Web-Based Knowledge Database Construction Method for Supporting Design

Kiyotaka Takahashi[1], Aki Sugiyama[1], Yoshiki Shimomura[1], Takeshi Tateyama[1], Ryosuke Chiba[1], Masaharu Yoshioka[2], and Hideaki Takeda[3]

[1] Faculty of System Design, Graduate School of System Design, Tokyo Metropolitan University, 6-6 Asahigaoka, Hino-shi, Tokyo 191-0065, Japan
takahashi-kiyotaka@sd.tmu.ac.jp

[2] Graduate School of Information Science and Technology, Hokkaido University Kita 14 Nishi 9, Kita-ku, Sapporo, Hokkaido, 060-0814, Japan
yoshioka@ist.hokudai.ac.jp

[3] National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan
takeda@nii.ac.jp

Abstract. In recent years, comprehensive problem solving by artefact designers has been required as demands related to artefacts become greater and more complicated. In relation to this background, we have proposed Universal Abduction Studio (UAS), a computer environment that synthetically supports a creative design. However, in general, it is difficult for a designer to acquire multiple domain knowledge because this knowledge is accumulated manually. This paper proposes a Web-based knowledge database construction method for supporting design using UAS.

Keywords: Creative Design, Knowledge Extraction, Analogy, Ontology.

1 Introduction

In recent years, the number of problems related to artefact design has increased for many reasons: the problem of manufacturers' responsibility, worsening environmental problems, the magnification and complication of artefacts and the diversification of consumers' values. Because of these issues, designers' roles and responsibilities have increased. In order to overcome these problems, the authors have developed a computer environment called *Universal Abduction Studio (UAS)* [13] to support creative design. The authors account for new knowledge generation processes in creative design; UAS realizes step-by-step knowledge extension by integrating different domains' knowledge groups.

In the above-mentioned knowledge integration, referring to a large amount of knowledge is effective. Therefore, designers need to store a huge amount of knowledge groups in multiple domains in the UAS design knowledge database. However, in the current UAS system, designers need to extend the knowledge stored in UAS by hand. Therefore, it is hard for them to acquire knowledge groups from multiple different domains. Therefore, the current prototype of UAS has a problem in

T. Yamaguchi (Ed.): PAKM 2008, LNAI 5345, pp. 173–184, 2008.

that designers only store a limited amount of a domain's knowledge in the design knowledge database. In addition, though we previously proposed a method to convert design knowledge data described in natural language into data with a format that could be processed in UAS [17], its processing accuracy is not so high. Therefore, designers currently need to correct design knowledge data by hand before conversion.

In this paper, for automatic and efficient knowledge construction, we propose a Web-based database construction method. For this purpose, we first propose a method of acquiring knowledge on the Web; this identifies design knowledge automatically, and extracts sentences that contain useful design knowledge information. Second, we propose a method to delete information that is useless for designers from the design knowledge information database. Third, we propose a method to change the acquired design knowledge information into available knowledge expression data for UAS.

We introduce the concept of UAS and the existing knowledge database construction method for UAS in section 2. In section 3, we propose the Web-based design knowledge construction method. In section 4, we verify the effectiveness of the proposed method. In section 5, we discuss the results of this verification. In the final section, we conclude the paper.

2 Universal Abduction Studio

2.1 Concept of UAS

UAS is a design support system based on abduction. Abduction, as proposed by C.S. Peirce, is a logical process by which an axiom can be found from a theorem [9]. Fig. 1 shows the system architecture of UAS. A workspace, a knowledge base group and a knowledge integration module group compose UAS. Designers put the design problem for operating and integrating with domain knowledge on the workspace. The knowledge integration module consists of multiple abductive reasoning mechanisms,

