対象とした曖昧検索手法である第 2.4.2 節で紹介した複数認識候補型検索手法と比較する。

この手法は、OCR の出力する検索対象テキストの方に複数の認識候補文字 (第 1 位候補文字以外) をもたせており、いうなれば検索対象テキストを拡張する手法である。文字列検索の実験において、ほぼ同性能と考えられる日本語活字 OCR の出力テキストに対して、提案手法と同じく 99% 以上の再現率及び適合率を実現している。一方複数認識候補型検索手法は、複数の候補文字を出力テキストに保持することでデータ量が第 1 位候補文字のみを出力する場合の 1.28 倍になっている。これは類似文字テーブルによって得られる検索語の平均置換候補文字数 1.26 とほぼ等しく、よって検索コストについては CMR 法と同程度といえる。しかし少なくとも以下の点について提案手法の優位性が主張できる。

- 文字切り出し誤りへの対応

複数認識候補型検索手法は OCR の出力する候補文字を直接利用して曖昧検索を行なうため、OCR が文字領域の抽出に失敗しているような場合には複数の候補文字の中にも正解が存在しないことが多い。一方提案手法では、トレーニングセットに対して認識誤りの抽出及び分類を OCR の認識過程とは独立に行なうため、ECMR 法によってこのような文字切り出しに起因する誤りにも対処できる。

- 曖昧検索のために必要な情報量

複数認識候補型検索手法では曖昧検索のために蓄積文書量が一定割合で増加するが、提案手法では曖昧検索に必要な類似文字テーブルなどは検索対象とする蓄積文書量に関係なく一定量で済む。

7.2 英文曖昧検索

本節では、英文曖昧検索特有の課題とその課題を考慮して第5章及び第6章で提案した英語の言語特性に適した曖昧検索手法についてまとめる。また和文の場合と同様に、検索効率と検索コストの両方の観点から提案手法の検索性能を評価して、代表的な曖昧検索手法である一般的な類似文字テーブルを用いる方法(第2.4.1節)や部分文字列照合(第2.4.3節)と比較して提案手法の有効性を示す。

7.2.1 課題と提案手法

英文曖昧検索では、第4章において提案したCMR法、ECMR法、BMR法を用いて和文の場合と全く同様に検索すると、

- 低適合率
- 膨大な数の拡張検索文字列の生成

という2つの問題が発生した。本研究ではこの2つの問題解決を課題として、日本語と異なる英語という言語の特徴を考慮した英文曖昧検索手法の提案を行なった。

まず低適合率の問題に対しては、大文字の小文字への変換(正規化)、デリミタによる単語の切り出しという前処理を行なうことで、和文と同程度の検索効率を得られることを示した(第5.2節)。また和文では考慮していなかった複合誤りをヒューリスティクスに基づいて分割し、その要素となる誤りの情報を類似文字テーブル及び拡張類似文字テーブルに反映させることで適合率が改善されることを示した(第5.3節)。

一方拡張検索文字列数の問題に対しては、英文曖昧検索において最も良い検索効率を示したECMR法を改良する形で、

- 拡張検索文字列中の認識誤りの数を制限するというヒューリスティクスを導入する(第5.5節)
- 拡張検索文字列の得点を逆確信度によって求め、その値に基づいてランキングを行ない、その上位 n 個を拡張検索文字列とする(第5.6節)

という2つの手法を提案し、検索効率及び拡張検索文字列数の両方の観点から評価した。

第6章では、認識前の元の文字及び2文字を状態とし、状態遷移の際に遷移後の状態が表す文字の認識結果を出力するようなHMMを構築し、このモデルに基づいた曖昧検索手法を提案した。このHMMでは、文字の接続確率が状態遷移確率で、文字の誤り確率がシンボル出力確率で表されており、このHMMに基づいた曖昧検索手法はECMR法とBMR法の1つの統合方法を示している。この手法は文字の接続情報を考慮するため、拡張検索文字列を生成する際の置換候補文字(列)が格段に減っており、結果として高い適合率と少ない拡張検索文字列数を期待でき、英文曖昧検索に適していると考えられる。

表 7.3: 検索効率と拡張検索文字列数 (英文トレーニングセット)

検索条件	再現率 (%)	適合率 (%)	拡張検索文字列数
完全照合	95.73	100.0	1
$P_E \geq 0.0001$ (ECMR 法)	99.05	99.63	171850.4
ヒューリスティクス 1 ($x=1$)	99.01	99.63	56.24
逆確信度によるランキング	99.01	99.63	20
HMM	100.0	99.83	24

表 7.4: 検索効率と拡張検索文字列数 (英文テストセット)

検索条件	再現率 (%)	適合率 (%)	拡張検索文字列数
完全照合	97.54	100.0	1
$P_E \geq 0.0001$ (ECMR 法)	99.01	99.47	92548.0
ヒューリスティクス 1 ($x=1$)	99.01	99.47	52.66
逆確信度によるランキング	99.01	99.80	9
HMM	97.60	100.0	7

7.2.2 検索性能

前節での議論を踏まえ本節では、提案した以下の3つの英文曖昧検索手法について、検索効率と拡張検索文字列の両方の観点から考察する。

1. 認識誤りの数を制限するためのヒューリスティクスを用いた検索手法
2. 逆確信度に基づくランキングによる検索手法
3. HMM に基づいた検索手法

なお、検索対象である英語活字 OCR の出力テキストには前処理がなされており、HMM に基づいた検索手法では複合誤りの分割が行なわれている。

このとき英文トレーニングセット及びテストセットそれぞれにおいて、これらの手法の最も良いと考えられる検索効率及び拡張検索文字列数をまとめると表 7.3、7.4 のようになる。また比較のためこれらの表には、完全照合及び確信度を閾値で評価するという従来の手法 (ECMR 法) による値も併記する (表 4.5、4.6 参照)。

表 7.3、7.4 に示したようにこの3つの手法は、従来の ECMR 法 ($P_E \geq 0.0001$) によって達成された検索効率を殆んど下げることなく拡張検索文字列数を大幅に削減したという点ではどれも有効である。但し唯一この有効性が示されていないのが、HMM に基づく手法のテストセットに

おける実験結果で、完全照合と比較しても殆んど再現率の改善がみられない。この理由は第6.5節で述べたように、トレーニングセットがHMMのパラメータ(遷移確率やシンボル出力確率など)を推定するのに十分でなかったことなどに原因があると考えられる。

認識誤りの数を制限するヒューリスティクスを用いた手法と逆確信度に基づいてランキング行なう手法を比較すると、以下の2つの理由から後者の方が前者よりも優れていると考えられる。1つは、後者の実現している検索効率は前者のそれに勝るとも劣らず、かつ後者の達成した拡張検索文字列数は前者のそのトレーニングセットにおいて約1/3、テストセットにおいては約1/5であること。もう1つは、表7.4から分かるように、テストセットにおいては完全照合と再現率の低過ぎるHMMに基づく手法を除くと後者が99.80%と最も高い適合率を実現していることである。表7.3、7.4は、逆確信度に基づいたランキング行なえば、完全照合の場合の20倍以下の検索コストで99%以上の検索効率が得られることを示している。

一方HMMに基づく手法は、トレーニングセットにおける検索では完全照合を除く他の3つの手法と比較して最も良い再現率と適合率を実現し、拡張検索文字列数も24と逆確信度による検索手法と同程度まで少なくなっている(表7.3)。また表6.1から、少し再現率を犠牲にすれば拡張検索文字列数は7まで下げられることが分かり、この場合の検索効率でも逆確信度による手法のそれを上回っている。よってトレーニングセットに対する検索のようにHMMのパラメータがうまく推定されている場合は、提案手法の中でこの手法が最も優れているといえる。一方、HMMに基づく検索手法の課題は表7.4のテストセットにおける実験結果に明確に示されており、適量のトレーニングセットを用いて学習を行なうことや、HMMのパラメータの補間や再推定を行なうHMMの改善を図ることが重要である。

7.2.3 他の英文曖昧検索手法との比較

英文テストセットに対する検索においてもっとも良い検索性能を示した、逆確信度に基づいてランキングを行なう手法を、英文を対象とした他の曖昧検索手法と比較する。この手法は表7.4に示した通り、99%以上の検索効率を完全照合の10倍以下の検索コストで実現している。

まず第2.4.1節でとりあげた類似文字の共通コード化(Character Classの生成)を行なう手法と類似文字テーブルを用いる手法について、検索効率に関するMykaらの実験結果[18]を表7.5に示す。この実験では英語活字OCRが実際に出力したテキストが用いられており、検索語長ごとの検索効率の値が得られている。本研究では英文トレーニングセット及びテストセットそれぞれに対して50の検索語を用意したがその平均の長さはそれぞれ7.42、7.30であったので、表7.5には検索語が7文字及び8文字のときの検索効率を示す。但し、実験環境が異なり、特にOCRの認識率がかなり悪い¹ため検索効率の絶対値の比較にはあまり意味がない。また類似文字テーブルを用いる手法は、表2.4などを用いて検索語を拡張するだけのものであり、本研究のように誤り易さに基づく確率は導入されていない。

¹完全照合を行なった際の再現率が認識率の1つの目安となるので、表7.4と表7.5を比較するとMykaらの実験で用いたOCRの認識率が本研究で用いた英語活字OCRに比べてかなり悪いことが分かる。

表 7.5: 検索効率の比較 (英文)

