

古典文学学習のデジタルトランスフォーメーション —源氏物語の若紫で学ぶくずし字アプリケーション—

林 愛梨[†] 須永 宏[†] 横山 恵理[†]

[†]大阪工業大学 情報科学部 〒573-0196 大阪府枚方市北山1丁目79番1号
E-mail: [†] e1b20087@st.oit.ac.jp , hiroshi.sunaga@oit.ac.jp , eri.yokoyama@oit.ac.jp

あらまし 現代の日本人でくずし字が読解できる人は数千人程度と言われており、日本に残された大量の歴史的資料をより使いやすくするには、くずし字を翻刻する必要がある。だが、くずし字を読める人が少ないため翻刻には非常に長い時間が必要となる。またくずし字を読みたいが学習が難しいとの声を聴きくずし字を読めるようになりたいと考えている方などに向けアプリケーションを開発することでくずし字の翻刻が行える人材の増加につながると考えられる。くずし字の翻刻を実践出来ること、翻刻したくずし字の内容の意味を知ることが必要である。また、翻刻のサポートを行えるくずし字で書かれた内容を予測することが出来る機能が求められる。それぞれの目標に合わせたアプリケーションを作成した。アンケート評価の結果からこれらのアプリケーションは有用性があるという評価がもたらされた。

キーワード 古典文学教育, くずし字, デジタルトランスフォーメーション

Digital Transformation of Japanese Classical Literature Learning —Kuzushi-ji Application Learning with Wakamurasaki Scroll of The Tale of Genji—

Airi HAYASHI[†] Hiroshi SUNAGA[†] and Eri YOKOYAMA[†]

[†] Faculty of Information Science, Osaka Institute of Technology 1-79-1 Kitayama, Hirakata City, Osaka, Japan
E-mail: [†] e1b20087@st.oit.ac.jp , hiroshi.sunaga@oit.ac.jp , eri.yokoyama@oit.ac.jp

Keyword Japanese Classical Literature Education, Kuzushi-ji, digital transformation

1. はじめに

私が古典を学習する際に興味を持った文献がくずし字で書かれており内容が読めず、インターネット上で翻刻を探しても見つからないという状況に陥ったことがある。実際に現代の日本人できちんとくずし字が読める人はくずし字がきちんと読める人は数千人程度と言われており、日本に残された大量の歴史的資料をより使いやすくするには、くずし字を翻刻する必要があるが、くずし字を読める人が少ないため翻刻には非常に長い時間が必要となる。[1]

自分自身も含めてゲームなどから日本文化やくずし字に興味がある人からくずし字を読みたいが学習が難しいとの声を聴くなど IT による解決に卒業研究を通して貢献できればと思うようになった。これを進めることで、くずし字を読めるようになりたいと考えている方などに向けアプリケーションを開発することでくずし字の翻刻が行える人材の増加につながると考えられる。

歴史に興味を持つとくずし字を目にする機会は多くあるが独学で学ぶことは難しく、興味を持った人が

くずし字を読めるようになるためのサポートがしたいと考えるようになった。有名な文献は多く存在するが、まずは義務教育でも触れる源氏物語の若紫を題材にくずし字を学ぶことが出来るアプリケーションを提案、構築して行く。

上記目的を達成するために、くずし字の翻刻を実践出来ること、翻刻したくずし字の内容の意味を知ることが必要である。また、翻刻のサポートを行えるくずし字で書かれた内容を予測することが出来る機能が求められる。また、基本的にスマートフォンではなく PC 利用を前提に、機能設計、画面設計を行う。

源氏物語は現存している文献はいくつか存在しているが、くずし字の翻刻を実践するのであれば過去に書かれた文献を翻刻すべきである。改変の許可が出ている文献であるべきである。翻刻した内容が合っているのかどうか判断できることも重要である。また、手軽に学ぶことができるようにダウンロードせずとも使用できるアプリケーションである必要がある。