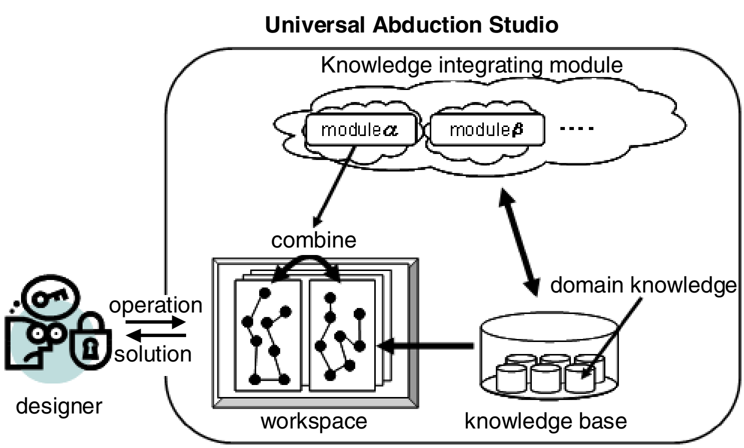

Fig. 1. System architecture of UAS [13]

and designers choose one or some of them depending on each design problem. The knowledge base consists of multiple domain knowledge bases that can potentially be integrated with the first domain's knowledge. The abductive reasoning system then performs knowledge integration. This fundamental concept requires a unified knowledge description among various domain knowledge bases.

2.2 Existing Knowledge Database Extension Method for UAS

In this section, we describe the existing knowledge database construction method for UAS.

1. Designers obtain design knowledge from paper texts that include the design knowledge and input design knowledge sentences to a text file line by line on a computer.
2. The text files are converted into a format that is processible in UAS.

In UAS, rule-type knowledge expressed in an If-Then form is used as a knowledge representation form (hereafter, we abbreviate this as If-Then-type design knowledge). We proposed an automatic document tagging method [17]. In this method, a computer automatically converts the design knowledge described by natural language into graph geometry as UAS knowledge.

The automatic document tagging system generates the design knowledge for UAS using the following procedures. The system:

1. Analyses a modification structure for the knowledge described by natural language using *Cabocha* [3], which is a parsing machine.
2. Defines a uniting relation between a predicate and an arbitrary case using the concept of *surface case* [14] with *GoiTaikei --- A Japanese Lexicon* [1].
3. Distinguishes a delimitation of the If-Part and Then-Part automatically, and divides them.
4. Outputs the result of procedure 3 (If-Then-type design knowledge) as an XML file.

The automatic document tagging system delimits as If-Parts and Then-Parts at the part including 10 patterns of words and Parts-of-Speech extracted from design documents [15] exist.

3 Proposed Method

3.1 Procedure of the Proposed Method

The information on the Web about multiple domain knowledge is updated hourly. Moreover, we can freely process that Web information using a computer, so we can extract information automatically. Therefore, the authors use Web information as the source in order to acquire multiple domain knowledge groups automatically and efficiently. Moreover, it is preferable that information stored in the design knowledge database is "New information that is highly reliable and useful for design objects". In this paper, the authors use press releases on the Web as the source for design knowledge, because press releases include design knowledge information such as the demand that a new product targets, the mechanism for achieving the function, and the function to satisfy the demand. Fig. 2 shows the construction method for the UAS design knowledge database. The details of the procedure are described as follows.

1. Acquire Web information including design knowledge using *Webstemmer* [7], a Web crawler for news sites, and store it in the Web information database.
2. Acquire design knowledge information from the Web information database using the method proposed in this paper.
3. Store the acquired design knowledge information in the design knowledge information database for UAS.
4. Convert the acquired design knowledge information into a text file with one sentence on each line using the proposed conversion system.
5. Convert the design knowledge information given as one sentence in each line into design knowledge for UAS using the automatic document tagging system, and store it in the design knowledge database for UAS.