検索条件	検索語長 (文字)	再現率 (%)	適合率 (%)
完全照合	7	82.2	100.0
	8	81.5	100.0
共通コード化	7	83.0	93.9
	8	82.4	94.2
類似文字テーブル	7	87.2	64.6
	8	86.1	71.5

表 7.5 から、完全照合を行なう場合に比べて両手法とも約 1~5% 再現率を改善しているものの、適合率の低下が甚だしいことが分かる。特に類似文字テーブルを用いる手法では、再現率を上げることのみを考えて、OCR の誤り易さを何ら考慮せずに検索語を拡張すると適合率の低下が無視できないものになることを示している。また検索コストに関しても、本研究のように確信度などに基づいて拡張検索文字列を選別することを行なわないため、逆確信度に基づく手法よりもかなり大きなものになっていると考えられる。一方共通コード化の場合は、使用する文字集合自体が小さくなることを除けば完全照合の場合と検索コストは変わらないが、再現率の改善効果が 1% 程度しか示されていないため実用性については疑問である。

次に第 2.4.3 節でとりあげた部分文字列照合のうち、本研究と同じ目的のために文書と検索語の類似度をエディット距離によって定義した Lopresti ら [24, 25] の実験について説明する。彼らは長さ m の検索語と長さ n の文書に対して、 $O(mn)$ のコストを費やして図 2.5 のような表を作成し検索語と文書の類似度を $[0, 1]$ の連続値として定義した。また論理演算にファジー理論を採り入れて、問い合わせを複数の検索語の論理式で表す論理演算モデルにおける曖昧検索手法を提案した。彼らの実験では、1 つの検索語に対する文字列検索の結果は示されていないため、誤り率 2% の疑似生成した OCR テキストに対する論理演算モデルによる文書検索の実験結果について述べると、類似度の閾値を 0.6 としたときに完全照合で約 90% の再現率を約 94% に改善するが、適合率が約 100% から約 92% に下がる。また類似度の閾値を 0.2 とした場合は、再現率をほぼ 100% に改善するが、適合率が 30% 未満に下がるという結果が得られている。部分文字列照合の場合は、OCR 認識誤りの特性を全く考慮しないため適合率の低下は避けられず、また $O(mn)$ の検索コストも無視できるものではない。

7.3 確信度及び逆確信度と確率

提案手法の特徴は、確率に基づいた確信度あるいは逆確信度を用いて拡張検索文字列の妥当性を評価し絞り込むことにあるので、その意味でこれらは非常に重要である。またこれらと確率の

関係を明らかにすることは、提案手法で仮定する言語モデル及び認識誤りモデルを説明するために必要で、それによって曖昧テキスト検索にふさわしい1つの検索モデルの確立が期待できる。そこで本節では、式(3.5)によって各手法が定める文字列の確信度と式(5.3)によって ECMR 法が定める文字列の逆確信度が、ベイズの定理より得られる確率とどのような関係にあるのか考察する[41]。

7.3.1 CMR 法の用いる確信度と確率

CMR 法では置換誤りしか扱わないため、長さ n の検索語 $A^{12...n}$ は同じく長さ n の検索文字列 $B^{12...n}$ に拡張される。このとき $P(A^{12...n}|B^{12...n})$ はベイズの定理より次式で求められる。

$$P(A^{12...n}|B^{12...n}) = \frac{P(B^{12...n}|A^{12...n})P(A^{12...n})}{P(B^{12...n})}. \quad (7.1)$$

式(7.1)で、 $P(A^{12...n})$ 、 $P(B^{12...n})$ は文字列 $A^{12...n}$ 、 $B^{12...n}$ の生起確率を表すいわゆる言語モデルと呼ばれるもので、一方 $P(B^{12...n}|A^{12...n})$ は OCR の認識モデルである。ここで言語モデルとして unigram モデルを仮定する²と、

$$\begin{aligned} P(A^{12...n}) &= P(A^1)P(A^2)\dots P(A^n). \\ P(B^{12...n}) &= P(B^1)P(B^2)\dots P(B^n). \end{aligned} \quad (7.2)$$

がいえ。また、OCR の認識モデルは置換誤りしか含まない場合は文字と文字の 1:1 の対応関係が保証されているので各文字の認識が独立に行なわれているとすると、

$$P(B^{12...n}|A^{12...n}) = P(B^1|A^1)P(B^2|A^2)\dots P(B^n|A^n). \quad (7.3)$$

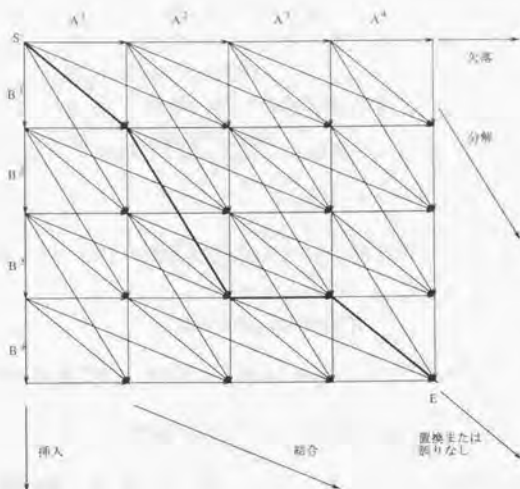
がいえ。式(3.1)、(7.1)、(7.2)、(7.3)より、式(3.5)が導けるので、CMR 法で計算した文字列の確信度はベイズ確率そのものである。

7.3.2 ECMR 法の用いる確信度と確率

ECMR 法の場合、検索語 $A^{12...m}$ とそれに対応する拡張検索文字列 $B^{12...n}$ が与えられても、 $A^{12...m} \rightarrow B^{12...n}$ という認識結果をもたらす誤りの組合せが複数考えられる(図 7.1 参照)。よって厳密に $P(A^{12...m}|B^{12...n})$ を求めるには、図 7.1 において S 地点から E 地点にたどり着くまでの枝に付与されている確信度を掛け合わせて 1 本のパス(文字列)の確信度を求め、さらにそれを全てのパスについて足し合わせる必要がある。つまりパスの 1 つを $(A_s^{12...m}, B_s^{12...n})$ 、考えられるパスの集合を L_p で表すことにすると、

$$P(A^{12...m}|B^{12...n}) = \sum_{(A_s^{12...m}, B_s^{12...n}) \in L_p} P(A_s^{12...m}|B_s^{12...n}). \quad (7.4)$$

² 文字の出現確率を、前後の文字(列)によらず独立とする。

図 7.1: ECMR 法における誤りの種類の曖昧性 ($m = n = 4$ の場合)

である。しかし ECMR 法ではこれを確信度が最大となるパスで近似することで計算の効率化を図っている。

$$\begin{aligned} \sum_{(A_z^{12\dots m}, B_z^{12\dots n}) \in L_p} P(A_z^{12\dots m} | B_z^{12\dots n}) &\simeq \max_{(A_z^{12\dots m}, B_z^{12\dots n}) \in L_p} P(A_z^{12\dots m} | B_z^{12\dots n}) \\ &= P(A_z^{12\dots m} | B_z^{12\dots n}). \end{aligned} \quad (7.5)$$

これは、図 7.1 において確信度が 0 を超えるパスは殆どの場合 1 つであることを利用したもので、第 4 章における実験ではこの近似による不都合は起こらなかった。

式 (7.5) からベイズの定理より、

$$P(A_z^{12\dots m} | B_z^{12\dots n}) = \frac{P(B_z^{12\dots n} | A_z^{12\dots m}) P(A_z^{12\dots m})}{P(B_z^{12\dots n})}. \quad (7.6)$$

がいえる。ここで $(A_z^{12\dots m}, B_z^{12\dots n})$ は最大の確信度をもつパスであるから、どの種類の誤りの連鎖によって構成されているかが特定されている。また unigram モデルを用いることで式 (7.2) が成り立っている。このとき、欠落・挿入・結合・分解誤りの確率を求め、それが第 3.3.6 節で述べたそれぞれの種類の誤りの確信度と一致することを以下に示す。これによって、ECMR 法における文字列の確信度もベイズ確率で説明することができる。

表 7.6: 文字の出現頻度

$N_{A^{all}}$	誤りのないテキストの長さ (文字列長)
N_{A^k}	A^k の出現回数
$N_{A_{miss}^k}$	全ての A^k のうち欠落誤りの数
$N_{A_{miss}}$	欠落誤りの総数

欠落誤り

$A^{12...m}$ 中の k 番めの文字が欠落誤りであるとする、式 (7.6) から求められるこの部分の確率 P_m は、

$$P_m = P(V_m|A^k)P(A^k). \quad (7.7)$$

となる。一方、ECMR 法では式 (3.6) に示したように $P_m = P(A^k|V_m)P_{m0}$ で定義しているため、

$$P(V_m|A^k)P(A^k) = P(A^k|V_m)P_{m0}. \quad (7.8)$$

のとき、確率と確信度が一致する。ここで、トレーニングセットにおいて表 7.6 のような文字の出現頻度を定義し、各確率をこれらの頻度から推定すると、

$$\begin{aligned} P(V_m|A^k)P(A^k) &= \frac{N_{A_{miss}^k}}{N_{A^k}} \frac{N_{A^k}}{N_{A^{all}}} = \frac{N_{A_{miss}^k}}{N_{A^{all}}}, \\ P(A^k|V_m)P_{m0} &= \frac{N_{A_{miss}^k}}{N_{A_{miss}}} \frac{N_{A_{miss}}}{N_{A^{all}}} = \frac{N_{A_{miss}^k}}{N_{A^{all}}}. \end{aligned} \quad (7.9)$$

