類推関係手法による漢字の文字構造の再発見

松下 浩太　　ルパージュ・イヴ
早稲田大学大学院 情報生産システム研究科
ips-takozo1008@akane.waseda.jp, yves.lepage@waseda.jp

1　はじめに

1.1　背景・目的

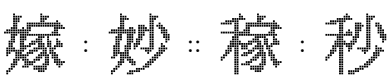
類推関係の概念は、近年の著者ら（[1]、[4]）によって、様々な方法で定義されている。本概念は従来の定義に遡って参照することで、以下の定義に基づき結論づけることができる：

> ４つの要素である A, B, C, D は類推関係 (proportional analogy) にある時、最初の要素は２番目の要素と同様に、３番目の要素は４番目の要素と同様の手法で導出する。類推関係は　$A : B :: C : D$　と表記する。

一般的に、２要素間の関係（”:”コロンで表記する）が比で表現される場合、類推関係とは２要素間の対の相似を指す。類推関係は、形のレベル、または意味のレベル、またその双方の単語間や文間に成立する。下記の例は日本語における類推関係の例である。

形レベル	残す : 残した :: あす : あした
意味レベル	残す : 残した :: 走る : 走った
形/意味レベル	残す : 残した :: 目指す : 目指した

また、下記の例は、４つの中日文字間の図像の類推関係を示している [6] 。



図像を文字として捉えると、黒と白のピクセルの間で類推関係が成立していることを示している。文字の左側の部分と右側の部分を明示的に構成要素を分解することによって、４つの異なる文字が生じるように転換でき、漢字間の相似を計算することが出来る。

例では、「嫁（配偶者)」「妙（奇妙)」「稼（稼ぐ)」「秒（時間)」となり、意味のレベルでは全く関係性はなく意味の類推関係は適用されない。また、いずれも発音のレベルでの類推関係は適用されない。

本論文では、類推関係の概念を用いて、対象となる対の構造を効率的に導出する方法を著したものである。類推関係は４つの要素を計算する際、$O(n^4)$ の計算量となる。類推関係のあるクラスターを計測する際、計算量が膨大となる本問題を改良し、$O(n^2)$ の複雑性に置き換えて計算量を減らすことによって、有益で無駄のない導出となる。

本研究では、漢字の学習における学びやすさに関わる広範な研究が存在する中、特に実用的な問題を、類推関係の概念を用いることによって、自動的に中国語漢字の図像構造を再発見する。

1.2　問題

類推関係は、　$A : B :: C : D$　の形式の他、７つの等しい形（下記を参照）が存在する。B と C の要素を転換したり、相似の対照（:: の印を基準とした両側の入れ換え）を行うことによって、次の８つの類推関係が表現出来る：

$$A : B :: C : D \qquad A : C :: B : D$$
$$C : D :: A : B \qquad C : A :: D : B$$
$$B : A :: D : C \qquad B : D :: A : C$$
$$D : B :: C : A \qquad D : C :: B : A$$

８つの類推関係の形式を取るため、導出する時間は８つの要素によって分けることが出来るが、$O(n^4)$ の複雑性を含んだ計算に基づいた確認作業が必要となる。

上記 $O(n^4)$ の計算量に沿って、36 の属性を使用し（属性の説明については、3.3 節を参照）４つの中日文字の類推関係の検証に必要な平均時間を推定した。平均時間は 0.8 ミリ秒となった。約 15000 の中日文字について（データの説明については、3.2 節を参照)、年間約 3.2×107 秒の時間を要するため、全ての可能な

関係性を得るために必要な時間は百万年を超える計算量となる。[1]

本論文では、上記の複雑性を変えることなく、計算時間を減らすことが出来る手法を提案する。各々の類推関係を導出するのではなく比の対として計算する方法を行う。以上に基づくと、計算量は基本的には$O(n^2)$となり、この出力はクラスタリングされた比の対として導出する。

2 類推関係クラスターについて

2.1 属性ベクトルとしての要素

本研究では、数値による属性のベクトルによって一つの要素を表す。また、全ての同じ要素に対して属性空間を表すので、各々が異なるベクトルとして２つの要素の間の比を定義することが出来る。

