

電子化した加点情報を用いた書き下し文生成ツールの試作

林 昌哉, 田島 孝治 (岐阜工業高等専門学校),
堤 智昭 (東京電機大学), 小助川貞次 (富山大学)

これまでに、国語研蔵『尚書(古活字版)』を対象として移点作業を行い、訓点(ヲコト点)の抽出を行ってきた。今回は、電子化したヲコト点データを使用した書き下し文の生成ツールを作成した。このツールは書き下し文を釈文に近い形式でブラウザから閲覧できる。用意したヲコト点電子化データや点図データ等を入力することで、書き下し文を動的に生成できるため、本文の画像と比較して、点図や本文の例外を発見する作業に活用できる。

Prototype a vernacular reading text generation tool using digitized glosses data

Masaya Hayashi / Koji Tajima (NIT, Gifu College)
Tomoaki Tsutsumi (Tokyo Denki University), Teiji Kosukegawa (University of Toyama)

We were transferring gloss on the "Syousyo (old type version)" and digitized its gloss. We created a generation tool for vernacular reading text using the digitized Wokototen. Vernacular reading text is in the form of a handwritten re-transcription of glossed text and can be compared with the original image. This tool can create a vernacular reading text by digitized Wokototen and Wokototenzu and browse it with a browser. By comparing the original image with the vernacular reading text, it is possible to find a variance or exception in Wokototenzu and the text.

1. まえがき

訓点資料の分析は、記述内容の正確な理解を目指し、加点内容を理解することを中心に行われてきた。これまでに、ヲコト点図や釈文の利用により、資料に付与された加点などによる注釈を解釈する方法は確立されている。この結果、現在では訓点資料の現代語訳を容易に手に入れることができる。

一方で、現代語訳の作成に際しては、訓点資料から、書き下し文を一度作り、解釈する作業が行われることが多い。書き下し文は、主に古典中国語で書かれた原文を日本語で読めるように、語順の調整や助詞・助動詞、読み方などを補って作られた文である。どのように原文を解釈するかは、ヲコト点や注釈により記されている。このため、原文から適切な書き下し文を作成するには、文献に関する知識に加え、ヲコト点と訓読方法に対する深い理解が必要である。

ヲコト点を解釈するためには、ヲコト点図だけでは不十分である。紀伝点や喜多院点など代表的なヲコト点に関しては、ヲコト点図に訓点記号と読みとの対応付けが記されている。それに加え、文献固有の対応付けが存在することも多く、経験や知識などにより意味を解釈しなければならないためである。

本研究では、訓点資料の書き下し文を機械的に生成するために、資料に付与されたヲコト点などの加点情報を電子化する手法を検討してきた¹⁾。

この結果、尚書(古活字版)のヲコト点の電子化に成功し、基礎計量を実現できた²⁾。書き下し文を作成する際に問題になる同じ文字に2つ以上の加点のあるものは、計量によっていくつかのパターンに絞られることが明らかになった。

今回は、このヲコト点電子データから、書き下し文の機械的な生成をするツールの作成を行った。このヲコト点電子データには、仮名点や返読点が含まれていない。よって、作成したツールは、本文とこのヲコト点電子データの情報から書き下し文を生成するため、返り点による語順転倒を行わない。

2. これまでの研究

2.1 対象とした資料

今回の研究は、電子化、統計処理、書き下し文の生成に至るまで、国立国語研究所蔵「尚書(古活字版)」を対象として実施した。尚書は書経とも呼ばれ、政治史・政教を記した中国最古の歴史書で、序文と58の通篇で構成される³⁾。電子データを作成した資料は1596[慶長元]-1615[慶長20]年刊のものであり、巻一から巻九までの画像データが公開されている。冊子本であるため、画像データは1丁に対し表裏が存在し、半丁あたり8行構成である。今回の電子化は本文である巻一から巻九の全てを対象とした。全144丁、約2300行に対するヲコト点情報が含まれている。

