Electronic Preprint for Journal of Information Processing Vol.20 No.1

Regular Paper

Incremental Construction of Causal Network from News Articles

Hiroshi Ishii[1,a) Qiang Ma[2] Masatoshi Yoshikawa[2]

Received: April 11, 2011, Accepted: September 12, 2011

Abstract: We propose a novel method for the incremental construction of causal networks to clarify the relationships among news events. We propose the Topic-Event Causal (TEC) model as a causal network model and an incremental constructing method based on it. In the TEC model, a causal relation is expressed using a directed graph and a vertex representing an event. A vertex contains structured keywords consisting of topic keywords and an SVO tuple. An SVO tuple, which consists of a tuple of subject,verb and object keywords represent the details of the event. To obtain a chain of causal relations, vertices representing a similar event need to be detected. We reduce the time taken to detect them by restricting the calculation to topics using topic keywords. We detect them on a concept level. We propose an identification method that identifies the sense of the keywords and introduce three semantic distance methods to compare keywords. Our method detects vertices representing similar events more precisely than conventional methods. We carried out experiments to validate the proposed methods.

Keywords: causal relation, network, increment, word concept, structured keywords

1. Introduction

News events are reported in television news programs, newspapers, and web pages every day. However, a deep understanding of a news event requires a background knowledge of that event. To obtain the necessary background knowledge, users can search for other articles about the event one by one. However, this can be arduous and users may still miss valuable components of the story. If relations between events (e.g., stock market events) are complex, understanding them is not easy. A system that provides background information of a news event is needed to improve our understanding of news. Causal relations are one way of providing the important knowledge that helps us understand the relations between events [7], [8].

Numerous technologies have been developed to extract and organize causal relations and construct causal networks [1], [2], [4], [5], [17], [18], [19]. These technologies constructed causal networks from documents on the Web. Conventional methods usually construct causal networks in a batch manner and rarely update them. Because new events are reported every day, for causal networks constructed from news articles, causal networks need to be constantly updated. To achieve this, we propose an incremental method for causal network construction.

Generally, for a causal network, a causal relation is expressed using a directed graph. A source vertex represents the cause event, and a destination vertex represents the result event. Usually, a vertex consists of keywords related to the event. In conventional methods, since keywords are extracted only from the phrases including a causal relation, each vertex consists of only a few words. Therefore, conventional approaches have two problems.

- We cannot understand what event the vertex represents.
- We cannot identify the original event of a vertex.

When a vertex and another vertex represent the same event, these vertices will be merged to construct a network. By this operation, we obtain a chain of causal relations and show the underlying cause and the result of an event. However, conventional methods using a set of keywords representing an event cannot efficiently merge similar vertices because of two problems.

- Vertices have to be compared too many times to detect similar vertices for network construction.
- Similar vertices are not detected accurately enough.

Since the similarity between each vertex pair for all vertices is calculated, times of comparison for detecting similar vertices are too many. Conventionally, the similarity between a vertices pair is computed by calculating the similarity with keywords of vertices. However, in this approach, the accuracy of the vertices similarity is low. Since a vertex generated by merging vertices repeated many times will include various events, the event imaged from the vertex after the merging operation is different from the event represented by the original vertex. Because of this, causal networks are different depending on the merging order and the consistency problem occurs.

As one of the solutions, we propose a topic-event causal network (TEC) model for representing causal relations and an incremental construction method of a causal network based on the model. In the TEC model, the topic and details of an event are separately represented by using topic keywords and an SVO tu-

[1] Corporate Software Engineering Center, Toshiba, Kawasaki, Kanagawa 212–8582, Japan
[2] Graduate School of Informatics, Kyoto University, Kyoto 606–8501, Japan
[a)] hiroshi11.ishii@toshiba.co.jp

ple. An SVO tuple is extracted by using a syntactic SVO structure. The SVO structure is used for extracting the basic element of an event [6]. An SVO tuple, which consists of a tuple of subject, verb and object keywords represents the details of the event. On the basis of a TEC model, we construct causal networks. We reduce the time taken to detect similar event vertices by restricting the calculation to topics using topic keywords. Since, in the calculation for detecting similar event vertices, we consider the role of keywords by using SVO syntactic structure and the concept of keywords by using WordNet, the proposed method detects similar event vertices more precisely.

The remainder of this paper is organized as follows. Section 2 introduces related work. Section 3 describes the TEC model and the method for extracting causal relations from articles. Section 4 introduces the method for constructing the causal network. Section 5 describes the experimental results. Section 6 concludes this paper.

2. Related Work

2.1 Extraction of Causal Relations

Various methods have been proposed to extract causal relations from Japanese documents, including using the clue phrase "tame" (because) [5]; and using clue phrases, in which a sentence represents a causal relation, and structural patterns [17]. In the research using a joint label [5], causal relations are extracted only from sentences including the word "tame". Sakaji et al. [17] extract causal relations using clue phrases (e.g., "tame", "wo haikei ni" (behind)). Using the Japanese syntax features, they divide clue phrases into four extracting patterns of causal relations. In contrast, to extract causal relations from English documents, Girju [4] focused on the concepts of a word and proposed a causal relation extraction method that uses WordNet. In this method [5], the causal relation is extracted only from complex or compound sentences. However, in Sakaji et al.'s method [17], the causal relation is also extracted from other types of sentences. In our work, we extracted causal relations using Sakaji et al.'s method [17].

2.2 Construction of Causal Network

By using the method for extracting the causal relations, various methods [1], [2], [18], [19] for constructing a causal network have been proposed. In other related research described by Feng [3], the methods for constructing a relation network of news are developed by using event threading in English. In case-frame dictionary methods [18], a causal network is constructed from documents to form a causal network of common knowledge. In another research [19], a causal network is constructed from Web pages. In a search-based method [1], [2], when a user inputs the query about an event into the system, the system extracts the cause relation about the query on the Web and constructs a causal network around the event queried. In all methods, important words are extracted from phrases including causal relations as keywords of vertices. Since keywords are extracted only from the phrases including a causal relation, each vertex consists of only a few words. Therefore, conventional approaches have problems as we mentioned in Section 1; the causal network is not easy for understanding and merge accuracy is low.

