T. Hirashima et al. (Eds.) (2011). Proceedings of the 19th International Conference on Computers in Education. Chiang Mai, Thailand: Asia-Pacific Society for Computers in Education

Development and Improvement of a Corpus-based Web Application to Support Writing Technical Documents in English

Yoshinori MIYAZAKI[a*], Shosaku TANAKA[b] & Yukie KOYAMA[c]
[a]*Faculty of Informatics, Shizuoka University, Japan*
[b]*College of Letters, Ritsumeikan University, Japan*
[c]*Center for Research and Development in Higher Engineering-Education, Nagoya Institute of Technology, Japan*
*yoshi@inf.shizuoka.ac.jp

Abstract: This paper discusses the development of a Web application that supports non-native speakers of English in writing technical academic documents in English. The application presents authentic English sentences from technical corpora on the basis of calculated similarities to the English input. The updates of this study are 1) highlighting relevant words/phrases of the output sentences and 2) generalizing structures of long output sentences to show the structure more explicitly. Results of experiments are also given to show possible usages and some shortcomings of the application.

Keywords: Technical corpus, English composition, *n*-gram model, MWE, generalization

Introduction

Because English is the most common language in science and technology, researchers including undergraduate or graduate students are often required to write technical documents in English. However, writing technical academic documents in English is quite challenging for non-native speakers (NNSs) especially in EFL settings[1], since technical documents usually contain specific words and phrases that we seldom use in general English. In other words, technical documents include certain templates or fixed patterns which appear quite often in the papers of a specific research field, but it is considerably difficult for NNSs to use these multi-word expressions naturally. In this light, the burdens of learners of English will be lessened if the following process is automated[2]: (a) retrieving English sentences from the authorized technical documents that are similar to the input sentence in English, and assuming the inappropriate wordings by NNSs, (b)[3] retrieving again with the keywords replaced by their synonyms. In addition, as stated above, technical documents have domain-specific conventional phrases and collocations. The application would be more useful if it had the further functions of (c) considering domain-specific expressions of the relevant field, and (d) summarizing long output sentences in order to clarify their significant parts.

[1] Evans, S. & Green, C. (2007). Why EAP is necessary: A survey of Hong Kong tertiary students. *Journal of English for Academic Purposes, 6*(1), 3–17.
[2]The basic concept of this system is to implement such a retrieving process.
[3]The actual cases involve not only word choices but their structures.

Our research group has worked on the development of software to help compose academic documents in English. In [1], as an application of this system, a Web application that has basic functions (a)-(c) was developed to help write technical documents using technical corpora[4]. In order to improve this application, this study reports

- (d) the function of generalization, which intensifies the structure of long sentences,
- the updated word replacement list and algorithm for the computation of similarities, and
- a highlighting function added for the output interface for better usability.

1. Literature Review

Thus far, several studies have attempted to develop systems to help users compose English [2][3][4][5][6]. [2][3] allowed users to input Japanese sentences to obtain sample-based English sentences similar to the corresponding English translations. In [2], a key notion was generated from a group of words in the Japanese sentence input, followed by the translation of the notion into English using a Japanese-English correspondence table. [3] first translated the Japanese input into English using commercial software. The drawback of this system was that the selection of keywords was limited to verbs and nouns only. In contrast, our study requires users to input English sentences, so that they can practice their English composition skills and users are not limited to speakers of Japanese only. [4][5] retrieved sentences from corpora that were similar to the English sentence input based on the phrase structure or dependency of the input. Although they are very promising work, the potential areas of application are limited at present. [6] prompted users to issue queries with wildcards (for the unclear parts), and directed these queries to Google engine in order to obtain matching expressions that appear frequently on the Web. However, the system was greatly affected by Google's limitations and many non-academic sentences appeared in the outputs. [7], which discusses user-friendly interface, is also of importance, since a sample-based Web application outputs a large volume of sentences for reference.