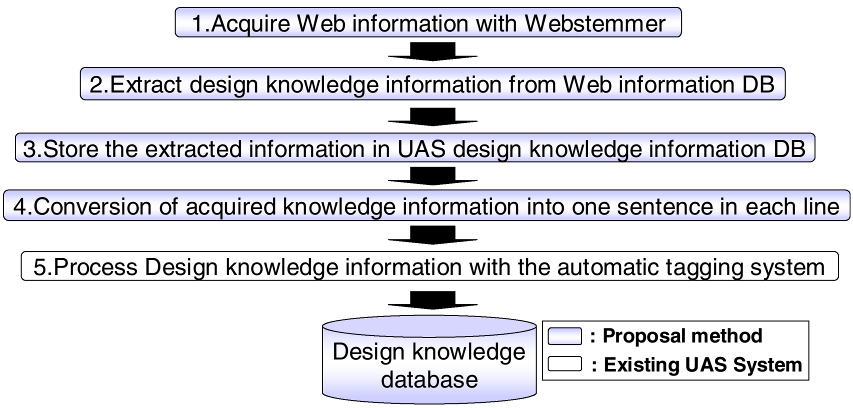

Fig. 2. Construction method for the UAS design knowledge database

3.2 Extracting Method of Design Knowledge Information from Web Information DB

In this section, the authors define the term design tag, introduced in this paper, explain how to create a design tag, and how to apply the design tag automatically to Web information. First, the authors define *design tag* as a tag for the efficient acquisition of useful design knowledge information from Web information. In this research, "The broader concept of the feature word of an article," is used as design tags, such as a physical phenomenon, an artificial material in GoiTaikei. The authors define a *feature word* is "A word existing with high frequency". In this research, a feature word is chosen by applying the *TF (Term Frequency) method* [6] to nouns, verbs, adjectives and adjectival verbs. We assume that a sentence that includes some design tags is a sentence that has useful design knowledge information. Therefore, we extract paragraphs that include design knowledge information sentences with design tags added. In addition, Web information includes information that is irrelevant to design such as sales information, etc. Therefore, we also use design tags as labels to distinguish relevant information. Hence, we create a design tag set by analysing press releases.

Next, we explain the *method for creating the design tag set*. First, we acquire multiple pieces of information from the Web and extract candidates for the design tag set from Internet. In this paper, we use press releases published in *Nikkei Net* [8] as an example of press releases about products. For example, we acquire articles on the four different domains of beverages, digital cameras, electric power equipments and Web services from Nikkei Net, one by one, to avoid bias in the knowledge domains. The design tag set is created using the following procedure. Fig. 3 shows an example of creating a design tag.

1. Execute a *morphological analysis* of sentences in the press releases and a *frequency analysis* for each morpheme.
2. Extract the concepts of the extracted feature words in procedure 1 based on GoiTaikei.
3. Abstract the extracted broader concept in procedure 2 to make it "a concept which does not depend on knowledge domains," and define the highest concepts as design tags.

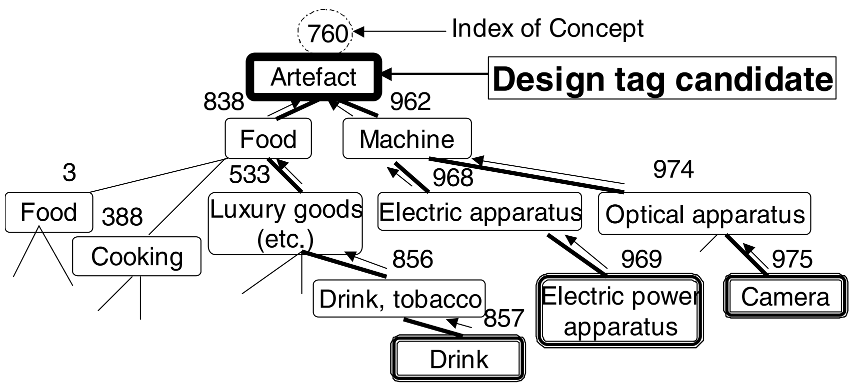

Fig. 3. Concrete example of the method for creating a design tag set

Procedure 3 makes it possible to acquire useful information for the design regardless of the domain of the press release. In addition, we applied procedure 3 to the concepts of all the listed feature words in procedure 1 and created multiple design tags.

For the frequency analysis for feature word extraction, we use a morpheme frequency analysis program produced by the authors. We adopted the name of a concept in GoiTaikei as the name for a design tag and we added an initial D, meaning "Design," to their heads. If a term had multiple broader concepts, we allocated the appropriate broader concept to the term by considering the contextual meaning of the term subjectively. Table 1 shows the design tag set created in this research.

