The Sixteenth International Symposium on Artificial Life and Robotics 2011 (AROB 16th '11),
B-Con Plaza, Beppu,Oita, Japan, January 27-29, 2011

An effective visualization of style inconsistencies for interactive text editing

Kazuki Shimamura
Department of Engineering Informatics,
Osaka Electro-Communication University
18-8, Hatsu-cho, Neyagawa,
572-8530, Japan
Email: shimason.1988@takelab.jp

Kazuhiro Takeuchi
Department of Engineering Informatics,
Osaka Electro-Communication University
18-8, Hatsu-cho, Neyagawa,
572-8530, Japan
Email: kazuh@takelab.jp

Kiyota Hashimoto
School of Humanities
and Social Sciences,
Osaka Prefecture University
1-1, Gakuen-cho, Naka-ku, Sakai,
599-8531, Japan
Email: hash@lc.osakafu-u.ac.jp

***Abstract* : Any authors, particularly learners, have much difficulty choosing the proper style for their particular writing and detecting style inconsistencies. We propose a new system that allows users to revise documents through human-system interaction. Although many systems for text writing have been proposed, most of the works focused mainly on automated techniques that detect human errors in texts. In contrast to those works, our study focuses on the visualization of multi-level style inconsistencies in texts to promote authors' awareness. In order to evaluate and visualize the differences in styles, we propose a model to compute the style similarity between a part and some genres. The similarity function that we propose is based on a model in which sentences are regarded as sequences of functional expressions. Applying the function, we develop a tryout system that shows which parts are inconsistent with the other parts from various viewpoints. Through interactions between users and the system, the user can repeat revising the text until the text maintains consistencies in various levels. Much has to be done towards a practically effective system, but our system helps to point out undesirable text should be conscious of stylistic differences in writing text.**

Keywords : **writing support, style consistency, corpora-based approach, awareness promoting visualization**

I. INTRODUCTION

We can easily distinguish a newspaper article and a newspaper editorial, or a textbook for graduate students and one for undergraduate or high school students, not just in terms of their contents but in terms of their style, though few of us can always make clear our criteria for this kind of distinction. How to narrate, or style, is highly related to the targeted audience and the author's communicative purpose of the text, and stylistic consistency is required in a text, though deliberate inconsistencies bring extra literary effects. [1]

Several studies have tackled with style; particularly it has been pointed out that basic stylistic consistency is held by the restrictive use of functional expressions particularly in the case of Japanese and other languages that have a rich variety of stylistic grammar forms. For example, the following pair has the same meaning ('This is a book' in English) but differs in the style:

- Kore-wa hon-desu.
- Kore-wa hon-da.

Both 'desu' and 'da' are auxiliary copular verbs and the difference is up to politeness, which in turn should be determined according to the targeted audience and the author's communicative purpose. So the mixed use of 'desu' and 'wa' causes undesirable stylistic inconsistency and thus should be avoided. Particularly for languages like Japanese, which has a rich set of style-sensitive functional expressions, the mere detection of the misuses of such functional expressions is useful to some degree for style consistency.

II. Environment for Text Editing

Stylistic consistency does not only depend on the inner consistency in the text, but also on the appropriate choice of style for the textual purpose. In other words, the targeted audience and the author's communicative purpose determine the desirable style; then the author, with his/her limited reading experiences, attempts to keep stylistic consistency: i.e., he or she tries to use as many appropriate stylistic features as possible and tries not to use inappropriate stylistic features. As a fundamental dimension of the features, it is necessary for him/her to be aware of the following two viewpoints.

