Vol.2019-NL-242 No.7
2019/10/26

契約書のOCR漢字誤り訂正における偏旁冠脚を考慮した編集距離の検討

阪本 浩太郎[1,a)]　阿部川 明優[2,b)]　岸川 至白[2,c)]　阪本 エリーザ[3,d)]　石下 円香[4,e)]　渋木 英潔[4,f)]
森 辰則[1,g)]

概要：会計業務の効率化を目的として，紙の契約書のスキャンデータから情報を自動抽出するシステムが求められている．我々は，契約書の構造により，会社名，住所，氏名などが書かれている範囲を推定し，推定された範囲ごとに OCR の結果からテキストを抽出するシステムを開発しているが，抽出されたテキストに OCR に起因する文字誤りが発生し，漢字の偏旁冠脚の誤りが観察された．誤りが検出されたテキストを自動訂正するため，本稿では，偏旁冠脚を考慮した編集距離を用いて辞書に登録されている語に訂正する手法を提案する．また，一般的な編集距離と比較した結果を報告する．

A study on the edit distance taking kanji radicals into account in contracts' OCR kanji error correction

1. はじめに

企業や行政政府などの決算に関して会計士が監査を行う会計監査業務において，証憑書類を突き合わせて調査すること（証憑突合）は重要な業務のひとつである．多くの証憑書類は電子的に作成されるが紙の印刷物を当事者間で取り交わされることが多い．証憑突合の際には，会計士が印刷物のスキャンデータから必要な情報を抽出する作業を行っており，大きな負担となっている．このような背景により，証憑突合の効率化のために，証憑書類のスキャンデータからの情報抽出を自動的に行うシステムが求められている．

証憑書類には，契約書，見積書，仕入伝票，納品書，領収書など様々なものが含まれ，そのすべてを網羅的に扱うシステムの構築を目指し，その第一歩として本稿では契約書を扱う．契約書は，複数の当事者が契約を締結する際に作成される文書であり，仕入先の会社名，住所，代表者の氏名など様々な情報が記述される．我々は，契約書の構造より，氏名，会社名，住所，法律・会計用語などが書かれている範囲を推定し，推定された範囲ごとに OCR の結果からテキストを抽出するシステムを開発している．

我々は研究開発に利用可能な契約書のデータを収集しているが，証憑書類は機密情報に当たるため，データの収集が困難であり，現在使用できるデータは少量となっている[*1]．収集済みの契約書に対し，開発しているシステムで情報抽出を行った結果，OCR に起因する文字誤りが発生し，「注文書」を「往文書」と認識するといった漢字の偏旁冠脚誤りが観察された．

このように検出された誤りを自動訂正するために，本稿では，OCR 文字誤りがあった場合の，誤り訂正手法について検討する．OCR 誤りの検出については，本稿の対象外とする．氏名，会社名，住所，法律・会計用語等抽出したいデータの辞書を作成し，OCR 結果を辞書に登録されている最も類似する語に訂正する．

2 つの文字列間の類似度を測る尺度として，編集距離が

1 横浜国立大学
Yokohama National University
2 Genial Technology, Inc.
690 Saratoga Avenue, Suite 100 ,San Jose, CA 95129, USA
3 無所属
No affiliation
4 国立情報学研究所
National Institute of Informatics
a) sakamoto@forest.eis.ynu.ac.jp
b) aki@genialtech.io
c) itaru.kishikawa@genialtech.io
d) sakamoto.e612@gmail.com
e) ishioroshi@nii.ac.jp
f) shib@nii.ac.jp
g) mori@forest.eis.ynu.ac.jp

*1 今後，研究開発に使用できるデータは増える予定である

ある．先の例の OCR 結果「往文書」を訂正する場合，一般的な編集距離（Levenshtein 距離）を用いた場合を考えると，辞書の登録語「注文書」と「公文書」への編集距離が同じであり，どちらも訂正可能性がある．しかし，OCR での結果であることを踏まえれば，より見た目が似ている「注文書」への訂正が望ましいと考えられる．そこで，見た目の一致を考慮に入れるため，偏旁冠脚の部分一致や画数による類似性の分だけ，「往」から「注」への置換操作の方が，「往」から「公」への置換操作よりも編集コストが減るように拡張した，OCR による漢字誤り訂正のための編集距離を提案する．