Table 1. Design tag set created in this research

Japanese concept	Design tag
Organization	<D-ORGANIZATION>
Artefact	<D-ARTIFACT>
Physical phenomenon	<D-PHENOMENA>
Property	<D-PROPERTY>
States	<D-STATES>
Right	<D-RIGHT>
Aspect (etc.)	<D-ASPECT>
Possible	<D-CAN>

Next, we explain the *automatic tagging method for design tags*. First, designers execute a morphological analysis by using *Chasen* [16] for each article acquired by Webstemmer. Second, they match each morpheme in the article and each term that should have a design tag added among the terms in GoiTaikei and add design tags to the morphemes matched to them.

In this thesis, objects that should be tagged are all terms that belong to the subordinate concept of each tag. Because prototypes and inflections exist in Japanese verbs, we perform the matching by using prototype information for vocabularies outputted as morphological analysis results by Chasen. Finally, all the paragraphs, including the sentence tagged by the above-mentioned procedure, are extracted. Each task in the automatic tagging method for design tags is completed automatically by a computer (processing with the grep command of Linux or an existing substitute software program).

3.3 Filtering Method

In the information included in press releases related to new products' release information, most of the sales information and corporate information is less likely to give designers new insights for design. Therefore, if we process the press releases that include this information using the proposed method, a large amount of useless information will be included in the design knowledge database.

As a result, computational complexity and processing time for the inference of UAS will increase pointlessly. Consequently, we propose the following *filtering method* to reduce information that is useless for design in the design knowledge information database. We assume that the following six *named entity tags* defined by *IREX* [12], <ORGANIZATION>, <PERSON>, <LOCATION>, <DATE>, <TIME>, and <MONEY> are added to sales information that seems not to be useful for the design (Refer to Table 2). First of all, designers create articles that include the named entity tag IREX by morphological analysis with Chasen. Next, they reduce the paragraphs including those tags. These tags are called "*Reduce tag*". This procedure is implemented using Cabocha because Chasen and the named entity analysis program are built into Cabocha.

Table 2. The named entity tag set [12]

Object	Start position tag	Finish position tag
Organization-name, Government organization-name	<ORGANIZATION>	</ORGANIZATION>
Person-name	<PERSON>	</PERSON>
Location-name	<LOCATION>	</LOCATION>
Artefact-name	<ARTIFACT>	</ARTIFACT>
Date-phrase	<DATE>	</DATE>
Time-phrase	<TIME>	</TIME>
Money-phrase	<MONEY>	</MONEY>
Percent-phrase	<PERCENT>	</PERCENT>

3.4 Automatic Conversion Method for Knowledge Representation

The following two kinds of problems exist in the automatic document tagging system.

- Erroneous handling of sentences that include two or more design knowledge domains
- Low processing accuracy. In particular, this system cannot correctly process a lot of If-Then type design knowledge data

The former is a problem such that sentences related to two or more design knowledge are not processed correctly. This is because we have developed the automatic document tagging system based on the assumption that designers usually describe only one piece of design knowledge in each line of the input text file. Therefore, when the system processes design knowledge information acquired through the above-mentioned proposed method, the possibility of generating a large amount of incorrect design knowledge is high. As a result, an increase of useless information in the design knowledge database for UAS and computational complexity in the inference of UAS will arise. Therefore, we need to convert design knowledge information into a text file in which one piece of design knowledge is described in each line. The latter is a problem given the low accuracy of the current automatic document tagging system. According to our experiments, when the automatic document tagging system processed 82 kinds of design knowledge information sentences, it could convert less than 10% of the design knowledge information into design knowledge for UAS correctly. Therefore, it is necessary to improve the accuracy of the automatic document tagging system.

Therefore, we propose following two methods in this research as *automatic conversion methods for knowledge representation*.

- A converting method to include one design knowledge sentence in each line
- An improved method of the automatic document tagging system

The Converting Method to Include One Design Knowledge Sentence in Each Line. The automatic tagging method for design tags acquires paragraph including design knowledge information. However, we think it is often case that the causal relationship between the If-Part and Then-Part is described in one sentence. Therefore, we think that it is appropriate to divide design knowledge information into one sentence for each line. Hence, in this research, we propose a system that can place a period "." to punctuate the sentence and divide design knowledge information into multiple sentences automatically with the regular expression. This system is implemented in Ruby.

