JAIST Repository
https://dspace.jaist.ac.jp/

Title	Text ComposTer: 文章作成の上流工程で生じる棄却テキスト断片を知的資源化するシステム
Author(s)	生田, 泰章; 西本, 一志
Citation	情報処理学会研究報告. HCI, ヒューマンコンピュータインタラクション研究会報告, 2016-HCI-168(6): 1-8
Issue Date	2016-05-26
Type	Journal Article
Text version	publisher
URL	http://hdl.handle.net/10119/14069
Rights	社団法人　情報処理学会, 生田 泰章, 西本 一志, 情報処理学会研究報告. HCI, ヒューマンコンピュータインタラクション研究会報告, 2016-HCI-168(6), 2016, 1-8. ここに掲載した著作物の利用に関する注意: 本著作物の著作権は（社）情報処理学会に帰属します。本著作物は著作権者である情報処理学会の許可のもとに掲載するものです。ご利用に当たっては「著作権法」ならびに「情報処理学会倫理綱領」に従うことをお願いいたします。 Notice for the use of this material: The copyright of this material is retained by the Information Processing Society of Japan (IPSJ). This material is published on this web site with the agreement of the author (s) and the IPSJ. Please be complied with Copyright Law of Japan and the Code of Ethics of the IPSJ if any users wish to reproduce, make derivative work, distribute or make available to the public any part or whole thereof. All Rights Reserved, Copyright (C) Information Processing Society of Japan.
Description	

JAIST
JAPAN
ADVANCED INSTITUTE OF
SCIENCE AND TECHNOLOGY

Japan Advanced Institute of Science and Technology

Vol.2016-HCI-168 No.6
Vol.2016-EC-40 No.6
2016/6/2

Text ComposTer：文章作成の上流工程で生じる棄却テキスト断片を知的資源化するシステム

生田泰章[†1]　西本一志[†1]

概要：文章作成過程において，執筆者は考えや思想の表出と表出内容の修正を繰り返し行う．その際，執筆者は不用と判断した本文の一部を棄却する．本稿では，棄却された本文の一部（棄却テキスト断片）を知的資源として捉え，棄却テキスト断片を収集可能な文章作成支援システム Text ComposTer を提案する．Text ComposTer は，ユーザがカード状のエレメントに作成対象の文章の一部を記入し，このエレメントを二次元平面上に配置することで本文を形成することができ，文章作成の上流工程から執筆者の文章作成を支援することができる．エレメントを配置するための二次元平面には，配置されたエレメントに記入された内容を本文に反映する領域と，反映しない領域が含まれる．Text ComposTer は，本文に反映しない領域に配置されたエレメントに記入されたテキスト断片を，後に知的資源として活用すべく収集する機能を有する．本稿においては，Text ComposTer を用いた実験を行い，有用性を確認することができた．

キーワード：棄却テキスト断片，文章作成支援システム，知識活用

Text ComposTer: A Text Processing System That Converts Unadopted Text Pieces to Intellectual Resource

HIROAKI IKUTA[†1]　KAZUSHI NISHIMOTO[†1]

Abstract: In a writing process, an author externalizes his/ her thought or idea and revises them recursively. In this process, the author sometimes decides not to adopt some parts of the written text because of, for instance, its inconsistency with the rest parts. However, such unadopted text pieces may be meaningful and useful in some different contexts. In this paper, in order to utilize the unadopted text pieces as intellectual resource, we propose a text processing system named “Text ComposTer”. A user of Text ComposTer firstly generates a “text element” that is a rectangle object and writes some text into the text element. Text ComposTer is equipped with three areas: a display area, an adopting area and a pending area. The display area shows an entire body text by concatenating the text pieces written in the text elements that are located in the adopting area from top to bottom, while the text pieces written in the text elements located in the pending area are not reflected to the entire body text shown in the display area. Finally, when the user finishes the writing, Text ComposTer outputs the body text in the display area as a text file as well as collects and stores the text pieces in the text objects located in the pending area to utilize them as intellectual resource afterwards. In this paper, we illustrate the setup of Text ComposTer and a pilot study to estimate whether useful unadopted text pieces can be efficiently collected by using Text ComposTer.

Keywords: Unadopted Text Pieces, Text Processing System, Knowledge Utilization

1. はじめに

今日まで，論文や技術資料等の文章化された知識が数多く創出されてきた．一般的に，論文や技術資料等の文章は，ある主題に基づいて論旨が展開されることにより，首尾一貫した内容で構成されることが好ましい．文章作成過程において，執筆者は，考えや思想の表出作業，表出内容の修正作業を再帰的に行い，首尾一貫性を高めた文章を作成する[1]．しかしながら執筆者は，文書作成作業の開始時点からいきなり首尾一貫した構成を持つ文章を作成できるわけではない．想定される多くの可能性を考慮しつつ多様なテキスト断片群を執筆し，その執筆過程で次第に全体的論旨を固め，それに沿って執筆したテキスト断片群を取捨選択し，さらに肉付けすることによって，最終的な文章を構成する．このような文章作成過程において，執筆者の表現したい主題と関連のないテキスト断片は，いったん創出されても最終的には棄却されてしまい，通常は活用されることはない．