In Sato and Horita's research [19], there is no method for merging the vertices. However, if the network has a lot of vertices, it is difficult to find the chain of causal relations. In our research, we overcome the difference in the word used and the lexical ambiguity by using WordNet, and we calculated the relatedness of all vertices pairs on concepts. We merge vertices with high relatedness and represent the chain of causal relations.

The major differences can be summarized as follows.

- Unlike the conventional methods, which pay more attention to how to construct causal networks in a certain period, our causal networks are incrementally constructed in order to update knowledge quickly.
- In conventional methods [1], [2], [18], a thesaurus is used to solve the lexical ambiguity. However, only words that are registered to the same concept can be combined. We measure the relatedness between concepts and combine concepts with high relatedness.
- In conventional methods, the grammatical role of a term is not considered to measure similarity between events. We consider that using Japanese syntax analysis and make a vertex to express the event specifically.

2.3 Semantic Similarity Method

Various methods [9], [10], [11], [12], [13], [14], [15], [20] for measuring semantic similarity have been proposed. Lee's method [11] calculates the conditional probability between Verb and Noun. This method uses some combination of the probability and some similarity metrics and evaluates each combination. Patwardhan's method [14] detects co-occurrence words for each word and create the vector based on these words and WordNet. Saruladha [10] surveyed some semantic similarity methods; Resnik's method [15], Jiang's method [9], etc. Liu [13] surveyed some semantic similarity methods based on WordNet and evaluated them; Resnik's method [15], Lin's method [12], Jiang's method [9] and Wu's method [20].

Our goal is to construct causal networks incrementally and a method of detecting the same event with different descriptions is necessary. In this paper, we propose a method of detecting same event with different descriptions by considering both the structure and content features of the descriptions of an event.

- These descriptions have a different representation using different terms. We need a method to estimate whether these different terms represent very similar (or the same) concepts. The conventional methods [11], [14], [15] based on co-occurrence relationships of terms are inappropriate and we select one semantic similarity method [20] which is introduced by Saruladha et al. [10].
- The descriptions of an event should make clear "who does what." In other words, the Subject, Object and Verb should be described. Thus, we represent a description of an event by using an SVO tuple and propose the method of comparing event descriptions by using it. That is to say, we propose a structure based approach.

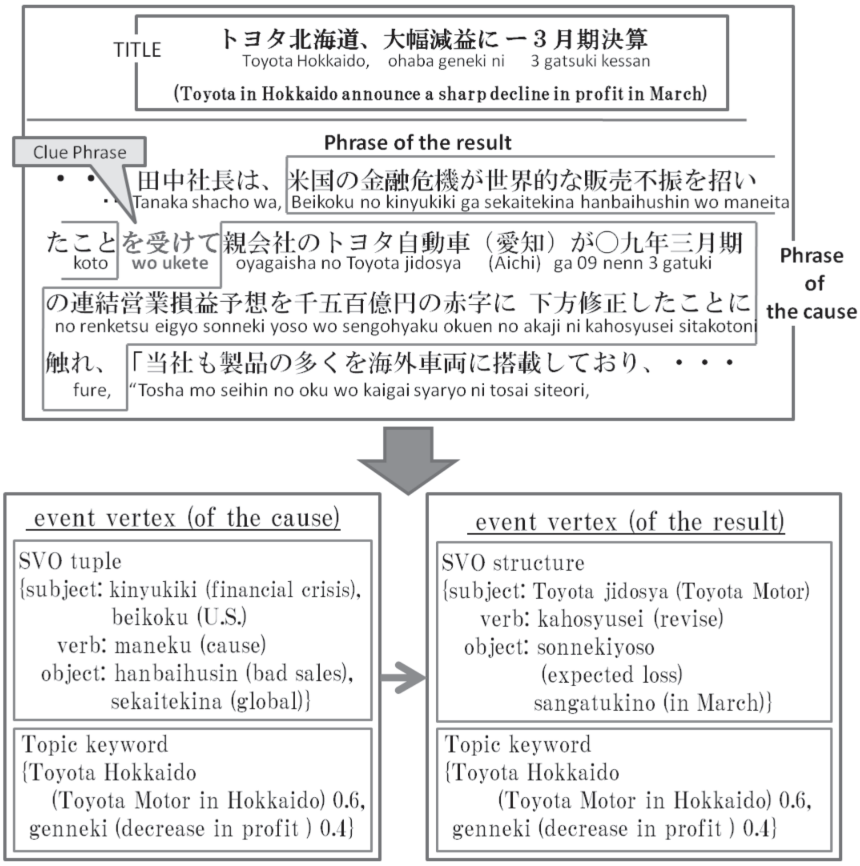
TITLE
トヨタ北海道、大幅減益に ー３月期決算
Toyota Hokkaido, ohaba geneki ni 3 gatsuki kessan
(Toyota in Hokkaido announce a sharp decline in profit in March)
Clue Phrase
Phrase of the result
・・・田中社長は、米国の金融危機が世界的な販売不振を招い
Tanaka shacho wa, Beikoku no kinyukiki ga sekaitekina hanbaihushin wo maneita
たことを受けて親会社のトヨタ自動車（愛知）が〇九年三月期
koto wo ukete oyagaisha no Toyota jidosya (Aichi) ga 09 nenn 3 gatuki
の連結営業損益予想を千五百億円の赤字に 下方修正したことに
no renketsu eigyo sonneki yoso wo sengohyaku okuen no akaji ni kahosyusei sitakotoni
触れ、「当社も製品の多くを海外車両に搭載しており、・・・
fure, "Tosha mo seihin no oku wo kaigai syaryo ni tosai siteori,
Phrase of the cause
event vertex (of the cause)
SVO tuple
{subject: kinyukiki (financial crisis), beikoku (U.S.)
verb: maneku (cause)
object: hanbaihusin (bad sales), sekaitekina (global)}
Topic keyword
{Toyota Hokkaido (Toyota Motor in Hokkaido) 0.6, genneki (decrease in profit) 0.4}
event vertex (of the result)
SVO structure
{subject: Toyota jidosya (Toyota Motor)
verb: kahosyusei (revise)
object: sonnekiyoso (expected loss) sangatukino (in March)}
Topic keyword
{Toyota Hokkaido (Toyota Motor in Hokkaido) 0.6, genneki (decrease in profit) 0.4}

Fig. 1 Causal relation extraction from articles.

