Trend Information Extraction based on Relative Expression participated on MuST T2N Subtask

Yasuhiro Uenishi† Fumito Masui† Tatsuaki Matsuba‡ Atsuo Kawai† Naoki Isu†
† Graduate School of Engineering, Mie University
‡ Faculty of Engineering, Mie University
1577, Kurimamachiya-cho, Tsu, Mie, 514-8507, Japan
{*uenishi, masui, matsuba, kawai, isu*}*@ai.info.mie-u.ac.jp*

Abstract

This paper describes a system participating in the MuST T2N subtask. To the participating system, we applied the method of implicit trend information extraction utilizing relative expressions such as "0.1%増 (grew 0.1%)", "前年 (previous year)", "過去最高 (maximum)" . Relative differences and numerical changes in trend information can be signified by relative expressions. The system extracts elements of four types by pattern-based rules considering the relative expression. The extracted element is compared with the query word by identifying the synonym of the elements utilizing an EDR dictionary and some synonym databases.

Some experiments were conducted with the MuST T2N formal run test collection. Although the results showed precision of 0.220 and recall of 0.029 totally, the outcomes of additional evaluations suggested the fundamental process performs effectively.

Keywords: *trend information, relative expression, quadruplet.*

1 Introduction

Information technology has expanded the variety of computerized text information. In modern society, one of the most serious problems is how to select useful information effectively amidst information overload. In order to extract the useful information, compiling information flexibly according to a user's interest [1] is necessary. Kato et al.[2] proposed "*Multimodal Summarization for Trend information*" to compile information[3][4][5]. "*Trend*" is defined as the first answer to the user's question such as "*how is it going in the game machine industry since 2006 ?*" or "*what changes have been seen in gasoline prices this year ?*". In trend information, time-series and geographical data are often included. Furthermore, interpretations, causes and forecasts of those data are also included in trend information. To extract and visualize the trend information, three stages are required as follows:

stage 1: Extraction of basic elements
stage 2: Selection of optimum visualization type
stage 3: Addition of annotations

Now, we try to solve the problem of stage 1. On the basis of the MuST definition[2], we defined a quadruplet, which is a set of four basic elements: *name*, *par*, *date* and *val*.

Ex1.
2007 年のアサヒのビール出荷量は **前年比0.1%増** の 1 億 8824 万ケースとなった。
(In 2007, the quantity of beer shipped by Asahi **grew 0.1% over the previous year** to 188.24 million cases).

Ex2.
パソコン出荷台数は **前年比 1%減** の 1414 万台だった。
(PC shipment volume **declined 1% from the previous year** to 1.414 million units.)

A relative expression signifies relative differences and a change of numerical values[6]. In many cases, the expression references not only one piece of trend information explicitly but also other trend information implicitly. Therefore, a relative expression is useful for collecting many more elements for *quadruplets* from identical documents easily.

Two examples of relative expressions are shown in Ex1 and Ex2. Explicitly, quadruplets are arranged from Ex1.

{ ビール出荷量 **(the quantity of beer shipped)**, アサヒ **(Asahi), 2007** 年 **(2007 year), 1** 億 **8824** 万ケース **(188.24 million cases)**}

Considering suggestion of relative expression "前年比 **0.1%**増 (grew 0.1% over the previous year)" , other quadruplets are inferred.

{ ビール出荷量 **(the quantity of beer shipped)**, アサヒ **(Asahi), 2006** 年 **(2006 year), 1** 億 **8800** 万ケース **(188 million cases)**}

Imaoka et al.[7] constructed trend information extraction rules with the tendency of relative expression in newspaper articles. They also evaluated their implemented extraction rules for trend information with experiments. And it was confirmed that the extraction based on a relative expression is well-performed. However, the appropriateness of extraction for the input query word has not been considered.