しかしながら，このようなテキスト断片などとしていったん創出されたにもかかわらず最終的に棄却された知識や情報が，元の主題とは関連のない別の新たな主題の下で，知識の創出に有効に活用される場合がまれにある．例えば，ポスト・イット®に使用される接着剤は，一般的な接着剤の開発過程において創り出された失敗作であり，本来棄却されるものであったが偶然保管されていた．そして，この接着剤は，付箋を接着面に付け外し自在とするための主要な構成要素として，今日においても活用されている[2]．また，本稿第 2 著者は，過去に実施したピアノ演奏の表現生成に関する研究[3]の中で，打鍵とそれによる発音のタイミ

†1 北陸先端科学技術大学院大学
Japan Institute of Science and Technology

ⓒ2016 Information Processing Society of Japan

ングに微小な遅延がある場合，演奏者はそれを発音の遅延としてではなく，鍵盤の重さの増加として認知することを見出し，論文[3]の執筆過程でいったんその発見を文章化した．しかし，最終的な論文では，この知見に関する文章は論文[3]の主題と関連しないために棄却された．後年，本稿第 2 著者は，この棄却された文章を読み返した際，当時進行中であったドラム演奏支援の研究への応用可能性を見出し，論文[4]として研究成果をまとめた．

以上のようなケースはきわめてまれであり，従来の文書作成過程においては，主題に関連のない知識や情報はほとんどすべて単純に棄却されてきたため，それらが新たな知識の創出につながる機会が失われているおそれがある．そこで，筆者らは文章作成過程で棄却された本文の一部（棄却テキスト断片）を収集するためのテキストエディタ（棄却テキスト断片収集エディタ）を実装し，収集した棄却テキスト断片の活用可能性についての検討をこれまで行ってきた[5]．しかしながら，作成した棄却テキスト断片収集エディタの表現系と操作系が，文章作成過程の最終工程（いわゆる清書工程）に適したものになっていたため，主として文章作成過程の上流工程で生じやすい，活用可能性の高い棄却テキスト断片を十分に収集することができていないことが判明した．

そこで本稿においては，活用可能性の高い棄却テキスト断片を収集可能な Text ComposTer を提案する．Text ComposTer は，執筆者による文章作成の上流工程をも支援することができる文章作成支援システムであり，この上流工程で生じた棄却テキスト断片を知的資源化すべく収集することができる．また，本稿においては Text ComposTer で収集した棄却テキスト断片について活用可能性を検討する．

以下，2 章では棄却テキスト断片収集エディタとその課題について述べ，3 章では提案手法である Text ComposTer について説明し，4 章では Text ComposTer の評価実験の結果を説明する．5 章では，4 章の実験結果を踏まえ，Text ComposTer の有用性について議論する．6 章では関連研究を概観し，7 章にてまとめを述べる．

2. 棄却テキスト断片収集エディタ

棄却テキスト断片収集エディタは，図 1 に示すような，Windows OS に付属するテキストエディタである「メモ帳」と同様の文書編集画面を有している．また，「コピー」，「切り取り」，「貼り付け」，「検索」，「置換」等の編集機能を実装し，執筆者はこれらの編集機能を用いながら文章を編集することができる．図 2 に，検索・置換機能画面を示す．執筆者が文章を編集中に，文字，単語，文，段落等のテキスト断片を削除した場合，棄却テキスト断片収集エディタは，削除されたテキスト断片を棄却テキスト断片として収集する．具体的には，棄却テキスト断片は執筆者が以下の 3 つの操作を実行したときに収集される．

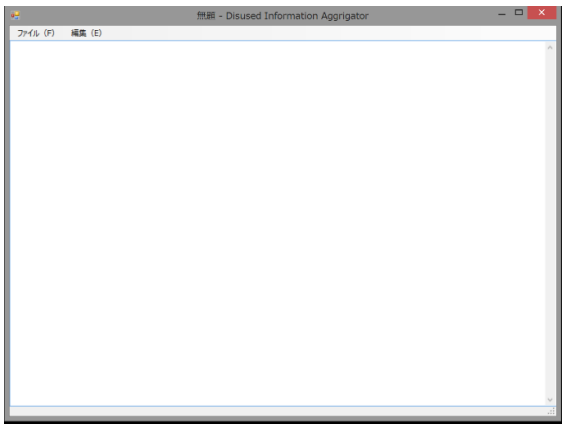

図 1. 棄却テキスト断片収集エディタの編集画面

図 2. 検索・置換機能画面

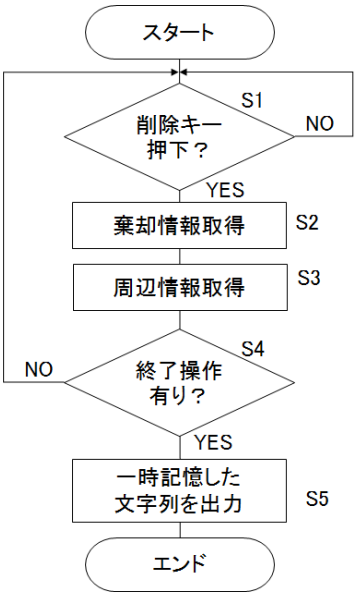

図 3. 棄却テキスト断片を収集するときの処理手順

(1) 削除キーの操作
(2) 文字列が範囲選択された状態での文字入力
(3) 置換機能の実行

ここで削除キーとは，「Delete キー」及び「Backspace キー」の両方を表す．

また，上記の操作が行われた際には，棄却テキスト断片に加え，当該棄却テキスト断片の周辺情報を併せて収集する．周辺情報とは棄却テキスト断片の前後の情報であり，

ⓒ2016 Information Processing Society of Japan

棄却テキスト断片とその前後それぞれにある区切り文字とで囲まれた文字列である．本稿における区切り文字は「.」「。」とした．例えば，ある文に含まれる単語が削除された場合，その文に含まれる，削除された単語を除く残りの文字列を周辺情報として収集する．