2. Proposed Algorithm

2.1 Outline

Fig. 1 shows the schema of the algorithm, with the sample input "Selected composition was excellent way to solve issue at this time". As one can see, the expression includes grammatical mistakes as well as inappropriate word choices (and this sentence was actually composed by a learner). See sections 2.1-2.5 for the definitions of symbols in the figure.
The proposed algorithm consists of three steps, for the retrieval of similar sentences:
Step 1. [Replacement by synonym groups] Applies the morphological analysis to the input sentence and extracts keywords from the result. If the obtained keyword is in some synonym group(s), replace the keyword by the synonym group ID(s) (e.g., *G1*, *G2*, …, in the figure). This step corresponds to (b) mentioned above.
Step 2. [Approximation of contextual-information] Computes *n*-gram components and applies [9][10] for multi-word expressions (MWEs). The use of *n*-grams and MWEs enables the application to compute the similarities between sentences based on approximate contextual-information in the sentence (c).
Step 3. [Extraction of similar sentences] Generates vectors from the input sentence and each of the corpus sentences, and quantifies the similarities (a). The technical corpora used in this

[4]Let us refer to corpora that collect technical documents as this in the sequel.

study were developed by Nagoya Institute of Technology (NIT)[5]. The extracted sentences appear with the option for its generalization (d).