がいえるため、式 (7.8) は成り立つ。なお $P_{m0} = P(V_m)$ である。

挿入誤り

$B^{12...n}$ 中の k 番めの文字が挿入誤りであるとする、式 (7.6) から求められるこの挿入誤りの部分の確率 P_i は、 B^k が挿入される確率が $P(B^k|V_i)P_{ins}$ と表せるので、

$$P_i = \frac{P(B^k|V_i)P_{ins}}{P(B^k)}. \quad (7.10)$$

となる。ここで、 P_{ins} は第 5.6.2 節で定めた挿入誤りの起こる確率であるから、 $P(V_i)$ に他ならない。一方 ECMR 法では挿入文字の確信度は $P(V_i|B^k)$ と計算するので、

$$P(V_i|B^k) = \frac{P(B^k|V_i)P(V_i)}{P(B^k)}. \quad (7.11)$$

のとき、確率と確信度が一致する。式 (7.11) は、 $P(V_i|B^k)$ をベイズの定理によって展開した式に他ならないので、挿入誤りを含む場合も確信度は確率である。

結合、分解誤り

$A^k A^{k+1} \Rightarrow B^l$ のような結合誤りを含む場合、 $A^k A^{k+1} \Rightarrow B^l$ となる確率は $P(B^l | A^k A^{k+1})$ と表せるので、式 (7.2)、(7.6) より確信度 $P(A^k A^{k+1} | B^l)$ が、

$$P(A^k A^{k+1} | B^l) = \frac{P(B^l | A^k A^{k+1}) P(A^k) P(A^{k+1})}{P(B^l)} \quad (7.12)$$

となる時確率と一致する。式 (7.12) は結合誤りの確信度 $P(A^k A^{k+1} | B^l)$ をベイズの定理と式 (7.2) を用いて展開したものに他ならないので、結合誤りを含む場合も確信度は確率といえる。

また $A^k \Rightarrow B^l B^{l+1}$ のような分解誤りを含む場合は、 $A^k \Rightarrow B^l B^{l+1}$ となる確率は $P(B^l B^{l+1} | A^k)$ と表せるので、式 (7.2)、(7.6) より同様に確信度 $P(A^k | B^l B^{l+1})$ が、

$$P(A^k | B^l B^{l+1}) = \frac{P(B^l B^{l+1} | A^k) P(A^k)}{P(B^l) P(B^{l+1})} \quad (7.13)$$

となる時確率と一致する。式 (7.13) は分解誤りの確信度 $P(A^k | B^l B^{l+1})$ をベイズの定理と式 (7.2) を用いて展開したものに他ならないので、分解誤りを含む場合も確信度は確率といえる。

7.3.3 ECMR 法の用いる逆確信度と確率

第 5.6 節で提案した文字列の逆確信度は、任意の文字列 $A^{012...n}$ が $B^{012...n}$ と認識される確率である $P(B^{012...n} | A^{012...n})$ の近似値となる。これは、元のテキストと認識結果のテキストを入れ換えて考えることで第 7.3.2 節と同様の議論が成り立つからである。

7.3.4 BMR 法の用いる確信度と確率

第 3.3.5 節で定めた $m_1(A_i^1)$ は CMR 法で用いる文字の確信度であり、ベイズ確率である。一方、 $m_2(A_i^1)$ は式 (3.3) の定義に示したように、ベイズ確率である $P(A_i^{1-1} | B^{i-1})$ と、文字の bigram から求めた $P(A_i^1 | A_i^{1-1})$ を組み合わせて、 B^{i-1} が観測されたとき A_i^1 である確率³を定めたものである。さらに式 (3.4) はこの 2 つの確率を Dempster の結合規則により統合して $m(A_i^1)$ を求めており、これも $\sum_i m(A_i^1) = 1$ を満たす確率となっている。よって BMR 法でいう文字の確信度は確率といえる。

次に、BMR 法で求めた文字列の確信度は式 (3.5) の定義より、

$$P(A_x^{12...n} | B^{12...n}) = m(A^1) m(A^2) \dots m(A^n) \quad (7.14)$$

で表される。使用する全ての文字種の数 N_{all} とすると、長さ n の文字列は N_{all}^n 通り考えられる。よって、 $P(A_x^{12...n} | B^{12...n})$ は $\sum_{x=1}^{N_{all}^n} P(A_x^{12...n} | B^{12...n}) = 1$ という意味では確率であるが、式 (7.1) のベイズ確率とは明らかに異なる。

³ $\sum_i m_2(A_i^1) = 1$ という意味で確率と呼ぶ。

7.3.5 確信度及び逆確信度とは何か

第7.3.1節では、CMR法が拡張検索文字列を選択するために計算する文字列の確信度が、式(7.1)によって計算されるベイズの定理から得られる確率そのものであることを示した。ECMR法の場合は、欠落・挿入・結合・分解誤りといった文字列の長さが変化する誤りを含むため、検索語 $A^{12..m}$ とそれに対応する拡張検索文字列 $B^{12..n}$ が与えられても、図7.1に示したようにそのような認識結果をもたらす要素誤りの組合せ(図7.1におけるパス)が複数考えられる。正確に $B^{12..n}$ が $A^{12..m}$ とみなせる確率を求めるには各パスの確率を足し合わせる必要があるが、ECMR法では第7.3.2節で述べたようにこれを最大となるパス1つで近似し、それを文字列の確信度としている。また第7.3.3節で述べたように、ECMR法の用いる逆確信度は $A^{12..m}$ が $B^{12..n}$ とOCRによって認識される確率の近似値といえる。BMR法の用いる文字列の確信度の場合は、第7.3.4節で述べたようにベイズの定理から導かれる確率とは異なるが、CMR法とは異なるモデルに基づいて $B^{12..m}$ が $A^{12..m}$ とみなせる確率を推定したものである。

以上の議論をまとめると、文字列の確信度は $B^{12..n}$ が $A^{12..m}$ とみなせる確率、もしくはその近似値といえ、逆確信度は $A^{12..m}$ が $B^{12..n}$ と認識される確率の近似値といえることができる。

7.4 確信度及び逆確信度と検索効率

英文曖昧検索では、検索語 $\alpha = A^{12..m}$ から複数の拡張検索文字列 $\beta = B^{12..n}$ を生成し、各 β を確信度、または逆確信度に基づいてランキングする手法を提案した。そこで本節では、確信度が近似している拡張検索文字列 β を検索語 α とみなせる確率 $P(\alpha|\beta)$ 、及び逆確信度が近似している α が β と認識される確率 $P(\beta|\alpha)$ と検索効率の関係について説明し、このランキングの意味について考察する。

7.4.1 文字列の出現確率

本研究でいう検索は文字列検索であり、検索語と一致する文字列を検索対象テキストの先頭から順に1文字ずつ比較しながら探していく。よってOCR認識前の元のテキストの大きさを $SIZE_{org}$ とおくと、長さ m の検索語 α と比較する文字列は全部で $SIZE_{org} - m + 1$ 存在する。よって検索語 α の元のテキストにおける出現確率 $P(\alpha)$ は次式で推定できる。

$$\begin{aligned} P(\alpha) &= \frac{\alpha \text{ が元のテキストに現れる頻度}}{SIZE_{org} - m + 1} \\ &\simeq \frac{\alpha \text{ が元のテキストに現れる頻度}}{SIZE_{org}} \end{aligned} \quad (7.15)$$

$SIZE_{org} - m + 1 \simeq SIZE_{org}$ は、一般に $SIZE_{org} \gg m$ のため成り立つ。

一方、OCR認識後のテキストの大きさを $SIZE_{recog}$ とおくと、 $P(\alpha)$ と同様に長さ n の拡張検

素文字列 β の認識後のテキストにおける出現確率 $P(\beta)$ は次式で推定できる。

$$\begin{aligned} P(\beta) &= \frac{\beta \text{が認識後のテキストに現れる頻度}}{SIZE_{recog} - n + 1} \\ &\simeq \frac{\beta \text{が認識後のテキストに現れる頻度}}{SIZE_{recog}} \end{aligned} \quad (7.16)$$

式 (7.16) でも、 $SIZE_{recog} \gg n$ により $SIZE_{recog} - n + 1 \simeq SIZE_{recog}$ と近似している。

すると検索語 α が元のテキストに出現しかつ、その文字列が β と認識されている確率 $P(\alpha \cap \beta)$ は、式 (7.17) で推定される。

$$\begin{aligned} P(\alpha \cap \beta) &= P(\beta|\alpha) \cdot P(\alpha) \\ &= \frac{\alpha \text{が}\beta \text{と認識されている頻度}}{\alpha \text{が元のテキストに現れる頻度}} \cdot \frac{\alpha \text{が元のテキストに現れる頻度}}{SIZE_{org}} \\ &= \frac{\alpha \text{が}\beta \text{と認識されている頻度}}{SIZE_{org}} \end{aligned} \quad (7.17)$$