図 3 は，削除キーが執筆者に操作されたときにおける棄却テキスト断片収集エディタの処理手順を示すフローチャートである．まず，執筆者によって削除キーが押下されたか否かを判定する（ステップ S1）．削除キーが押下されていないと判定した場合（ステップ S1：NO），執筆者から受け付けたその他の処理を実行し，削除キーが押下されるまで処理を待つ．削除キーが押下されたと判定した場合（ステップ S1：YES），棄却テキスト断片を取得する（ステップ S2）．ステップ S2 において削除キーに応じた文字又は文字列を切り取り，一時的に記憶しておく．削除キーに応じた文字とは，「Delete キー」が削除キーであった場合は，キャレット直前の 1 文字又は範囲選択された文字列であり，「Backspace キー」が削除キーであった場合は，キャレット直後の文字又は範囲選択された文字列である．その後，周辺情報を取得する（ステップ S3）．すなわち，キャレットの直前及び直後にある区切り文字までの文字列をコピーし，一時的に記憶しておく．

次いで，削除動作の終了を示す操作があったか否かを判定する（ステップ S4）．削除動作の終了を示す操作とは，削除動作が継続して行われないことを示す操作であり，削除キー以外のキーボード操作，及びマウスクリックの操作として実装されている．削除動作の終了を示す操作がなかったと判定した場合（ステップ S4：NO），ステップ S1 に処理を戻す．そして，再びステップ S2 の処理を行うに際して，先に一時記憶した文字列又は文字に追加するように対象の文字列又は文字の一時記憶を行う．また，再びステップ S3 の処理を行うに際して，周辺情報の更新を行う．削除動作の終了を示す操作があったと判定した場合（ステップ S4：YES），ステップ S2 で取得された棄却テキスト断片及びステップ S3 で取得された周辺情報それぞれを表す文字列をファイルに出力する（ステップ S5）．ステップ S1～ステップ S4 の処理を行うことによって，複数回にわたって削除キーが押下された場合に，押下された回数分の文字列を一単位としてファイルに出力することができ，最後に押下されたときにおける周辺情報をファイルに出力することができる．

ステップ S5 の処理を行った後，棄却テキスト断片を収集する処理を終え，「日時」，「出力動作」，「文字列」，「周辺情報」を属性とする xml 形式にてファイル出力を行う．出力動作は，ファイルに出力するときに執筆者が行った動作を表し，本稿においては「削除」，「置換」等である．

文献[5]では，以上のように構成された棄却テキスト断片収集エディタを被験者に使用してもらう実験を行い，その結果，棄却テキスト断片収集エディタには 2 つの課題があり，各課題が生じる原因が以下のように考察されている．

(1) 課題：収集される棄却テキスト断片がきわめて雑多であること．
原因：収集された棄却テキスト断片のうち，誤記の訂正及び表現の修正により生成された活用可能性の低い棄却テキスト断片が大半を占めていたため．

(2) 課題：他の主題において活用されるかもしれない棄却テキスト断片の数が少ないこと．
原因：活用可能性の高い棄却テキスト断片そのものがそもそも少ないうえに，棄却テキスト断片収集エディタが，文章作成過程における最終工程に適した表現系と操作系を持つ，いわゆる清書用のエディタであったため．

3. Text ComposTer

3.1 課題克服のためのアプローチ

2 章で示した 2 つの課題を克服するための方法として，文章作成支援システム Text ComposTer を提案する．Text ComposTer は，文章作成の上流工程から下流工程まで，すなわち，文章の構想段階から構成段階，さらには清書段階までを一貫して支援し，各段階における文書作成行為に適した表現系と操作系を提供する．Text ComposTer は，それらの表現系と操作系を利用して，誤記訂正や表現の修正によって生じた棄却テキスト断片と，文脈や構成上の不整合などによって生じた棄却テキスト断片とを分別して収集することを可能とする．特に，Text ComposTer は，上流工程に適した文書の表現系と操作系を提供することによって，執筆者による文章の構想を練っている段階における多角的・発散的な知識の表出化の促進が期待できるため，最終的に本文に採用されないような知識をより多く収集することができるようになると考えられる．

3.2 システム構成

図 4 は，Text ComposTer の操作画面である．Text ComposTer は，テキスト断片が記入されたカード状のエレメントの配置位置に従って本文全体を形成することにより，執筆者の文章作成を支援するシステムであり，C#言語にて Windows 上に実装されている．Text ComposTer は，本文全体を表示する表示領域と，エレメントの配置が可能な配置領域を備える．また，配置領域は，反映領域と非反映領域を有しており，反映領域に配置されたエレメント内のテキスト断片は表示領域に反映され，非反映領域に配置されたエレメント内のテキスト断片は表示領域に反映されない．ここで，表示領域に反映されるテキスト断片の順序は，反映領域の上下方向におけるエレメントの配置位置に対応している．すなわち，反映領域の最上部に配置されたエレメント内に記入されたテキスト断片が表示領域にて最先に表示され，反映領域の最下部に配置されたエレメント内に記