- intra-sentential consistency
- passage-level consistency

For dealing with the first consistency in computer systems, many previous works have been developed. And now some of the modules that detect errors (and inconsistencies) of sentences are provided through WEB-API services in the Internet. The followings are an example of the list which we can use through of the such WEB-APIs. These kinds of the features, which detects sentence-level consistencies such as spelling errors and simple syntactic errors, have been implemented in some word processors such as *Microsoft Word.*

a) Spelling errors
b) Inappropriate use of Synonyms
c) Inappropriate use of Kanji-characters
d) Inconsistency of Proper Nouns

e) Inconsistency of Okurigana
f) Use of Double Negations
g) Use of Redundant Expressions
h) Inappropriate use of Abbreviations

On the other hand, some discussions still remain concerning visualization of the inconsistency from the second viewpoint (passage-level consistency). With this in mind, we have been investigated on the visualization of passage-level inconsistency based on a set of textual corpora that consists of two or more subgenres and extract stylistic features of each subgenre. These sets roughly correspond to our reading experiences but the larger size is naturally expected to contribute to a better detection of stylistic features.

For the first approximation, let us consider that stylistic features of a subgenre are based on the use of preferable expressions for the subgenre and the lack of undesirable expressions for it. The definition of 'expression' may vary, but it should be noted here that a large number of words, including misspelled ones.

A scatter diagram between these two subgenres is shown in Fig.1, where samples of atomic expressions are distributed on the corresponding two axis. The determination of the proper (range of) *n* is a heuristic issue, and we first adopt the range of *n* as two to four, mainly because most of the Japanese words consist of one or two characters. To conduct the first experiment, we used a dumped file of Japanese Wikipedia[2] and a sample of 2ch BBS (http://www.2ch.net/) as sample contrastive data set, the sizes of which are shown in Table 1.

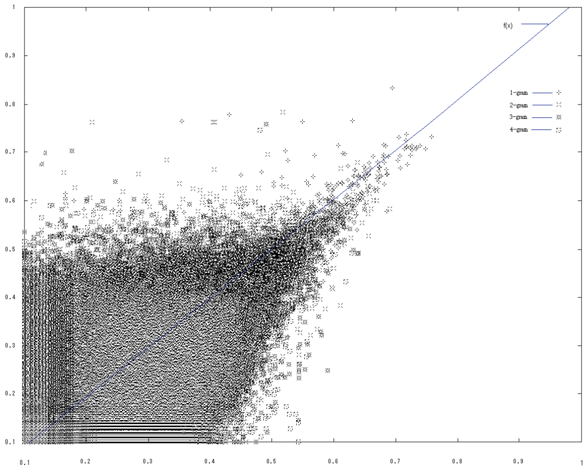


Fig.1. Scatter Diagram of 2- to 4-gram expression

Table1 Data SET

Data	Number of characters
Wikipedia	161,223,892
2ch	108,031,243

The frequency of each atomic expression in Wikipedia and 2ch classifies them roughly into three classes: (a) atomic expressions frequently used in Wikipedia but not frequently used in 2ch, (b) atomic expression frequently used in both data, and (c) atomic expressions not frequently used in Wikipedia but frequently used in 2ch. This classification means that the class (b) consists of rather neutral atomic expressions, whose use may well not characterize a text as either of the two, while the class (a) and (c) are to be considered to be stylistic features for Wikipedia and 2ch, respectively.

With these classes representing the resemblance of a given text *T* with the model set of texts of the targeted genre, we have been able to visualize the textual characteristics on this aspect[3] . Suppose that the function *freq(a, X)* is the frequency of occurrence of atomic expression *a* in the corpus *X*. For computing the stylistic similarity o f *a* to the style of Wikipedia, we compare *freq(a, Wikipedia)* and *freq(a, Y)*, in which *Y* is another corpus(for example, corpus of BBS articles).

This stylistic similarity of atomic expressions whose the size is three or four characters shown in Fig.1, is too short to represent the passage-level consistency. For the problem, we have to extend the function *freq (a, X)* to a function *freq(w, X)* for computing the appearances of desirable expressions for the targeted subgenre *X*, where w is a function that returns the set of expressions included from certain span in the text. With this function, we compute the style similarity of any span in the text to a certain subgenre.

III. Inconsistency Analysis Based on Templates

As we have summarized in Section II, there are expressions which are successfully used in identifying the genre of a text. Those *n*-grams are often parts of content words and perhaps they would be better to find frequently occurring words. Content words are defined as the words which are not functional expressions. In contrast with them, functional expressions are those that have little lexical meaning or have ambiguous meaning. For example, dictionaries always define the specific meanings of content words, but only describe the general usages of function words. Instead of the complexities to describe the meaning of function words, these words serve to express grammatical structure of a sentence or clause and specify the attitude or mood of the writer. In order to extend a simple consistency analysis based on content words to adopt more various degree of consistency, we adopt a method based on occurrence of function words.