2.2 ヲコト点のデータ構造

電子化したヲコト点は RFC8259 に準拠した軽量化オブジェクト記法である JSON 形式で記述した。任意のキーに対して値を関連付ける key-value 型のデータ構造にすることで、必要に応じて要素を追加することが容易である。さらに、統計処理は各種のプログラミング言語を利用して行うため、多数のプログラミング言語で標準的に読み込みが可能な点もこの形式を選択した理由である。特に計算機による処理は Python や JavaScript などのインタプリタで実行可能な言語で処理したため、これらの言語において命令一つでデータ構造を含めて一括して読み込める JSON は利便性が高い。電子化データの key には文字の場所を示す place, 文字の形状を示す character, 行の位置を示す lineno, 付与されている訓点記号を示す elements がある。電子化ツールに尚書の白文のテキストファイルを入力すると、この key に value として各文字の位置や形状が入力された JSON ファイルが生成される。生成時は訓点記号の elements の部分が全て空になっており、入力者は電子化ツールを通じて加点情報を入力していくことで、データを追加できる。elements の key は 3 つで、それぞれ文字に対する訓点記号の位置を示す position, 訓点記号の色を示す style, 訓点記号の形状を示す mark がある。

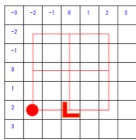
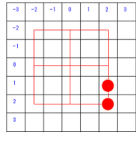
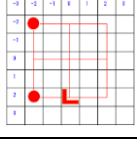
過去の構造化で二文字にまたがる訓点記号に対応するために「語要素」を区別して電子化していた。今回の電子化では、この語要素が電子化されていないため、二文字にまたがる訓点記号の訓合符や音合符は、前の文字の下部に付与されているものとして扱った。段落要素も除外したため、段落の頭を示す科段点は文字の上部に付与されているものとして扱った。

2.3 複数の点が同じ文字に付与されている場合

書き下し文の機械的な生成をする上で、複数の点が付与されている文字の解釈は機械には難しい。同じ二点でも、付与されている文字によって訓読の順番が異なる可能性もある。複数の点に優先順位があり、常にそれが守られているかは、統計的に処理しなければ判断できない。このため、書き下し文を生成する前段階として、ヲコト点の読み順に関係する点を集計し、細かく比較する必要があった。

ヲコト点の読み順に関係ない点は、墨点など一部の特別なヲコト点である。具体的には、合符、音訓読みを表す点、声点、科段点、句読点、返り点兼用の「テ」である。返り点兼用の「テ」の順番は、返り点を読んだ後なので、ヲコト点の読み順には関係がない。

表 1 複数の加点があるパターン

	パターン	総数
(a)		70
(b)		64
(c)		50

2.4 複数の点が付与されているパターン

文字に関係なく、ヲコト点の読み順に関係がない点を除いた、複数の点が加点されているパターンを、登場回数の多い順に調査した³⁾。表 1 に示すように最も多かったパターン(a)は(0,2)「シ」と(-2,2)「テ」であり、～シテと順に読める。2 番目に頻出したパターン(b)の(2,1)「ト」と(2,2)「ハ」の点は、順に～トハとなり、割注などで「AトハB也」と使われることが多い。3 番目のパターン(c)はそれぞれ(-2,-2)「ニ」、(-2,2)「テ」、(0,2)「シ」であり、順に「～ニシテ」と読める。

文字に関係なく、複数加点のあるパターンを集計すると、全 140 パターンに集中していることが確認できた。このように頻度の多い 3 種のパターン含む上位 40 パターンを確認したところ、付与されている文字に関係なく、ヲコト点の読み順が決定されるものであった。

3. 書き下し文生成ツール

3.1 概要

今回試作した書き下し文の機械的な生成をするツールは、必要なデータをユーザーが入力することで書き下し文を Html として自動的に生成し表示するものである。本文に対してヲコト点の解釈は小さく、合符や読みを表す線点は形状がそのままのため、釈文に近い表示形式とした。また、助詞、助動詞を表す点と、文字の読み方を表す点の区別はできないため、どちらも並列に表示される。表示ページの左半面を原本の画像、右半面を生成した書き下し文にすることで、比較しながら読むことが可能である。

過去の研究で作成した JSON 形式のヲコト点電子データには、本文の文字データも含まれている。この JSON データを JavaScript で読み込むことで、

ヲコト点を平仮名に翻訳した文章を Html として生成することができる。図 1(a)に示すように語順転倒のない書き下し文として、そのままヲコト点を文章として追加した場合、本文画像との差が激しく、可読性が低い。また、ヲコト点の中でも合符や科段点は、特性が異なるため翻訳できない。よって、書き下し文は、より本書に近い釈文の形式を使用する。図 1(b)に示すように釈文は本文に比べ、ヲコト点の翻訳文が小さく文の概形が変化せず読み比べやすい。合符や科段点も違和感なく原形を保つことが可能である。