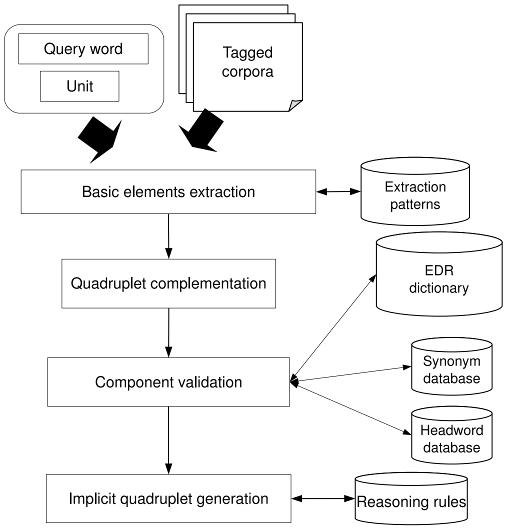

Figure 1. Overview of Our System

In this paper, we propose a trend information extraction system based on relative expression. We describe the overview of our participating system at the MuST T2N subtask in section2. In section3, performance of our system is evaluated. In section4, the results are discussed.

2 System Overview

This section describes an overview of our implemented system (Figure1). On the basis of relative expressions, quadruplets for trend information are extracted from the MuST corpus, which has been tagged with several kinds of tags. The system consists of the four modules below.

module 1: Basic elements extraction
module 2: Quadruplets complementation
module 3: Component validation
module 4: Implicit quadruplet generation

The detail of each process of the four modules is illustrated in the following subsections.

2.1 Basic Elements Extraction

Utilizing extraction patterns, basic elements for quadruplets and a relative expression are extracted from documents. The pattern has been manually generated with abstraction of relative expressions frequently in the MuST corpus. Ex3 and Ex4 show some examples of extraction patterns.

Ex3.
<date> の <par> の <name> は <date> 比 <rel> 増の <val>
(In <date>, <name> by <par> grew <rel> over <date> to <val>.)

Ex4.
<name> は <date> 比 <rel> 減の <val>
(<name> declined <rel> from <date> to <val>.)

In search of a document in corpus[1] , tagged characters matched with the extraction pattern are extracted as an element relating each tag type. By collecting the elements, a quadruplet is arranged.

Ex5.
<date>2007 年 </date> の <par> アサヒ </par> の <name> ビール出荷量 </name> は <date> 前年 </date> 比 <rel>0.1%</rel> 増の <val>1 億 8824 万ケース </val> となった。
(In <date>2007</date>, <name>the quantity of beer shipped</name> by <par>Asahi</par> grew <rel>0.1%</rel> over <date>the previous year</date> to <val>188.24 million cases</val>.)

⇑

P_1: <date> の <par> の <name> は <date> 比 <rel> 増の <val>
(In <date>, <name> by <par> grew <rel> over <date> to <val>.)

⇓

name =	ビール出荷量 (the quantity of beer shipped)
par =	アサヒ (Asahi)
date =	2007 年 (2007)
val =	1 億 8824 万ケース (188.24 million cases)

⇓

Q_i =
{ ビール出荷量 (the quantity of beer shipped), アサヒ (Asahi), 2007 年 (2007), 1 億 8824 万ケース (188.24 million cases)}

In Ex5, the extraction pattern P_1 extracts four elements, which are "ビール出荷量 (the quantity of beer shipped)" as *name*, "アサヒ (Asahi)" as *par*, "2007 年 (2007)" as *date* and "1 億 8824 万ケース (188.24 million cases)" as *val* from the sentence ex1. In the end, a quadruplet Q_i is arranged.

2.2 Quadruplet Complementation

In some cases in the process of the basic elements extraction, not enough elements are extracted to fill a quadruplet.

To glean such missing elements, complementation rules are applied. Details of complementation rules are explained below.

name: The nearest *name* element ahead of a part matched with the extraction pattern
par: The nearest *par* element ahead of a part matched with the extraction pattern from a sentence

[1] In T2N subtask, a document unit is defined as one newspaper article.