One of the problems when we examine the occurrence of functional expressions is that the use of them strongly depends on the own style of the writers. If we adopt a corpus in which sentences are corrected from a free Internet forum to which a lot of writers can freely post their articles, the use of functional expressions must quite vary. In other words, it is difficult to collect a large corpus in which the use of functional expressions is expected to be consistent.

In order to overcome the problem of the shortage, we prepare templates that are sequence patterns of the functional expressions. We extract these templates from students' essays that we assigned them to write the essays referring to some Wikipedia articles. The purpose of this setting is that the students will use their familiar functional expressions even if they refer to the articles (Wikipedia articles) in which the consistency of the use of functional expressions is maintained.

The process to extract templates is conducted based on comparison between the sentences which the student writes and their corresponding sentences in the Wikipedia articles. The measures for the comparison is computed based on how many content words are common between the sentences. For example, each pair of sentences in (s1) and (s2) is compared because of more than half content words are commonly used between them.

(s1)(Wikipedia)テキストをデータ構造に変換する。
(Student)テキストにデータ構造は変換されます。
(Template) ○に ○は変換されます。

(s2)(Wikipedia)北米は家庭に約2台のパソコンがある。
(Student)北米は家庭の パソコンがあります。
(Template) ○は ○の ○があります。

As a result of comparing process between them, the words which are anything other than the functional expressions in (s1) and (s2) are replaced with meta word '○', and we obtain a ordered sequence of functional expressions like {'に','は変換されます'} and {'は','の','があります'}. In this paper, we refer to the sequences as "templates." Because each template mainly shows a kind of features of the syntactic structure of the sentence, the occurrences of infrequent content words in the sentence are ignored with the computations which adopt the templates.

By using the collection of these templates, we extend the function appropriately, which we have mentioned in Section II. For the purpose of evaluating the passage-level consistency, the first argument *w* in the function *freq(w,X)* to be extended. In our proposal, our algorithm to evaluate the consistencies from the viewpoint of the passage-level processes sentence by sentence. This evaluation is conducted by the algorithm shown in Fig.2.

Regarding a target text as the set of sentences like {s1, s2, ..., sn}, each sentence is evaluated whether it holds the expected style by applying the function *freq (w, X)*, where *X* is the corpus in which texts are assumed to be written in the expected style. Note that the algorithm needs one or more corpora to compare the style similarities of the target sentences to the various styles. As outputs of the algorithm, either of three characters {'+', '*', '-'} is to output for each unit by the *print* command. This output does not need to be displayed directly, it means that we have to devise an effective way to promote users' awareness of the inconsistency in the text.

```
{s1,s2,...,sn}.each do |w|
   y = freq(x, Wikipeida) - freq(w, 2ch)
   if y > 0 then
          print "+"
   else if y < 0 then
          print "-"
   else
          print "*"
   endif
end
```

Fig.2. Algorithm for Evaluating Passage-Level Consistency

IV. Proposal for Interactive Editing

Systems for interactive text editing are designed to allow one to use pre- and post-processing of subparts of the text when he/she is imputing or editing it. A system we propose here has the following features:

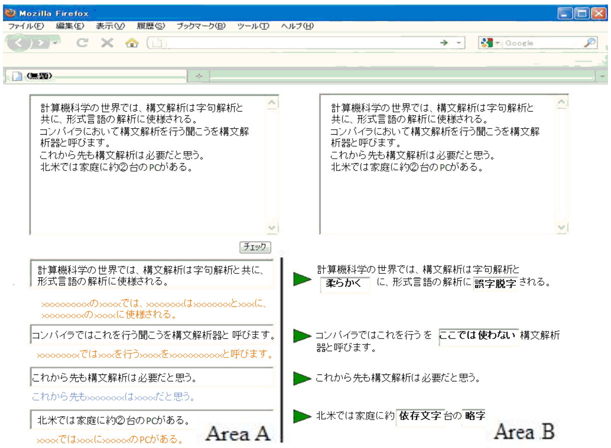
- detecting the errors in the inputted phrases and suggest the exact parts to correct
- visualizing the multi-level consistencies (from intra-sentential and passage-level viewpoints)