このような分析を行い、ブラウザで動作する Web アプリケーションとして、JavaScript プログラミングに

より実装を進める。

先程の文献に対する要求からの国文学研究資料館蔵「源氏物語」doi:10.20730/200003803[2]とその源氏物語からくずし字一文字ずつ切り抜いた「日本古典籍くずし字データセット」doi:10.20676/00000340[3]を使用する。理由は江戸時代「源氏物語」版本として最も流布し、現在の研究の基盤となっている。実際にこの文献を使用した源氏物語研究書がある。次に保存状態が良く、ライセンスの問題をクリアしているからである。この文献のライセンスは「CC BY-SA 4.0」となっており、教育・研究目的を含む、あらゆる目的で使用許可が出ているためこの文献を使用することに決定した。以下、本論文で使用する源氏物語の画像は国文学研究資料館蔵「源氏物語」を使用する。

これらの背景を踏まえ、第2章はアプリケーションの仕様・実装方法について述べる。第3章ではアプリケーションの評価を述べ、第5章にて結論として全体と総括する。

2. アプリケーションの仕様・実装方法

本章では本研究の構成および各機能のアルゴリズムについて述べる。本研究は、はじめに記述した目的と課題を解決するため、くずし字の翻刻を実践出来る翻刻練習アプリ、翻刻したくずし字の内容の意味を知るテキスト集アプリ。また、翻刻のサポートを行えるくずし字で書かれた内容を予測することが出来るくずし字 DB アプリを制作した。また、テキスト集では現代語訳を行った全文も確認できるようにした。問題集は答える際は全文平仮名で答えるように設定しているが、利用者の意見などに応じて変更して行けるように考えている

2.1 源氏物語翻刻練習アプリ

源氏物語翻刻練習アプリの画面構成を説明する。

テキストボックスに数字を入力すると任意ないしは指定行の部分の画像を表示する。行の画像を表示しているとき読みをひらがなで解答すると正誤判定がされる。表示している画像のサイズを変更できる。ランダムボタンでランダムに問題が出題される。



図 2.1.1 翻刻練習アプリの起動画面

画像の表示

width と height を画像の幅と高さから計算して設定する。次に size に設定された値に合わせて getElementById で取得した id の padding に style で width と height を設定する。このとき canvas の高さとも幅も設定する。この関数内の最後に draw という関数が呼び出される。この関数を呼び出すことにより画像を表示することができる。drawImage で画像の幅や高さ、座標などを指定し、表示する。Swich 文で大きさを変更できるようにしている。

機能説明



図 2.1.2 源氏物語翻刻練習アプリの上メニュー

左上から順に説明する。見開きページ数の入力と書かれているテキストボックスに特定の範囲の数字を入力し、隣の変更ボタンを押すと、下の canvas に見開きページの画像が表示される。

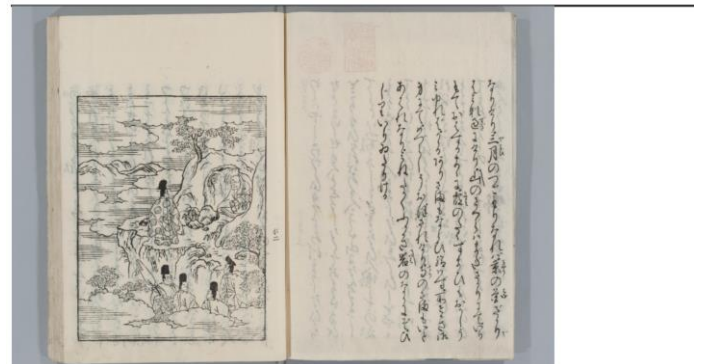
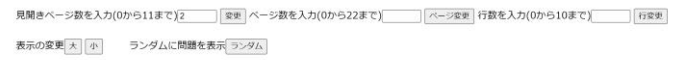


図 2.1.3 見開きページの画像の表示

図 2.1.3 の見開きページの画像の表示は imgchange という関数により表示されている。テキストボックスに入力された数字を取得し、その数字を使って画像を探しつけた画像を入手しロードした際に次に進むように onload を使用し描画をし直すという方法を取っている。次にページ数を入力と書かれているテキストボックスに特定の範囲の数字を入力し、隣のページ変更ボタンを押すと、下の canvas に 1 ページ分の画像が表示される。



図 2.1.4 ページの画像の表示

図 2.1.4 のページの画像の表示は `imgchangeimage` という関数により表示されている。ページ数のテキストボックスに入力された数字を取得し、その数字を使って画像を探し見つけた画像を入手しロードした際に次に進むように `onload` を使用し描画をし直すという方法を取っている。