If a *par* element is not found in a sentence, the quadruplet does not have a *par* element.

date: The complementation rules are applied in the following order. The element matched with the rules is complemented.
(1) The nearest *date* element ahead of a part matched with the extraction pattern in a sentence
(2) The *date* element which is close to the head of an article
(3) The date when the newspaper article was written

In complementation rules (1) and (2), the *date* element matched with patterns "*<date> の (<date> no)*" or "*<date> における (<date> ni-okeru)*" is complemented.

The *val* element is extracted by only extraction patterns and is not complemented.

In Ex6, extraction pattern: "<name> は <date> 比 <rel> 減の <val> (<name> declined <rel> from <date> to <val>.) " is applied. Then, *name*:"パソコン出荷台数 (the PC shipment volume)" and *val*:"1414 万台 (1.414 million units)" are extracted. Because the *par* element and *date* element could not be extracted by the extraction pattern, the two elements are complemented by the complementation rules. In this case, there is not a *par* element in the sentence matched with the extraction pattern. Therefore, the quadruplet does not have a *par* element. On a *date* element, "2007 年" is complemented by the second complement rule. Finally, quadruplet: { パソコン出荷台数 (the PC shipment volume), ϕ, 2007 年 (2007), 1414 万台 (1.414 million units)} is extracted.

Ex6.
日本電子工業振興協会は <date>9 日 </date>、
<date>2007 年 </date> のパソコン国内実績を発表した。
<name> パソコン出荷台数 </name> は <date> 前年 </date> 比 <rel>1%</rel> 減の <val>1414 万台 </val> だった。
(On <date>9th</date>, JEIDA reported the PC shipments in **<date>2007</date>**. *<name>The PC shipment volume</name> declined <rel>1%</rel> from <date>the previous year</date> to <val>1.414million units</val>.*

2.3 Component validation

In this module, the quadruplets that relate to the query word are selected. To select the quadruplets related to a query word, components of the *name* element and the query word are validated.

We assumed that both a *name* element and a query word also consist of a headword (a unit of trend) and a specifier (a subject of trend). Some examples are shown in Table1. For example, " パソコン出荷台数 (PC shipment volume)"(*name* element) is divided into " パソコン (PC)"(specifier) and " 出荷台数 (shipment volume)"(headword).

Figure 2 shows an overview of component validation. For validation, three processes are applied as follows:

(1) generating suffix set
(2) distinguishing components
(3) checking components

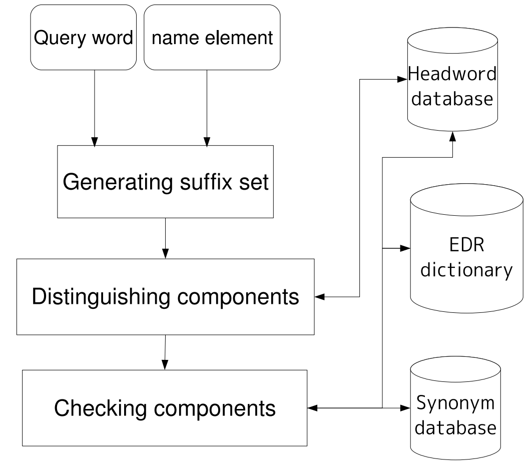

Figure 2. Overview of Component validation

If checking components succeeds, the *name* element relates to the query word. Through the three processes, the quadruplets relevant to the query word are selected. Each process is illustrated below.

Table 1. Component of *name* Element

name element	specifier	headword
パソコン出荷台数 (PC shipment volume)	パソコン (PC)	出荷台数 (Shipment volume)
ビール出荷数量 (The quantity of beer shipped)	ビール (Beer)	出荷数量 (The shipped quantity)
政党支持率 (Approval rating for political parties)	政党 (Political parties)	支持率 (Approval rating)

In order to validate components, the EDR dictionary[2],the synonym database and the headword database are utilized. The synonym database and the component database are constructed manually. The synonym database includes 54 associate relations to check a specifier. The headword database includes 142 relations to check a headword.