提案した誤り訂正手法を実験により評価するため実際の契約書のデータを利用したいが，現在利用可能な契約書のデータ量が少なく，実験データとしては適さない．そこで，本稿では実験用のデータを独自に作成し，実験を行った．実験に使用したデータについては 4 章で述べる．偏旁冠脚を考慮した編集距離や一般的な編集距離を用いた自動訂正手法を比較した結果を報告する．

なお，使用した漢字編集距離を計算するプログラムは Github のリポジトリ[*2]に公開し，偏旁冠脚データは，そのリポジトリ内に設置した．

2. 本研究の対象

想定する全体の処理の流れを図 1 に示す．OCR 結果と，氏名，会社名，住所，用語といったデータ型を入力し，OCR 文字誤り検出を行う．誤りが検出されない場合は OCR 結果をそのまま出力する．誤りが検出された場合は，OCR 文字誤り訂正を行う．OCR 文字誤り訂正では辞書から OCR 結果と最も類似する登録語を取り出し，それを訂正結果として出力する．OCR 文字誤り訂正はデータ型によって処理の流れが異なる．

本稿では，OCR の誤りが検出された場合の OCR 結果の訂正部分のみに着目する．

使用した辞書データについては，3 章，偏旁冠脚データについては 5 章で述べる．2 つの文字列間の類似度を測る尺度として使用した変種距離については 8 章，手案する偏旁冠脚を考慮した編集距離を 9 章で述べる．データ型ごとの訂正方法については，6 章で述べる．

3. 辞書データ

本章では，作成した辞書データについて説明する．

3.1 人名辞書

人名辞書を NEologd[*3]の辞書から作成した．重複を削除し，アルファベット含むものとカタカナのみのものも削除

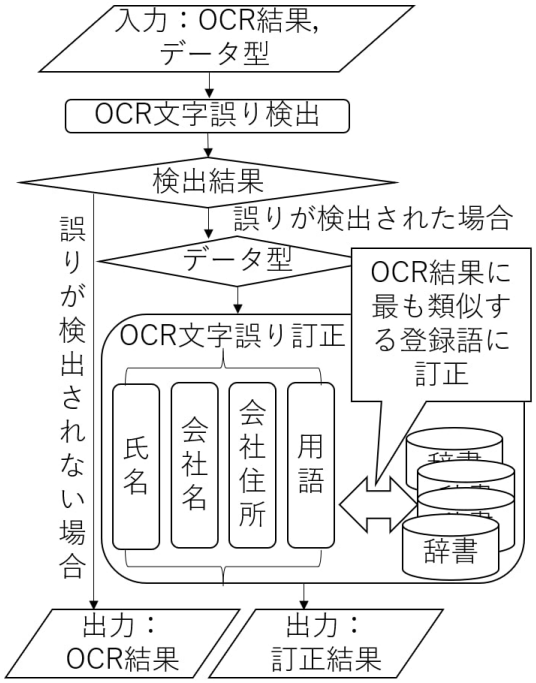

図 1 全体の処理の流れ

した．登録した苗字は 74,744 件，名前は 82,704 件である．なお，これらの内，ひらがなのみで構成された苗字は 0 件，名前は 408 件である．

3.2 法人名・法人住所辞書

法人名とその住所を登録した法人名・法人住所辞書を法人登記データ（令和元年 8 月 30 日更新版）[*4]から作成した．登録した法人数は 4,783,578 件である．法人登記データには 4,786,054 件の法人が登録されているが， 以下の理由により 2,476 件を削除した．

- 解散した法人は法人番号付与時と解散時のレコードがあり重複しているため、この重複を削除
- 国外のみに住所がある外国法人は削除

法人登記データから得られた法人情報は，法人住所を，都道府県，市区町村，番地・建物の 3 つに分解した上で，法人名から番地・建物，番地・建物から市区町村，市区町村から都道府県への参照を残す形で，それぞれ辞書に登録した．

3.3 法律・会計用語辞書

政府機関「日本法令外国語訳推進会議」による「法令用語日英標準対訳辞書」[*5]の見出し語 3,810 件と，経営者と士業者のマッチングサイト「士業ねっと！」による「会計用語キーワード辞典」[*6]の見出し語 932 件から作成した． 2 つの辞書の見出し語の重複は 117 件あり，重複を除き，合計 4,625 語を法律・会計用語辞書に登録した．