行数を入力と書かれているテキストボックスに特定の範囲の数字を入力し、ページ数を入力するテキストボックスにも数字を入力している状態で、隣の行変更ボタンを押すと、下の `canvas` に指定された行の部分の画像が表示される。



図 2.1.5 行の画像の表示

図 2.1.5 の行の画像の表示は `imgchangeline` という関数により表示されている。ページ数と行数のテキストボックスの値を取得し、`switch` でページ数によって表示する画像を変える。設定している数字は挿絵のあるページであるため行がないのでそのままのページを表示する。そうでないなら取得した数字を使用し、行の画像を取得する。正誤判定を分かりやすくする画像を見えないようにし、その後見つけた画像を入手しロードした際に次に進むように `onload` を使用し描画をし直すという方法を取っている。

表示の変更の隣にある大ボタンと小ボタンを押す

とそれぞれの画像の表示される大きさが変更される。



図 2.1.6 画像の表示サイズの変更

図 2.1.6 の画像の表示サイズの変更は大ボタンが `imglarge` という関数、小ボタンが `imgsmall` 関数により変更されている。双方 `size` の値を変更している。描画し直す際に `size` の値により画像のサイズが変更される。それぞれ `size` の値に合わせて変更がされるようになっている。`canvas` の大きさと画像の大きさが変更されている。`Canvas` の大きさを変更しているのは画像だけが小さくなると下メニューがそのまま下に残されてしまい見づらいから変更を行っている。

上メニューの最後のランダムに問題を出題はランダムボタンを押すとランダムに行の画像を表示する。まれに挿絵の画像が表示される。



図 2.1.7 ランダムに行の画像を表示

図 2.1.7 のランダムに行の画像の表示するのは `qrandom` という関数により表示されている。

この関数はページ数のテキストボックスと行数のテキストボックスにランダムに数字を入力し、`imgchangeline` 関数で表示するという方式を取っている。

次に下メニューについて説明する。

回答はひらがなのみ

回答欄

答え

回答

もう一度挑戦

図 2.1.8 源氏物語翻刻練習アプリの下メニュー

回答欄は回答を入力するテキストボックスである。これは表示されている画像が行の画像のときその行の翻刻した内容を回答することが出来る。そして回答欄に答えを入力した後、回答ボタンを押すと答えと正誤判定が画像で表示される。画像が○なら正解、×なら不正解である。

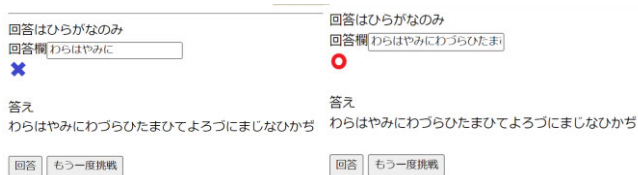


図 2.1.9 回答後の正誤判定

図 2.1.9 の正誤判定は `input` という関数により表示されている。この関数はまずページ数と行数のテキストボックスの値を取得し、`switch` でページ数によって表示する画像を変える。ページ数が挿絵のページの場合は何もせずに終了する。挿絵ではないとき、ページの番号を 1 から順に挿絵ページなしの連番として数字を振っている。ページに降った連番と行数のテキストボックスの値を組み合わせで配列からデータを取り出す。そして、取り出した答えを確認し合っていた場合 `change` 関数で○の画像を表示し、そうでないなら×の画像を表示し答えを表示する。

問題の管理

問題の答えのデータは多重配列で管理している。データは膨大なため一部を示す。

```
var ans=[["わらはやみにわづらひたまひてよろづにま  
じなひかぢ","などまいらせたまへどしるしなくてあ  
またたび",・・・],・・・]
```

ページごとの配列に、行ごとの配列が中にあるというような状態で管理している。そのため、ページ数と行数の両方が答えを取り出す際に必要となる。

もう一度挑戦ボタンを押すと `imgchangeline` 関数が動き、答えと正誤判定の画像を見えないようにする。

2.2 テキスト集アプリ

次にテキスト集アプリの確認できる画面について説明する。テキスト集は `index` で選択したページの内

容を確認できるアプリである。全文の確認もできる。

Genji Monogatari Wakamurasaki

画像をクリックしてください



図 2.2.1 テキスト集のインデックス起動画面

図 2.2.1 のテキスト集のアプリは源氏物語の若紫は本であるため、ページ順でデータを並べている。起動画面は左上の表紙から順に主要な場面までを左に並べたものとなっている。このアプリケーションを起動したとき、`initialize` という関数がロードされる。`createElement` で `table` を作り、さきほど設定した `width` と `height`, `GYOU` と `RETSU` を参照し計算、動的生成をしながら画像が順に並ぶように `appendChild` を使う。最後に `divelm.appendChild` でこの関数で作成した `table` を `appendChild` すると図 3.2.2.1 のように並んだ起動画面となる。