ⓒ2016 Information Processing Society of Japan

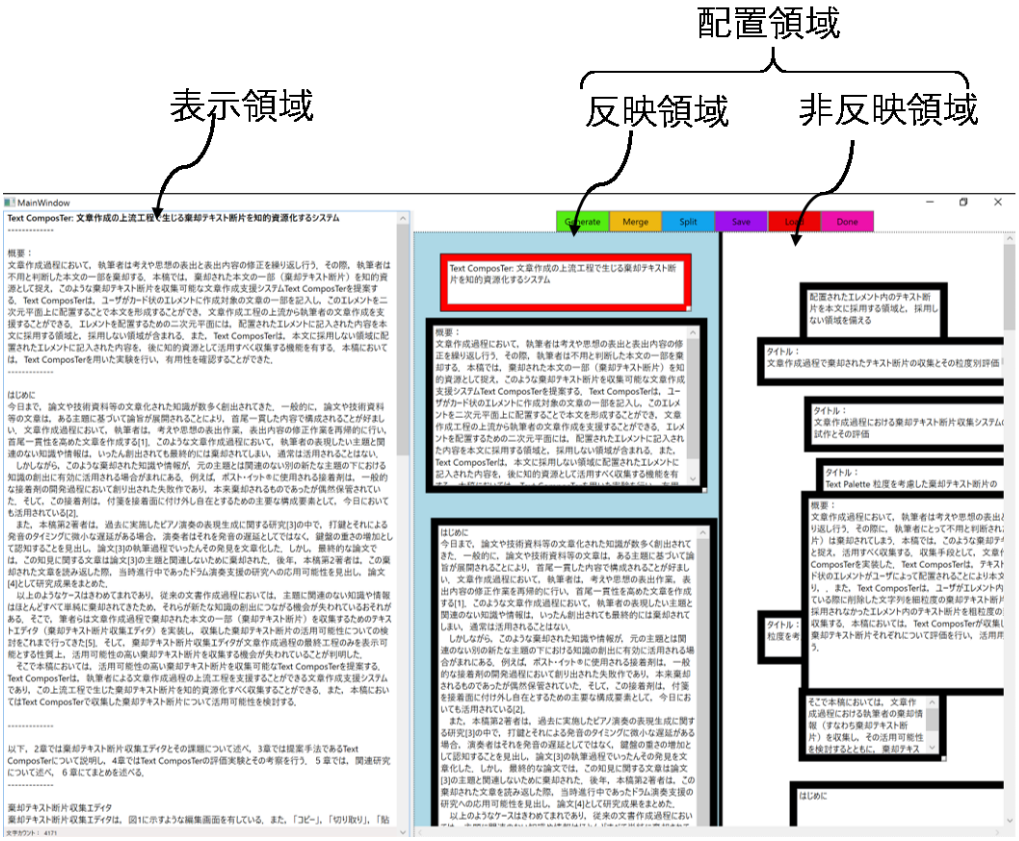

図 4. Text ComposTer の操作画面

入されたテキスト断片が表示領域にて最後に表示される．ユーザは，エレメントの生成，エレメント内へのテキスト断片の記入，反映領域又は非反映領域へのエレメントの配置を行うことによって，本文全体を作成する．

次に，Text ComposTer が有する各機能について説明する．

Generate 機能

Generate 機能は，配置領域内にエレメントを生成する機能である．ユーザは，配置領域上部に設けられた Generate ボタンを押下するか，配置領域内のコンテキストメニューにて Generate 機能を選択するか，又は Generate 機能を実行するためのショートカットキー（Ctrl＋G）を用いることで，配置領域の任意の位置にエレメントを生成することができる．

Merge 機能

Merge 機能は，反映領域に配置された複数個のエレメントを 1 つのエレメントに統合する機能である．Merge 機能にて複数個のエレメントが統合される際，各エレメント内に記入されているテキスト断片が 1 つのエレメント内に統合される．テキスト断片の統合順序はエレメントの配置位置に対応しており，統合対象となるエレメントにおいて，最上部に配置されたエレメント内に記入されたテキスト断片から最下部に配置された配置されたエレメント内に記入された文章断面まで順次統合される．ユーザは，統合対象となる 2 つ以上のエレメントを選択（Shift キーを押下しながらエレメントをクリック）し，配置領域上部に設けられた Merge ボタンを押下するか，配置領域内のコンテキストメニューにて Merge 機能を選択することで，選択した複数のエレメントを 1 つのエレメントに統合する．ユーザは Merge 機能を使用することで，例えば，同一のトピックについて書かれているものの複数のエレメントに分かれて記入されているテキスト断片を，そのトピックについてのテキスト断片として 1 つのエレメントに統合することができる．