[*2] https://github.com/genial-technology/KanjiDamerauLevenshteinDistance/
[*3] https://github.com/neologd/mecab-ipadic-neologd
[*4] http://www.houjin-bangou.nta.go.jp/download/zenken/
[*5] http://www.japaneselawtranslation.go.jp/dict/download/
[*6] https://kaikei-yougo.sigyo.net/

4. OCR コーパス

本章では，実験に用いた OCR コーパスについて説明する．

人名辞書，法人名・法人住所辞書，法律・会計用語辞書からしたそれぞれ 1,100 件ずつをランダムに選択し，10.5pt で紙に印刷し，原則 300dpi でスキャンし，OCR にかけた．印刷用紙は，富士ゼロックス社のモノクロ用上質紙 V-paper（坪量 64g/㎡，紙厚 88 ㎛，白色度 82%）の A4 サイズを使用した．OCR は，LSTM に基づく OCR である Tesseract [*7]（4.1.0 版），Adobe Acrobat Pro DC (Continuous Release のバージョン 19.012.20040) に搭載されている OCR，富士ゼロックス社のプリンタ複合機 ApeosPort-VI C5571 に搭載されている OCR を用いた．Tesseract の学習モデルは，公開されている日本語 OCR 用の学習済みモデル（tessdata_best[*8]の jpn.traineddata）を用いた．フォントは和文契約書でよく使用される MS 明朝，MS P 明朝，メイリオ，游ゴシックの 4 種類を使用した．各辞書の見出し語を 4 等分した 275 件ずつ異なるフォントを用いた．人名辞書には，苗字と名前がそれぞれ別々に登録されているが，苗字と名詞を組み合わせて「苗字＋半角スペース文字＋名前」を氏名とし，氏名を単位としてコーパスを作成した．

表 1 は，OCR ごとの文字列レベルの正答率である．OCR A，B，C は，上記に述べた 3 つの OCR のいずれかにそれぞれ対応する．正解情報と OCR 結果に対し文字列正規化を行った後に，２つの文字列の表層完全一致により，正答率を調査した．文字列正規化として，文字列から空白文字，半角縦線記号「| (U+007C)」，半角ピリオド「. (U+002E)」等の記号を除去をしたり，横棒系の記号を半角ハイフン「- (U+002D)」に統一した上で[*9]，Unicode の NFKC フォームによる文字列正規化を行った．

氏名は OCR B で 0.945，法人名，法人住所，用語は OCR A でそれぞれ 0.945，0.974，0.991 が最大正答率である．また，フォントによって最大正答率となる OCR が異なることがなく，最大正答率の揺れも小さいため，フォントによる正答率への影響は低いものと考えられる．

5. 偏旁冠脚データ

偏旁冠脚とその画数を考慮した編集距離を計測するために処理の内部で用いる，偏旁冠脚の情報を登録したデータを，漢字オンライン[*10]で公開されている漢字構成から作成した．漢字辞典オンライン上の漢字ごとの HTML ファイルをダウンロードし，漢字構成に記載されている偏旁冠脚についての情報を抽出し，図 3 のような JSON ファイルに変換した．偏旁冠脚データの JSON 構造について説明する．この JSON の根は配列であり，一つの漢字に対応するオブジェクトが要素である．一つの漢字に対応するオブジェクトは，original 属性や steps 属性を持つ．original 属性の値は対応する漢字である．steps 属性については，漢字辞典オンラインの漢字構成が図 2 のように表示されており，上から下にかけて，元の漢字から少しずつ変更されているため，各行を step とみなしそれに対応する step オブジェクトを変更順に配列 steps に納めた．step オブジェクトの中の step 属性の値は漢字構成の何行目かを指し，最初の行は元の漢字のままであるがその行を 0 行目として，以降，下に進むにつれて 1 行ずつ増える．step オブジェクトの中の hbkkList 属性の値は，偏旁冠脚のオブジェクトの配列となっており，各オブジェクトは hbkk 属性と numOfStrokes 属性を持つ．hbkk 属性の値は偏旁冠脚であり，numOfStrokes 属性の値はその偏旁冠脚の画数である．