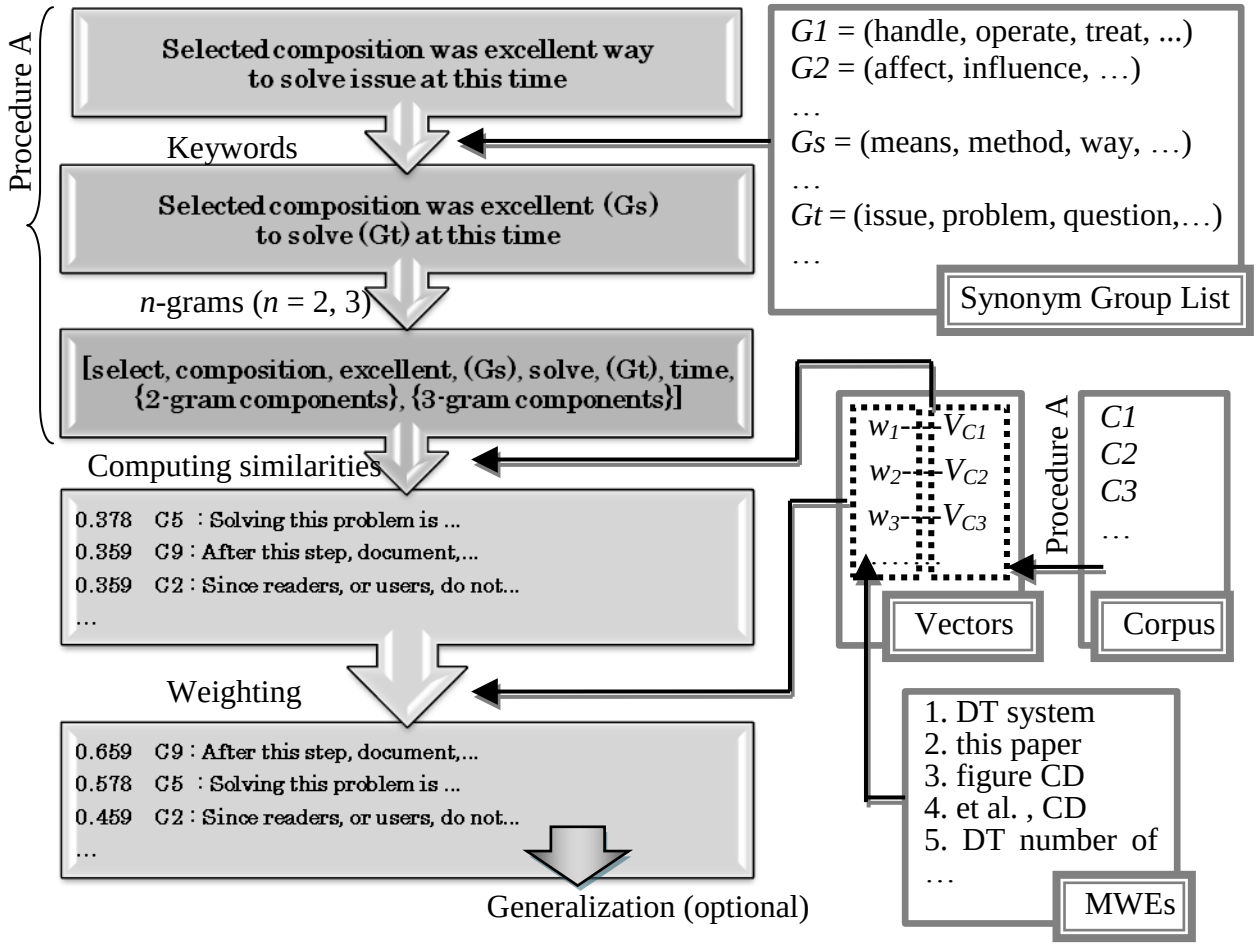

Fig. 1: The schema of the proposed algorithm

2.2 Selection of Keywords

First, the morphological analysis using Tree Tagger[6] is applied to an input sentence in order to obtain the part of speech and the lemma of each word. For the purposes of this study, we defined a keyword as a word whose part of speech is either a noun, adjective, adverb, or a verb. A word judged as either a preposition, determiner, or a cardinal number, retains the information of its part of speech (see 2.4). The delexical verbs are eliminated because these words are used quite often with different meanings and hence they might negatively affect the sentence identification process[7].

2.3 Keyword Replacement by a Synonym Group

In composing technical documents, learners of English might end up using inappropriate words, phrases or collocations. Synonym groups are capable of working well even in such situations. Synonym groups are used as a union set of those collected from a book and a dictionary. [7] introduces synonyms essential to writing technical documents, and we

[5] NIT Concordancer, http://lang.cc.nitech.ac.jp/conc/conc.pl
[6] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger
[7] We tentatively selected the 7 words as delexical verbs:"be", "do", "give", "take", "have", "make", and "get".

selected 102 groups. The number is small whereas the reliability is high, since the book is intended for helping to write technical documents. To reinforce the small number, the adoption of WordNet[8] was attempted. The synonyms in WordNet are much larger in number, but with the treatment of various meanings. Therefore, the straightforward use of the synonyms might produce excessive or redundant results. To prevent this, we regarded a word as an input suitable for reference to WordNet only when the word was considered specific to technical documents. The Chi-square test for homogeneity was carried out after extracting words from a technical corpus and English textbooks in (junior)-high schools[9]. The resulting words which were significant at 1% level were intended to be input words to WordNet (1,957 words in the case of ACL[10], the technical corpus compiled by NIT). The number of the generated groups was 2,943.

2.4 N-gram and Computation of Similarities

An *n*-gram is a subsequence of *n* items from a given sequence. For example, a set of 2-grams of "Corpora consist of documents." are "corpora consist", "consist of", and "of documents". In this study, these are generated after a synonym group (ID) or information of part of speech replaces the original sentence.

The degree of similarity is given by cosine similarity (vector space model) and expressed as a value between 0 and 1. First a vector v is composed. As a component of v, each of the keywords and *n*-grams (*n* = 2, and 3) is respectively added, while these keywords and *n*-grams are both obtained from the input sentence and corpus sentence, in the process of Procedure A in Fig. 1. If the same component is already in v, the component is skipped. Next, let vectors v_L be defined as having 1 (0) in the *i*-th element if the *i*-th component of v is (not) found in the input sentence (*i* = 1, 2, …, the last component of v). Likewise, each vector of v_C for sentences in the corpus (*C* = *C1*, *C2*, …) is determined. Now the cosine similarity is computed as $\cos(v_L, v_C) = (v_L, v_C)/(\|v_L\| \cdot \|v_C\|)$, where $(\cdot,\cdot)$ denotes the inner product of the two vectors, and $\|\cdot\|$ the length of the vector (2-norm).