画面左半分は、国語研の公開している尚書(古活字版)の画像を表示している。この画像は解像度が高く、ヲコト点の表記は微細であるため、画像自体の拡大とスクロールができるようになっている。

画面右半分には読み込んだ JSON データから、本文の文字データに、ヲコト点を平仮名や句読点にしたものを付与してして表示している。本文は 1 丁が 8 行で構成され、右半分の表示は各行ごとに項として区切られ構成される。1 つの項の表示は 1 行に限らず、表示部に収まらない場合は改行する。Html で表示しているため、ブラウザのウィンドウサイズが変化した場合にも、見切れることなく表示部に合わせた改行ができる。

JSON データには仮名点や返り点は含まれてい

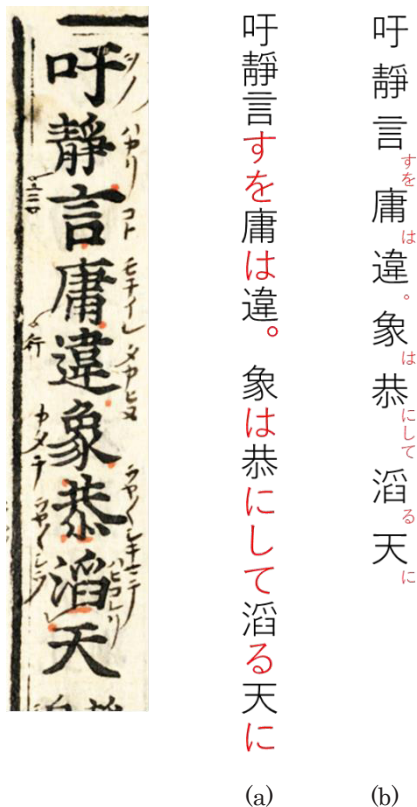


図 1 書き下し文の表示形式

ないことから、ヲコト点のみを反映した語順転倒を行わない書き下し文が生成される。語順転倒が無い場合、返り点兼用の「テ」は「(て)」と表記した。原本では文章の終わりには句点や科段点の他に、割注の切り替わりがあるため文字の大きさから判断できる。同様に生成文も割注の文字サイズを僅かに下げ、背景色を変化させることで区別した。

3.2 想定される使用用途

このツールの主な用途は、本文の画像データと点図から解釈の比較である。ほとんどの点図は訓点資料の読み方を後からまとめ、作成されたと考えられている。尚書(古活字版)のデータ化の際に使用した点図は「論語」に対応したものであり、尚書用の点図に最も近いと考えられている。このように、訓点資料と点図が完全に対応していないため、齟齬が発生することがある。そこで、釈文を作成することで本文と電子化データを比較し、例外を抽出できる。

そのほかに、点図との違いを探す中で、ヲコト点電子データの入力ミスや、解釈による違いを発見することができる。ヲコト点の電子化は、画像処理による自動化が難しいほどに文字の形状や前後の文字に影響され、間違いは多くなる。電子化したデータを、さらに訓読に知識のある方に確認して頂くことで、より精度の高いデータ化が望める。

3.3 利用するデータ構造

Html ページのヘッダ部に以下の 4 つの電子データを入力することで書き下し文を生成できる。

- A) 点図データ
- B) 本文のデータ
- C) 複数点のある文字の読み順データ
- D) ヲコト点電子データ

A.点図データは堤氏らが作成したヲコト点図⁴⁾をそのまま使用している。この点図データには点の位置、ヲコト点の形状、意味(読み)が記述されている。

B.本文のデータは各行ごとに行 ID と本文、本文中に割注を表す括弧が記入されている。

C.複数点のある文字の読み順データにはこれまでの研究で明らかになった登場頻度の高い 40 パターンの読み順の対応表を JSON 形式で記述したものである。

D.ヲコト点の電子データは過去の研究で作成したもので、本文の文字、ヲコト点の位置や形状、色が含まれている。

表2 変換時のデータ構造

key	value 例	表示例
character	“思”	
type	main	
glossR	“にし”	
glossL	“(て)”	
mark	音読み	

3.4 データの変換

入力された JSON ファイルの「D.ヲコト点電子データ」は文字ごとに表2のようなオブジェクトに変換される。本文の文字の入る character, 本文か割注を表す type, 通常のヲコト点の読みを表す glossR, 返り点兼用の「テ」を表す glossL, 合符等を表す mark をそれぞれ key とするオブジェクトを作成する。character にはヲコト点電子データの本文をそのまま挿入する。