Generating suffix set: This process is applied to the query word and the extracted *name* element. The object character unit is divided into a set of morphemes. The set of morphemes is refined by deleting words such as particles and auxiliary verbs. Based on the refined morphemes set, suffixes are generated for each element. In Ex7, "台数 (volume)","出荷台数 (shipment volume)" and "パソコン出荷台数 (PC shipment volume)" as suffix are generated form "パソコンの出荷台数 (PC shipment volume)".

[2] http://www.jsa.co.jp/EDR/index.html?

Ex7.
パソコンの出荷台数
(PC shipment volume)
⇒ パソコン / 出荷 / 台数
(PC) (shipment) (volume)
⇒suffix: { 台数, 出荷台数, パソコン出荷台数 }
(volume, shipment volume, PC shipment volume)

Distinguishing components: In this process, each one of the query word and the *name* element is divided into a headword from a specifier.

First, on the query word, the following steps are applied.

step 1: Each suffix is compared with the entries in the headword database.
step 2: If exactly the same characters as the suffix are found in the database, the matched suffix is recognized as a headword. The other part of a query word is also recognized as a specifier.
step 3: If exactly the same characters as the suffix are not found in the database, the process ends.
step 4: To the *name* element, the same step is also applied.

If the headword is identical to the *name* element, the *par* element becomes a specifier.

Ex8 shows the process when ”パソコンの出荷台数 (PC shipment volume)” is divided into ”出荷台数 (shipment volume)”(headword) and ”パソコン (PC)”(specifier).

Ex8.
headword db:{ 出荷数量, 出荷台数, 支持率, . . . }
(the shipped quantity, shipment volume, approval rating)
suffix: { 台数, 出荷台数, パソコン出荷台数 }
(volume, shipment volume, PC shipment volume)

suffix		headword db
台数 (volume)	⇒	“出荷数量” : miss (the shipped quantity)
台数 (volume)	⇒	“支持率” : miss (approval rating)
	⋮	
出荷台数 (shipment volume)	⇒	“出荷台数” : match (shipment volume)

⇓
headword: { 出荷台数 (shipment volume)}
specifier: { パソコン (PC)}

Checking components: The identity of the headwords and the specifiers are checked respectively. The *name* element that passed both checks is determined to relate to the query word.

(1) Checking headword:

step 1: The ids for two headwords, which are query-derived and element-derived, are retrieved from the headword database and compared.
step 2: If they are identical, go to the next step; if they are not the same, the process ends.

(2) Checking specifier:

step 3: The ids for two specifiers, which are query-derived and element-derived, and the id of the upper concept from the element-derived specifier are retrieved from the EDR dictionary and compared.
step 4: In the following cases, the element-derived specifier is relevant to the query-derived specifier and the process ends.
(a) Both specifiers had the same id (Ex9).
(b) The id of the query-based specifier is identical to the id of the upper concept from the element-derived specifier (Ex10).
step 5: If they are not identical, go to the next step.
step 6: The ids for two specifiers, which are query-derived and element-derived, and the id of the relevant concept to the element-derived specifier are retrieved from the synonym database and compared.
step 7: In the following cases, the element-derived specifier is relevant to the query-derived specifier.
(a) Both specifiers have the same id, and they are recognized as identical specifiers (Ex11).
(b) The id of the query-based specifier matches the id of the associative concept from the element-based specifier (Ex12).

Ex9.
query: パソコン出荷台数
(PC shipment volume)
[パソコン (PC)⇒3c677f]
name: パーソナルコンピュータの出荷台数
(Personal computer shipment volume)
[パーソナルコンピュータ
(Personal computer)⇒3c677f]
Ex10.
query: 政党支持率
(Approval rating for political parties)
[政党 (Political parties)⇒0f95e0]
name: 支持率
(Approval rating)
par: 自民党
(LDP)
[自民党 (LDP)⇒1f7f26⇒0f95e0]
Ex11.
query: デジタルカメラ出荷台数
(digital camera shipment volume)
[デジタルカメラ (digital camera)⇒1]
name: デジカメの出荷台数
(digital camera shipment volume)
[デジカメ (digital camera)⇒1]
Ex12.
query: パソコンの出荷台数
(shipment volume of PC)
[パソコン (PC)⇒50]
name: 出荷台数
(shipment volume)
par: NEC [NEC⇒74⇒50]