2.5 Weighting

2.5.1 Weighting by Domain-Specific MWEs

A technical corpus does not always include sentences that contain typical technical expressions (e.g., introductory paragraphs). This shows that weightings should be effective by the degrees of the sentences having likelihood of being technical. Tanaka&Koyama have successfully generated a list of domain-specific MWEs in technical corpora [9][10], and obtained 2,677 MWEs in ACL corpus. Some examples are:

- The subjects : "DT system", "this paper", "figure CD", "et al., CD", "DT number of"
- The predicates : "However", "for example", "case", "therefore", "finally"

where "DT" stands for a DeTerminer, and "CD" a CarDinal number. The authors provisionally set a weight on C_i to

[*some constant*] × [*number of MWEs in* C_i] / ([*number of keywords in* C_i] + 1).

[8] A large-scale English conceptual dictionary, http://wordnet.princeton.edu/

[9] English textbooks in high and junior-high schools in Japan used 2002 to 2005 academic year, which was compiled by ELPA (Association for English Language Proficiency Assessment).

[10] The Association for Computational Linguistics, http://www.aclweb.org/

2.5.2 Weighting by Vector Components

It is not unusual that a word belongs to more than one synonym group. It turns out that some of them are the members of more than 10 groups. In such cases, distributing the weightings (such that they sum up to 1) is probably a reasonable solution. In 2.4, 0 or 1 was given as each of the vector components. In contrast, *1/k* is allocated to the word belonging to *k* synonym groups. When more than one word in the *n*-grams belong to synonym groups, $1 \Big/ \left(\sqrt[m]{\prod_{w \in S} k(w)} \right)$ was weighted, where *S* is the set of such words, *m* is the number of elements of *S*, and *k(w)* is the number of synonyms of the group the word *w* belongs to.

2.6 Generalization

What are extracted from corpora have authenticity since they are actually used and refereed by journal referees. At the same time, they are sometimes too complicated to learn from due to their authenticity. Generalization is the procedure to summarize such sentences and show the general form of each authentic sentence by replacing specific words with their grammatical markers[11] such as part-of-speeches. This procedure puts frequently observed subsequences into chunks and enables us to show structures of long sentences by repetition (examples are shown in the next section). The basic idea of generalization is as follows: (1) high-frequent words in a set of sentences should be retained, and (2) words occurring in various sentences should be also retained even though the frequency is not so high. On the other hand, (3) low-frequency words should be replaced to grammatical markers, and (4) words, whose frequencies probably are low, occurring in specific sentences should be replaced. Based on the idea, words to be replaced are selected by the following score[12]: itf-isf(w) $= \frac{1}{tf(w)} \cdot \log \frac{|S|}{sf(w)}$, where *S*, *tf*($w$) and *sf*($w$) are a set of sentences shown in the system, the term- and sentence-frequency of the word *w* in *S*, respectively. The generalizing procedure repeatedly replaces words in each sentence in *S* with grammatical markers, in descending order of itf-isf(w). One of the main problems of this procedure is to determine the proper degree of replacement. This system regards an occurrence of sentences as a probabilistic phenomenon, and adopts a class-based *n*-gram model as the statistical language model. As a result, the procedure can be implemented as model selection of information criterion. For more details on this formalization, see [11].

3. Implementation

This application was implemented using PHP, JavaScript, and Perl (as CGI), with the intention of developing it as a Web application. MySQL was used as the database system. An initial screen offers options for various needs of users. They are:

- Target Corpus : Choose a technical corpus from ACL, Nature, Scientific American, and Biology-related Journal, each compiled by NIT.
- Weighting : Weighting algorithm in 2.5.2 is applied when “yes” is selected.
- Retrieval Mode : When “speedy retrieval” is on, only the top 5,000 ranked corpus

[11] Grammatical markers are symbols on the grammatical information such as part-of-speeches, syntactic categories (e.g., *N and &CD; described below are a category of noun phrase and a number, respectively).
[12] “itf-isf” is the abbreviation for “inverse term frequency and inverse sentence frequency.”

sentences are the targets for quick retrieval[13].