次にテキストが確認できる画面について説明する。

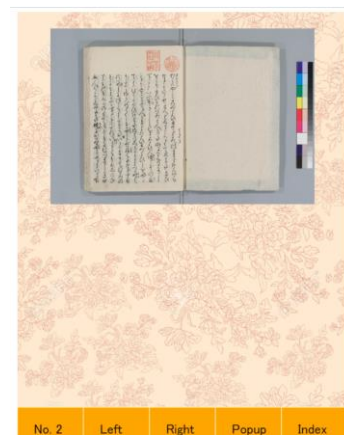


図 2.2.2 テキスト集の起動画面

図 2.2.2 のテキスト集のアプリはインデックスの画面から画像を選択すると図の画面が起動する。真ん中の画像はインデックスで選択した画像が表示されている。このアプリケーションを起動したとき、`initialize` という関数がロードされる。`jsonread()` で `Json` 形式のデータを読み出す。また、ボタンの高さもここで設

定する。Json 形式のデータは膨大なため一部を以下に示す。

```
<textarea id="json" cols=60 rows=10>
{ "title": "源氏物語 5 帖 若紫",
  "data": [ { "key": 1, "file": "g001.jpg", "scene": "表紙",
    "all": "表紙",
    "all2": "",
    "right": [],
    "left": [{"col": 1,
      "posi": [56,18,70,56],
      "yomi": "わか紫",
      "imi": ""}],
    {"key": 2, "file": "g002.jpg",
      "scene": "第一段 三月晦日、加持祈祷
のため、北山に出向く (その 1)",
      "all": "瘡病にわづらひたまひて、よろ
づにまじなひ加持など参らせたまへど、しるしなくて、
あまたたびおこりたまひければ、ある人、「北山になむ、
なにがし寺といふ所に、かしこき行ひ人はべる。去年
の夏も世におこりて、人びとまじなひわづらひしを、
やがてとどむるたぐひ、あまたはべりき。ししこらか
しつる時はうたてはべるを、とくこそ試みさせたまは
め」など聞こゆれば、召しに遣はしたるに、「老いかが
まりて、室の外にもまかです」と申したれば、「いかか
はせむ。いと忍びてものせむ」とのたまひて、御供に
むつましき四、五人ばかりして、まだ暁におはす。や
や深う入る所",
    "all2": "",
    "right": [],
    "left": [
      {"col": 1,
        "posi": [90,33,98,119],
        "yomi": "わらはやみにわづらひ
たまひてよろづにまじなひかぢ",
        "imi": ""
      }, . . .
    ]
  ]
}
```

キーと画像、シーン、全文などを設定している。本のページを右と左に分けてそれぞれ配列で行についての情報を管理している。jsonread()関数で読み出した後はJsonmapで一括管理される。

initialize00 関数は背景の画像を入手し画像をロードした際に次に進むように onload を使用する。Initialize00 関数の中で定義されている initialize0 関数はインデックス画面で選択した画像を入手し画像をロードした際に次に進むように onload を使用する。Initialize0 関数の中で定義されている initialize1 関数は

新たに canvas という HTML 要素を生成し、その中に高さと幅の属性を設定する。その後画像の大きさや位置などを見やすいよう値を計算し設定している。この関数内の最後に draw という関数が呼び出される。この関数は draw1 関数と draw2 関数を呼び出すものになっている。draw 関数の中で定義されている draw1 関数はこの関数は haikei 関数を呼び出す。次に lfrg の値によってインデックスで選択した画像の表示が変わる。draw1 関数の中で定義されている haikei 関数はこの関数は背景の画像を表示する関数となっている。そして drawbt 関数を呼び出す。この関数は canvas 内にボタンとなる四角を設定する。次に四角の中に文字を memopoint 関数で設定する。drawbt 関数の中で定義されている memopoint 関数はテキストの文字の色とフォントを設定し replace でテキスト内の¥n や gmなどを{br}に変更する。次にテキストを前から順に一文字ずつ取り出し、配列に追加する。文字数などによって次の行に移る。そして文字を追加し終わると fillText で引数の座標に一文字ずつ順に書く。