Split 機能

Split 機能は，1 つのエレメントを 2 つのエレメントに分割する機能である．ユーザは，分割対象となるエレメント内に記入されたテキスト断片の一部を範囲選択し，配置領域上部に設けられた Split ボタンを押下するか，エレメント内のコンテキストメニューにて Split 機能を選択する．そうすることで，Text ComposTer は分割対象のエレメントから選択されたテキスト断片を削除し，このテキスト断片を記入してある新たなエレメントを生成することで Split 機能を実現する．ユーザは，例えば 1 つのエレメント内に記入されている棄却テキスト断片が 2 つのトピックについて書かれている場合に，Split 機能を使用することで，2 つのエレメントそれぞれを 1 つのトピックについて書かれたテキスト断片が記入されたものとすることができる．

Save 機能

ⓒ2016 Information Processing Society of Japan

Save 機能は，操作画面内の情報を保存する機能である．ユーザは，操作画面内の情報の保存を所望する任意のタイミングで, 配置領域上部に設けられた Save ボタンを押下するか, 配置領域内のコンテキストメニューにて Save 機能を選択する．その後，ユーザは操作画面上に表示された保存用のダイアログボックスにて保存するファイル名の入力，保存先を選択し，保存ボタンを押下する．Text ComposTer は，保存ボタンが押下されたことを検知し，操作画面に係る情報を選択された保存先に入力されたファイル名にて XML ファイル形式で保存する．

Load 機能

Load 機能は，Save 機能で保存された操作画面を現在の操作画面に反映する機能である．ユーザは，保存された操作画面の反映を所望する任意のタイミングで，配置領域上部に設けられた Load ボタンを押下する．その後，ユーザは操作画面上に表示されたファイルを開くためのダイアログボックスにて保存された XML ファイルを選択し，開くボタンを押下する．Text ComposTer は，開くボタンが押下されたことを検知し，選択された XML ファイルが表す操作画面を表示する．

Done 機能

Done 機能は, 表示領域に表示された本文全体を別の外部ファイルとして出力する機能である．ユーザは，当該機能の実行を所望する任意のタイミングで，配置領域上部に設けられた Done ボタンを押下するか，配置領域内のコンテキストメニューにて Done 機能を選択する．その後，ユーザは操作画面上に表示された保存用のダイアログボックスにて保存するファイル名の入力，保存先を選択し，保存ボタンを押下する．Text ComposTer は，保存ボタンが押下されたことを検知し，操作画面に係る情報を選択された保存先に入力されたファイル名にてテキストファイル形式で保存する．ユーザは例えば，文章作成が完了したときに Done 機能を実行することにより，外部ファイルに本文全体を出力することができる．

3.3 棄却テキスト断片の収集

Text ComposTer は，棄却テキスト断片を収集する機能を有しており，粗粒度と細粒度の 2 種類の棄却テキスト断片を収集する．Text ComposTer は，Done 機能が実行されたときに非反映領域内に配置された各エレメントに記入されているテキスト断片それぞれを粗粒度の棄却テキスト断片として収集する．具体的には， Done 機能が実行された場合に，非反映領域内に配置された各エレメントに記入されているテキスト断片それぞれを 1 つの XML ファイル形式で出力する．また，Text ComposTer は，各エレメント内のテキスト断片の編集中に削除された文字列を細粒度の棄却テキスト断片として収集する．なお，Text ComposTer が細粒度の棄却テキスト断片を収集するときの処理手順は，2 章で示した棄却テキスト断片収集エディタが棄却テキスト断片を収集するときの処理手順と同様である．

ユーザは，以上のような GUI 及び機能を備える Text ComposTer を用いることによって，エレメントの生成・テキスト断片の編集・エレメントの配置をインタラクティブに行い，本文を作成することができる．このように，Text ComposTer は，単に文章作成過程の最終状態を表示するだけでなく，文章作成過程全体を支援する機能を持つ．したがって Text ComposTer は，文章作成の上流工程で創造されたものの，最終的に本文に採用されなかった棄却テキスト断片を，2 章で述べた棄却テキスト断片収集エディタに比べてより多く収集することができると考えられる．また，粗粒度の棄却テキスト断片は，文章作成過程の上流工程から下流工程にわたって創造されたが，最終的に本文に採用されなかった知識であり，誤記の訂正によって生じた棄却テキスト断片でもないし，表現の修正によって生じた棄却テキスト断片でもない．しかも Text ComposTer は，粗粒度の棄却テキスト断片と細粒度の棄却テキスト断片を区別して収集することができる．ゆえに，Text ComposTer を用いることで，棄却テキスト断片が整理された状態で，包括的に棄却テキスト断片を収集することができると期待される．

4. 実験

4.1 実験 1: 粗粒度の棄却テキスト断片と細粒度の棄却テキスト断片の比較実験

Text ComposTer が収集した粗粒度の棄却テキスト断片と細粒度の棄却テキスト断片を比較するために，4 名の被験者それぞれに Text ComposTer を用いて文章作成を行ってもらい，粗粒度の棄却テキスト断片と細粒度の棄却テキスト断片を収集する実験を行った．被験者はすべて，修士課程の学生であり，対象となる文章は所定の書式にて記載する必要のある研究計画書であった．なお，今回の実験における各被験者は，研究計画書の構想段階から第 1 稿又は第 2 稿を完成させるために Text ComposTer を用いて文章作成を行っており，完成稿までは Text ComposTer を使用していない．