Fig. 2 is a screenshot of the output screen showing the result of computations. The corpus sentences with high similarities are displayed in a descending order. The number above each sentence is its cosine similarity. The highlighting function enables us to identify the sentences having corresponding expressions to the input sentence. When "highlight all the sentences" button is on, all the words are highlighted in the same color as that of the corresponding word in the input sentence, if they are in the same synonym group. If the input sentence has more than one such word, each of them is highlighted in different colors. Pushing "highlight this sentence" colors corresponding *n*-gram components of the corpus sentence as well as the input sentence in the order of n = 1, 2, and 3.

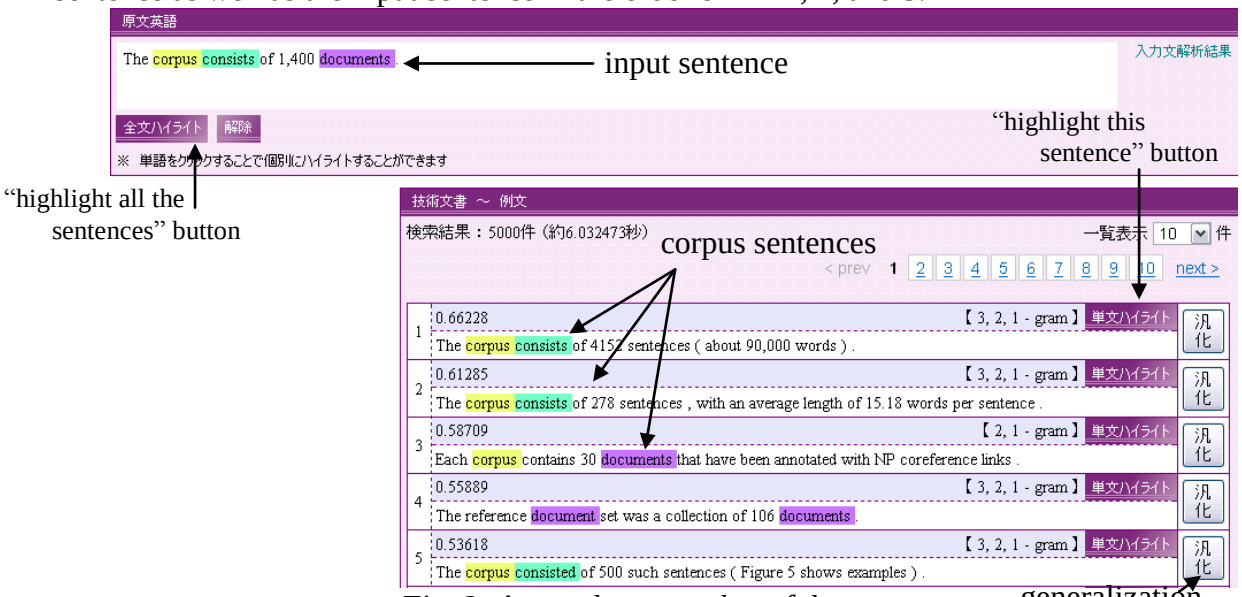

Fig. 2: A sample screenshot of the output

Displaying a generalized corpus sentence is optional. If users want to change the degree of generalization, they are asked to specify the number of generalizations. Fig. 3 shows an example. After the generalization 20 times, the input sentence "The corpus consists of 278 sentences, with an average length of 15.18 words per sentence." is simplified as "The corpus consists of &CD; sentence, with an *N of &CD; *N per sentence.", clarifying the structure of the sentence (where "*N" is for Noun phrase, and "&CD;" for CarDinal number).