上記の設定の中、相似はベクトル間の等しさとして還元される。次の式は４つの次元空間内のベクトル間の類推関係の可能な場合を示している。

$$\begin{pmatrix}3\\6\\10\\7\end{pmatrix}-\begin{pmatrix}2\\6\\2\\3\end{pmatrix}=\begin{pmatrix}10\\8\\9\\5\end{pmatrix}-\begin{pmatrix}9\\8\\1\\1\end{pmatrix}$$

2.2 相似の推移性：類推関係クラスター

等しい値の関係の中で、推移性は自然に再帰性と対称性を加えた要素を保持している。類推関係については、相似の推移を意味している。

$$A:B::C:D \quad \text{かつ} \quad C:D::E:F \quad \Rightarrow \quad A:B::E:F$$

指定された範囲において、全要素間の類推関係を列挙する現在の処理では、相似の推移性によって導出される列挙を大幅に削減することができる。

$$\begin{array}{c}A:B\\C:D\\E:F\end{array}$$

結果、相似としての推移性を仮定とする中、指定されたクラスターで、全要素間の類推関係を列挙する問題に変換できる。前者の問題は、$O(n^4)$を複雑性がある一方、後者は$O(n^2)$の複雑性があることがわかる。

[1] $14,655^4 \times 0.8$ ミリ秒 $> 14^4 \times 10^{12} \times 0.8$ ミリ秒
$> 48 \times 10^{12}$ 秒
$> 48 \times 10^{12} / (3.2 \times 10^7)$ 年
$> 1.5 \times 10^6$ 年

2.3 属性木

本論文では上記に加え、属性木の概念も用いる。図１は属性木の一例を示している。今回用いる全ての漢字を属性ベクトルとして表現し、ベクトルの最初の値から最後の値まで属性を含んだノードが生成される。また、ベクトルの最初の値を読み込んだ際、同じ値であれば、一つの値として結合する（レベル 1）。以上の過程を全ての場合に適用することによって、ベクトルの最初の値から最後の値までが一致すれば、対の相似として表現する。

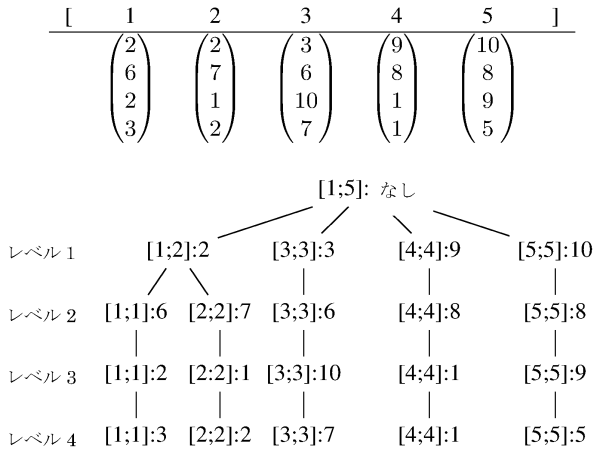


図 1: 属性木の一例

2.4 類推関係クラスターの列挙

全ての属性ベクトル間の対の計算は、幅優先順で並行して同じ属性木を探索し（デカルト積のように）、ノード各々の対の値の差を計算する。属性木の一般ノードは一つの要素だけではなく、限定的な段階で同じ属性の値の連続間隔を表す（図１に参照）。与えられた限定的な段階で同じ値の差がある時、全ての間隔の対はブロックとして記憶される。上記の一連の処理は、限られた段階で、各々の違った値を最終的な段階まで再帰的に適用する。最終的な段階で、ブロックはクラスターに変換される。

ある要素の１対 $A:B$ において、重複する $A:B$ を排除する。このようなクラスターは $A:B::A:B$ の類推関係以外には、いかなる類推関係も生じさせないため、出力データとしては無価値なものである。処理時間を削減する際、このような余剰を早期に発見することは重要である。

上記のように、不要なクラスターを早期発見した上で実装をする際、属性木のデータ構造（図１の例を参照）に基づいて行う。

漢字数	なし	あり	割合
1,000	9	14	+55%
2,000	39	36	-7%
3,000	92	82	-10%
4,000	173	142	-17%
5,000	277	219	-20%
6,000	426	313	-26%
7,000	605	438	-27%
8,000	739	557	-24%
9,000	944	702	-25%
10,000	1204	836	-30%
11,000	1517	1123	-25%
12,000	1864	1302	-30%
13,000	2265	1342	-40%
14,000	2646	1791	-32%
14,655	2873	1889	-34%