If the *name* element consists of only a headword and a quadruplet does not include the *par* element, the specifier is checked based on co-occurrence in a sentence. The sentence where the *val* element has been extracted is searched for the query-derived specifier ahead of the extracted point. If the specifier is found, the *name* element is relevant to the query word. In Ex13, because the query-derived specifier:"パソコン (PC)" appear in the sentence where the *val* element:"1249 万台 (12.49 million units) has been extracted, *name* element is relevant to the query word.

Ex13.
query: パソコン出荷台数
(PC shipment volume)
name: 出荷台数 (shipment volume)
sentence: **パソコン** 出荷金額は前年比 1.9%減の 1 兆 7360 億円で, **出荷台数** は同 4.2%増の **1249 万台** となった
(**PC** shipment value declined 1.9 % from the previous year to 1.736 trillion yen and **shipment volume** grew 4.2 % over the year to **12.49 million units**.)

2.4 Implicit Quadruplet Generation

In this module, to reason the other quadruplet, some reasoning rules are applied for every relative expression. Applying the elements in an explicit quadruplet to the reasoning rules, other implicit elements for *date* and *var* are calculated.

For instance, the following four basic elements are extracted from the first sentence in Ex1 and arranged as a quadruplet.

$$\begin{aligned} name_{exp} &= \text{ビール出荷量} \\ &\quad \text{(The quantity of beer shipped)} \\ par_{exp} &= \text{アサヒ (Asahi)} \\ date_{exp} &= \text{2007 年 (2007)} \\ val_{exp} &= \text{1 億 8824 万ケース} \\ &\quad \text{(188.24 million cases)} \end{aligned}$$

$$\Downarrow$$

$$\{ \text{ビール出荷量 (the quantity of beer shipped),} \\ \text{アサヒ (Asahi), 2007 年 (2007),} \\ \text{1 億 8824 万ケース (188.24 million cases)} \}$$

As the rule related with " 前年比0.1%増 (grew 0.1% over the previous year)", (a) and (1) in Figure3 are selected and applied to $date_{exp}$ and val_{exp}. Consequently, the other quadruplet is arranged.

$$\begin{aligned} date_{imp} &= \text{2007 年 (2007 year) - 1 年 (1 year)} \\ &= \text{2006 年(2006 year)} \end{aligned}$$

$$\begin{aligned} val_{imp} &= \frac{1 \text{ 億 } 8824 \text{ 万ケース } (188.24\ million\ cases)}{1 + \frac{0.1}{100}} \\ &= 1 \text{ 億 } 8800 \text{ 万ケース } (188\ million\ cases) \end{aligned}$$

$$\Downarrow$$

$$\{ \textbf{ビール出荷量 (the quantity of beer shipped),} \\ \textbf{アサヒ (Asahi), 2006 年 (2006 year),} \\ \textbf{1 億 8800 万ケース (188 million cases)} \}$$

3 Experiments

In Table2, evaluation results of our system for the MuST T2N formal run are shown. Because we could not completely implement the function described in section2, the results show precision of 0.220 and recall of 0.029 on the micro-average. Precision and recall on the macro-average are 0.093 and 0.021 respectively. There is remarkable difference between the micro-average and macro average in precision. The result indicates that the system still has uneven performance.