6. OCR 結果の訂正

本章では，データの型ごとの訂正処理の流れを説明する．

表 1　OCR の正答率

OCR	フォント	氏名	法人名	法人住所	用語
A	総合	0.901	**0.945**	**0.975**	**0.991**
	(/1,100)	991	**1,039**	**1,073**	**1,090**
	MS 明朝	0.884	**0.935**	**0.971**	**0.993**
	(/275)	243	**257**	**267**	**273**
	MS P 明朝	0.898	**0.964**	**0.967**	**0.989**
	(/275)	247	**265**	**266**	**272**
	游ゴシック	0.927	**0.945**	**0.993**	**0.989**
	(/275)	255	**260**	**273**	**272**
	メイリオ	0.895	**0.935**	**0.971**	**0.993**
	(/275)	246	**257**	**267**	**273**
B	総合	**0.945**	0.860	0.905	0.940
	(/1,100)	**1,040**	946	996	1,034
	MS 明朝	**0.945**	0.895	0.942	0.942
	(/275)	**260**	246	259	259
	MS P 明朝	**0.960**	0.865	0.895	0.975
	(/275)	**264**	238	246	268
	游ゴシック	**0.938**	0.884	0.916	0.916
	(/275)	**258**	243	252	252
	メイリオ	**0.938**	0.796	0.869	0.927
	(/275)	**258**	219	239	255
C	総合	0.596	0.821	0.783	0.927
	(/1,100)	656	903	861	1,020
	MS 明朝	0.535	0.760	0.753	0.891
	(/275)	147	209	207	245
	MS P 明朝	0.531	0.778	0.698	0.938
	(/275)	146	214	192	258
	游ゴシック	0.673	0.884	0.847	0.942
	(/275)	185	243	233	259
	メイリオ	0.647	0.862	0.833	0.938
	(/275)	178	237	229	258

[*7] https://github.com/tesseract-ocr/tesseract/
[*8] https://github.com/tesseract-ocr/tessdata_best/
[*9] 漢数字の「一 (U+3192)」と長音記号「ー (U+30FC)」は除く
[*10] https://kanji.jitenon.jp/

図 2　漢字オンラインの漢字構成の例

```
[
  ...
  {
    "original": "勇",
    "steps": [
      {
        "step": 0,
        "hbkkList": [
          {
            "hbbk": "勇",
            "numOfStrokes": 9
          }
        ]
      },
      {
        "step": 1,
        "hbkkList": [
          {
            "hbbk": "甬",
            "numOfStrokes": 7
          },
          {
            "hbbk": "力",
            "numOfStrokes": 2
          }
        ]
      },
      ...
    ]
  },
  ...
]
```

図 3　偏旁冠脚データの例

6.1　氏名の訂正

人名辞書を用いて OCR 結果を訂正する．OCR 結果と最も類似する苗字を辞書から選択する．この選択の際には，登録語の文字列長だけ OCR 結果の先頭から切り取った部分文字列と登録語の間で計測した類似度を用いる．選択された苗字の字数分だけ OCR 結果の先頭文字を削除する．苗字を削除した OCR 結果と最も類似する名前を人名辞書から選択する．選択された苗字と選択された名前をこの順で結合したものを訂正結果とする．

6.2　法人名の訂正

法人名・法人住所辞書を用いて OCR 結果を訂正する．OCR 結果と最も類似する会社名を辞書から選択し，訂正結果とする．

6.3　法人住所の訂正

法人名・法人住所辞書を用いて OCR 結果を訂正する．まず，OCR 結果と最も類似する都道府県を辞書から選択する．この選択の際には，登録語の文字列長だけ OCR 結果の先頭から切り取った部分文字列と登録語の間で計測した類似度を用いる．選択された都道府県の字数だけ OCR 結果の先頭文字を削除する．次に，その都道府県に紐づく市区町村の中で，都道府県を削除された OCR 結果と最も類似するものを辞書から選択する．この選択の際にも，登録語の文字列長だけ都道府県を削除された OCR 結果の先頭から切り取った部分文字列と登録語の間で計測した類似度を用いる．選択された市区町村の字数だけ都道府県を削除された OCR 結果の先頭文字を削除する．最後に，その市区町村に紐づく番地・建物の中で，都道府県と市区町村を削除された OCR 結果と最も類似するものを辞書から選択し，選択された都道府県，市区町村，番地・建物をこの順で結合したものを訂正結果とする．