機能説明

このアプリケーションを利用するにあたってインデックス画面で画像を選択することで内容などを確認できるようになる。まずインデックスを起動し画像を選択すると図 2.2.2 の起動画面となる。この画面の機能を説明する。



図 2.2.3 テキスト集のメニュー

テキスト集のメニューについて左から順に説明する。最左に書かれているナンバーはインデックスで選んだ画像に設定されている番号が表示される。次の Left と Right はそれぞれ見開きページで表示されている画像の左側、または右側を表示する。Popup は行を選択しているときポップアップを表示する。これらはタッチイベントにより制御されているため、ボタンのようにクリックすると動く。onmousedown や onmouseup で座標を取得する。座標の高さがボタンの範囲のときそれぞれのボタンの範囲内であるとき特定の数字を ii に入れる。そして switch 文でそれぞれのボタンにあった関数を動かす。case の上から順がメニューの左順と同じとなって居り、順に使用されている関数について説明する。leftside 関数は Left を押すと見開きページの画像の左側を表示する。rightside 関数は Right を押すと見開きページの画像の右側を表示する。



図 2.2.4 Left, Right クリック後の表示

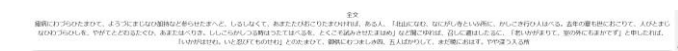


図 2.2.5 全文の表示

見開きページの画像から左側または右側だけを大きく見せることが出来る。また、下にページの図 3.3.2.3 のように全文が表示される。 leftside 関数, rightside 関数はそれぞれフラグを立てて、全文を表示し、draw1 関数を呼び出している。draw1 関数 Switch 文を使いフラグに合わせて、drawImage で画像の表示を変更している。

次に Popup ボタンは行を選択している必要がある。Left, Right クリック後の画像の行を直接クリックするとクリックした行が赤い枠に囲まれ選択状態になる。このタッチイベントは initialize1 関数に設定されている。initialize1 関数は座標を取得し、getcolmap で右左どちらが拡大されているかを確認する。その後、画像をクリックした場所が行であるかを判定し、行であるならその行の番号を atarinum に入れ、draw 関数で描画する。draw 関数は draw1 関数は画像を描画する関数だが、draw2 が行の赤い枠を描画する関数となっている。draw2 関数は行をクリックした場合、どちらのページが拡大されているかを確認後に colmap にある posi を呼び出しそれぞれ合うように座標を設定する。そして strokeRect で設定した座標と高さ、幅で行を囲うように赤い枠を表示する。この枠が表示されているとき Popup ボタンを押すとタッチイベントにより popup 関数が呼び出される。popup 関数は createElement で動的にポップアップを動的に生成する。Canvas を生成し選択した行の画像を表示し、table を生成する。これにより画像の横に縦書きで翻刻した内容を表示している。

タッチイベントで呼び出されるもう一つのボタンである Index ボタンはインデックス画面に移動するボタンである。クリックすると図 2.2.1 の起動画面になる。

2.3 くずし字 DB アプリ

このアプリケーションを利用するにあたってメインとなる機能を説明する。それはタッチイベントによってドラッグアンドドロップでくずし字のイメージなどを動かすことが出来る。

くずし字DB

文字をドラッグして配置、修正・消去も可能。選択ボタンで検索に設定。くずし字解読の際に文面や単語の予測に使用出来ます



図 2.3.1 くずし字 DB の起動画面

図 2.3.1 のくずし字 DB アプリは源氏物語内にある文字の画像を一文字ごとに五十音順を基本としてデータとして並べているため、起動画面は図 2.3.1 のようになっている。このアプリケーションを起動した際、initialize という関数がロードされる。dbread 関数で Json 形式のデータを読み出す。次に dbread 関数によって Kuzusizidata に入ったくずし字のデータを確認していき、すべての画像データを表紙の画像を入手し画像をロードした際に次に進むように onload を使用する。Json 形式のデータは膨大なため一部を示す。

```
<textarea id="kuzusizidata" cols="120" rows="10">
{"tablename": "kuzushizi_db","data":
  [{"key": "1",
    "type": "1",
    "moziposition": "",
    "name": "あ",
    "kanzi": "安",
    "memo": "",
    "information": {"pagenum": "2",
                    "xpozi": "351",
                    "ypozi": "581",
                    "wid": "77",
                    "hei": "91"},
    "filename": "a1.jpg"},
  . . .
```