表 1: 処理時間の比較（同一対のクラスター出力を排除した場合（なし）・含めた場合（あり））

木構造の下位の段階まで、出来うる全ベクトル間の全ての相違を計算した後に、値が重複する属性ベクトルが導出された場合、各々は一つの要素に縮小される。以上の過程でクラスターを直ちに削減することによって、過剰な計算を止める。

上記に挙げた表 1 は重複したクラスターの排除を行った場合・行わなかった場合の、双方の実行したプログラムの結果の比較を示している。今回、中日文字を構造化する特別な場合において、計算時間を約 3 分の 1 に削減することが達成可能であることを示している。

3 実験

本研究では、漢字を学ぶ学習者が、漢字の各々の形と発音を同時に覚えることができることを測定する研究が広大にある中、漢字の形とその発音の間の一致・規則性に着眼する。

中国語漢字はいくつかの構造を持ち、いくつかの図像の意味や、発音のいずれかで、図像としての要素から成り立っていることが知られている。

本研究の第一段階として、等幅フォントを使用し、中国と日本の漢字間における、出来うる類推関係を抽出し、以下に得られた結果の一部を示す。

3.1 漢字の構造

数は、前述の構成要素に隠された構造を示す。

漢字の最も知られている構造として、2 つの要素で構成される漢字である。一方は発音、もう一方は意味の関連性があると言われている。上記の関連性を図 2 [2]に示す。

本論文では、図像形式との関係にのみ着眼し、目標は、自動化による漢字の図像構造の抽出に限定されている。また、上記の構造体は一般的ではあるが、全ての漢字に対して有効な構造ではない。

京:先	左 / 共通部 (意味)
涼:洗	氵 [水]
凉:洗	冫 [氷]

氵:亻	右 / 共通部 (発音)
冼:伴	半 [PÀN]
凉:倞	京 [LIÀNG]

図 2: 左右の部分に意味及び発音の関連性を持っている

3.2 等幅フォントの漢字

等幅フォント（固定幅または固定サイズ）は、一定の高さと幅の黒と白の図像等で表す文字である。本研究では、フォントは knj10B.bdf[3]を使用し、漢字フォントとして使用可能な 14655 字を使用する。図 3 はこれらの文字のサンプルとして、3 つのランダムに選択された文字を可視化したものである。図に示すように、このフォントの文字は 18 行の固定の高さと 24 ピクセルと固定幅を持っている。