また確率 $P(\alpha \cap \beta)$ は、拡張検索文字列 β が認識後のテキストに出現しかつ、その文字列に対応する元の文字列が α である確率といえる。よって $P(\alpha \cap \beta)$ は、式 (7.18) によっても推定できる。

$$\begin{aligned} P(\alpha \cap \beta) &= P(\alpha|\beta) \cdot P(\beta) \\ &= \frac{\alpha \text{が}\beta \text{と認識されている頻度}}{\beta \text{が認識後のテキストに現れる頻度}} \cdot \frac{\beta \text{が認識後のテキストに現れる頻度}}{SIZE_{recog}} \\ &= \frac{\alpha \text{が}\beta \text{と認識されている頻度}}{SIZE_{recog}} \end{aligned} \quad (7.18)$$

式 (7.17) と式 (7.18) の推定値は通常 $SIZE_{org} \neq SIZE_{recog}$ のため一致しないが、 $SIZE_{org} \simeq SIZE_{recog}$ が一般的に成り立つためほぼ等しくなる。例えば本研究の実験で用いた英文トレーニングセットでは、認識前の元のテキスト 80,985 文字に対して、認識後のテキストは 80,884 文字であった。よって、 $SIZE_{org}$ と $SIZE_{recog}$ の平均値などを $SIZE$ とおくと、

$$P(\alpha \cap \beta) \simeq \frac{\alpha \text{が}\beta \text{と認識されている頻度}}{SIZE} \quad (7.19)$$

がいえる。また同様に、式 (7.15) 及び式 (7.16) から、

$$P(\alpha) \simeq \frac{\alpha \text{が元のテキストに現れる頻度}}{SIZE} \quad (7.20)$$

$$P(\beta) \simeq \frac{\beta \text{が認識後のテキストに現れる頻度}}{SIZE} \quad (7.21)$$

がいえる。

7.4.2 拡張検索文字列数と検索効率

検索効率の定義は、式 (4.1) 及び (4.2) に示した通りである。よって、 $P(\alpha|\beta_2)$ あるいは $P(\beta_2|\alpha)$ に基づいてランキングされた拡張検索文字列のうち、上位 n 個による検索で期待される再現率

Recall_nは、式(7.19)及び式(7.20)を用いて次式で計算できる。

$$\begin{aligned}\text{Recall}_n &= \frac{SIZE \sum_{z=1}^n P(\alpha \cap \beta_z)}{SIZE \cdot P(\alpha)} = \sum_{z=1}^n \frac{P(\alpha \cap \beta_z)}{P(\alpha)} \\ &= \sum_{z=1}^n P(\beta_z | \alpha).\end{aligned}\quad (7.22)$$

一方、 n 個の拡張検索文字列を用いて検索した場合に期待される適合率 Precision_nは、式(7.19)及び式(7.21)を用いて次式で計算できる。

$$\begin{aligned}\text{Precision}_n &= \frac{SIZE \sum_{z=1}^n P(\alpha \cap \beta_z)}{SIZE \sum_{z=1}^n P(\beta_z)} \\ &= \frac{\sum_{z=1}^n P(\alpha | \beta_z) \cdot P(\beta_z)}{\sum_{z=1}^n P(\beta_z)}.\end{aligned}\quad (7.23)$$

$$= \frac{\sum_{z=1}^n P(\beta_z | \alpha) \cdot P(\alpha)}{\sum_{z=1}^n P(\beta_z)}.\quad (7.24)$$

また $n=1$ 、すなわち拡張検索文字列が1つのときは、

$$\text{Precision}_1 = P(\alpha | \beta_1).\quad (7.25)$$

が成り立ち、適合率はその1つの拡張検索文字列の $P(\alpha | \beta_1)$ に等しくなる。

式(7.22)は各拡張検索文字列の $P(\beta_z | \alpha)$ (逆確信度) の和が再現率となることを示している。式(7.23)は、拡張検索文字列が $P(\alpha | \beta_z)$ (確信度) に基づいてランキングされているとき、適合率が拡張検索文字列数の増加とともに低くなることを示している。これは以下のようにして証明できる。

証明 まず式(7.23)を用いて $\text{Precision}_k - \text{Precision}_{k+1}$ を展開すると式(7.26)が得られる。ここで k は自然数であり、このとき $\text{Precision}_k - \text{Precision}_{k+1} \geq 0$ が成り立てば、適合率が拡張検索文字列数の増加とともに低くなることがいえる。

$$\begin{aligned}\text{Precision}_k - \text{Precision}_{k+1} &= \frac{\sum_{z=1}^k P(\alpha | \beta_z) \cdot P(\beta_z)}{\sum_{z=1}^k P(\beta_z)} - \frac{\sum_{z=1}^{k+1} P(\alpha | \beta_z) \cdot P(\beta_z)}{\sum_{z=1}^{k+1} P(\beta_z)} \\ &= \frac{(\sum_{z=1}^{k+1} P(\beta_z))(\sum_{z=1}^k P(\alpha | \beta_z) \cdot P(\beta_z)) - (\sum_{z=1}^k P(\beta_z))(\sum_{z=1}^{k+1} P(\alpha | \beta_z) \cdot P(\beta_z))}{(\sum_{z=1}^k P(\beta_z))(\sum_{z=1}^{k+1} P(\beta_z))}.\end{aligned}\quad (7.26)$$

式(7.26)の分母は明らかに0以上なので、式(7.26)の分子が0以上であることを示すために、これを次式で展開する。

$$\begin{aligned}(\text{式(7.26)の分子}) &= (P(\beta_1) + \dots + P(\beta_{k+1})) \cdot (P(\alpha | \beta_1) \cdot P(\beta_1) + \dots + P(\alpha | \beta_k) \cdot P(\beta_k)) - \\ &\quad (P(\beta_1) + \dots + P(\beta_k)) \cdot (P(\alpha | \beta_1) \cdot P(\beta_1) + \dots + P(\alpha | \beta_{k+1}) \cdot P(\beta_{k+1})) \\ &= P(\beta_{k+1}) \sum_{z=1}^k P(\beta_z) (P(\alpha | \beta_z) - P(\alpha | \beta_{k+1})) \geq 0.\end{aligned}\quad (7.27)$$

このとき拡張検索文字列は $P(\alpha|\beta_2)$ に基づいてランキングされており、 $i \leq k$ のとき $P(\alpha|\beta_2) - P(\alpha|\beta_{k+1}) \geq 0$ であるので、式 (7.27) の不等号が成り立つ。(証明終り)

7.4.3 ランキングの意味

$P(\beta_2|\alpha)$ に基づくランキング

拡張検索文字列が $P(\beta_2|\alpha)$ (逆確信度) に基づいてランキングされている場合、検索に用いる文字列を1増やすということはもともと再現率を上げる効果の高い拡張検索文字列を選ぶということに他ならない。よって、 $P(\beta_2|\alpha)$ に基づくランキングを行なうということは、再現率重視の検索を行なうことといえる。

しかし適合率に関しては式 (7.24) に示したように、 $P(\alpha)$ 及び $P(\beta_2)$ が影響するため、 $P(\beta_2|\alpha)$ に基づくランキングが高適合率になるとは限らない。逆に図 5.10 から分かるように、英文曖昧検索では拡張検索文字列の増加は直ちに適合率の低下をもたらしている。しかし、再現率が最も効果的に改善されるため、結果的に適合率の低下が起こる前に少ない文字列で高い再現率を実現しており、英文曖昧検索に適した手法といえる。

$P(\alpha|\beta_2)$ に基づくランキング

一方拡張検索文字列が $P(\alpha|\beta_2)$ (確信度) に基づいてランキングされている場合、拡張検索文字列を1増やすということは最も $P(\alpha|\beta_2)$ が大きい拡張検索文字列を選ぶということである。またこの $P(\alpha|\beta_2)$ は、式 (7.25) に示されている通りその拡張検索文字列のみで検索を行なった場合の適合率でもある。ここで、各拡張検索文字列 β_2 の出現確率 $P(\beta_2)$ が等しく $P(\beta)$ であるという仮定をおけば、式 (7.23) は次のように変形できる。

$$\text{Precision}_n = \frac{\sum_{i=1}^n P(\alpha|\beta_2) \cdot P(\beta)}{n \cdot P(\beta)} = \frac{\sum_{i=1}^n P(\alpha|\beta_2)}{n} \quad (7.28)$$

式 (7.28) は、 n 個の拡張検索文字列の出現確率が全て等しい場合の適合率が、各文字列の $P(\alpha|\beta_2)$ の平均になることを意味している。よって拡張検索文字列が $P(\alpha|\beta_2)$ に基づいてランキングされている場合、それは最も適合率が高くなるように拡張検索文字列を n 個選んでいることになる。しかし再現率は、これらの文字列の $P(\beta_2|\alpha)$ によって決定されるため高くなるとは限らない。

よって $P(\alpha|\beta_2)$ に基づくランキングを行なうということは、適合率重視の検索を行なうことといえる。故に、図 5.5 などからも分かるようにかなり多くの拡張検索文字列を用いても高い適合率が保たれる。しかし英文曖昧検索においては、これらの多くの拡張検索文字列の大部分は再現率の改善に寄与しておらず、検索コストの増加だけをもたらしている。これはすなわち、拡張検索文字列の多くが認識結果のテキストには現れず、実際の検索に役立たない無駄な検索文字列ということになるので、例えば第 5.5 節で提案したヒューリスティクスなどを用いて拡張検索文字列を絞り込む必要がある。