The Improved Method of the Automatic Document Tagging System. It is necessary to clarify the method of distinguishing between the If-Part and Then-Part in order to improve the accuracy of delimiting the If-Part and Then-Part. We describe related work on this measure as follows. Inui et al. analyzed newspapers, and categorised words that showed a causal relation as *causal markers*. In addition, they proposed a method for acquiring cause and effect knowledge including "*tame* (one of the *conjunctive particles* indicating causality)"[2]. On the other hand, Sakaji [10] researched a method of acquiring knowledge about causal relations using causal markers for all cause and effect knowledge. Based on the above-mentioned related works, we propose effective causal markers as the measure for distinguishing between the If-Part and Then-Part based on the result of analysing design knowledge information in press releases. To be specific, we acquired design knowledge from 50 press releases, converted them into the If-Then forms by hand. Herewith, we enumerated the description seen in the If-Then delimitation. As a result, we identified the following two heuristics for the If-Then delimitation.

- Seen in [2][10], "conjunctive particle" indicates causality often exist.
- "*infinitive form of the verb*" indicates causality with comma often exist.

Based on the above heuristics, we use the causal markers in Table 3. The method of distinguishing a concrete delimitation between the If-Part and Then-Part is described as follows. This system is implemented in Ruby.

1. Match the causal markers and each morpheme using the morphological analysis machine Mecab [4], working in the automatic document tagging system. The output data describe the morpheme and Part-of-Speech information.
2. If the causal markers correspond to an arbitrary morpheme, the system divides the sentence into the If-Part and Then-Part before and behind the morpheme.

Table 3. Causal markers in this paper

	Term "The romanized Japanese word":(the anglicized Japanese word)	Kind of term
Item 1	"yori, ":(by,) "de, ":(by,) "niyoru":(by) "niyotte":(by) "noga":(by) "ba,":(if…,) "baai":(when) "node":(because) "houga":(with) "kotoha":(that is)	Word
	"toki":(when) "tame":(because)	Reading
Item 2	"soubi si,":(is installed, so) "jitugen si,":(is achieved, so), etc.	Infinitive form of verb

4 Verification

4.1 Verification of Automatic Tagging Method of Design Tags and Filtering Method

In our experiment, we verify the proposed method by applying it to 20 press release articles acquired by Webstemmer. First, we mark the paragraphs including useful sentences for design in each acquired press release. In this experiment, we use press releases for a time clock, a wool coat, a game controller, etc. We assume these marked paragraphs to be the correct answers. We apply the proposed method to the articles and examine whether the method can output the correct answers.

The method is evaluated according to *F-measure* [11], which gives a value in assessing both *accuracy* and *recall ratio*. The accuracy evaluates how much unnecessary information a system can remove. The recall ratio evaluates how much necessary information a system can acquire without omitting necessary information. F-measure is a value that shows the effectiveness of an information extraction system. The value ranges from 0 to 1. We aim to maximize the F-measure in this paper. A computer executes the processes (tagging and filtering) automatically.

Next, we give the experimental conditions and results, as well as our discussion. In this research, we assume that several kinds of design tags are put to use in finding information for design. Because of this assumption, if the number of kinds of design tags in a sentence is greater than a threshold value, we regard the paragraph which includes such a sentence as useful information for design. We define the threshold value as the *design tag acquisition threshold*. We classify the *design tag acquisition threshold* into four stages, as shown in experiment 1-8 in Table 4. Moreover, we verify the automatic tagging method for design tags with and without the filtering method.