ヲコト点電子データには type に該当するものや glossR, L に該当するヲコト点の読み方は記入されていない。Type は「B.本文データ」のテキストファイルを参照し本文であれば「main」、割注であれば「sub」変換する。ヲコト点の読み方は点の位置と形状、色から「A.点図データ」を参照し、読みを文字列として glossR または L に格納する。点図には音合符、訓合符、音読み、訓読みも他の点と同様に記入されているので、それらの点は mark に格納する。

3.5 複数の加点のある文字

これまでの研究で尚書卷一から卷三に含まれる、ヲコト点の読み順に関係がない点を除いた、複数の点が加点されているパターンを調べたところ、140 個であった。文中に高い頻度で登場する 40 パターンは、読み順が一意に定まることを確認した。それ以外の、登場頻度が低いパターンも助詞、助動詞の点と読み仮名の点の二点のものがほとんどで、今後詳しく調査する予定である。

頻度の高い 40 個のパターンの読み順を一つ一つリストアップしたファイルを作成し、読み順の制御に利用する。これは、ツールの作成中に新たなパターンの追加や削除が容易である。しかし、電子データの増大に伴い、入力量が膨大になる可能性もある。今回作成した 40 パターンの「C.複数点のある文字の読み順データ」は卷一から卷三までの読み順の対応表であるため、今後卷四以降の読み順を調査し、追加する必要がある。

複数加点のある文字の正しい読み順に変換する流れを図2に示す。複数点のある文字は表1のオブジェクト作成時に glossR の key に複数の読み方が一時的に代入される。図2の例のように「思」という character に対して「に」「て」「し」

「。(句点)」といったヲコト点が付与されているとする。作成時 glossR 内の順番は「D.ヲコト点電子データ」の作成時に入力した順になるため、同じ文字や点でも順番が異なる可能性がある。一意に定めるため、glossR を 50 音順にソートすると「。 , し, て, に」となる。この時、glossR には句読点のみ語順に影響しない点のため取り出し、後から付与する必要がある。句読点を省き一つの文字列として合体した「してに」の形で「C.複数点のある文字の読み順データ」を参照し、変換する。「してに」の場合は「にして」と一意に決定できる。最後に取り出してある句読点を後ろに追加することで、「に」「て」「し」「。(句点)」の4点は「にして。」として glossR に保存される。

返り点兼用の「テ」や合符、音読み訓読みは glossR には挿入されない。「C.複数点のある文字の読み順データ」には句読点を除いた複数のヲコト点を 50 音順にソートしたものとその変換先が一覧として記入されている。