また，被験者それぞれに，自身の文章作成で生じた粗粒度の棄却テキスト断片と，細粒度の棄却テキスト断片が有用であるか否かの評価をしてもらった．具体的には，被験者には粗粒度の棄却テキスト断片であるか細粒度の棄却テキスト断片であるかを示さず，各棄却テキスト断片が「有用」であるか「無用」であるか「どちらでもない」かの 3 段階で評価してもらった．

実験結果：数の比較

表 1 は，各被験者から収集された棄却テキスト断片の統計値を粒度別にまとめたものであり，「数」と，「平均文字数」と，文字数の「標準偏差」が示されている．表 1 に示されているように，どの被験者から収集された粗粒度の棄却テキスト断片であっても，細粒度の棄却テキスト断片に

ⓒ2016 Information Processing Society of Japan

表 1. 収集された棄却テキスト断片の統計値

		数	平均文字数	標準偏差
粗粒度	被験者1	5	240.6667	233.2600
	被験者2	8	86.6250	51.8361
	被験者3	0	ー	ー
	被験者4	6	97.6667	69.8538
	全体	19	106.4737	84.7984
細粒度	被験者1	196	5.3163	14.4821
	被験者2	309	3.9029	5.7124
	被験者3	38	10.1842	14.4495
	被験者4	84	6.5176	28.7859
	全体	627	4.6571	10.0065

表 2. 棄却テキスト断片の有用性の評価結果

		有用	無用	どちらでもない
粗粒度	被験者1	3 (50%)	3 (50%)	0 (0%)
	被験者2	5 (62.5%)	3 (37.5%)	0 (0%)
	被験者3	0 (-%)	0 (-%)	0 (-%)
	被験者4	3 (50%)	0 (0%)	3 (50%)
	全体	11 (55%)	6 (30%)	3 (15%)
細粒度	被験者1	1 (約0.5%)	195 (約99.5%)	0 (0%)
	被験者2	5 (約1.6%)	303 (約98.1%)	1 (約0.3%)
	被験者3	5 (約13.2%)	12 (約31.6%)	21 (約55.3%)
	被験者4	0 (0%)	84 (100%)	0 (0%)
	全体	11 (約1.8%)	594 (94.7%)	22 (約3.5%)

()内は全評価に対する割合

比べ数が少なく，平均文字数が多く，文字数のばらつきが大きかった．

実験結果：質の比較

表 2 は，各被験者が自身の文章作成で生じた棄却テキスト断片の有用性についてどのように評価したのかを，棄却テキスト断片の粒度別にまとめたものである．表 2 に示されているように，各被験者における有用性の判断結果全体に対する有用と判断された数の割合は，粗粒度の棄却テキスト断片の方が大きく上回る結果となった．また，有用と判断された棄却テキスト断片は粒度の種類を問わず，文字数が多いか，あるいは専門用語が含まれている傾向にあった．

4.2 実験 2: Text ComposTer と棄却テキスト断片収集エディタの比較実験

Text ComposTer と棄却テキスト断片収集エディタを比較するために，8 名の被験者それぞれに 2 つの文章を順に作成する課題に取り組んでもらった．表 3 は，各被験者に作成してもらう文章のテーマと，使用してもらうシステムの組み合わせを示したものである（なお，表 1 の被験者番号と表 3 の被験者番号は対応していない）．各被験者には，Text ComposTer（表 3 中「C」）を用いて一方の文章を作成してもらい，棄却テキスト断片収集エディタ（表 3 中「E」）を用いて他方の文章を作成してもらった．また，作成してもらう文章のテーマは，「10 年後の携帯電話はどうなっているかを予想してください．その際外観，機能面のそれぞれの観点から述べてください」（表 3 中 T_1）と「JAIST のことを世間にもっと知ってもらうにはどのようにすれば良いか．その方法とその方法のメリット・デメリットを少なくとも一組書いてください」（表 3 中 T_2）と設定した．さらに，各レポート課題の作成時間は 30 分程度と設定し，作成文字数は 100 字以上 400 字以下になるように各被験者に指示した．また，被験者には 1 回目の文章作成と 2 回目の文章作成の間に最大 5 分の休憩をとってもらった．

表 3. 文章作成テーマ及び使用システムの組み合わせ

	1回目	2回目
被験者1	T_1，E	T_2，C
被験者2	T_1，E	T_2，C
被験者3	T_2，C	T_1，E
被験者4	T_2，C	T_1，E
被験者5	T_1，C	T_2，E
被験者6	T_1，C	T_2，E
被験者7	T_2，E	T_1，C
被験者8	T_2，E	T_1，C

また，2 つの文章作成後の各被験者にインタビューを行った．主な質問項目は，「Text ComposTer と棄却テキスト断片収集エディタの使用感」と，「どのようにシステムを使って文章作成を進めていったか」とした．

実験結果

表 4 は，被験者が Text ComposTer と棄却テキスト断片収集エディタそれぞれを使用した際に収集された棄却テキスト断片の統計値をテーマ毎にまとめたものである．統計値は，各被験者から収集された棄却テキスト断片の「平均個数」，「平均文字数」，文字数の「標準偏差」である．表 4