6.4　用語の訂正

法律･会計用語辞書を用いて OCR 結果を訂正する．OCR 結果と最も類似する用語を辞書から選択し，訂正結果とする．

7.　関連研究

日本語の OCR 誤り訂正手法については，Nagata[1] が，文字混合確率モデル及び言語モデルを用いた手法が提案している．この手法は文字の置換誤りにのみ対応可能である．Neubig ら [7] は，文字誤り訂正のための雑音のある通信路モデルを提案し，文字の融合や分離の誤りにも対応している．増田 [8] は，文字列中の各箇所ごとに訂正候補を作成するのではなく，大域的な情報を用いて，特定の文字に対し出現箇所によらない訂正候補の作成を行った．

我々の研究では，証憑書類を扱っており，大局的な情報は用いることができないという点でこれらの研究と異なる．

Peng ら [2] は中国語の名前照合の方法として，文字列照合と学習によるアプローチの両方の方法の評価を行っている．文字列照合として，Levenshtein 距離と Jaro-Winkler 距離を用いて比較・検討を行っている．本研究では，OCR の誤り訂正を目的として，編集距離を使用する．そして，漢字の内部情報の類似性により編集距離を補正している．

8. 編集距離

本章では，2 つの文字列間の類似度の計測に用いた編集距離について述べる．

8.1 Levenshtein 距離

Levenshtein 距離は，1 つの文字列を別の文字列に変形するのに必要な手順の最小コストとして定義される [4]．操作は文字の挿入 (insertion)，削除 (deletion)，置換 (substitution) であり，それぞれを行うためのコストは一般的に 1 を用いる．その場合の距離は，手順の最小回数と一致する．なお，置換は挿入と削除の組でも実現できるため，挿入のコストと削除のコストの合計が置換のコストを下回ると置換は起こらない点に留意されたい．

8.2 Damerau-Levenshtein 距離

Damerau-Levenshtein 距離は，Levenshtein 距離と同様に，1 つの文字列を別の文字列に変形するのに必要な手順の最小コストとして定義される．Damerau-Levenshtein 距離での操作には，Levenshtein 距離で定義する 3 つのものに加え、操作として隣接文字交換 (transposition) が含まれている [3]．Levenshtein 距離と同様に，隣接文字交換で発生するコストは一般的に 1 が用いられる．なお，隣接文字交換は挿入と削除の組でも実現できるため，挿入のコストと削除のコストの合計が隣接文字交換のコストを下回ると隣接文字交換は起こらない点に留意されたい．

8.3 Jaro 距離

Jaro 距離 Φ は文字列 s_1,s_2 の長さと，文字列中の部分的な間違いの数から構成され，以下の式で定義される [5]．

$$\Phi = W_1 \cdot \frac{c}{d} + W_2 \cdot \frac{c}{r} + W_t \cdot \frac{c - \tau}{c} \tag{1}$$

ただし，$c = 0$ で $\Phi = 0$ とする．ここで，各記号は以下のように定義される．

W_1：文字列 s_1 に掛かる重み $[0, 1]$
W_2：文字列 s_2 に掛かる重み $[0, 1]$
W_t：置き換えに掛かる重み $[0, 1]$
d：文字列 s_1 の長さ $[1,)$
r：文字列 s_2 の長さ $[1,)$
τ：「置換」が必要な文字数 $[0,) \ / \ 2$
c：s_1 と s_2 のある区間内で一致する文字数 $[0,)$

$W_1 + W_2 + W_t = 1$ であり，$W_1 = W_2 = W_t = \frac{1}{3}$ とするのが一般的である．

8.4 Jaro-Winkler 距離

Jaro-Winkler 距離 Φ_w は Jaro 距離 Φ を使って，以下の式で定義される [6]．

$$\Phi_w = \Phi + i \cdot p \cdot (1 - \Phi) \tag{2}$$

i は s_1 と s_2 で一致する接頭辞 (先頭文字からの連続する文字列) で何文字目までを考慮に入れるかの文字数であり，最大値は 4 である．p は i に対するスケーリング因数（0.25 以下の定数）であり，$p = 0.1$ が標準的な値として用いられる．

9. 漢字 Damerau-Levenshtein 距離

OCR による漢字誤り訂正を目的として，漢字置換の際に偏旁冠脚の部分一致や画数といった内部情報の類似性の分だけ漢字置換コストを和らげ編集距離が縮まるように拡張した漢字 Damerau-Levenshtein 距離について説明する．