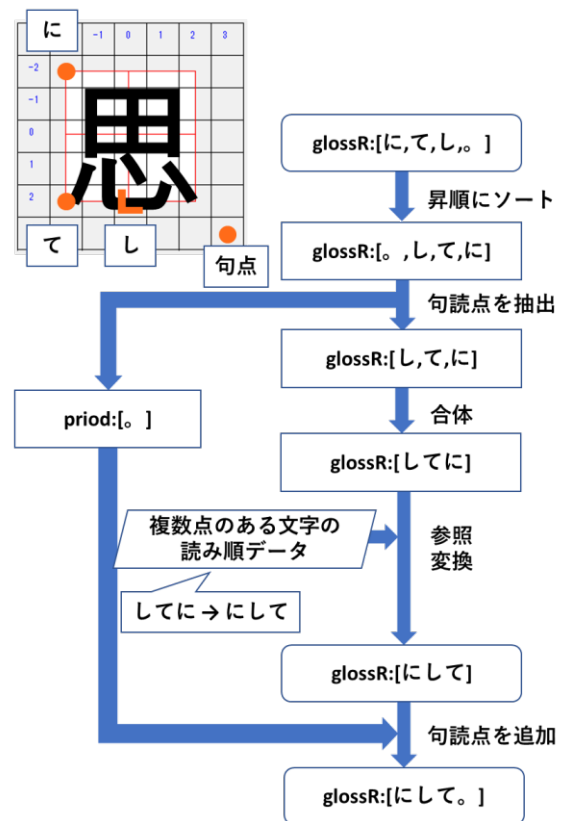


図2 読み順変換の流れ例

3.6 表示方法

作成した Html をブラウザで表示したものを図 3 に示す。図 3 は尚書卷一の第一丁表の行頭から表示したものである。左半分に表示されている画像は国立国語研究所が公開している尚書(古活字版)の同丁のものである。ページのヘッダ部は必要ファイルの入力欄であり、利用者が用意したファイルの一つずつ入力する必要がある。A~D に各ファイルは専用の組み合わせではないため、同じヲコト点電子データを複数の点図データで確認することも可能である。すべてのファイルを入力すると自動的にページ右半分に書き下し文が表示される。

書き下し文は表 2 のオブジェクトを読み込むことで、各 key に対応した位置に文章や記号を表示する。glossR のヲコト点を翻訳した文字列は文字の右下に、glossL の返り点兼用の「テ」は文字または glossR の下、左部に表示される。合符や音読み訓読みは釈文と同じく線点のまま文字に直接対応する位置に表示される。科段点も翻訳せず、そのまま朱の点として文字上部に表示する。

書き下し文表示方式は二種類あり、変換した文の最後まで全て一続きにするものと、1 ページずつ表示するものがある。この二種はヘッダ下部のボタンで切り替えられる。変換時は全て一続きで表示され、左半分の画像は尚書卷一の第一丁表の画像になる。書き下し文を左右にスクロールすることで丁を跨いで確認できる。画像はボタンで手動で追従させる必要がある。1 ページずつ表示のボタンを押すことで書き下し文を半丁ずつ確認することができる。この時、左半分の画像は表示されている丁の画像に遷移し、自動で追従する。

書き下し文は上部には項 ID があり、卷一の第

一丁表一行目であれば卷 1,1 オ 01 である。ブラウザの拡大率に依存するが、1 項は 1~4 行ほどで構成され、本文と割注が混在している。

割注は文頭に「(割注)」とあり、文全体に背景色がついている。割注の文自体をクリックすることで割注の表示を切り替えられ、文章が消え「(割注)」の表示のみになる。割注の非表示時に残った「(割注)」をクリックすることで再度表示できる。ページヘッダ下部の割注をすべて表示するボタンを押すことで表示文章全体の割注表示を切り替えられる。表示される書き下し文はテキストとしてコピーが可能であり、図 1(a)のような書き下し文としてテキスト化できる。文章中の合符や科段点、「(割注)」は選択対象にならないため、簡単にコピーすることができる。

3.7 残された課題

この書き下し文生成ツールの開発の中で不足している要素として、「返読点」があげられる。ヲコト点電子データを作成する際に、数ある加点点要素の中からヲコト点のみを電子化したため、完成した書き下し文を読み下す難易度が高い。ヲコト点と仮名点は加点点時期や人物の違いからか、解釈の齟齬はたびたび発生する。しかし、返読点はヲコト点加点点時に前提として存在しており、返り点兼用の「テ」など依存度が高いものもある。円滑な読み下しに返読点は必須であり、電子化難易度の低さから、早急にデータに追加する必要がある。

その他に、ヲコト点電子データは非常に量が多く読み込みに時間がかかるため、データ自体の軽量化やデータの分割化などを視野に入れる必要がある。現在「尚書(古活字版)」を対象として

