Causal Network Construction to Support Understanding of News

Hiroshi ISHII
ishii@db.soc.i.kyoto-u.ac.jp

Qiang MA
qiang@i.kyoto-u.ac.jp

Masatoshi YOSHIKAWA
yoshikawa@i.kyoto-u.ac.jp

Department of Social Informatics,
Graduate School of Informatics, Kyoto University
Honmachi, Yoshida, Sakyo-ku, Kyoto, 606-8501, Japan

Abstract

To support understanding of news, we propose a novel TEC model (Topic-Event Causal relation model) and describe the method to construct a Causal Network in the TEC model. The model includes two types of keywords to represent causal relations: topic keywords, which describe topics, and event keywords, which describe events. In the TEC model, causal relations are represented by an edge-labeled directed graph. A source vertex represents the cause of an event, and a destination vertex represents the result of that event. Each vertex contains event keywords and topic keywords and an importance score for each keyword. The edge label is the importance score of that causal relation. To construct a causal network, we extract causal relations from articles based on ‘clue phrases’, merge similar event vertices and reduce the size of the causal network based on the importance score of each causal relation. Preliminary experiments to assess the validity of the proposed method demonstrated its usefulness.

1. Introduction

Fully understanding an event reported in a television news program, newspaper, or a web page requires background knowledge of the event. Background knowledge is particularly important in understanding an event that is complicated or at a delicate stage. Without such knowledge, we can gain only a superficial or even false understanding of the event. To obtain the necessary background knowledge, users can search other articles or TV News programs related to the event one by one. However, this can be a burden and users may still miss valuable components of knowledge. A system that provides background knowledge of a news event is needed to improve our understanding of news.

As one kind of background information, the causal relation, i.e. the reason why the event happened, is crucial. Numerous technologies have been developed to extract causal relations from news articles and documents [3, 5, 7, 8, 9, 10]. However, for most, processing is performed per article and too much fragmented knowledge is extracted to facilitate understanding. Extracting useful background knowledge presents difficulties. To use causal relations as a basis for background knowledge requires, we need the whole causal relation related to the event, the root cause of the event, and other events affected by the event. To achieve this aim, we propose a method of constructing a causal network by merging similar causal relations. Moreover, we propose a method for deleting unimportant causal relations.

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. We call the vertices on the causal network event vertices. To merge similar causal relations and reduce the causal network, we propose a model of the causal network for understanding background of the news called a TEC model (Topic-Event Causal model). An event vertex consists of sets of important words extracted from the original article. In most of conventional work [7,8,9], keywords for an event vertex are extracted from sentences that include causal relations. The keywords are obtained by using the Japanese Dependency Structure Analyzer and the case-frame dictionary. These keywords are not weighted. In these methods, since keywords are only extracted from sentences representing causal relations, and words judged to constitute prolixity are eliminated, the event vertex only has two or three keywords. In this state, there are too few keywords, which mean that:

1. we can’t understand what the event means well, and
2. we can’t specify the original event from the event vertex

without topic information. For example, one vertex consists of the extracted keyword “kanrei(customary)” [10], but the vertex does not describe what ‘customary’ is. Moreover, if there are vertices that only have the keyword “kanrei(customary)”, these vertices are judged to be similar, even though this is not necessarily correct. To solve this problem, we extended the types of keyword used and propose a novel TEC (Topic-Event Casual) model.

In the TEC model, an event vertex contains the following two types of keywords and their importance scores:

- event keywords: Event keywords are extracted from an effect phrase or a cause phrase in news and represents what happened as an event including a causal relation.

- topic keywords: Topic keywords extracted from news titles and represent the subject of news.

To reduce the causal network, we calculate the importance score of the causal relation. In the TEC model, an edge has a label, which is the importance score of the edge. This score is calculated from the number of similar causal relations. Using this score, we calculate the importance of the relation. The causal network is constructed in three steps:

1. the system extracts causal relations from articles based on ‘clue phrases’ in Japanese, such as “Wo Haikei ni (background)”\(^1\), and “Tame (for the sake of)” ,
2. the system merges similar event vertices and
3. the system reduces the causal network based on the importance score of each causal relation.

We extract the causal network from news articles every day and incrementally merge a new network with the existing one. That is, we construct the causal network in an incremental manner and this process is shown in Figure 1.

In this paper, we describe our implementation of the system based on the proposed method and evaluate the method of constructing the causal network. The experimental results showed that the TEC model produced event vertices that facilitated understanding and improved the accuracy of the merging process.

The remainder of this paper is organized as follows. Section 2 introduces related work. Section 3 describes the TEC model. Section 4 introduces the method of constructing the causal network. The experimental results are described in Section 5 and we conclude this paper in Section 6.

\(^1\)In this paper, we describe a Japanese term and its English translation in the form of “Japanese (English)”.

2 Related Work

2.1 Method of Extracting Causal Relations

Various methods have been proposed to extract causal relations from Japanese documents, including using the joint label “tame(because)” [5]; using a case frame dictionary [8, 9, 10]; and using ‘clue phrases’, in which a sentence represents a causal relation, and structural patterns [7]. In contrast, to extract causal relations from English documents, the concept of word plays an important role[3]. In the research using a joint label [5], the authors focus on the word “tame(because)”, which is often used in sentences representing causal relations. They extract causal relations only from complex sentences. In the research using a case-frame dictionary [8, 9], if a sentence has two or more case frames, and there is a connection relation, the keywords of the causal relation are extracted from the sentence. Focus-
ing on the concepts of a word, Girju[3] proposes a casual relation extraction method that uses WordNet as a large list of semantic hierarchies the words to make the rules of extracting causal relations using the extractive patterns among the verb and the concept of subject, the concept of object. When using the case-frame dictionary method [8, 9], important keywords are extracted from both the phrase representing the cause event and the phrase representing the result event.