キーと画像、ひらがな、漢字などを設定している。

Informationはその画像が実際の文献内のどこから切り抜かれた画像なのかの情報を管理している。jsonread()関数で読み出した後はkuzusizidataで一括管理される。initialize関数の中で定義されているinitialize2関数はファイル名を取得しimgmapにsetしていく。canvasの高さと幅の属性を設定する。mobileがtrueであれば専用の高さと幅を返す。その後いくつかの値を計算し設定している。seiretsu2関数でくずし字を順に並べて表示する。initialize2関数の中で定義されているseiretsue2関数はSortarrayでキーの数字順に並べたkuzusizidataを二重for文で順にくずし字の幅と高さを使い並べていく。最後にhaikai関数とdraw関数で描画を行う。seiretsu2関数の中で定義されているdraw関数はhaikai関数で背景の描画を行う。for文内でそれぞれのtypeに合わせて色を振り分ける。そしてstrokeRectで四角を書きmemoprint関数でくずし字の画像の上に表示されている文と画像を表示する。draw関数と組み合わせることで図2.3.1の起動画面が描画される。

機能説明

このアプリケーションを利用するにあたってメインとなる機能を説明する。それはタッチイベントによってドラッグアンドドロップでくずし字のイメージなどを動かすことが出来る。図2.3.2のようにくずし字を動かすのはinitialize2関数にあるタッチイベントにより動かすことが出来る。またクリックすると図3.3.3.2のようにメニューが表示される。これらがタッチイベントで実装されている。



図 2.3.2 くずし字の移動後

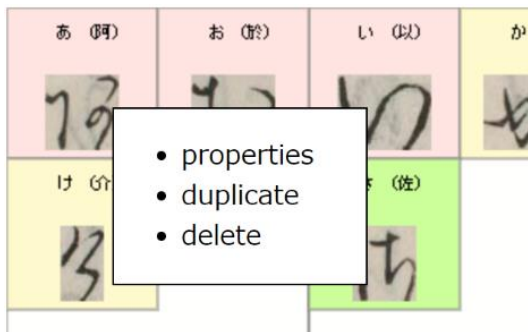


図 2.3.4 くずし字をクリック後

タッチイベントのクリックで座標とその場所にあるくずし字を判定する。判定はその後動きがあるかないかで動作が変わる。クリックの場合メニューが表示され時間経過で消える。ドラッグアンドドロップの場合くずし字が動く間draw関数で描画し続ける。また、ドロップした場所によってmozipositionの値が変更される。メニューに表示されている文字をクリックするとそれぞれの機能が動くので、上から順に説明する。



図 2.3.5 くずし字の詳細を表示

propertiesをクリックするとくずし字の詳細が表示される。これはitem_propertiesという関数が呼び出されることにより表示される。この関数はポップアップを表示し、kuzusizidataの中にある情報をidに合わせて代入する。最後にclearconextmenuでタッチイベントによって表示されるメニューを見えないようにする。

duplicateをクリックするとそのくずし字が複製される。duplicate_itemという関数が呼び出されることにより表示される。また、deleteをクリックするとそのくずし字が削除される。

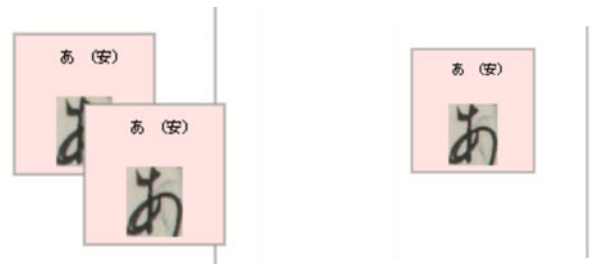


図 2.3.6 くずし字を複製, 削除

duplicate_item関数は新しいitemを生成し、選択したくずし字の内容すべてを新しいitemに代入する。座標のみ幅を4分の1した分ずらし、kuzusizidataにpushしdraw関数で描画する。

deleteをクリックするとdelelm関数を呼び出し、メニューを見えないようにする。atariに-999を代入しておく。delelm関数はatariのデータを除外し、もう一度情報を集めてelmarrayにまとめ、Json形式のデータ