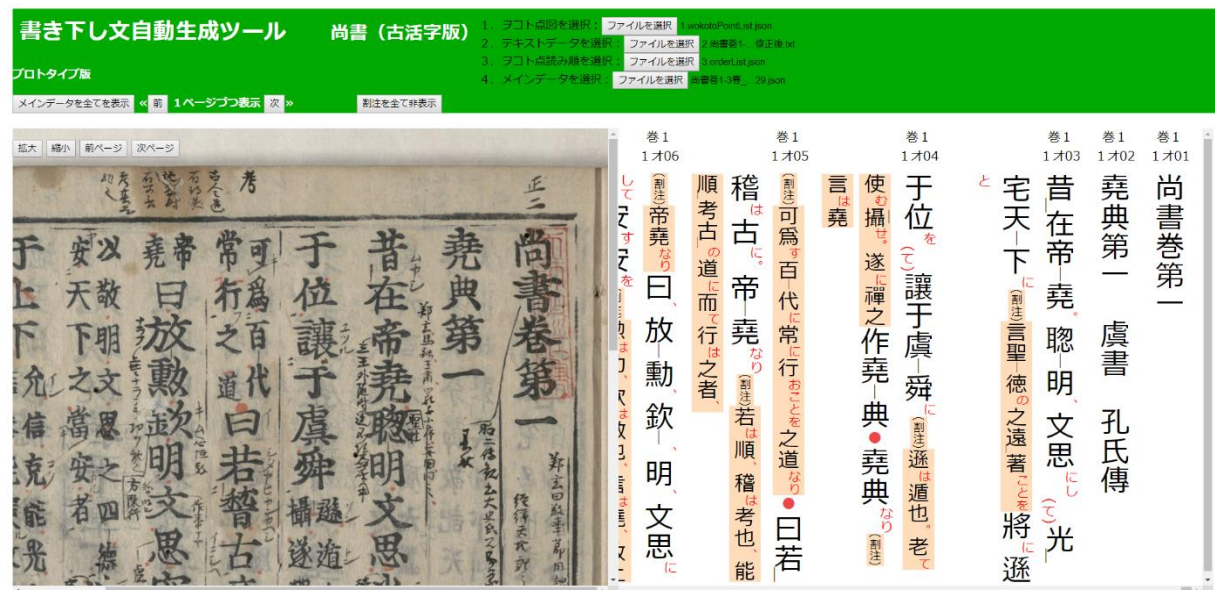


図 3 作成したツールの表示画面

書き下し文を生成しているが、ほかの文書にも対応することで、ツールの汎用性が上がると考えられる。

4. あとがき

本稿では国立国語研究所蔵「尚書(古活字版)」を対象として、電子化したヲコト点データを使用した書き下し文の生成ツールを作成した。書き下し文は釈文に近い形式で、本文画像との比較が可能である。Htmlとしてブラウザで閲覧でき、用意したヲコト点電子化データや点図データ等を入力することで、書き下し文を生成できる。全体を一続きに並べるか、半丁ごとに本文画像を追従させながらの表示が可能である。割注は部分的にも全体的にも表示の有無を切り替えられ、本文のみの表示もできる。

ツールの出力結果を数丁分、細かくチェックし文中に現れる例外的な翻訳を調べた。文中の不自然な点や、専門家から見た、ヲコト点電子化の際に生じた曖昧な点や入力ミスの発見があった。しかし、ヲコト点電子データは返読点を含んでいないため、読みづらさの課題が残る。返読点や仮名点の追加によって、より釈文に近い形式となり、より広い層の利用者にも触れやすいツールになると考えられる。

謝辞

本研究は JSPS 科研費 17K1850606 の助成を受けたものである。また、人間文化研究機構広領域連携基幹研究プロジェクト「異分野融合による総合書物学」の国語研ユニット「表記情報と書誌形態情報を加えた日本語歴史コーパスの精緻化」による成果の一部である。

参考文献

- 1) 林昌哉 他4名：訓点資料の加点情報計量のためのデータ構造 — 国立国語研究所蔵「尚書（古活字版）」を対象として—, じんもんこん 2017 論文集, Vol.2017, pp.45-52 (2017.12)
- 2) 林昌哉, 田島孝治, 高田智和：尚書（古活字版）の訓点データの基礎計量, 第118回 人文科学とコンピュータ研究会発表会.
- 3) 赤塚忠(翻訳)：中国古典文学大系(1)書経・易経(抄), (1972.1)
- 4) 堤 智昭 (東京農工大学), 田島 孝治 (岐阜工業高等専門学校), 高田 智和 (国立国語研究所): 点図情報入力支援ツールによるヲコト点図の電子化 じんもんこん 2015 論文集, Vol.2015, pp.185-190 (2015-12)