第8章

おわりに

本論文を総括し、今後の展望について述べる。

8.1 本論文のまとめ

8.1.1 曖昧検索手法の提案とその背景

主に文書を扱う電子図書館では、閲覧に文書の再現性の高い文書画像を用い、検索には文書画像を解析し、さらに OCR 処理して得られる低コストかつ不完全なテキストがそのまま活用されるようになりつつある。このような文書画像を主体とした電子図書館は、過去の膨大な文書遺産を蓄積・提供することを考えると最も現実的な実現形態といえ、特に電子図書館構築側の要求であるデータ入力自動化への道を開くものである。本研究ではこのような電子図書館において必要不可欠な、不完全なテキストを認識誤りを考慮しながら検索する曖昧検索手法を和英文についてそれぞれ複数提案しその評価を行なった。

8.1.2 本研究のまとめ

本研究で提案した曖昧検索手法は全て検索語拡張という形態をとり、検索語 α を元に生成される検索文字列 β の取捨選択を、 β を α とみなせる確率 $P(\alpha|\beta)$ 及びその近似値としての確信度、または α が OCR によって β と認識される確率 $P(\beta|\alpha)$ 及びその近似値としての逆確信度に基づいて行なう点に特徴がある。よって提案手法の成否は、これらの確率あるいはその近似値の推定方法や精度に依るところが大きく、本研究ではその推定方法が異なる複数の曖昧検索アルゴリズムを提案した。

まず第3章において CMR 法、ECMR 法、BMR 法の3つの曖昧検索手法を提案した。CMR 法は置換誤りを、ECMR 法は置換誤りに加えて欠落・挿入・結合・分解誤りを、BMR 法は置換誤りに加えて文字の bigram に基づく接続確率を検索時に考慮する手法である。第4章においてこれらの手法を、現行の日本語及び英語活字 OCR の出力テキストに対して適用してその性能を評価した。その結果和文においては検索効率、検索コストともに良好な結果を得たが、英文では低適合率、膨大な検索コストという2つの問題が発生した。この原因は、日本語と比べて単語が長く、また OCR に混同される可能性のある文字が多いという英語の言語特性によるものである。

そこで第5章では英文曖昧検索におけるこれらの課題を解決するために、英文に特化した処理及び曖昧検索アルゴリズムを新たに提案した。低適合率の改善については、大文字の小文字への変換(正規化)、デリミタによる単語の切り出しという前処理を行なうこと、あるいは和文では考慮していなかった複合誤り情報を類似文字テーブル及び拡張類似文字テーブルに反映させることが有効であった。一方検索コストの目安となる拡張検索文字列数を削減するためには、拡張検索文字列中の認識誤りの数を制限するというヒューリスティクスを導入する手法と、拡張検索文字列の逆確信度に基づいてランキングを行なう手法の2つを提案しその有効性を示した。

一方第4章における実験から、第3章で提案した3つの手法の中では文字切り出し誤りに対応した ECMR 法が最も高い検索効率を実現できること、CMR 法と BMR 法の比較から文字の接続確率を考慮することが検索効率の改善と検索コストの削減に有効であることが明らかになった。そこでこの結果を踏まえて第6章では、認識前の元の文字及び2文字を状態とし、状態遷移の際

に遷移後の状態を表す文字の認識結果を出力するような HMM を構築し、このモデルに基づいた曖昧検索手法を提案した。この HMM では、文字の接続確率が状態遷移確率で、文字の読み確率がシンボル出力確率で表されており、この HMM に基づいた曖昧検索手法は ECMR 法と BMR 法の 1 つの統合方法といえる。

最後に第 7 章で検索語 α と拡張検索文字列 β について、本研究で定めた拡張検索文字列の確信度及び逆確信度と、ベイズの定理から導かれる $P(\alpha|\beta)$ 及び $P(\beta|\alpha)$ との関係について考察した。また拡張検索文字列を確信度あるいは逆確信度に基づいてランキングすることの意味を検索効率の観点から考察し、実験結果について定性的に説明した。

8.1.3 本研究の成果

実験により実用的な日本語及び英語活字 OCR の出力したテキストに対する文字列検索において、日本語では CMR 法または BMR 法によって通常の高々 10 倍程度、英語では逆確信度に基づくランキングを行なう手法によって通常の高々 20 倍程度の検索コストで、99% 以上の再現率及び適合率を達成できることを示した。また文字の切り出し誤りに起因する欠落・挿入・結合・分解誤りと文字の接続確率を同時に扱える HMM を用いた手法が、99% 以上の高い検索効率を維持したまま英文トレーニングセットにおける検索コストを、通常の 10 倍未満にまで抑えることができることを示した。よってこの HMM に基づく手法が、提案した英文曖昧検索手法の中で検索効率と検索コストの双方の点で最も優れた性質をもつと考えられ、テストセットにおける性能改善が期待される。

8.2 今後の展望

本研究で提案した曖昧検索手法は、文字列検索において現実的な検索コストで高い検索効率を実現している。しかし実際の文書検索では、検索対象の規模や検索語数の点で本論文の実験条件よりも厳しくなることが予想されるため検索コストが非常に重要な要素となる。すなわち、実用的な検索システムに本研究で提案した曖昧検索手法を実装する場合には、検索システムの性能によって自ずと拡張検索文字列数の上限が定められるであろうから、その制限内で如何に有効な拡張検索文字列を生成するかが重要な課題となる。とりわけ検索コストが大きくなりがちな英文曖昧検索ではこの問題への配慮は不可欠であるが、本研究で提案した HMM に基づいた手法がこの要件を満たす非常に有望な手法であると考えられる。現時点ではこの手法はテストセットにおいて成果をあげるに至っていないが、第 6 章で指摘した学習量の問題を検討し、HMM パラメータの補間、再推定、状態の統合といった方法で HMM を改善することで英文曖昧検索の 1 つの解法となることを確信している。

また本研究で示した曖昧検索モデルは確率を基礎に考えておりその意味で汎用性をもつ。よって活字 OCR による認識誤りのみでなく、例えば音声認識など他の認識系においても、元のシンボル系列とそれに対応する認識結果のシンボル系列がトレーニングセットとして得られれば原理

的に適用可能である。勿論現時点における音声認識の認識率は活字 OCR による文字認識と比較するとかかなり低いため、提案手法の認識率に対する頑健性が問われることになるし、また認識系が異なれば、置換・欠落・挿入・結合・分解誤りといった認識誤りの種類についても見直す必要があるかもしれない。しかし電子図書館などデータベースに蓄える情報の質と量について検索の利便性とデータ入力コストの観点から考えたとき、曖昧検索が有力かつ現実的な解法であることに疑いはない。本研究の成果が様々な認識系の出力結果を曖昧検索する際に広く活用されることを期待している。

謝辞

本研究をすすめるにあたって指導教官である安達淳教授並びに学術情報センターの高須淳宏助教授には、多くの御指導と貴重なアドバイスを頂きました。実に5年間という大学院生活の長きに渡り、安達先生には貴重な時間を割いて学問の基礎から研究全般に至るまで粘り強く御教示頂きました。高須先生には、特に理論面から本研究に関する有益な御指摘並びに助言を賜りました。ここに、心から感謝の意を表します。

安達研究室の方々には、プログラミングをはじめ計算機に関する多くのことを教えて頂き、また日頃の生活でも大変お世話になりました。特に学術情報センター助手の片山紀生氏、日立の西澤格氏、NECの酒井乃里子氏の諸先輩方には、ひとかたならぬお世話になりました。ここに深謝致します。

研究室での生活は、同じ学術情報センターの濱田研究室、浅野研究室の皆様との合同イベントなどあり非常に楽しく充実したものでした。ここに改めて感謝の意を表したいと思います。

そして最後になりましたが、あらゆる面で支援してくれた家族並びに特に精神的な面で支えてくれた妻に感謝します。

参考文献

- [1] 安達淳, 橋爪宏達, 片山紀生: 学術情報センターの電子図書館システムの概要と試行実験, 情報処理学会研究報告, 95-FI-37, pp. 23-30 (1995).
- [2] 石川徹也: フルテキスト・データ検索機能の検討, デジタル図書館, No. 3 (1995).
- [3] 太田学, 高須淳宏, 安達淳: 認識誤りを含む和文テキストにおける全文検索手法, 情報処理学会論文誌, Vol. 39, No. 3, pp. 625-635 (1998).
- [4] Ohta, M., Takasu, A. and Adachi, J.: Reduction of Expanded Search Terms for Fuzzy English-text Retrieval, *Proc. of ECDL'98 (Second European Conference on Digital Libraries)*, LNCS (Lecture Notes in Computer Science) 1513, Crete, Greece, pp. 619-633 (1998).
- [5] 秋山照雄, 増田功: 書式指定情報によらない紙面構成要素抽出法, 電子通信学会論文誌, Vol. J66-D, No. 1, pp. 111-118 (1983).
- [6] 黄瀬浩一, 杉山淳一, 馬場口登, 手塚慶一: レイアウトモデルに基づく文書構造解析, 電子情報通信学会論文誌, Vol. J72-D-II, No. 7, pp. 1029-1039 (1989).
- [7] Taghva, K., Condit, A., Borsack, J., Kilburg, J., Wu, C. and Gilbreth, J.: The MANICURE Document Processing System, Technical Report 95-02, Information Science Research Institute, University of Nevada, Las Vegas (1995).
- [8] Taghva, K., Condit, A. and Borsack, J.: An Evaluation of an Automatic Markup System, *Proc. of the IS&T/SPIE 1995 International Symposium on Electronic Imaging Science and Technology*, San Jose, CA, pp. 317-327 (1995).
- [9] 西野文人: 文字認識における自然言語処理, 情報処理, Vol. 34, No. 10, pp. 1274-1280 (1993).
- [10] Esakov, J., Lopresti, D. P., Sandberg, J. S. and Zhou, J.: Issues in Automatic OCR Error Classification, *Proc. of SDAIR'94 (3rd Annual Symposium on Document Analysis and Information Retrieval)*, Las Vegas, Nevada, pp. 401-412 (1994).