に戻すことで削除を実装している。

3. 評価

3.1 評価結果

文学の授業で実際に生徒の方に使用していただきアンケートを取った。

アンケート評価の結果からこれらのアプリケーションは有用性があるという評価がもたらされた。使いやすさでも使いやすいと答えてくれた方の割合が高かった。ビジュアル評価に関しても見やすいと答えてくれた方の割合が高かった。

今回アンケート評価の中でいただいた意見の中で重要なものを以下に示す。説明部分と回答部分が見えづらい。画像が大きすぎて見づらい。翻訳の答えと写真を並べて表示されるのは見やすいですが、途中からずれているし、ともに全文を見られるようにしてほしい。index でいちいち戻らなくてもいいようになったらテキスト集から直接移動できるようにすると思う。上下つながった形で見られるようにしてほしい。下の方のくずし字が使いづらい。読もうとしている文章の画像を張り付けられたりすることが出来たら良いと思う。

源氏物語翻刻アプリで1つ目の意見は多かったため、今後わかりやすく表示できるようテキストボックスのサイズ変更などの工夫が必要である。2つ目の意見は行の画像の表示が大きすぎるため感じるとのことであった。下の回答欄とともに見えるくらいにまで大きさを変更できるよう改善する必要がある。3つ目、4つ目はテキスト集のアプリケーションの利便性がさらにあがる工夫であり、今後の課題である。5つ目の意見はくずし字とは上下の繋がりが重要であるので画像を繋げられるようにしてほしいというものである。そのため画像のみを大きく表示できるような工夫が必要である。6つ目の意見はくずし字 DB においてくずし字を動かす際に下の方のくずし字が使いづらいというものであり、検索などのソート機能が必要である。7つ目の意見はくずし字 DB アプリの利便性がさらにあがる工夫であり、今後の課題である。

4. 結論と今後の課題

4.1 結論

本研究で開発したアプリケーションは源氏物語を使用したくずし字学習アプリケーションができた。また、アンケート調査の結果から、本研究の目的であるくずし字の翻刻が行える人材を増加するため、アプリケーションを開発している際には見えなかった改善点や思いつかなかった工夫点が多数存在し、今後さらに良いアプリケーションにできるような意見が多く寄せられ

た。

4.2 今後の課題

源氏物語翻刻アプリはアンケート評価の意見にもあったように回答欄と画像を見やすくする必要がある。また、工夫点として背景色を変更すれば良いのではという意見があったため、それを参考に見やすくしていく。

テキスト集はアンケート評価の意見にあったように文字を画像の位置に合わせての表示、index に戻らなくとも他のページに移動できるような機能を実装する必要がある。また、全文をとともに確認できるような機能を実装できるよう調査していく。

くずし字 DB は下の方くずし字が使いづらいというものがあるためアンケート評価の意見にあったように検索やソートが出来ることが必要である。また、くずし字の画像の大きく変更できるような機能を実装すべきである。アンケート評価の意見にある読みたい文章の画像が見える位置に張り付けるという機能が追加すると利便性が上がると考えられる。しかし、今回開発したアプリケーションでは画像の任意の取得や表示などの機能は一切ないため根本的なプログラムの修正が必要である。

今回のアンケート評価で良いと感じた点についても多く意見をいただいたため、それらを残しつつ今後はさらなる学習の効率化を目指すため、誰にでも利用しやすいよう改善していくことが課題となる。

参考文献

- [1] ROIS-DS 人文学オープンデータ共同利用センター, 「みを (miwo): AI くずし字認識アプリ」, <http://codh.rois.ac.jp/miwo/> (参照 2024-02-09)
- [2] ROIS-DS 人文学オープンデータ共同利用センター, 「源氏物語 | 日本古典籍データセット」, <http://codh.rois.ac.jp/pmjt/book/200003803/>, doi:10.20730/200003803(参照 2024-02-09)
- [3] ROIS-DS 人文学オープンデータ共同利用センター, 「日本古典籍くずし字データセット | ROIS-DS 人文学オープンデータ共同利用センター (CODH)」, <http://codh.rois.ac.jp/char-shape/>, doi:10.20676/00000340(参照 2024-02-09)