Table 2. Formal Run Evaluation Results

	Precision	Recall	F-measure
micro-ave.	0.220	0.029	0.051
macro-ave.	0.093	0.021	0.031

Table 3. Evaluation Results for Relative Expressions

	Precision	Recall	F-measure
system1	0.220	0.068	0.108
system2	0.638	0.226	0.333

In the formal run, recall for evaluation is defined as the proportion of the target elements that the system extracted from all documents. The target elements are pairs of *date* element and *val* element. Our system extracts elements only for relative expression. Recall for exact evaluation should be defined as the proportion of the target elements that the system extracted from documents for relative expression. We also evaluated performance of our system for relative expression. In Table 3, performance results are shown for the participating system in the formal run (**system1**) and the system implementing all functions described in Section 2(**system2**). Precision and recall on the micro-average are 0.638 and 0.226 respectively.

4 Discussion

Regarding performance of the whole system, the result shows precision of 0.638 and recall of 0.226. This result indicates that basic elements are not sufficiently extracted.

Table 4. Detailed Evaluation Results of Our System

	Precision	Recall	F-measure
module1	1.000	0.669	0.802
module2	0.718	–	–
module3	0.882	0.455	0.600

module1: basic elements extraction
module2: quadruplet complementation
module3: implicit quadruplet generation

Table 4 shows detailed evaluation results of our system. Regarding the extraction of basic elements process, the result shows precision of 1.00 and recall of 0.699. It can be said that the extraction pattern could not extract enough basic elements. It is conspicuous for these three topics: "デジカメ (digital camera)", "通信機器 (communication device)" and "ガソリン (gasoline)".

In the topics "デジカメ" and "通信機器", the main cause is that many specific expressions out of the training data appeared in those articles. We can enlarge the

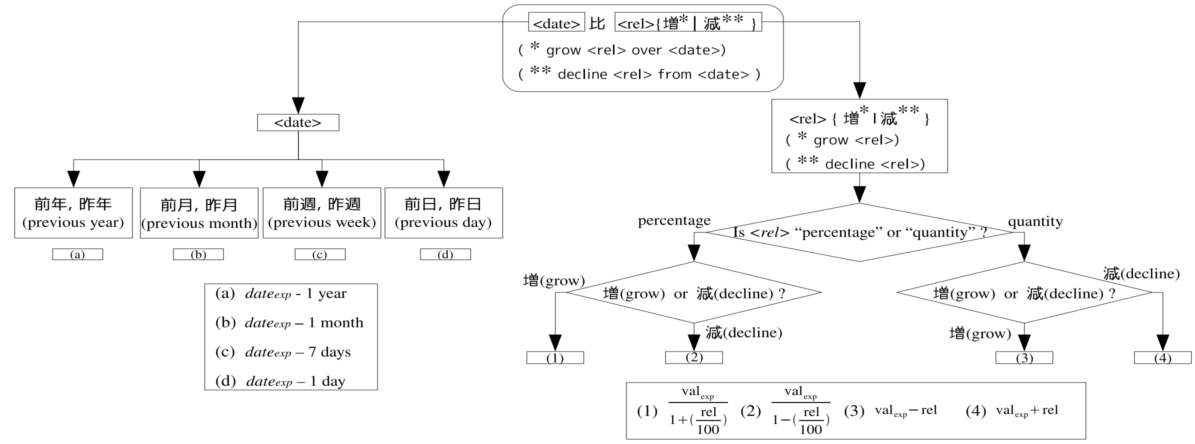
<date> 比 <rel>{増* | 減** }
(* grow <rel> over <date>)
(** decline <rel> from <date>)
<date>
前年, 昨年 (previous year)
前月, 昨月 (previous month)
前週, 昨週 (previous week)
前日, 昨日 (previous day)
(a)
(b)
(c)
(d)
(a) date_exp - 1 year
(b) date_exp – 1 month
(c) date_exp – 7 days
(d) date_exp – 1 day
<rel> { 増* | 減** }
(* grow <rel>)
(** decline <rel>)
percentage
Is <rel> "percentage" or "quantity" ?
quantity
増(grow)
増(grow) or 減(decline) ?
減(decline)
増(grow) or 減(decline) ?
減(decline)
増(grow)
(1)
(2)
(3)
(4)
(1) val_exp / (1+(rel/100)) (2) val_exp / (1−(rel/100)) (3) val_exp − rel (4) val_exp + rel

Figure 3. Example of Reasoning Rules

training data to solve this problem. In the topic of "ガソリン"", It is difficult to abstract relative expressions in articles because there is no regular appearance of relative expressions.