- [11] Taghva, K., Borsack, J. and Condit, A.: An Expert System for Automatically Correcting OCR Output, *Proc. of the IS&T/SPIE 1994 International Symposium on Electronic Imaging Science and Technology*, San Jose, CA, pp. 270-278 (1994).
- [12] Wiedenhofer, L., Hein, H.-G. and Dengel, A.: Post-Processing of OCR Results for Automatic Indexing, *Proc. of ICDAR'95 (3rd International Conference on Document Analysis and Recognition)*, Montreal, Canada, IEEE Computer Society Press, pp. 592-596 (1995).
- [13] 山岡正輝, 岩城修: 文書画像の SGML 文書への変換に関する一検討, 電子情報通信学会技術研究報告 PRU94-36, Vol. 94, No. 241.242, pp. 73-80 (1994).
- [14] Fujisawa, H. and Marukawa, K.: Full-Text Search and Document Recognition of Japanese Text, *Proc. of SDAIR'95 (4th Annual Symposium on Document Analysis and Information Retrieval)*, Las Vegas, Nevada, pp. 55-80 (1995).
- [15] Taghva, K., Borsack, J. and Condit, A.: Results of Applying Probabilistic IR to OCR Text, *Proc. of the 17th Annual International ACM/SIGIR Conference on Research and Development in Information Retrieval*, Dublin, Ireland, pp. 202-211 (1994).
- [16] Taghva, K., Borsack, J. and Condit, A.: Evaluation of Model-Based Retrieval Effectiveness with OCR Text, *ACM Trans. on Information Systems*, Vol. 14, No. 1, pp. 64-93 (1996).
- [17] Taghva, K., Borsack, J. and Condit, A.: Effects of OCR Errors on Ranking and Feedback Using the Vector Space Model, *Information Processing & Management*, Vol. 32, No. 3, pp. 317-327 (1996).
- [18] Myka, A. and Guntzer, U.: Fuzzy Full-Text Searches in OCR Databases, *Advances in Digital Libraries (Preliminary Version)*, SPRINGER-VERLAG, New York, chapter 7, pp. 87-100 (1995).
- [19] 丸川勝美, 藤澤浩道, 嶋好博: 認識機能の出力あいまい性を許容した情報検索手法の一検討-認識誤り特性に着目した検索手法の分析評価-, 電子情報通信学会論文誌, Vol. J79-D-II, No. 5, pp. 785-794 (1996).
- [20] 丸川勝美, 藤澤浩道, 嶋好博: 文書認識と全文検索の融合技術に関する実験的検討, 情報処理学会研究報告, 95-FI-39, pp. 65-72 (1995).
- [21] 長尾眞, 森信介: 大規模日本語テキストの n グラム統計の作り方と語句の自動抽出, 情報処理学会研究報告, 93-NI-96, pp. 1-8 (1993).

- [22] Kantor, P. B. and Voorhees, E.: Report on the TREC-5 Confusion Track, *Proc. of TREC-5 (5th Text REtrieval Conference)*, Gaithersburg, Maryland, NIST Special Publication 500-238, pp. 65-74 (1996). URL <http://trec.nist.gov/pubs.html>.
- [23] Takasu, A., Katayama, N., Yamaoka, M., Iwaki, O., Oyama, K. and Adachi, J.: Approximate Matching for OCR-Processed Bibliographic Data, *Proc. of ICPR'96 (13th International Conference on Pattern Recognition)*, Vol. III, Vienna, Austria, pp. 175-179 (1996).
- [24] Lopresti, D. and Zhou, J.: Retrieval Strategies for Noisy Text, *Proc. of SDAIR'96 (5th Annual Symposium on Document Analysis and Information Retrieval)*, Las Vegas, Nevada, pp. 255-269 (1996).
- [25] Lopresti, D. P.: Robust Retrieval of Noisy Text, *Proc. of ADL'96 (Forum on Research and Technology Advances in Digital Libraries)*, Library of Congress, Washington, D. C., pp. 76-85 (1996). URL <http://dlt.gsfc.nasa.gov/adl96/>.
- [26] Jardino, M.: Multilingual Stochastic N-gram Class Language Models, *Proc. of ICASSP'96 (IEEE International Conference on Acoustics, Speech and Signal Processing)*, Vol. 1, pp. 161-163 (1996).
- [27] 石塚満: Dempster と Shafer の確率理論, 電子通信学会誌, Vol. 66, No. 9, pp. 900-903 (1983).
- [28] 小林邦勝, 鈴木伸明, 根元義章, 佐藤利三郎: Dempster と Shafer の確率理論に基づく情報量に関する一考察, 電子通信学会論文誌, Vol. J68-A, No. 8, pp. 741-747 (1985).
- [29] Croft, W. B., Harding, S. M., Taghva, K. and Borsack, J.: An Evaluation of Information Retrieval Accuracy with Simulated OCR Output, *Proc. of SDAIR'94 (3rd Annual Symposium on Document Analysis and Information Retrieval)*, Las Vegas, Nevada, pp. 115-126 (1994).
- [30] Elsevier Science: About Us. <http://www.elsevier.nl/homepage/about.html>.
- [31] 森大毅, 阿曾弘具, 牧野正三: 2重マルコフモデルを用いた日本語文書認識後処理, 情報処理学会研究報告, 94-NL-102, pp. 89-96 (1994).
- [32] 永田昌明: 文字類似度と統計的言語モデルを用いた日本語文字認識誤り訂正法, 電子情報通信学会論文誌, Vol. J81-D-II, No. 11, pp. 2624-2634 (1998).
- [33] Rabiner, L. R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition, *Proc. of the IEEE*, Vol. 77, No. 2, pp. 257-286 (1989).
- [34] Charniak, E.: *Statistical Language Learning*, The MIT Press (1993).

- [35] 大河内正明: Hidden Markov Model に基づいた音声認識, 日本音響学会誌, Vol. 42, No. 12, pp. 936-941 (1986).
- [36] 大河内正明: マルコフモデルによる音声認識, 電子情報通信学会誌, Vol. 70, No. 4, pp. 352-358 (1987).
- [37] 鹿野清宏: 統計的手法による音声認識, 電子情報通信学会誌, Vol. 73, No. 12, pp. 1276-1285 (1990).
- [38] 竹内孔一, 松本裕治: 隠れマルコフモデルによる日本語形態素解析のパラメータ推定, 情報処理学会論文誌, Vol. 38, No. 3, pp. 500-509 (1997).
- [39] 鹿野清宏: 音声認識のための統計的言語モデル (1996). 音声情報処理 (講義資料) 音声認識の第5章.
- [40] 森信介, 長尾眞: 形態素 bi-gram と品詞 bi-gram の重ね合わせによる 形態素解析, 情報処理学会研究報告, 96-NL-112, pp. 37-44 (1996).
- [41] Lu, S.-Y. and Fu, K.-S.: Stochastic Error-Correcting Syntax Analysis for Recognition of Noisy Patterns, *IEEE Transactions on Computers*, Vol. c-26, No. 12, pp. 1268-1276 (1977).

発表文献

1. 学会誌論文

- 太田 学, 高須 淳宏, 安達 淳: 認識誤りを含む和文テキストにおける全文検索手法, 情報処理学会論文誌, Vol. 39, No. 3, pp. 625-635 (March 1998).
- Ohta, M., Takasu, A. and Adachi, J.: Reduction of Expanded Search Terms for Fuzzy English-text Retrieval, *the special issue of the IJODL (International Journal on Digital Libraries)*, Springer (submitted).

2. 国際会議論文

- Ohta, M., Takasu, A. and Adachi, J.: Probabilistic Retrieval Methods for Text with Miss-Recognized OCR Characters, *Proc. of IROL'96 (the Workshop on Information Retrieval with Oriental Languages)*, Taejon, Korea, pp. 35-41 (June 1996).
- Ohta, M., Takasu, A. and Adachi, J.: Retrieval Methods for English-Text with Mis-recognized OCR Characters, *Proc. of ICDAR'97 (Fourth International Conference on Document Analysis and Recognition)*, Vol. 2, Ulm, Germany, pp. 950-956 (August 1997).
- Ohta, M., Takasu, A. and Adachi, J.: Reduction of Expanded Search Terms for Fuzzy English-text Retrieval, *Proc. of ECDL'98 (Second European Conference on Digital Libraries)*, Crete, Greece, LNCS (Lecture Notes in Computer Science) 1513, pp.619-633, Springer (September 1998).

3. 研究会論文

- 太田 学, 高須 淳宏, 安達 淳: 英文曖昧検索における拡張検索文字列数の削減, 電子情報通信学会技術研究報告 DE98-9, Vol. 98, No. 42, pp. 63-70 (May 1998).

4. 大会論文

- 太田 学, 高須 淳宏, 安達 淳: 文字誤りを含む英文検索手法, 第54回情報処理学会全国大会講演論文集 (3), pp. 163-164 (March 1997).