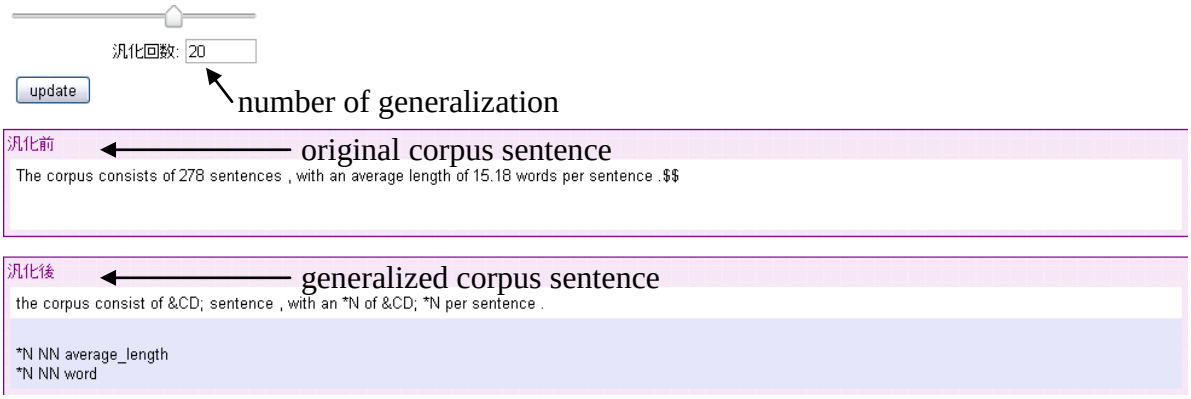

Fig. 3: A sample screenshot for before/after generalization

4. Experiment and Discussion

[13] In terms of the high accordance of the keywords (or synonym group IDs) with that of an input sentence.

Experiments were conducted to assess the validity of the implemented Web application (options: ACL Corpus, with weighting on, with speedy retrieval off). Due to the limitation of the space, only two examples are shown. The first example is when the following input was made: "The result is showed by Table 1". Table 1 shows the output ranked top 10 by the application. The word in bold represents the existence of a corresponding word (belonging to the same synonym group), and the same type of lines are drawn under the corresponding words. The cosine similarities of the sentences are omitted.

Table 1: Result of Experiment (Example 1)

Rank	Top 10 Output Sentences (Input: The **result** is **showed** by **Table** 1)
1	The categorization **results** are **demonstrated** in **table** 5
2	The **result** of projecting is as **table** 4 below
3	The **results** are summarized in **table** 2 below
4	The **results** are **shown** in the following **table**
5	The **results** we obtained are summarized in **table** 1
6	In **table** 2 , we illustrate our experimental **results**
7	The **results** are **presented** in the following section
8	While the examples in **table** 2 have a low mutual information
9	The **results** are **shown** in the **table** below
10	In **table** 1 , we **present** the **results** obtalned for these 52 sentences

From the results, learners can become aware that the expression "be showed in (table)" is more appropriate than "be showed by (table)". What is more, it was made clear that there are few examples using "showed", as a past participle of "show", although it is grammatically correct. By scanning output examples, learners are also given opportunities to learn similar expressions. However, the corpus comes from journals in which real sentences are used. It is admitted that not all the mistakes such as typos can be prevented even though they are refereed. One example is the 10th sentence. The "obtalned" should obviously be "obtained".

Table 2: Result of Experiment (Example 2)

Rank	Top 10 Output Sentences (Input: We **view** **several** **ways** to **solve** this **problem**)
1	We **view** **several** **ways** to **solve** this **problem**
2	We **considered** using the example-based **method** to **solve** this **problem**
3	There are **several** **methods** to estimate model parameters
4	We address this **problem** in **several** **ways**
5	**Several** **methods** have been proposed with regard to this **problem**
6	There are **several** **ways** in which external processes can be used
7	**Several** special cases arose
8	There are **various** **ways** to determine the filtering range
9	**Several** variants on this base scheme can be thought of
10	To overcome these kinds of **problems** , **several** **methods** to **resolve** zero pronouns which **consider** appl/ cations for a practical machine translation system with an n~]imJted translation target area , have been proposed