Table 4. Results of the verification of the automatic tagging method for design tags and the filtering system

	Design tag's acquisition threshold	Filtering	Average accuracy	Average recall ratio	F-measure
Experiment 1	One kind of design tag	Non-executed	9.19%	97.14%	0.17
Experiment 2		Executed	9.25%	60.00%	0.16
Experiment 3	Two kinds of design tag	Non-executed	14.95%	91.43%	0.26
Experiment 4		Executed	19.00%	54.29%	0.28
Experiment 5	Three kinds of design tag	Non-executed	22.83%	82.86%	0.36
Experiment 6		Executed	33.33%	48.57%	0.40
Experiment 7	Four kinds of design tag	Non-executed	27.06%	65.71%	0.38
Experiment 8		Executed	39.39%	37.14%	0.38
Further experiment Experiment 9	Three kinds of design tag	Executed to only the sentences put three kinds of design tag	27.72%	80.00%	0.41

When we compare the F-measure of each experiment without the filtering method, we confirm that the F-measure improves proportionally as the number of kinds of design tag increases. Therefore, the automatic tagging method for design tags can result in the efficient acquisition of useful information for designs by setting the design tag's threshold to be high. Therefore, we confirm that the method tends to tag multiple design tags as representing useful information for design. The experimental results show the effectiveness of the information extraction method based on the linguistic features of design knowledge information. As a result of applying filtering system to automatically tagged sentences, the accuracy and the F-measure improve the design tag's threshold in most cases. We can confirm that applying the filtering system can reduce the noise in the Web information database, and the effectiveness of the proposed method has been demonstrated.

In addition, we carry out experiment 9 in order to examine about adding the condition to the filtering system based on the result of automatic tagging system of design tags. In experiment 9, design tag's acquisition is three kinds of design tag, and we add condition that designers apply the filtering system to only the sentences put three kinds of design tag are applied. As a result, in comparison with experiment 6, the accuracy improves without decreasing the recall ratio. Therefore, we consider that we can construct a more effective system by combining the result of the automatic tagging method for design tags and the filtering system.

4.2 Verification of the Automatic Conversion Method for Knowledge Representation

First, we verified the effectiveness of the converting method for one design knowledge sentence in each line. When we applied this proposed method to 100 press releases, it was automatically able to convert the design knowledge from all the press releases to one sentence in each line using punctuation. As mentioned above, the effectiveness of the converting method for placing one design knowledge sentence on each line is shown.

Next, we verify the effectiveness of the improved method of the automatic document tagging system. As a validation methodology, we obtain the correct data for the delimitation of If-Parts and Then-Parts for 32 kinds of design knowledge information that describes customer demands, the functions to satisfy the demand and the mechanism to achieve the function by hand. As a result, we verified whether the improved automatic document tagging method could correctly process If-Then-type knowledge information. We analysed 82 kinds of design knowledge information extracted from 17 kinds of press releases. This system extracted 32 kinds of If-Then-type knowledge information.

The part of the press release regarding a motherboard is shown below as an example of an experimental result.

<If-Part> By adopting low generation of heat and low RDS (on) MOSFET
<Then-Part> Longevity of the capacitor has improved by six times or more. [5]

The verification result for the improved automatic document tagging system is shown in Table 5. As shown, 65.60 % of the If-Then-type knowledge information can be converted to If-Then-type design knowledge. The improved automatic document tagging method can correctly convert more than half of 32 kinds of If-Then-type design knowledge information.

Table 5. Result of the verification of the improved method of the automatic document tagging systems

	Quantity	Proposal method	
		Accuracy before it applies	Accuracy after it applies
If-Then type knowledge	32	6.30%	65.60%

As mentioned above, the effectiveness of the improved method of the automatic document tagging system is shown.

Finally, when we apply the design knowledge acquired by the proposal method to inference mechanism of UAS, it operates correctly.

5 Discussion

We confirmed the feasibility of performing a series of operations for constructing the design knowledge database for UAS automatically and efficiently by the proposed methods. The following are the main results:

- *Automatic tagging method of design tags:* This method could extract design knowledge information automatically on the basis of linguistic features.
- *Filtering method:* This method could reduce the amount of redundant information such as sales information (sale dates, offer prices, etc.) using the named entity analysis automatically. In addition, it could improve the effectiveness of the automatic tagging method of design tags.
- *Automatic conversion method for knowledge representation*: This method could automatically convert more than half of the extracted If-Then-type information into If-Then-type design knowledge for UAS.

We found that the proposed methods have the following drawbacks:

- Because of errors commonly occurring in the named entity analysis, the filtering method sometimes erroneously reduces the amount of design knowledge information by including reduce tags.
- When If -Then-type information includes active sentences and itemized sentences, the improved automatic document tagging method can not process it. This is because such information does not include causal markers.