3. Topic Event Causal Model

In the TEC model, causal relations are represented by a directed graph, i.e., the causal network. The source and destination vertices denote the causal and result events of a causal relation, respectively. In the TEC model, a vertex represents the topic and details of an event separately. A topic is represented by using topic keywords, and the details are represented by using an SVO tuple.

(a) Topic keywords T

Topic keywords T represent the topic of an event. By using Chasen [*1], we analyze the titles of related articles that describe the same topic and extract words as keywords for further processing. The words that appear frequently are extracted as the topic keywords T. Here, related articles are those on a certain topic obtained from Google News.

(b) Extracting SVO tuples E_{SVO}

An SVO tuple E_{SVO} represents the details of an event. We used the Japanese syntactic parsing provided by the Institute of Language Engineering [*2] to construct the SVO tuple. We extract keywords of subjects, verbs and objects on the basis of grammatical frames (e.g., "[subject words] + ga + [verb words]", "[object words] + wo + [verb words]").

We construct event vertices and edges from phrases including events with causal relations. To extract causal relations, we used the method based on clue phrases (in Japanese) and four syntax patterns [17]. **Figure 1** shows an example of extracting causal relations and creating event vertices.

4. Construction of Causal Network Using TEC model

4.1 Overview

We construct the causal network on the basis of the TEC model. In our method, we construct a causal network by the following two-step operation. First, a causal network is constructed from the news articles over a certain period (one day, one week, etc.). We extract causal relations (from news articles) and make a causal network (**Fig. 2** (a)). Next, we merge the previous network with the current one to update the causal network (Fig. 2 (b)). The cycle in Fig. 2 shows how the network is updated by repeating extraction and merging operations for the causal relations.

4.2 Detecting Similar Vertices

We detect similar event vertices in order to merge them. First, we group vertices into topics. Then, we calculate the relatedness between SVO tuples per topic to detecting similar vertices.

4.2.1 Grouping Vertices

We calculate the topic similarity and group vertices into topics by using topic keywords. Words used frequently in each related news articles organized by Google News articles are used as topic keywords.

All vertices extracted from the same set of related news articles have the same topic keywords. Therefore, the topic similar-

[*1] Chasen (Japanese morphological analyzer): http://chasen-legacy.sourceforge.jp/

[*2] http://www.gengokk.co.jp/

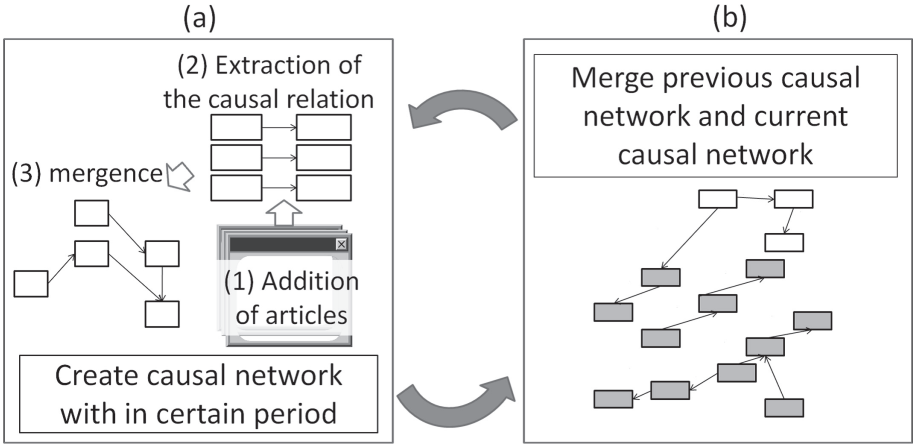
(a)
(2) Extraction of the causal relation
(3) mergence
(1) Addition of articles
Create causal network with in certain period
(b)
Merge previous causal network and current causal network

Fig. 2 Incremental construction of network.

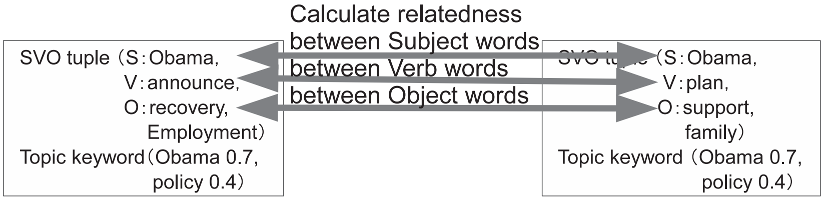
Calculate relatedness
between Subject words
between Verb words
between Object words
SVO tuple（S：Obama, V：announce, O：recovery, Employment）
Topic keyword（Obama 0.7, policy 0.4）
SVO tuple（S：Obama, V：plan, O：support, family）
Topic keyword（Obama 0.7, policy 0.4）

Fig. 3 Relatedness between SVO tuples.