Damerau-Levenshtein 距離は，文字レベルでの編集距離を計測するが，漢字 Damerau-Levenshtein 距離は「往文書」と「注文書」の例のように，置換操作が発生する「往」と「注」に対して，単純な置換コストを与えるのではなく，それらの漢字の内部情報の類似性に注目し，類似する分だけ置換コストを和らげるように Damerau-Levenshtein 距離に対し補正する．そこで，漢字の置換コストを和らげるための漢字の内部情報の類似度は偏旁冠脚列間の Damerau-Levenshtein 距離を基に算出する．この偏旁冠脚の編集距離を計測する際に，「往」と「注」の例のように，置換操作が発生する「氵（さんずいへん)」と「彳（ぎょうにんべん)」に対して，単純な置換コストを与えるのではなく，それらの偏旁冠脚の内部情報の類似性に注目し，類似する分だけ置換コストを和らげるように偏旁冠脚列間の Damerau-Levenshtein 距離に対し補正する．そこで，偏旁冠脚の置換コストを和らげるための偏旁冠脚の内部情報の類似性として，線に注目し画数の差を基に算出する．

文字の配列をそれぞれ K_1 と K_2 と表し，それらの要素である文字 k_1 と k_2 の偏旁冠脚の配列をそれぞれ H_1 と H_2 と表し，それらの要素である偏旁冠脚 h_1 と h_2 の画数をそれぞれ S_1 と S_2 と表す．

漢字 Damerau-Levenshtein 距離は，次のように補正された Damerau-Levenshtein 距離 $DL_k^*(K_1, K_2)$ のことである．

文字の配列 K_1 と K_2 の間の Damerau-Levenshtein 距離 $DL_k(K_1, K_2)$ から，補正された Damerau-Levenshtein 距離 $DL_k^*(K_1, K_2)$ への変換式は次である．

$$DL_k^*(K_1, K2) = DL_k(K_1, K_2) - Sim_k(K_1, K_2) \tag{3}$$

ここで，$Sim_k(K_1, K_2)$ は，次の式のように，文字の配列 K_1 と K_2 の間で文字置換コスト R_k が発生するペアのすべてに対して，偏旁冠脚の編集距離に注目した文字間の類似度を計測し，それらを総和したものである．なお，文字置換コストが発生するペアが漢字のペアではない場合は，類似度は 0 である．

$$Sim_k(K_1, K_2) = \sum (R_k \times sim_k(k_1, k_2)) \tag{4}$$

漢字 k_1 と k_2 の間の類似度は次の式のように求める．

$$sim_k(k_1, k_2) = 1 - \frac{DL_h^*(H_1, H_2)}{max(L_{H_1}, L_{H_2})} \quad (5)$$

L_{H_1} と L_{H_2} はそれぞれ H_1 と H_2 の配列長である．また，$DL_h^*(H_1, H_2)$ は，偏旁冠脚の配列 H_1 と H_2 の間の Damerau-Levenshtein 距離 $DL_h(H_1, H_2)$ を補正したものであり，変換式は次である．

$$DL_h^*(H_1, H_2) = DL_h(H_1, H_2) - Sim_h(H_1, H_2) \quad (6)$$

ここで，$Sim_h(H_1, H_2)$ は，次の式のように，偏旁冠脚の配列 H_1 と H_2 の間で偏旁冠脚置換コスト R_h が発生するペアのすべてに対して，画数に注目した偏旁冠脚間の類似度を計測し，それらを総和したものである．

$$Sim_h(H_1, H_2) = \sum (R_h \times sim_h(h_1, h_2)) \quad (7)$$

偏旁冠脚 h_1 と h_2 の間の類似度は次の式のように求める．

$$sim_h(h_1, h_2) = 1 - min\left(\frac{|S_1 - S_2| + \alpha}{max(S_1, S_2)}, 1\right) \quad (8)$$

ここで，各記号は以下のように定義される．

S_1：h_1 の画数

S_2：h_2 の画数

α：h_1 と h_2 が不一致だが S_1 と S_2 が一致する場合にコストが 0 にならないようにするための補正

ところで，偏旁冠脚データには，ひとつの漢字に対し複数の偏旁冠脚の配列が登録されているため，漢字 Damerau-Levenshtein 距離の計測のためにどの偏旁冠脚の配列を使用するかの決定方法について説明する．漢字 k_1 が持つ偏旁冠脚の配列の集合と漢字 k_2 が持つ偏旁冠脚の配列の集合の間で，総当たりで偏旁冠脚配列間の類似度を計算し，最も類似度が高い偏旁冠脚の配列のペアをそれぞれ k_1 と k_2 の偏旁冠脚の配列 H_1 と H_2 として採用する．類似度としては，偏旁冠脚の配列の間の要素の重複数に対し，長い方の配列長で割った値を用いる．