We will solve the above problems in the future.

6 Conclusion

In this thesis, we proposed a Web-based knowledge database construction method for supporting design. The verification results demonstrated the possibility of constructing a design knowledge database for UAS automatically and efficiently. We will improve the database construction method on the basis of the results obtained in this study. In addition, we will develop the system including all proposed method in order to automate the manual processing part, and verify about it intimately.

References

1. Ikehara, S., Miyazaki, M., Yokoo, A., Shirai, S., Nakaiwa, H., Ogura, K., Ooyama, Y., Hayashi, Y.: Nihongo Goi Taikei—A Japanese Lexicon. Iwanami Shoten 5 (1997) (in Japanese)
2. Inui, T., Okumura, M.: Investigating the Characteristics of Causal Relations in Japanese Text. In: The 43rd Annual Meeting of the Association for Computational Linguistics, Workshop on Frontiers in Corpus Annotation II: Pie in the Sky (2005)

3. Kudo, T., Matsumoto, Y.: Japanese Dependency Analysis Using Cascaded Chunking. In: Proc. 6th Conference on Natural Language Learning (CoNLL 2002), pp. 63–69 (2002)
4. Kudo, T.: Mecab: Yet Another Part-of-speech and Morphological Analyser, http://mecab.sourceforge.net/
5. Links International, Inc.: GA-MA790X-DS4 GIGABYTE Motherboard—Product Information (This press release is available in NIKKEI NET [8] dated (November 21, 2007) (in Japanese), http://www.gigabyte.com.tw/Products/Motherboard/Products_Overview.aspx?ProductID=2695
6. Luhn, H.P.: The Automatic Creation of Literature Abstracts. Journal of Research and Development 2(2), 159–165 (1958)
7. Niiyama, Y.: Webstemmer, http://www.unixuser.org/~euske/python/Webstemmer/index-j.htm
8. Nikkei Inc.: NIKKEI NET, http://release.nikkei.co.jp
9. Peirce, C.: Collected Papers of Charles Sanders Peirce, vol. 5. Harvard University Press, Cambridge (1935)
10. Sakaji, Y., Takeuchi, K., Sekine, S., Masuyama, S.: Causative Relation Extraction Using Syntactic Pattern. In: 14th Annual Meeting of the Japanese Association for Natural Language Processing, Tokyo, Japan (2008)
11. Sarwar, B., Karypis, G., Konstan, J., Riedl, J.: Analysis of Recommendation Algorithms for E-Commerce. In: Proc. of the 2nd ACM Conference on Electronic Commerce (EC 2000), pp. 285–295 (2000)
12. Sekine, S., Eriguchi, Y.: Japanese Named Entity Extraction Evaluation—Analysis of Results. In: Proc. of The 18th International Conference on Computational Linguistics, Saarbrucken, Germany, July 31–August 4, pp. 1106–1110 (2000)
13. Takeda, H., Sakai, H., Nomaguchi, Y., Yoshioka, M., Shimomura, Y., Tomiyama, T.: Universal Abduction Studio—Proposal of a Design Support Environment for Creative Thinking in Design. In: The Fourteenth International Conference on Engineering Design, ICED 2003 (2003)
14. The National Institute for Japanese Language: Relation between Surface Case and Deep Case in Japanese, Sanseido (1997) (in Japanese)
15. Watanabe, H.: Zoku Mechanical Design Acquaintanceship Note, The Nikkan Kogyo Shimbun (1988) (in Japanese)
16. Yamashita, T., Matsumoto, Y.: Language Independent Morphological Analysis. In: Proceedings of 6th Applied Natural Language Processing Conference, pp. 232–238 (2000)
17. Yoshioka, M., Satoh, T., Morimoto, K., Takeda, H., Shimomura, Y.: Proposal of Automatic Document Tagging Method for Hypothetical Knowledge Generation. In: Creative Design Support System, 20th Annual Conference of the Japanese Society for Artificial Intelligence, 3B3-02, pp. 1–4 (2006)