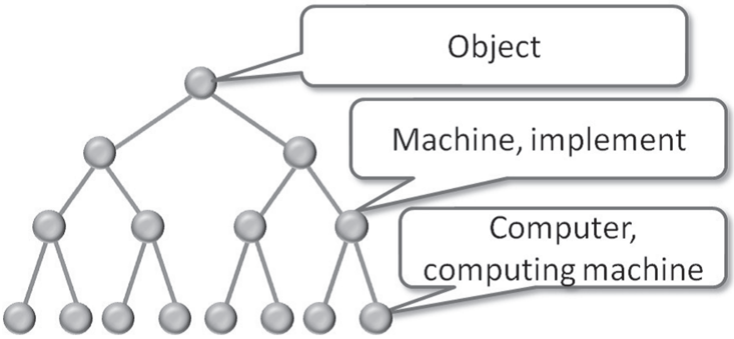

Fig. 4 WordNet structure.

ity is calculated once for a pair of related article sets. Because we incrementally merge causal networks, we have two methods for computing the topic similarity between news articles: one is for computing topic similarity of article sets in a certain period to construct the current network (Fig. 2 (a)), and the other is for computing the topic similarity of article sets in previous and current networks (Fig. 2 (b)).

(1) Grouping vertices per topic in a certain period

By using topic keywords and their frequencies, we compute the cosine similarity as the topic similarity based on a vector space model. We calculate topic similarities between each pair of related articles sets from a certain period. We group the pair of related news article sets as similar topic news articles in descending order of the topic similarity while they exceed the threshold.

(2) Grouping vertices in different causal networks

The topic similarity between a related article set in the previous causal network and that in the current causal network is calculated. In general, the content of news reports continuously changes over time. As a result, the topic keywords will change. The topic keywords of the previous and current networks may be different although they are related to the same topic. In order to keep pace with the change of topic keywords, in the cosine similarity calculation for this topic similarity we use only the words included in the topic keywords of the related article set of the current causal network. The pair of related article sets is grouped as similar topic news articles in descending order of the topic similarity while they exceed the threshold. To update topic keywords of related article sets that have been merged, we give the union of topic keywords of related article sets that have been merged for topic keywords updated.

4.2.2 Relatedness between SVO Tuples

After vertices are grouped into topics, we calculate relatedness between SVO tuples of vertices in each topic to detect similar events. As shown in **Fig. 3**, to calculate the relatedness of SVO tuples, we calculate the relatedness between their subject, verb, and object keywords, respectively. Relatedness between two keywords is computed at the concept level by using Japanese WordNet. As shown in **Fig. 4**, super-subrelation or inclusive relations

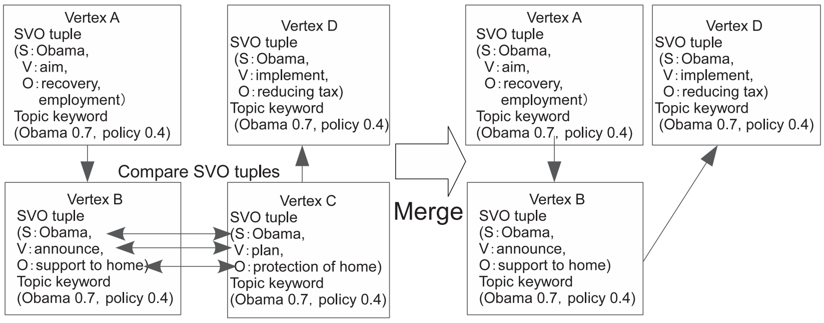

Fig. 6 Merging vertices.

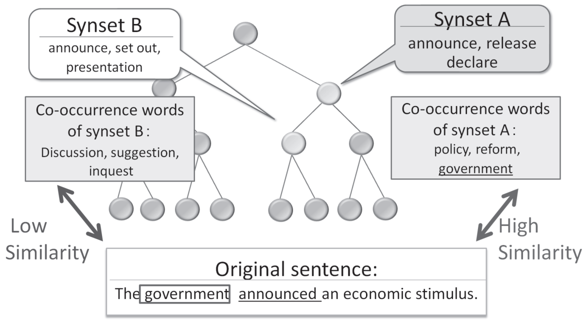
Synset B
announce, set out, presentation
Synset A
announce, release declare
Co-occurrence words of synset B:
Discussion, suggestion, inquest
Co-occurrence words of synset A:
policy, reform, government
Low Similarity
High Similarity
Original sentence:
The government announced an economic stimulus.

Fig. 5 Identifying sense of word.

between concepts are represented by the tree structure of concepts in WordNet. Words that fit into the same concept are grouped, and each word group is called a synset. A distance of 2 synsets on the tree structure is defined as the distance of 2 synsets.

In general, a word has multiple meanings. When a word has multiple meanings, a word is registered on multiple synsets in WordNet. First, we identify the synset of a word by using the context (surrounding text) of original articles. To identify the synset, we created a co-occurrence word dictionary of synsets. We collected co-occurrence words for each synset by using the co-occurrence word database provided by ALAGIN Forum[*3]. We calculate the cosine similarity between words that appear around the keyword in the original articles and the co-occurrence word of each synsets where the keyword is registered by using that dictionary. The synset with the highest cosine similarity in the cosine similarity for all synsets is considered as the synset representing the meaning of the keyword used in the original article. **Figure 5** shows identifying the synset of the Verb word "announce" in Fig. 3 as an example. There are the synsets A and B containing "announce." The phrase around the keyword is "The government announced an economic stimulus)." The phrase and the occurrence words of synset A include "government." We calculate the cosine similarity between the words extracted from the phrase and the occurrence word set each synset. The similarity related to synset A is high and synset A is considered as the synset representing the sense of "announce"used in the phrase.

After the synset of keywords is identified, we calculate relatedness between the pair of synsets. Relatedness between synset A and synset B, which are the synsets of two given keywords, is calculated using the three method as follows.

Method (A) Focusing on Only The Same Synsets

If A and B are the same, the relatedness is 1; otherwise it is 0.