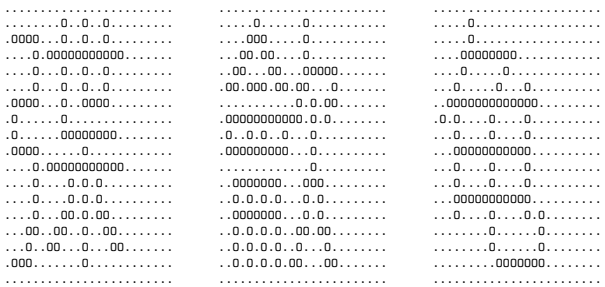


図 3: ドットと 0 によって可視化された文字の例

3.3 属性の使用

今回使用する属性は、各行と各列の黒のピクセル数を表している。1 文字あたり 36 の属性数の合計を構成し、18 行と 18 列から成り立つ。

例として、図の左端の文字の最初の属性を図 3 に示す。

0, 3, 7, 12, 4, 4, 9, 2, 9, 5, など

[2]フォントの表示に問題があるため、中国語漢字ではなく日本語漢字を使用した。

[3]長尾氏 (snagao@tkb.att.ne.jp) によって生成されたものを使用した (バージョン 1.1、1999 年)。

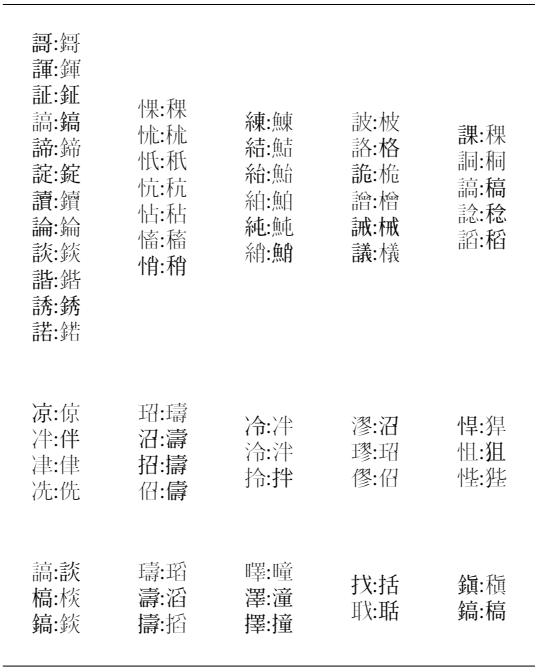


図 4: 出力されたクラスターの例

上記に挙げた 36 の属性は、実験で使用した 14655 文字の間で各々の文字を属性ベクトルとし、実際に各々の文字を 1 つの要素として表現した。

3.4 得られた類推関係クラスター

今回、選択された等幅フォントで 14655 文字の図像構造を抽出する方法を適用した。1.7GHz の CPU、4GB 内部メモリ、Intel Core i5 のプロセッサを搭載したマシンの条件下、プログラミング言語 Python で書かれたプログラムは、30 分未満の時間を要した。

本手法によって得られた図 4 は 15 のクラスターの例を示している。目視による検査では、左と右の部分に分解された文字の典型的な構造を見ることが出来る。

要素の対の数（36 の属性）によるクラスターの分布は図 5 に示した。対の少ないクラスターの数が多く出力された一方、クラスター数が少数のものは、対の数が非常に大きいことがわかる。

4 関連研究と結論

本論文は属性ベクトルの対から、可能な限り類推関係を自動的に導出する手法を提案した。先行研究では、短文における手法が提案されているが [5]、対称の文字列間における類推関係の計算で、制約の必要性があることを確認する必要がある。例外の計算として、相似の対における編集距離の制約を初めて言及した。

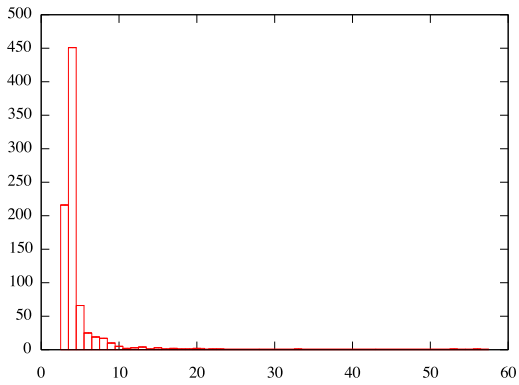


図 5: 対毎のクラスターの経緯。縦軸はクラスター数、横軸は同じ対のクラスターの総数を表している。55 対が最も多いが、 13〜55 対の間では 16 対のクラスターしか存在しない

今回、本問題を解決する方法として、各々の図像の形の中日文字間の類推関係を導出するための扱いやすくさせる問題を示した。

類推関係の図像は既に単語の広義の上では強調されている [2]。一方、直接的に形のレベルのデータを用いて、黒と白のピクセルの間の類推関係を解決した例は本論文が最初の試みである。本問題はすでに未解決な状態で言及されている [3]。

今後、中国語、また中日文字の間における形・音の類推関係の一致した漢字対を同時に学ぶことによって、学習者に対して記憶することがより容易になることを示し、仮説を被験者に対して実証していく予定である。

参考文献

[1] Dedre Gentner, ‘Structure mapping: A theoretical model for analogy’, *Cognitive Science*, **7**(2), 155–170, (1983).

[2] Esa Itkonen, ‘Iconicity, analogy, and universal grammar’, *Journal of Pragmatics*, **22**(1), 37–53, (1994).

[3] Yves Lepage, *Of that kind of analogies capturing linguistic commutations (in French)*, Habilitation thesis, Joseph Fourier Grenoble University, May 2003.

[4] Yves Lepage, ‘Analogy and formal languages’, *Electronic notes in theoretical computer science*, **53**, 180–191, (April 2004).

[5] Yves Lepage, ‘Lower and higher estimates of the number of “true analogies” between sentences contained in a large multilingual corpus’, in *Proceedings of COLING-2004*, volume 1, pp. 736–742, Genève, (August 2004).

[6] Lars Yencken and Timothy Baldwin, ‘Measuring and predicting orthographic associations: Modelling the similarity of Japanese kanji’, in *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pp. 1041–1048, Manchester, UK, (August 2008). Coling 2008 Organizing Committee.