10. 実験

提案した誤り訂正手法の性能を比較評価するため，OCR コーパスの OCR 結果がすべて誤りであると検出された前提で，誤り訂正を行った結果に対し，正答率で評価する．正答率は，文字列正規化を行った上で，訂正結果と正解情報の間の文字列の表層完全一致による一致率で計算する．誤り訂正の手法としては，2 章で提案した処理の流れに沿って，類似度として次の 3 つの編集距離を基にした値を用いる．

- 漢字 Damerau-Levenshtein 距離
- Damerau-Levenshtein 距離
- Jaro-Winkler 距離

Jaro-Winkler 距離は 0 以上 1 以下で値が大きいほど類似していることを示す類似度であるためそのまま用いるが，漢字 Damerau-Levenshtein 距離，及び Damerau-Levenshtein 距離は，0 以上の実数を取り，値が小さいほど類似していることを示す非類似度である．そのため，非類似度から類似度に変換するため次の式を用いる．

$$sim(s1, s2) = 1 - \frac{distance(s1, s2)}{max(length(s1), length(s2))} \quad (9)$$

漢字 Damerau-Levenshtein 距離，Damerau-Levenshtein 距離の両方について，編集操作により発生するコストはその種類に寄らずすべて 1 とする．漢字 Damerau-Levenshtein 距離のパラメタ α は 1 とする．また，Jaro-Winkler 距離の計算では，Jaro 距離が閾値以上となると Jaro 距離に補正項を加えるが，その閾値を 0.7 とし，スケーリング因子を 0.1 とする．

使用する辞書については，3 章で述べた辞書のうち，人名辞書と法律・会計用語辞書はそのまま用いたが，法人名・法人住所辞書については登録されている件数が多く，処理に時間が掛かりすぎたため，代わりに OCR コーパスの正解情報のリストを簡易辞書として用いて実験を行った．

何も行わず OCR 結果をそのまま出力した場合の正答率（表 1 に示したもの）とも比較する．

11. 結果と考察

実験の結果を表 2 に示す．訂正手法が (none) は誤り訂正をしなかった場合の正答率であるが，訂正を行った方がいずれも正答率は高まった． 3 つの訂正手法を比較すると，漢字 Damerau-Levenshtein 距離と Damerau-Levenshtein 距離のどちらかあるいは両方が最も高い正答率となった．漢字 Damerau-Levenshtein 距離と Damerau-Levenshtein 距離を比較すると，法人名の訂正もしくは OCR B の OCR 結果の訂正においては Levenshtein 距離の方が正答率が高く，それ以外は近い値となった．

誤り訂正の結果を観察すると，OCR A による「念治 二奈」の OCR 結果「..o.... ムー本芯 I ローカ」のような文字や漢字の内部情報を用いても訂正することは困難だと思われる誤りが見つかった．このような誤りは OCR A の氏名の結果に 1 件，用語の結果に 3 件，OCR B の氏名の結果に 4 件，用語の結果に 28 件，OCR C の用語の結果に 4 件観察され， 2 ～ 4 文字程度の短い語に対して起こりやすい現象だと考えらえる．OCR B による「ヘッジ会計」の OCR 結果「ノ＼ツジ会!言十」は，隣接する文字を組み合わせることで元の文字に復元できるような誤りであり，本稿で提案した編集距離ではこの類似性を考慮することができていないことがわかった．