Method (B) Forcusing on Sibling Synsets

Only when the parent synset or ancestor synset (of distance 2) of both A and B is the same, relatedness between synsets A and B is 1; otherwise the relatedness is 0.

Method (C) Calculating The Distance of The Synset

We use Wu and Palmer's semantic distance method [20]. In this method, relatedness is the distance between synsets on the tree structure of WordNet. First, the common ancestor synset of both synsets A and B is detected. Then, the relatedness is calculated as follows.

$$relatedness = \frac{2 * c}{a + b + 2 * c} \tag{1}$$

where, the distance between the ancestor synset and synset A is a. The distance between the ancestor synset and synset B is b, and the distance between the ancestor synset and the root synset is c.

We clarify which method is most suitable for the incremental construction of causal networks in Section 5 based on the experimental results. After the relatedness between keywords in SVO tuple pairs is calculated, the relatedness D-rel between SVO tuple E_{SVO_a} and E_{SVO_b} is calculated as follows.

$$\begin{aligned}\text{D-rel}(E_{SVO_a}, E_{SVO_b}) = \alpha * relatedness(S_a, S_b) \\ + \beta * relatedness(V_a, V_b) \\ + \gamma * relatedness(O_a, O_b)\end{aligned} \tag{2}$$

where, α, β and γ are weight parameters ($\alpha + \beta + \gamma = 1$).

4.3 Merging Vertices

It is necessary to merge similar vertices to obtain the chain of causal relations. **Figure 6** is an example of merging vertices. In the example, by means of a merging operation we can obtain the chain of causal relations from the news story about the aim of a recovery in employment to the story about the tax cut program announced by the government.

We calculate the relatedness scores between the SVO tuples of each vertices pair. The vertices pair is merged in descending order of the relatedness scores while it exceeds the threshold. We delete vertices that are merged and create a new vertex representing the event represented by them. Edges connected to vertices that are merged re-connect to the new vertex. The SVO tuple of the new vertex is given the SVO tuple of either original vertex. If

[*3] Advanced language information forum (ALAGIN): http://www.alagin.jp/

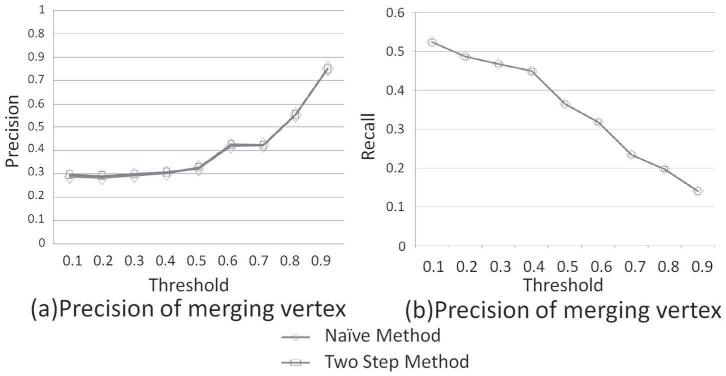

Fig. 7 Precision and recall of merging similar vertices.

the new vertex and another vertex have high relatedness after that, we will create a new vertex whose SVO tuple is given as that of the former new vertex. This is because the SVO tuple of vertices merged first represents the event more relevantly (recall that we merge vertices in descending order), and we consider that to be more justifiable keywords that represent the event.

5. Experiment

5.1 Evaluation for the TEC Model

In our causal network construction method based on the TEC model, vertices are grouped per topic. The number of calculations done to detect similar vertices for each topic is less than the number of calculations for all vertices. However, when there are similar event vertices across the topic, we may miss them as similar event vertices. We evaluated vertices merging whether we miss pairs of vertices representing similar events by restricting the calculation for topic using topic keywords. As the baseline, we use a naïve method which calculates the similarity of event keywords (keywords contained in the details) based on the vector space model and detects similar event vertices among all vertices. As one variation of our proposed method, a two-step method is used for the experiment. In the two-step method, we first group the news articles into topics and extract causal relations per topic. Then, to merge vertices, we compute the similarity of event keywords as the same as that of the naïve method. We compared the naïve method and the two-step method.

We used 7 topics, 60 articles in Japanese. These articles were collected from the economic section of Google News and were collected on ten days from January 11, 2009 to January 20, 2009. We extracted 43 causal relations, 86 event vertices based on the TEC model from the sets. We search the event vertices for the pairs of vertices which represent similar events and 107 pairs of vertices are judged to represent similar events by a user. The precision of detecting similar vertices is defined as in Eq. (3).

$$Precision = \frac{\begin{array}{l}\#\ of\ \ similar\ vertex\ pair\ judged\ by\ human\\ in\ vertex\ pairs\ merged\ by\ system\end{array}}{\#\ of\ \ vertex\ pairs\ merged\ by\ system} \qquad (3)$$

Recall of detecting similar vertices is defined as in Eq. (4).

$$Recall = \frac{\begin{array}{l}\#\ of\ \ similar\ vertex\ pair\ judged\ by\ human\\ in\ vertex\ pairs\ merged\ by\ system\end{array}}{\#\ of\ \ similar\ vertex\ pair\ judged\ by\ human} \qquad (4)$$

Figure 7 shows the accuracy and the recall of detecting similar event vertices. In Fig. 7, the threshold means the threshold for the relatedness for topic similarity (described in Section 4.2.1). In Fig. 7, there is almost no difference in the results of these two methods. The result shows that the grouping operation of topics reduces the time of calculation of detecting similar vertices without decreasing the accuracy and the recall.