The second example is the case when one of the corpus sentences was input. As was expected, the input sentence appeared on top of the list, and its similarity was 1. Again, several paraphrasable expressions as well as collocations such as "solve … problem" were observed. The 10th is a good example of a long sentence. The sentence structure is difficult to comprehend although several words match the input sentence with the implication of

high similarity. Then, the generalization was applied to the sentence to make it simple.
The result: To *V these *N of problem, several method to *V &CD; *N which *V *N for a *N with an *N, have be *V. ("*V" is for Verbal phrase) [number of generalization :17]
Thankfully, a long noun phrase is made compact to be "*N". It is expected that the understandings of sentence structures are deepened, comparing with its original corpus output sentence. Further data and discussion will be shown in the succeeding publication.

5. Conclusion and Future Research Plans

In this study, a Web-based application was developed to present sentences, which are similar to a user's input, from a technical corpus. This application is intended for NNSs to help them compose technical documents in English. The advantages of this application are (1) broadening the range of retrieval by synonym groups, (2) treating sentences based on approximate contextual-information in the sentence, and (3) generalizing long sentences to show their structures explicitly. In the experiments, it was confirmed that even grammatically incorrect input sentences could present similar sentences from a technical corpus, and that long sentences from the corpus were made fairly compact, helping learners grasp the sentence structure easily.
In the future, we plan to increase technical corpora in terms of quantity as well as quality. Increasing technical corpora is also indispensable for devising a better algorithm to weight the words/phrases (in MWEs) with greater accuracy. Enabling faster computations is also an important related issue. Our final goal is to design this application so that it can be customized according to users' preferences or their log data. Therefore, a comprehensive update of the internal design of the application is essential. Lastly, we hope to establish the objective criteria to evaluate the precision of the resulting output list, because that will prove instrumental in accelerating the pace of research in this field.

References

[1] Miyazaki, Y., Ikemoto, T., & Tanaka, S. (2010). Development of Web application to help write technical documents in English: using corpus for language teaching. *Proc. of the ICTATLL2010* (pp. 149-158).
[2] Takeda, A., & Furugori, T. (1994). A sample-based system for helping Japanese write English sentences. *Journal of IPSJ, 35*(1), 53-61.
[3] Takakura, S., & Furugori, T. (2002). TransAid : a writer's aid system for translating Japanese into English. *Technical Report of IEICE (NLC2002-11), 102*(199), 7-14.
[4] Miyoshi, Y., et al. (2003). An illustrative-sentences search tool using phrase structure for English writing. *Journal of Japan Society for Educational Technology*, *27*(3), 283-294.
[5] Matsubara, S., Egawa, S., & Kato, Y. (2007). English sentence retrieval system: library service using article database. *INFOPRO-2007* (pp. 125-129).
[6] Oshika, H., Satou, M., Ando, S., & Yamana, H. (2004), A translation support system using search engines, *Institute of Electronics, Information, and Communication Engineers, DE2004-88*, 237-242.
[7] Yamanoue, T., et al. (2004). Learning usage of English KWICly with WebLEAP/DSR. *Proceedings of the 2nd International Conference on Information Technology and Applications*.
[8] Yokota, H., & Tsumatori, C. (2007). *Synonyms for Scientific Papers*, Beret Publishing.
[9] Tanaka, S., & Koyama, Y. (2010). Extracting domain-specific expressions from ESP corpora in view of syntactic structures. *The Institute of Statistical Mathematics Cooperative Research Report 239*, 13-30.
[10] Tanaka, S., & Koyama, Y. (2010). Towards extracting domain-specific expressions on a variety of linguistic levels based on a sentence classification model. *The Institute of Statistical Mathematics Cooperative Research Report 233*, 21-33.
[11] Tanaka, S., Miyazaki, Y., Ikemoto, T., & Koyama, Y. (2011). Towards generalization of set of sentences to support sentence composition in English using large-scale n-gram database, *The Institute of Statistical Mathematics Cooperative Research Report 254*, 1-19.