- 太田 学, 高須 淳宏, 安達 淳: 英文認識誤り特性に基づいた曖昧検索手法, 第 55 回情報処理学会全国大会講演論文集 (3), pp. 123-124 (September 1997).
- 太田 学, 高須 淳宏, 安達 淳: OCR 認識誤りの学習方法について, 第 57 回情報処理学会全国大会講演論文集 (2), pp. 41-42 (October 1998).

5. その他

- 太田 学, 高須 淳宏, 安達 淳: 認識誤りを含むテキストにおける検索手法, 学術情報センター紀要 第 9 号, pp. 161-172 (March 1997).

付録 A

HMM に基づいた英文曖昧検索手法の生成する拡張検索文字列

第 6 章において提案した HMM に基づいた英文曖昧検索手法が、第 4 章に示した英文トレーニングセットを検索する際に 50 の検索語から生成した拡張検索文字列とその得点を示しておく。具体的には以下に、検索語 1 語につき最大 10 の拡張検索文字列を得点の大きい順に列挙する。この中には表示桁数の関係で得点が 0.00000 となっているものもあるが、ここに列挙した拡張検索文字列の得点は全て 0 よりも大きい。

検索語	[ランク] 拡張検索文字列 (得点)	
estimation	[1]estimation(0.74631)	[2]estimadon(0.13039)
	[3]estimabon(0.06520)	[4]estimahon(0.04346)
	[5]estlmation(0.00249)	[6]estimatlon(0.00249)
	[7]stimation(0.00163)	[8]estmation(0.00125)
	[9]estimaton(0.00125)	[10]esimation(0.00123)
flow	[1]flow(0.90718)	[2]f ow(0.05040)
	[3]tlow(0.02105)	[4]ilow(0.00574)
	[5]llow(0.00383)	[6]l'low(0.00383)
	[7] low(0.00191)	[8]low(0.00191)
	[9]t'low(0.00191)	[10]t ow(0.00117)
image	[1]image(0.99182)	[2]imag<(0.00395)
	[3]limage(0.00312)	[4]dmage(0.00052)
	[5]imadge(0.00035)	[6]im*age(0.00011)
	[7]i_image(0.00006)	[8]i_mage(0.00006)
	[9]limg<(0.00001)	[10]dmag<(0.00000)
adaptation	[1]adaptation(0.75178)	[2]adaptadon(0.13135)
	[3]adaptabon(0.06567)	[4]adaptahon(0.04378)
	[5]adaptatlon(0.00251)	[6]adaptaton(0.00126)
	[7]adap^tation(0.00063)	[8]adapwtation(0.00063)
	[9]adaptateon(0.00063)	[10]udaptation(0.00048)
filter	[1]filter(0.93526)	[2]tilter(0.02170)
	[3]filler(0.00964)	[4]iilter(0.00592)
	[5]file(0.00428)	[6]l'filter(0.00395)
	[7]iilter(0.00395)	[8]fliter(0.00382)
	[9]filtr(0.00277)	[10]t'ilter(0.00198)

検索語	[ランク] 拡張検索文字列 (得点)	
drawings	[1]drawings(0.96913)	[2]drawin6s(0.00925)
	[3]drawtngs(0.00763)	[4]d*rawings(0.00246)
	[5]dradwings(0.00235)	[6]lrawings(0.00230)
	[7]drawillgs(0.00224)	[8]drawintjs(0.00185)
	[9]drawilgs(0.00075)	[10]drawisgs(0.00075)
motion	[1]motion(0.98960)	[2]snotion(0.00456)
	[3]motlon(0.00331)	[4]moton(0.00165)
	[5]moteon(0.00083)	[6]mot'ion(0.00003)
	[7]snotlon(0.00002)	[8]snoton(0.00001)
	[9]snoteon(0.00000)	[10]snot'ion(0.00000)
field	[1]field(0.93652)	[2]tielfd(0.02173)
	[3]fie d(0.01362)	[4]ielfd(0.00593)
	[5]l'ield(0.00395)	[6]lielfd(0.00395)
	[7]f ield(0.00382)	[8]fielfd(0.00341)
	[9]ti'ield(0.00198)	[10]ielfd(0.00198)
experiments	[1]experiments(0.72252)	[2]expenments(0.21907)
	[3]expedments(0.04694)	[4]experments(0.00406)
	[5]expeiments(0.00331)	[6]xperiments(0.00157)
	[7]experimelts(0.00096)	[8]xpenments(0.00048)
	[9]expenmelts(0.00029)	[10]experimen ts(0.00024)
stage	[1]stage(0.99247)	[2]stag<(0.00395)
	[3]sage(0.00164)	[4]tage(0.00113)
	[5]stadge(0.00035)	[6]s tage(0.00017)
	[7]s*tage(0.00017)	[8]st'age(0.00011)
	[9]sag<(0.00001)	[10]tag<(0.00000)

検索語	[ランク] 拡張検索文字列 (得点)	
technique	[1]technique(0.97014)	[2]techniquc(0.01540)
	[3]technique(0.00936)	[4]tchnique(0.00287)
	[5]tl`chnique(0.00144)	[6]techn ique(0.00032)
	[7]teechniquc(0.00015)	[8]te'chnique(0.00005)
	[9]te`chnique(0.00005)	[10]tec hnique(0.00005)
matching	[1]matching(0.97187)	[2]matchin6(0.00927)
	[3]snatching(0.00448)	[4]match;ng(0.00343)
	[5]matchlng(0.00343)	[6]matchillg(0.00225)
	[7]matchintj(0.00185)	[8]matchilg(0.00075)
	[9]matchisg(0.00075)	[10]matchiog(0.00075)
geometry	[1]geometry(0.96314)	[2]geometr,v(0.02007)
	[3]6eometry(0.00535)	[4]g<ometry(0.00384)
	[5]geosnetry(0.00304)	[6]geomefry(0.00166)
	[7]geomeffry(0.00166)	[8]geome'try(0.00034)
	[9]geome`try(0.00034)	[10]geomet'ry(0.00012)
algorithm	[1]algorithm(0.75695)	[2]algonthm(0.21998)
	[3]algorthm(0.00425)	[4]a]gorithm(0.00397)
	[5]algonithm(0.00227)	[6]algonithm(0.00227)
	[7]algoithm(0.00227)	[8]algorihm(0.00133)
	[9]a]gonthm(0.00115)	[10]aigorithm(0.00099)
scene	[1]scene(0.98349)	[2]seene(0.01058)
	[3]scvne(0.00149)	[4]scxvne(0.00149)
	[5]scele(0.00131)	[6]cene(0.00112)
	[7]s cene(0.00018)	[8]s*cene(0.00018)
	[9]sce`ne(0.00004)	[10]sce'ne(0.00004)

検索語	[ランク] 拡張検索文字列 (得点)	
enhancement	[1]enhancement(0.97402)	[2]enhancement(0.00660)
	[3]enhancement(0.00425)	[4]enhancement(0.00318)
	[5]enhancement(0.00212)	[6]enhancement(0.00148)
	[7]enhancement(0.00148)	[8]enhancement(0.00130)
	[9]enhancement(0.00130)	[10]enhancement(0.00106)
detection	[1]detection(0.70062)	[2]detection(0.16354)
	[3]detection(0.05451)	[4]detection(0.05451)
	[5]detection(0.00676)	[6]detection(0.00234)
	[7]detection(0.00207)	[8]detection(0.00166)
	[9]detection(0.00158)	[10]detection(0.00135)
stereo	[1]stereo(0.97977)	[2]stereo(0.00606)
	[3]stereo(0.00448)	[4]stereo(0.00290)
	[5]stereo(0.00202)	[6]stereo(0.00161)
	[7]stereo(0.00145)	[8]stereo(0.00111)
	[9]stereo(0.00016)	[10]stereo(0.00016)
advantages	[1]advantages(0.99133)	[2]advantages(0.00395)
	[3]advantages(0.00108)	[4]advantages(0.00108)
	[5]advantages(0.00108)	[6]advantages(0.00063)
	[7]advantages(0.00035)	[8]advantages(0.00033)
	[9]advantages(0.00011)	[10]advantages(0.00004)
results	[1]results(0.97708)	[2]results(0.01007)
	[3]results(0.00604)	[4]results(0.00231)
	[5]results(0.00231)	[6]results(0.00201)
	[7]results(0.00006)	[8]results(0.00002)
	[9]results(0.00002)	[10]results(0.00002)