For the complementation of elements process, only precision is meaningful to evaluate because of the task of complementing four elements. As a result of this process, precision marked 0.718. The failure of the complementing of the *date* element is remarkable in particular. An excessively strict condition for extracting the *date* element caused the failure of the complement.

The outcome of synonymy identification shows 0.882 for precision and 0.455 for recall. Mainly, identification was unsuccessful in the case where the *name* element is verbose. For example, "国内パソコン出荷台数 (domestic PC shipments)" is divided into "国内パソコン (domestic PC)" as a specifier and "出荷台数 (shipments)" as a headword. But no synonym is found for "国内パソコン (domestic PC)" because of verboseness. In order to apply identification properly, the specifier should be recognized as "パソコン (PC)" with the estimating structure of a specifier.

In the other instance, "携帯電話と PHS の合計加入台数 (the total number of cellular phone and PHS subscribers)" was divided into "携帯電話と PHS(cellular phone and PHS)" as a specifier and "合計加入台数 (the total number of subscribers)" as a headword. However, strictly, two specifiers such as "携帯電話 (cellular phone)" and "PHS" should be recognized. To solve this problem, the method to extract elements considering parallel noun structure is necessary.

5 Conclusion

A trend information extraction system using *relative expressions* was proposed. We defined relative expressions and described the overview of our participating system at the MuST T2N subtask in section2. Our system was evaluated by MuST T2N subtask evaluation. We could not obtain satisfactory results in comprehensive performance because of the implementation problem.

Therefore, we also conducted additional experiments and each module of the system was evaluated. The results showed fairly good precision and promising recall and suggested that the fundamental process performs effectively. We believe there is good potential in our proposed method.

In the future, we need to improve the recall of the proposed method. Moreover, for applying the method to non-tagged documents, we should try to modify the extraction process with the named entity extraction method.

Acknowledgment

This work was supported by KAKENHI(20500833).

References

[1] Kato, T., and Matsushita, M.: Toward Information Compilation, *The 20th Annual Conference of Japanese Society for Artificial Intelligence*, 1D3-2, 2006 (in Japanese).

[2] Kato, T., Matsushita, M., and Kando, N.: MuST: A Workshop on Multimodal Summarization for Trend Information, *Proceedings of the Fifth NTCIR Workshop Meeting*, pp. 556-563, 2005.

[3] Nanba, H., Okuda, N., and Okumura, M.: Extraction and Visualization of Trend Information from Newspaper Articles and Blogs, *Proceedings of the Sixth NTCIR Workshop Meeting*, pp. 243-248, 2007.

[4] Takama, Y., Yamada, T., and Nakano, J.: Visualization of Earthquake Trend Information from MuST Corpus, *Proceedings of the Sixth NTCIR Workshop Meeting*, pp. 249-255, 2007.

[5] Saito, H., Kawai, H., Tsuchida, M., Mizuguchi, H., and Kusui D.: Extraction of Statistical Terms and Co-occurrence Networks from Newspapers, *Proceedings of the Sixth NTCIR Workshop Meeting*, pp. 256-263, 2007.

[6] Nanba, H., Kunimasa, Y., Fukushima, S., Aizawa, T., and Okumura, M.: Extraction and Visualization of Trend Information Based on the Cross-document Structure, IPSJ SIG Technical Report, 2005-NL-168, pp.67-74, 2005 (In Japanese).

[7] Imaoka, H., Masui, F., Kawai, A., and Isu, N.: Effectiveness of Relative Expression for Trend Information Extraction, *Journal of Japan Society for Fuzzy Theory and Intelligent Informatics*, Vol.18, No.5, pp.735-744, 2006 (in Japanese).