表 4. 収集された棄却テキスト断片の統計値

		T_1			T_2		
		平均個数	平均文字数	標準偏差	平均個数	平均文字数	標準偏差
Text ComposTer	粗粒度	5.75	40.2727	50.1789	2.25	63.3333	47.8774
	細粒度	74.75	4.7715	9.2677	42.25	4.5444	6.8937
棄却テキスト断片収集エディタ		41.5	6.0663	19.1707	97	6.4897	13.2141

ⓒ2016 Information Processing Society of Japan

に示されているように，各テーマとも棄却テキスト断片収集エディタで収集された棄却テキスト断片と，細粒度の棄却テキスト断片は平均文字数が同じであったが，棄却テキスト断片収集エディタで収集された棄却テキスト断片の方が文字数のばらつきが大きかった．これは，棄却テキスト断片収集エディタが収集した棄却テキスト断片には，粗粒度に対応する棄却テキスト断片と，細粒度に対応する棄却テキスト断片が混在しているからである．また，実験 1 と同様，粗粒度の棄却テキスト断片は，平均文字数が多く，かつ文字数のばらつきも大きかった．

インタビューの結果，一部の被験者は，文章作成を行う手順が確立されており，その手順に従って両システムを同様に使用していることが確認された．被験者 1 は，文章の流れを頭の中で形成されてから，本文を作成する手順を採用している．そのため，被験者 1 は，Text ComposTer 及び棄却テキスト断片収集エディタの両者を文章の最終状態を表現するためのみに使用しており，棄却テキスト断片の収集数が他の被験者に比べ極端に少なかった．

被験者 2，3，4，5，及び 8 は文章作成に際して，まず本文に採用する候補となるキーワードやアイデアを羅列し，羅列したキーワードやアイデアの中から本文に採用するものを選定して本文を作成する手順を採用している．そのため，被験者 2，3，及び 8 は，Text ComposTer の使用により文章を形成するに際して，エレメントを複数生成してキーワード及びアイデアを書き出し，その後，本文を清書するためのエレメントを生成し，書き出したキーワード及びアイデアの中から本文に採用するものを参照しつつ，本文の清書を行っていた．最終的に，キーワード及びアイデアが記入されたエレメントは非反映領域に配されていた．被験者 4 及び 5 は，Text ComposTer を使用するに際して,被験者 2，3，8 と同様に複数のエレメントにキーワード及びアイデアを書き出していた．その後，被験者 4 は，本文に採用するキーワード及びアイデアのエレメントに肉付けを行うことで文章化し，最後に文章化したテキスト断片を含む各エレメントを Merge 機能により統合することにより本文を形成した．

一方，被験者 2，3，4，5，及び 8 は，棄却テキスト断片収集エディタの使用により文章を形成するに際して，編集画面の上部又は下部にキーワード及びアイデアを書き出し，その後，書き出したキーワード及びアイデアを参照しつつ本文を形成していた．最終的に，書き出されたキーワード及びアイデアは，削除されていた．ここで，被験者 3 及び 4 は,「もったいないなと思いながら書き出したキーワード及びアイデアを削除した」と報告している．

被験者 6 及び 7 は，Text ComposTer を使用するときと，棄却テキスト断片収集エディタを使用するときで文章の作成態様を変えていた．具体的には，被験者 6 及び 7 は，Text ComposTer を使用したときは，被験者 2，3，4，5 及び 8 と同様の使用態様を採用し，棄却テキスト断片収集エディタを使用したときは，被験者 1 と同様の使用態様を採用していた．

さらに，Text ComposTer は，棄却テキスト断片収集エディタと比べ,「文章作成をするときのアイデアを出しやすい」,「頭を整理させて文章作成を行うことができるため，構造的に文章が書ける」という優位性を示すインタビュー結果が得られた．反面,「エレメントに文章を記入していくことで本文を作成することに違和感がある」という劣位性を示すインタビュー結果も得られた．

5. 議論

以上の実験結果を踏まえ，Text Composter の有用性について検討する．まず，Text ComposTer が棄却テキスト断片収集エディタの課題であった「収集される棄却テキスト断片がきわめて雑多であること」を克服できたかどうかについて議論する．Text ComposTer が収集する粗粒度の棄却テキスト断片は，非反映領域に配置されたエレメントに記入されたテキスト断片である．そのため，上述したように，粗粒度の棄却テキスト断片は誤記の訂正に起因して生じたものでもないし，表現の修正に起因して生じたものでもない．また，実験 1 における表 2 に示されたように，粗粒度の棄却テキスト断片は，被験者が有用と判断する割合が高く，細粒度の棄却テキスト断片は，被験者が有用と判断する割合が低い．したがって，粗粒度の棄却テキスト断片と細粒度の棄却テキスト断片を区別して収集可能な Text ComposTer は，上述の課題を克服することができたものと考えることができる．

次に，棄却テキスト断片収集エディタのもう一方の課題であった「他の主題において活用されるかもしれない棄却テキスト断片の数が少ないこと」を克服できたかどうかについて議論する．実験 2 で示したように，被験者が文章作成を行う手順を確立している場合，使用するシステムによらず,同様の手順にて文章作成が行われることが判明した．例えば，実験 2 の被験者 2，3，4，5，及び 8 が棄却テキスト断片収集エディタを使用した際に収集された棄却テキスト断片には，本文に採用されていないキーワードやアイデア等の活用可能性の高い棄却テキスト断片が，粗粒度の棄却テキスト断片数と同程度存在することが確認された．したがって，これらの被験者に関しては，Text ComposTer によって当該課題を克服できたとは言い難い. しかしながら，Text ComposTer は，粗粒度の棄却テキスト断片と細粒度の棄却テキスト断片を区別して収集しているため，活用可能性の高い棄却テキスト断片をより簡便に取り出すことができる．文献[5]で議論されているように，収集された棄却テキスト断片が雑多であった場合，その中から，活用可能性の高い棄却テキスト断片を自動的に取り出すことは難しい．したがって，Text ComposTer は棄却テキスト断片収集エデ