12. まとめと今後の課題

会計業務の効率化を目的として，紙の契約書のスキャ

表 2　実験結果の正答率

OCR	訂正手法	氏名	法人名	法人住所	用語
A	漢字 DL	**0.906 (997/1,100)**	0.970 (1,067/1,100)	**1.000 (1,100/1,100)**	**0.995 (1,094/1,100)**
	DL	0.903 (993/1,100)	**0.976 (1,074/1,100)**	**1.000 (1,100/1,100)**	0.994 (1,093/1,100)
	JW	0.903 (993/1,100)	0.975 (1,072/1,100)	0.998 (1,098/1,100)	0.994 (1,093/1,100)
	(none)	0.901 (991/1,100)	0.945 (1,039/1,100)	0.975 (1,073/1,100)	0.991 (1,090/1,100)
B	漢字 DL	0.946 (1,041/1,100)	0.926 (1,019/1,100)	0.995 (1,095/1,100)	0.958 (1,054/1,100)
	DL	**0.951 (1,046/1,100)**	**0.965 (1,061/1,100)**	**0.998 (1,098/1,100)**	**0.972 (1,069/1,100)**
	JW	0.950 (1,045/1,100)	0.963 (1,059/1,100)	0.994 (1,093/1,100)	0.969 (1,066/1,100)
	(none)	0.945 (1,040/1,100)	0.860 (946/1,100)	0.905 (996/1,100)	0.940 (1,034/1,100)
C	漢字 DL	**0.615 (677/1,100)**	0.917 (1,009/1,100)	**0.998 (1,098/1,100)**	**0.977 (1,075/1,100)**
	DL	0.615 (676/1,100)	**0.963 (1,059/1,100)**	0.994 (1,093/1,100)	**0.977 (1,075/1,100)**
	JW	0.610 (671/1,100)	0.962 (1,058/1,100)	0.985 (1,084/1,100)	0.976 (1,074/1,100)
	(none)	0.596 (656/1,100)	0.821 (903/1,100)	0.783 (861/1,100)	0.927 (1,020/1,100)

ンデータから情報を自動抽出するシステムの改良に向け，OCR により抽出されたテキストの OCR 文字誤りを自動訂正する手法を提案した．偏旁冠脚に注目した漢字 Damerau-Levenshtein 距離，Damerau-Levenshtein 距離，Jaro-Winkler 距離を用いた 3 つの手法を提案し，実験したところ，自動訂正による正答率の向上が認められ，データによって漢字 Damerau-Levenshtein 距離と Damerau-Levenshtein 距離のいずれかまたは両方が最大の正答率となった．より詳細な結果の考察は今後の課題である．

本来は実際の契約書のスキャンデータを用いて実験したいが，今回は十分な量が集まっていないため，OCR コーパスを作成し，それ対し提案する誤り訂正手法の正答率を比較した．契約書のスキャンデータを集めることは今後の課題である．OCR コーパスについて正答率による簡単な調査を行っているが，誤り分析を行っていないため，これは今後の課題である．今回の実験では編集コストを操作の種類に寄らず 1 に固定したが，訓練データから最適なコストを自動推定することは今後の課題である．コストを自動推定した結果，正答率がテストデータに依存していないかを交差検定の結果の分散値により調べたり，データ量による正答率の変化を見て収束するか，訓練データにテストデータで使用されるフォントが含まれていない場合の正答率への影響，OCR，氏名や住所といったデータ型，氏名の表記法，フォントなどに寄らず推定されるコストが安定するか等を調査したい．

謝辞　漢字辞典オンラインの漢字構成のデータ利用について快諾してくださった運営者の吉岡祐樹様に深謝する．

参考文献

[1] Masaaki Nagata.: *Japanese ocr error correction using character shape similarity and statistical language model* Proceedings of the 17th international conference on Computational linguistics-Volume 2. Association for Computational Linguistics, pp. 922–928, 1998.

[2] Nanyun Peng, Mo Yu, and Mark Dredze.: *An Empirical Study of Chinese Name Matching and Applications* Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), pages 377–383, Beijing, China, July 26–31, 2015.

[3] Damerau, Frederick J.: *A technique for computer detection and correction of spelling errors* Communications of the ACM 7(3): pages 659-664, 1964.

[4] Levenshtein, Vladimir Iosifovich: *Binary codes capable of correcting deletions, insertions, and reversals* Soviet Physics Doklady. 10 (8): pages 707–710, February 1966.

[5] Jaro, M. A. *Advances in record linkage methodology as applied to the 1985 census of Tampa Florida* Journal of the American Statistical Association. 84 (406): pages 414–20, 1989.

[6] Winkler, W. E. *String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage* Proceedings of the Section on Survey Research Methods. American Statistical Association: pages 354–359, 1990.

[7] Graham Neubig, 森信介, 河原達也. 重み付き有限状態トランスデューサーを用いた文字誤り訂正. 言語処理学会第 15 回年次大会, pp. 332–335, 2009.

[8] 増田 勝也. 大域的情報を用いた OCR 文字誤り訂正. 言語処理学会 第 21 回年次大会, pp. 127–130, 2015.