検索語	[ランク] 拡張検索文字列 (得点)	
surface	[1]surface(0.88429)	[2]surface(0.04780)
	[3]surtace(0.04780)	[4]surfaee(0.01028)
	[5]sumface(0.00352)	[6]surfacy(0.00134)
	[7]surfacy(0.00134)	[8]urface(0.00101)
	[9]surtaee(0.00056)	[10]surlaee(0.00056)
reconstruction	[1]reconstmction(0.48116)	[2]reconstruction(0.21309)
	[3]reconstmcdon(0.11231)	[4]reconstrucion(0.04974)
	[5]reconstmcson(0.03744)	[6]reconstmchon(0.03744)
	[7]reconstruchon(0.01658)	[8]reconstrucion(0.01658)
	[9]reconstmction(0.00464)	[10]reconstruction(0.00352)
control	[1]control(0.97659)	[2]eontrol(0.00849)
	[3]contro](0.00708)	[4]contr`l(0.00524)
	[5]centrol(0.00195)	[6]con trol(0.00032)
	[7]cont`rol(0.00012)	[8]eontro](0.00006)
	[9]econtr`l(0.00005)	[10]contr`](0.00004)
region	[1]region(0.95513)	[2]re6ion(0.01619)
	[3]reg;on(0.00724)	[4]reglon(0.00724)
	[5]rregion(0.00591)	[6]legion(0.00226)
	[7]legion(0.00226)	[8]region(0.00197)
	[9]re`gion(0.00055)	[10]re'gion(0.00055)
constraint	[1]constraint(0.92244)	[2]constrahnt(0.05309)
	[3]econstraint(0.00802)	[4]constrahnt(0.00664)
	[5]constraiilt(0.00214)	[6]censtraint(0.00184)
	[7]constraint(0.00152)	[8]constraist(0.00071)
	[9]constraiot(0.00071)	[10]constraiilt(0.00071)

検索語	[ランク] 拡張検索文字列 (得点)	
development	[1]development(0.97074)	[2]deve]opment(0.01412)
	[3]deve]opment(0.00353)	[4]develop`ment(0.00274)
	[5]developwment(0.00274)	[6]levelopment(0.00230)
	[7]dcvelopment(0.00187)	[8]developmelt(0.00129)
	[9]developmen t(0.00032)	[10]developme`nt(0.00004)
method	[1]method(0.78186)	[2]mdhod(0.20879)
	[3]snethod(0.00360)	[4]method(0.00135)
	[5]meffhod(0.00135)	[6]sndhod(0.00096)
	[7]mete`od(0.00050)	[8]metbod(0.00050)
	[9]me`thod(0.00027)	[10]me`thod(0.00027)
contours	[1]contours(0.98279)	[2]eontours(0.00855)
	[3]contoums(0.00392)	[4]conturs(0.00236)
	[5]centours(0.00197)	[6]con tours(0.00032)
	[7]eontoums(0.00003)	[8]eonturs(0.00002)
	[9]eentours(0.00002)	[10]contums(0.00001)
test	[1]test(0.99361)	[2]tst(0.00294)
	[3]tes(0.00164)	[4]tl`st(0.00147)
	[5]tes*t(0.00017)	[6]tes t(0.00017)
	[7]ts(0.00000)	[8]tl`s(0.00000)
	[9]ts*t(0.00000)	[10]ts t(0.00000)
curve	[1]curve(0.98745)	[2]eurve(0.00859)
	[3]cumve(0.00393)	[4]eumve(0.00003)

検索語	[ランク] 拡張検索文字列 (得点)	
structure	[1]stmctux(0.65341)	[2]structux(0.28937)
	[3]stmcture(0.03321)	[4]structure(0.01471)
	[5]structux(0.00478)	[6]smctux(0.00108)
	[7]tmctux(0.00074)	[8]sructux(0.00048)
	[9]tructux(0.00033)	[10]structure(0.00024)
camera	[1]camera(0.98642)	[2]eamera(0.00858)
	[3]camea(0.00451)	[4]cadmera(0.00012)
	[5]c amera(0.00011)	[6]cnamera(0.00011)
	[7]came'ra(0.00006)	[8]came`ra(0.00006)
	[9]eamea(0.00004)	[10]eadmera(0.00000)
implementation	[1]implementation(0.74092)	[2]implementadon(0.12945)
	[3]implementabon(0.06472)	[4]implementahon(0.04315)
	[5]impementation(0.00315)	[6]imp]ementation(0.00315)
	[7]implementation(0.00248)	[8]implen`entation(0.00242)
	[9]Implementation(0.00233)	[10]implementaton(0.00124)
issues	[1]issues(0.97482)	[2]issucs(0.01547)
	[3]is5ues(0.00439)	[4]Issues(0.00307)
	[5]issues(0.00156)	[6]dssues(0.00051)
	[7]is5ucs(0.00007)	[8]lssucs(0.00005)
	[9]isucs(0.00002)	[10]ls5ues(0.00001)
relationship	[1]relationship(0.72557)	[2]reladonship(0.12677)
	[3]relabonship(0.06338)	[4]relahonship(0.04226)
	[5]re]ationship(0.01055)	[6]rlationship(0.00449)
	[7]retationship(0.00264)	[8]relationshp(0.00256)
	[9]relationshp(0.00256)	[10]relatlonship(0.00242)

検索語	[ランク] 拡張検索文字列 (得点)	
object	[1]object(0.98441)	[2]objeet(0.00950)
	[3]olvject(0.00399)	[4]bjeet(0.00095)
	[5]ob ject(0.00045)	[6]objeent(0.00026)
	[7]objec t(0.00026)	[8]obje`ct(0.00005)
	[9]obje`ct(0.00005)	[10]olvjeet(0.00004)
recognition	[1]recognition(0.82665)	[2]recogndion(0.09178)
	[3]recognidon(0.04265)	[4]recognition(0.00797)
	[5]reco6nition(0.00656)	[6]rcognition(0.00511)
	[7]recognition(0.00276)	[8]lrecognition(0.00195)
	[9]lrecognition(0.00195)	[10]rccognition(0.00170)
system	[1]system(0.87543)	[2]syskm(0.11483)
	[3]system^(0.00286)	[4]systm(0.00259)
	[5]sysem(0.00144)	[6]systl`m(0.00130)
	[7]ysystem(0.00100)	[8]sys*tem(0.00015)
	[9]sys tem(0.00015)	[10]yskm(0.00013)
concept	[1]concept(0.98032)	[2]econcept(0.00852)
	[3]conecept(0.00428)	[4]cencept(0.00196)
	[5]concxvpt(0.00149)	[6]concvpt(0.00149)
	[7]concep`t(0.00082)	[8]concepwt(0.00082)
	[9]con cept(0.00015)	[10]econcept(0.00004)
paper	[1]paper(0.99497)	[2]pape(0.00455)
	[3]p`aper(0.00018)	[4]pwaper(0.00018)
	[5]pape`r(0.00006)	[6]pape`r(0.00006)
	[7]pwape(0.00000)	[8]p`ape(0.00000)
	[9]p`ape`r(0.00000)	[10]p`ape`r(0.00000)

検索語	[ランク] 拡張検索文字列 (得点)	
class	[1]class(0.98693)	[2]class(0.00858)
	[3]clas5(0.00445)	[4]elas5(0.00004)
range	[1]range(0.97677)	[2]ran6e(0.00932)
	[3]rang<(0.00389)	[4]lange(0.00231)
	[5]lange(0.00231)	[6]rantje(0.00186)
	[7]rarlge(0.00106)	[8]rallge(0.00106)
	[9]raoge(0.00106)	[10]ran ge(0.00013)
data	[1]data(0.99693)	[2]lata(0.00236)
	[3]d*ata(0.00050)	[4]dat'a(0.00011)
	[5]dadta(0.00010)	[6]l*ata(0.00000)
	[7]lat'a(0.00000)	[8]ladta(0.00000)
	[9]d*at'a(0.00000)	[10]d*adta(0.00000)
edge	[1]edge(0.85561)	[2]ed6e(0.13369)
	[3]edg<(0.00341)	[4]dge(0.00186)
	[5]eclge(0.00158)	[6]elge(0.00158)
	[7]ed*ge(0.00091)	[8]ed6<(0.00053)
	[9]d6e(0.00029)	[10]el6e(0.00025)
vision	[1]vision(0.93272)	[2]vijon(0.05888)
	[3]v:sion(0.00442)	[4]vion(0.00205)
	[5]viion(0.00149)	[6]v:jon(0.00028)
	[7]vis ion(0.00007)	[8]vis*ion(0.00007)
	[9]v:son(0.00001)	[10]v:ion(0.00001)

検索語	[ランク] 拡張検索文字列 (得点)	
combination	[1]combination(0.73973)	[2]combinadon(0.12924)
	[3]combinabon(0.06462)	[4]combinahon(0.04308)
	[5]eombination(0.00643)	[6]combination(0.00247)
	[7]cosnbination(0.00233)	[8]combillation(0.00171)
	[9]cembination(0.00148)	[10]combinaton(0.00124)
model	[1]model(0.97575)	[2]mode (0.01419)
	[3]suodel(0.00450)	[4]modet(0.00355)
	[5]mocl(0.00188)	[6]suode (0.00007)
	[7]modc (0.00003)	[8]snodet(0.00002)
	[9]suodcl(0.00001)	[10]modet(0.00001)
view	[1]view(0.99455)	[2]v;ew(0.00471)
	[3]vie~w(0.00037)	[4]vie'w(0.00037)
	[5]v;e~w(0.00000)	[6]v;e'w(0.00000)
problem	[1]problem(0.58276)	[2]pmblem(0.40725)
	[3]pr~blem(0.00313)	[4]prolvem(0.00236)
	[5]problen~(0.00190)	[6]pmblen~(0.00133)
	[7]paoblem(0.00102)	[8]pwroblem(0.00005)
	[9]p~roblem(0.00005)	[10]proble'm(0.00003)
position	[1]position(0.89467)	[2]posidon(0.04616)
	[3]posaion(0.04497)	[4]p(v)sition(0.00432)
	[5]positlon(0.00299)	[6]postion(0.00197)
	[7]posiion(0.00157)	[8]positon(0.00149)
	[9]positeon(0.00075)	[10]p(v)sidon(0.00022)