When the process of detecting similar vertices checks all vertices, the number of calculations N is represented as Eq. (5).

$$N = V \cdot (V - 1)/2 \qquad (5)$$

where, V represents the number of vertices. When the similar vertices detection is performed only for vertices between similar topic, the number of calculations is the sum of the comparison time of each topics using the Eq. (5). When there are enough numbers of topics S in the vertices, the rough number of calculation N' is represented as Eq. (6).

$$N' = S * (V/S) \cdot (V/S - 1)/2 \approx N/S \qquad (6)$$

Since there are many topics in the articles, S is a large number and the topic grouping method is effective for constructing causal networks. As a case study, we describe the number of calculations for detecting similar event vertices on December 1, 2010. We extracted 455 causal relations and 910 event vertices. When similar vertices detection is performed for all vertices, the number of calculations times is about 410,000. When similar vertices detection is performed between similar topics, the number of calculations times is reduced to about 7,600. That is to say, the time taken to detect similar vertices will be reduced by the grouping operation for similar topics.

5.2 Evaluation of Similar Vertices Detection

We carried out the experiment of calculating relatedness between SVO tuples to evaluate the method for detecting similar event vertices. We compare the three methods described in Section 4.2.2 and a baseline method. In the experiment, as the baseline, we use a naïve method that merges vertices on the basis of the similarity of keywords (keywords contained in the details of an event) based on the vector space model. We use three related article sets (101 articles) collected from the economics section of Google News from July 31 to August 5, 2010. 45 causal relations were extracted from them. These articles reported on the BP oil

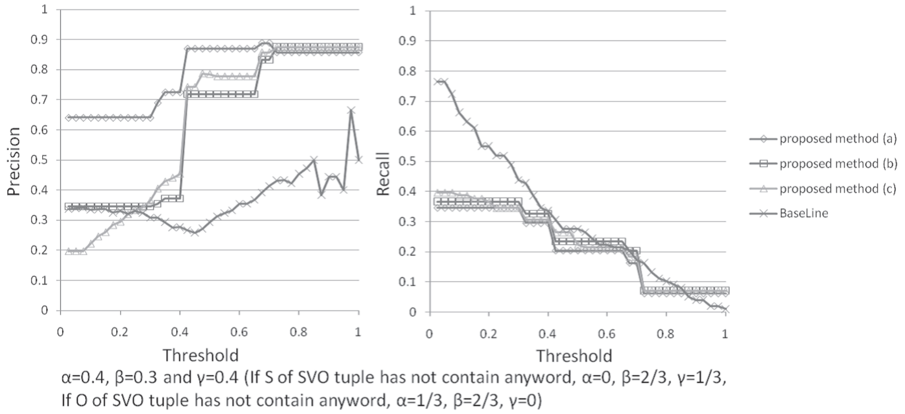

Fig. 8 Precision and recall of detecting similar vertices (the parameter of Verb is heavy).

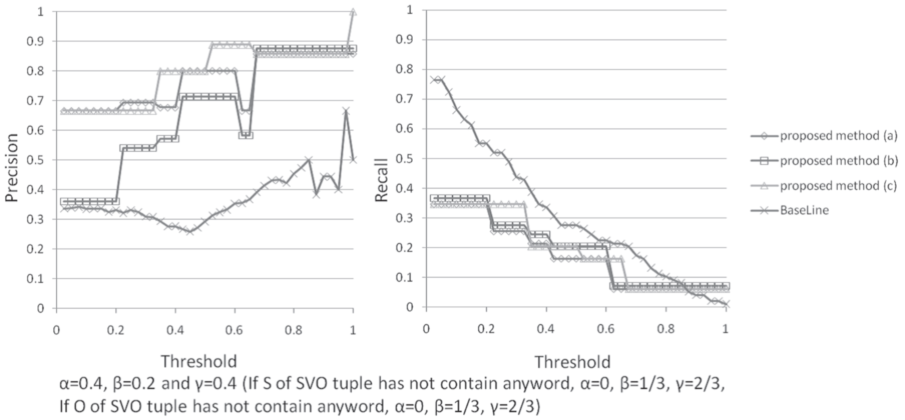

Fig. 9 Precision and recall of detecting similar vertices (the parameters of Noun are heavy).

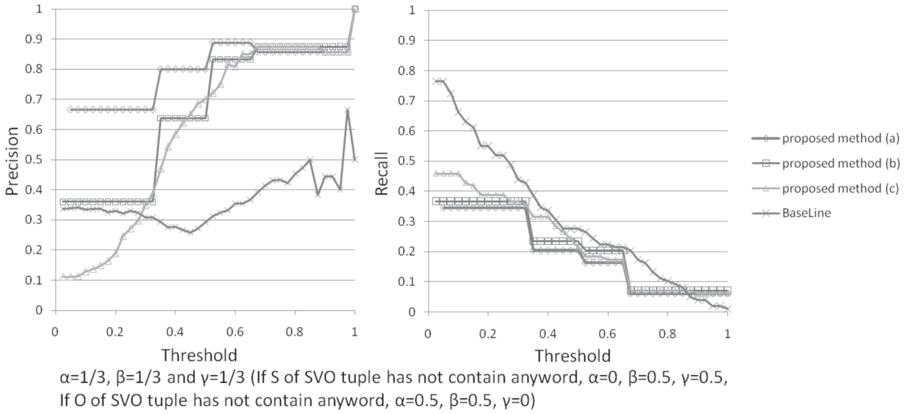

Fig. 10 Precision and recall of detecting similar vertices (the parameters have same weight).

spill. We searched the event vertices for the pairs of vertices that represent similar events in order to create the relevant set. In the set, 95 pairs of vertices were judged to represent similar events.

The system calculated relatedness between each SVO tuple pair in event vertices on each topic by using the three methods. We experimented by the parameters in 3 cases for Eq. (2): the parameter of Verb β is heavy; the parameter of Noun α and γ are heavy; the parameters α, β and γ have the same weight.