ⓒ2016 Information Processing Society of Japan

ィタに比べ，棄却テキスト断片，特に粗粒度の棄却テキスト断片を知的資源化することができる可能性が高い．

また，以上の実験から Text ComposTer が有する課題も確認された．Text ComposTer では，ユーザが文章を記入する対象はエレメントのみである．つまり，ユーザが文章作成に際してキーワードやアイデアを書き出すことと，本文を作成することがエレメントにテキスト断片を記入する行為として同一視されている．しかしながら，キーワードやアイデアを記入する行為は発散的であり，本文を形成する行為は収束的であるため,そもそもの思考プロセスが異なる．そのため，実験 2 において，被験者は，清書用のエレメントを生成したり，本文の作成に際しての違和感を覚えたりしたものと考えられる．このような課題を克服するためには，文章作成に際して，発散的思考による行為と，収束的思考による行為とを明確に切り分け，かつこれらの行為を再帰的かつシームレスに行うことができる仕組みが必要であると思われる．従来，文章作成過程において文章全体の構造を決定するトップダウンとテキスト断片を組み立てていくボトムアップの切り替えが重要という指摘されてきた[7]が，ボトムアップの工程を支援する仕組みがさらに必要であると考えられる．

6. 関連研究

本稿で提案した Text ComposTer のように，テキスト断片を二次元平面上に配置することで文章作成を行うシステムが従来から多くの研究がなされている[6, 7]．Art#001[6]は，Text ComposTer と同様，エレメントを線形的に配置することで文章作成を行うシステムである．Art#001 は，Text ComposTer の表示領域及び反映領域により構成されたシステムに対応しているため，生成されたエレメントはすべて文章全体に反映される．iWeaver[7]は，MapView と呼ばれる領域で作成された章立て等の項目を配置可能なように構成され，本文に採用する項目を OutlineView に追加することで，文章全体の構造化する．iWeaver は，MapView 内に項目を追加したものの本文に採用しない項目は OutlineVew に追加しないことで，Text ComposTer の非反映領域にエレメントを配置することと同様に機能する．しかしながら，Art#001，iWeaver ともに棄却された情報を活用する目的で収集する機能は有しておらず，棄却された情報の知的資源化を行ってはいない．これに対し，Text ComposTer は，棄却された情報を知的資源化すべく, Done 機能を有している点に特徴がある．

7. まとめ

本稿においては，文章作成過程における上流工程から下流工程で創造されたものの，最終的に本文に採用されなかった棄却テキスト断片を収集可能な Text ComposTer を実装した．Text ComposTer は，粗粒度と細粒度の 2 種類の棄却テキスト断片を収集可能であり，実験により粗粒度の棄却テキスト断片に有用なものが含まれている可能性が高いことを確認した. 今後は, 5 章で述べたように, Text ComposTer に文章作成過程におけるボトムアップ工程を支援する機能を実装し，より多くの棄却テキスト断片を収集するシステムを構築予定である．また，Text CopmosTer で収集した棄却テキスト断片の活用法を検討していく．

謝辞 本研究は，JSPS 科研費 15K12093 の助成を受けたものです．本稿の執筆に当たり，実験に協力下さった被験者の方々に謝意を表します．

参考文献

[1] 丸野俊一：対話の視点から捉えた書く時の推敲過程，人工知能学会誌，Vol. 23 No. 1, pp.293-302 (2008).
[2] 宮永博史：セレンディピティと MOT，三菱 UFJ リサーチ＆コンサルティング 季刊 政策・経営研究 2009 vol.3 pp. 50-60 (2009).
[3] 大島千佳，西本一志，阿部明典：ピアノ演奏における離鍵速度の重要性と特性に関する考察, 情報処理学会論文誌, Vol.47, NO.5, pp.1546-1557 (2006).
[4] 池之上あかり，小倉加奈代，鵜木祐史，西本一志：微小遅延聴覚フィードバックを応用したドラム演奏フォーム改善支援システム, Vol.1, No.1, pp. 15-24 (2013).
[5] 生田泰章，才記駿平，西本一志：文章作成過程における棄却テキスト断片の活用に関する一検討，インタラクション 2016 論文集，1B35, pp. 302-305 (2016).
[6] Nakakoji, K., Yamamoto, Y., Reeves, B.N., Takada, S., Two-Dimensional Positioning as a Means for Reflection in Design, Proceedings of Design of Interactive Systems (DIS'2000), ACM Press, New York, NY, pp.145-154, (2000).
[7] 柴田博仁，堀浩一：デザインプロセスとしての文章作成を支援する枠組み，情報処理学会論文誌，Vol. 44 NO. 3, pp.1000-1012, (2003).

ⓒ2016 Information Processing Society of Japan