We designed our system to be run on the server computer and to be used through web browsers. In the internet, some of the useful modules which analyze some errors of intra-sentential inconsistencies are available for such kind of text processing services which are used from Web browsers. The algorithm that we described in Section III is implemented as a module like them and our system is developed combining these various modules.

Interactive input systems are familiar with Japanese people because these systems have been developed to help to input Japanese characters. Intelligent interactive text editing is also investigated as applications of the machine translation. For example, some works focus on interactive text pre-editing to improve the output quality of machine translation[3][4], in which the user can input or edit the source-language sentence interactively referring to the obtained sentence by translating it. Like the works, in which the users edit his/her writing text according to the messages from the system, our system provides an interface that allows the user to check his/her writing text from two viewpoints (passage-level consistency and intra-sentence consistency).

Fig.3 shows the screenshot of our interface, in which there are two big input boxes on the upper side of the page and under these boxes there are two areas for interactive editing to the inputted text.

An original text is inputted in the left hand side of the big input boxes and then the user pushes the check button to edit the inputted text. The box on the right hand side displays the log for the editing history.


Mozilla Firefox
ファイル(F) 編集(E) 表示(V) 履歴(S) ブックマーク(B) ツール(T) ヘルプ(H)
Google
(無題)
チェック
Area A
Area B

計算機科学の世界では、構文解析は字句解析と共に、形式言語の解析に使様される。
コンパイラにおいて構文解析を行う聞こうを構文解析器と呼びます。
これから先も構文解析は必要だと思う。
北米では家庭に約②台のPCがある。

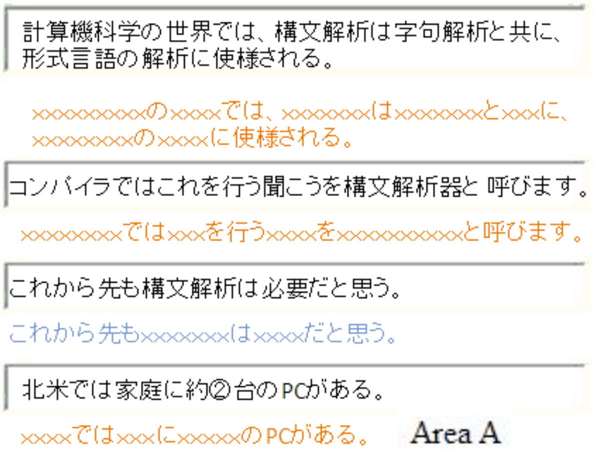
計算機科学の世界では、構文解析は字句解析と共に、形式言語の解析に使様される。
コンパイラにおいて構文解析を行う聞こうを構文解析器と呼びます。
これから先も構文解析は必要だと思う。
北米では家庭に約②台のPCがある。

計算機科学の世界では、構文解析は字句解析と共に、形式言語の解析に使様される。
xxxxxxxxxのxxxxでは、xxxxxxxはxxxxxxxとxxxに、xxxxxxxxのxxxxに使様される。
コンパイラではこれを行う聞こうを構文解析器と 呼びます。
xxxxxxxxではxxxを行うxxxxをxxxxxxxxxxと呼びます。
これから先も構文解析は必要だと思う。
これから先もxxxxxxxはxxxxだと思う。
北米では家庭に約②台のPCがある。
xxxxではxxxにxxxxxのPCがある。

計算機科学の世界では、構文解析は字句解析と 柔らかく に、形式言語の解析に 誤字脱字 される。
コンパイラではこれを行う を ここでは使わない 構文解析器と呼びます。
これから先も構文解析は必要だと思う。
北米では家庭に約 依存文字 台の 略字