The precision and recall of detecting similar vertices are defined in Eq. (3) and Eq. (4). The results are shown in **Fig. 8**, **Fig. 9** and **Fig. 10**. In these figures, the threshold means the threshold for the relatedness for SVO tuples (described in Section 4.2.2). The results shows that the recall score of the proposal methods is lower than that of the baseline method. However, the precision score of the proposal methods is higher than that of the baseline method. In constructing a causal network, the precision score is more important than the recall score. This is because the incorrect merging makes an incorrect chain of causal relations and one incorrect merging damages the correctness of all the networks. On the whole, our proposed methods are more precise than the baseline method.

In the experimental results on the 3 parameters, there are no big differences. The value of method A is steady and good at each parameter experiment. It shows that method A is a better choice.

We noticed that some SVO tuples do not represent the origi-

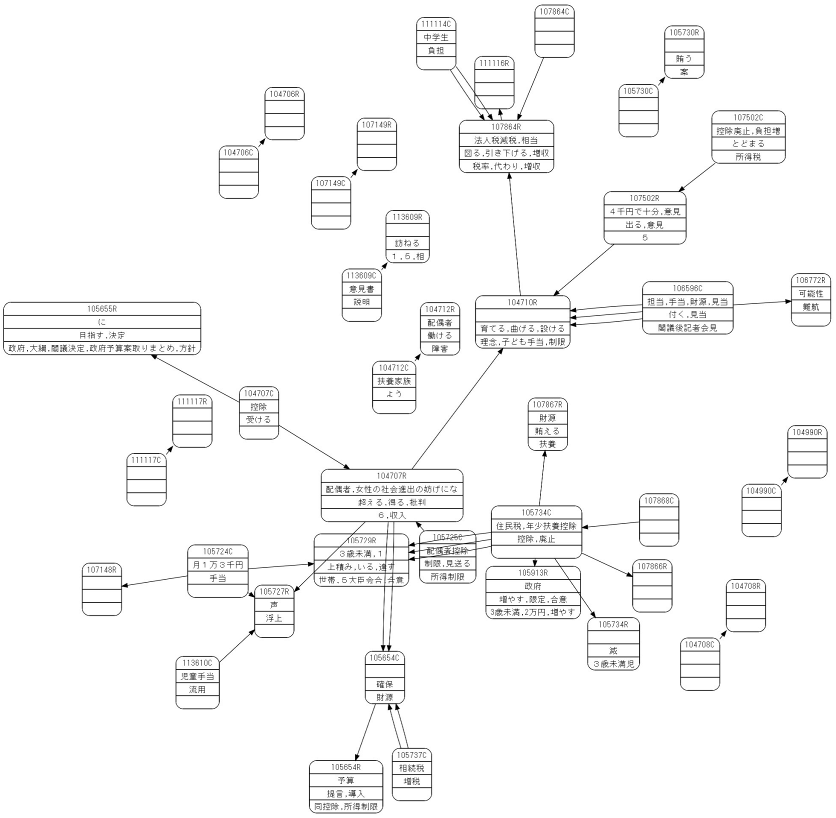

Fig. 11 Constructed causal network.

nal event precisely. When the original phrase includes a negative word, these kinds of SVO tuples will be extracted. For example, from both "Oil spilled" and "Obama did not spill" the same SVO tuple will be extracted. As a solution, extending keywords of an event vertex is considered. That is to say, we will add negative words into the SVO tuple when they appear in the causal phrase.

5.3 Evaluation of Incremental Construction of Causal Networks

We constructed causal networks incrementally using articles collected from the politics section of Google News from December 1 to 31, 2010. First, we grouped topics based on the proposal method. We chose the topic of "child benefit payments" for our evaluation. In this topic, there are 110 news articles and 42 causal relations. **Figure 11** shows the constructed network. Each vertex is represented by a vertex ID and SVO tuples.

To evaluate the construction method for causal networks, we created a relevant causal network manually. First, we searched the event vertices for similar event vertices. The similar event vertices were assigned the same label. The relevant network had 33 vertices and 41 edges. We calculated the graph edit distance [16] between the networks created by the system and the relevant network. The smaller the value of the graph edit distance, the more similar the two networks. Each vertex of the network created by the system is given the label that is given to the relevant network. **Table 1** shows their results. In Table 1, the threshold means the threshold for the relatedness for SVO tuples (described in Section 4.2.2). We consider that similar events are detected precisely and construct the causal network reliably when the threshold is high. We consider that many chains of causal relation are found when the threshold is low, but we need to delete the wrong chain of that manually. When the threshold is 0.4, our method achieves the best performance.

Table 1 The graph edit distance.

Threshold	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Edit distance	20.5	13.5	17.0	18.5	23.5	25.0	25.0

Table 2 The graph edit distance.

Method	Constructed network on incremental manner	Constructed network on batch manner
Edit distance	13.5	16.5
Construction time	163 sec	376 sec

We also constructed a causal network in the batch manner where we detect similar vertices from all vertices at once. **Table 2** shows the results of values of the edit distance and the time it took to construct the network in the proposal method and the batch manner. In the result, the construction time of the incremental manner is less than that of the batch manner. This is not associated with the reducing computational time by the topic clustering (Section 5.1), because articles used for this experiment represented just one topic about "child benefit payments." Since the merge calculation is performed everyday in this experiment, the construction time of the incremental manner has a low score. In event vertices extracted by articles reported the same day, there may many similar events. After similar event vertices reported in the same day are merged, vertices will decrease. Then we detect similar event vertices across days. This makes the construction time of the incremental manner a low score. The results show that our incremental construction method shortens the construction time and makes causal networks more relevant.

6. Conclusion

We proposed a TEC model and an incremental construction method based on it in order to construct causal networks from news events. The experimental results of similar vertices detection show that our proposal method more precisely detects similar vertices by using the roles and the concepts of keywords. The experimental results of constructing a causal network show that our proposal method reduces the time taken to detect similar vertices and makes causal networks more relevant.

Further study on the construction of the causal network is necessary. The precision of the constructed causal network depends on the efficiency of the merging similar events method. In this paper, we propose the method especially targeting incremental construction. We will compare our method with other previous works that are not incremental methods to show the effectiveness on the merging precision. The time-series features of events will also be considered to improve the construction of the causal network. The correlation coefficient between cause and result events will also be studied in future work. How to create a large data set for evaluation is a kind of future work.

Reference

[1] Aono, H. and Ohta, M.: Construction of a Causal Network by Searching Factors (in Japanese), IPSJ SIG Technical Report (IPSJ SIG Technical Report), pp.1–8 (2009).

[2] Aono, H. and Ohta, M.: Construction of a Causal Network and Acquisition of Causal Knowledges by Searching Factors (in Japanese), *DEIM Forum 2010* (2010).

[3] Feng, A. and Allan, J.: Finding and linking incidents in news, *Proc. 16th ACM Conference on Conference on Information and Knowledge Management*, pp.821–830 (2007).

[4] Girju, R.: Automatic detection of causal relations for Question Answering, *Proc. ACL 2003 Workshop on Multilingual Summarization and Question Answering*, Vol.12, pp.76–83 (2003).

[5] Inui, T., Inui, K. and Yuji, M.: Acquiring Causal Knowledge from Text Using the Connective Marker tame (Natural-Language Processing) (in Japanese), *ACM Trans. Information Processing Society of Japan*, Vol.45, No.3, pp.919–933 (2004).

[6] Ishida, S., Ma, Q. and Yoshikawa, M.: Analysis of news agencies' descriptive feature by using SVO structure, *Proc. 4th International Conference on Digital Information Management, 2009* (*ICDIM 2009*), pp.1–8 (2009).

[7] Ishii, H., Ma, Q. and Yoshikawa, M.: An Incremental Method for Causal Network Construction, *Proc. Web-Age Information Management 11th International Conference* (*2010*), pp.495–506.

[8] Ishii, H., Ma, Q. and Yoshikawa, M.: Causal Network Construction to Support Understanding of News, *Proc. Hawaii International Conference on System Sciences*, pp.1–10 (2010).

[9] Jiang, J.J. and Conrath, D.W.: Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy, *Proc. International Conference on Research in Computational Linguistics*, pp.19–33 (1997).

[10] Saruladha, K. Aghila, G. and Raj, S.: A Survey of Semantic Similarity Methods for Ontology Based Information Retrieval, *Proc. 2nd International Conference on Machine Learning and Computing*, pp.297–301 (2010).

[11] Lee, L.: Measures of Distributional Similarity, *Proc. 37th Annual Meeting of the Association for Computational Linguistics*, pp.25–32 (1999).

[12] Lin, D.: An Information-Theoretic Definition of Similarity, *Proc. 15th International Conference on Machine Learning*, pp.296–304 (1998).

[13] Liu, P.Y., Zhao, T.J. and Yu, X.F.: Application-Oriented Comparison and Evaluation of Six Semantic Similarity Measures Based on Wordnet, *Proc. 2006 International Conference on Machine Learning and Cybernetics*, pp.2605–2610 (2006).

[14] Patwardhan, S. and Pedersen, T.: Using WordNet-based Context Vectors to Estimate the Semantic Relatedness of Concepts, *Proc. EACL Workshop Making Sense of Sense: Bringing Computational Linguistics and Psycholinguistics Together*, pp.1–8 (2006).

[15] Resnik, P.: Using Information Content to Evaluate Semantic Similarity in a Taxonomy, *Proc. 14th International Joint Conference on Artificial Intelligence*, pp.448–453 (1995).

[16] Riesen, K. and Bunke, H.: Approximate graph edit distance computation by means of bipartite graph matching, *Image Vision Comput.*, Vol.27, pp.950–959 (2009).

[17] Sakaji, H., Sekine, S. and Masuyama, S.: Extracting Causal Knowledge Using Clue Phrases and Syntactic Patterns, *Proc. 7th International Conference on Practical Aspects of Knowledge Management* (*PAKM2008*), pp.111–122 (2008).

[18] Sato, H., Kasahara, K. and Matsuzawa, K.: Rertrieval of Simplified Causal Knowledge in Text and its Application (in Japanese), Technical report of IEICE, Thought and Language, Vol.98, pp.27–32 (1999).

[19] Sato, T. and Horita, M.: Assessing the plausibility of inference based on automated construction of causal networks using web-mining (in Japanese), *Sociotechnica*, pp.66–74 (2006).

[20] Wu, Z. and Palmer, M.: Verb semantics and lexical selection, *Proc. 32nd Annual Meeting of the Association for Computational Linguistics*, pp.133–138 (1994).

Hiroshi Ishii was born in 1986. He received his master's degree from Department of Social Informatics, Graduate School of Informatics, Kyoto University in 2011. He has been working at TOSHIBA Corporation since 2011.

Qiang Ma received his Ph.D. degree from Department of Social Informatics, Graduate School of Informatics, Kyoto University in 2004. He was a research fellow (DC2) of JSPS from 2003 to 2004. He joined National Institute of Information and Communications Technology as a research fellow in 2004. From 2006 to 2007, he served as an assistant manager at NEC. From October 2007, he joined Kyoto University and has been an associate professor since August 2010. His general research interests are in the area of databases and information retrieval. His current interests include multimedia database, Web mining and multimedia information retrieval.

Masatoshi Yoshikawa received his B.E., M.E. and Ph.D. degrees from Information Science from Kyoto University in 1980, 1982 and 1985, respectively. From 1985 to 1993, he was with Kyoto Sangyo University. In 1993, he joined Nara Institute of Science and Technology as an associate professor of Graduate School of Information Science. Currently, he is a professor of Graduate School of Informatics, Kyoto University. His current research interests include XML information retrieval, databases on the Web, and multimedia databases. He is a member of ACM, IPSJ and IEICE.