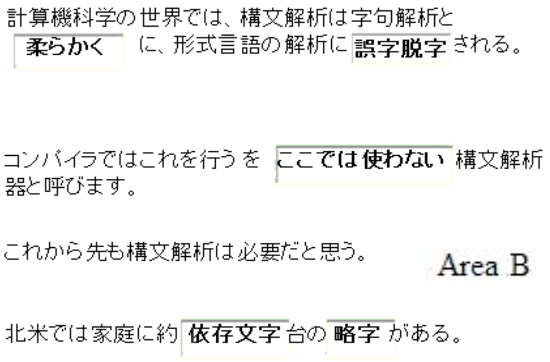
Fig.3. Screenshot of Our System

The interactive edit area is divided into the two areas as shown in Fig. 4 and Fig. 5.



計算機科学の世界では、構文解析は字句解析と共に、形式言語の解析に使様される。
xxxxxxxxxのxxxxでは、xxxxxxxはxxxxxxxとxxxに、xxxxxxxxのxxxxに使様される。
コンパイラではこれを行う聞こうを構文解析器と 呼びます。
xxxxxxxxではxxxを行うxxxxをxxxxxxxxxxと呼びます。
これから先も構文解析は必要だと思う。
これから先もxxxxxxxはxxxxだと思う。
北米では家庭に約②台のPCがある。
xxxxではxxxにxxxxxのPCがある。

Fig.4.Area A



計算機科学の世界では、構文解析は字句解析と 柔らかく に、形式言語の解析に 誤字脱字 される。
コンパイラではこれを行う を ここでは使わない 構文解析器と呼びます。
これから先も構文解析は必要だと思う。
北米では家庭に約 依存文字 台の 略字 がある。

Fig.5.Area B

We designed that each sentence is displayed with the information of its style in the edit area A. Under each sentence box, the template which the sentence contains is colored with the color which represents the style similarity for user to check the consistency from the passage-level viewpoint. The computation of the style similarity follows the algorithm described in Section III.

On the other hand, the user can concentrates on correcting the intra-sentential errors in the edit area B. In the edit area, some parts of the sentences are replaced to the input boxes, to which the system promote the users' awareness of errors and mistakes that are detected from intra-sentential viewpoint to correct them. For each box, the reason why the corresponding part should be correct is also informed.

In this way, the user can be aware of the changes of the consistencies in the passage-level as soon as he/she revises the errors in area B. On the other hand, he/she rewrites a whole sentence in the edit area A to hold the passage-level consistency of the context, the system interactively evaluate the intra-sentential consistencies of the corresponding parts.

V. CONCLUSION

In this paper, we proposed a system for interactive text editing that allows one to check the multi-level consistencies in his/her writing text. In the system, we used the sequences of functional expressions to compute one of the consistencies from the passage-level viewpoint. In order to promote awareness on the multi-level consistencies of the writing text, we combined it with some algorithms that previous works proposed. Much has to be done towards a practically effective tool, but any advanced tool to help to detect undesirable expression should be conscious of stylistic differences among subgenres and our proposed methods are fundamentally effective.

ACKNOWLEDGMENT

This work was partly supported by KAKENHI (21500113).

REFERENCES

[1] Schiffrin, D., D. Tannen, and H.E. Hamilton (2001) *The Handbook of Discourse Analysis*, Blackwell
[2] Japanese Wikipedia dumped file. http://download.wikimedia.org/jawiki/20071013/jawiki-20071013-pages-articles.xml.bz2
[3] K. Hashimoto, K. Takeuchi, H. Ando (2010) A Corpora-based detection of stylistic inconsistencies of text in the targeted subgenre, AROB15, pp.110-113
[4] K. Uchimoto, N. Hayashida, T. Ishida and H. Isahara (2005) Automatic Rating of Machine Translatability, In Proceedings of 10th Machine Translation Summit, pp. 235-242
[5] N. Hayashida, T. Ishida (2005) Performance Prediction of Supporting Self-Initiated Repair by Translation Agents (in Japanese)
The transactions of the IEICE. D-I J88-D-I(9),pp.1459-1466