In several of the methods [8, 9, 10], the causal relation is extracted from only complex or compound sentences. However, in other methods [7], the causal relation is also extracted from other types of sentences. In our work, we extracted causal relations using the case-frame dictionary method [7].

2.2 Construction of Causal Network Model

By using the method of extracting the causal relations, various methods [8, 9, 10] of constructing a causal Network are approached. In the other related research described in [2], the methods of constructing a relation network of news are approached by event threading. In case-frame dictionary methods [8, 9], a causal network is constructed from documents to form a causal network of common knowledge. In other research [10], a causal network is constructed from web pages. However, in conventional approaches, the similarity of event vertices is not easy to compute because keywords are obtained only from causal phrases. Hence, merge and reduction of vertices becomes difficult. To solve this problem, we extended the types of keyword used.

In the research described in [10], there is no method of merging the vertices, and the distance between each pair of vertices represents the similarity between them. However, if the network has a lot of vertices, there is difficulty in finding the chain of causal relations. In our research, we represent the chain of causal relations by merging vertices.

In methods based on the case-frame dictionary [8, 9], a thesaurus dictionary is used to combine keywords that provide similar concepts. When keywords of different vertices are similar in concept to each other, these vertices are judged to be merged. This is considered to be their purpose which is to construct a network of common knowledge. In our research, we estimate the similarity of vertices not by fully matching keywords but by the similarity of keyword vectors. In case-frame dictionary methods [8, 9], there is no method of organizing a causal network. However, if the causal network is complex, users have difficulty in understanding it. An example of a network constructed from web documents [10] is shown in Figure 2. This network is too complex. In our research, we used the importance score of the edge to reduce the number of causal relations and organize the causal network.

3 TEC model

We propose a TEC model to represent and construct network of causal relations, i.e., a relation between a causal event and a result event. Within the TEC model, causal relations are represented by an edge-labeled directed graph. A source vertex expresses the cause event of that causal relation, and a destination vertex expresses the result event of that causal relation. The label of an edge is the importance score of that causal relation. An edge-labeled directed graph $G$ consists of vertices $V$, edges $E$ and labels of edges. Graph $G$ has a mapping as

$$E \rightarrow V \times V$$

and is expressed by $G := (V, E, f)$. Each edge $e$ is expressed as follows.

$$e := (v_s, v_t, h_e)$$

$v_s$ is a source vertex, $v_t$ is a destination vertex, and $h_e$ is the label denoting the importance score of that edge. The importance score of an edge is determined by the frequency of causal relations. When a causal relation is extracted, the default importance score of the edge (causal relation) is 1. Intuitively, the score becomes larger when the frequency of the causal relation is high. The method of calculating the importance score of an edge is described in Section 4.3.

In the TEC causal relation model, one event vertex has event keywords, topic keywords, and an importance score for each keyword. Event keywords represent an event, and topic keywords represent a news topic. Event vertex $v$ is a...
set of the pair of a keyword $k$ and the score of the keyword $w_k$ as

$$ v := \{(k, w_k)\} + $$

Since there are two kinds of keywords (an event keyword $k_e$ and topic keyword $k_t$), the event vertex $k$ is represented as

$$ v := \{(k_e, w_{k_e})\} +, (k_t, w_{k_t})\} + $$

‘+’ means that the number of keywords is 1 or more. Figure 3 shows the example of representing a causal relation in the model. The calculation method used to obtain the importance score of a keyword is described in Section 4.2. Next, $k_e$ and $k_t$ are described.

(a) Event keyword $k_e$. An event vertex has one or more event keywords representing an event. Event keywords are words extracted from phrases including causal relations.

(b) Topic keyword $k_t$. An event vertex has one or more topic keywords representing the topic of the event. Topic keywords are extracted from the title of the original article.

4 Construction of Causal Network Using TEC model

We construct the causal network using the TEC model. Causal relationships are extracted as edges/sub-graphs of the model; similar vertices are combined and unnecessary causal relations are deleted in the network constructing on process.

Using ‘clue phrases’ (in Japanese), causal relations are extracted from articles and event vertices and edges are created. Based on the similarity between a pair of vertices, vertices that represent a similar event are merged. The importance score of edges is used to reduce the number of causal relations and thus the size of the causal network.

4.1 Collecting news articles

We collect news articles in preparation for the construction of causal relation network. We use “Google News” [4] for collecting articles. In Google news, news articles are grouped into topic. We collect the news articles per topic. In the processes of vertices merging and incremental construction of network are performed inside of a topic and among different topics, perspectively.

4.2 Extracting Causal Relations

The process of extracting causal relations to create event vertices is shown as follows.

1. Extracting causal relations from news articles

2. Forming event vertices and edges

   (a) Extracting event keywords

   (b) Extracting topic keywords

3. Calculating the importance score of keywords

Figure 4 shows a causal relationship extracted from an article and the associated event vertices.

4.2.1 Extracting Phrase Representing Causal Relation from Articles

To extract causal relations from articles, we used the method based on clue phrases (in Japan) and syntax patterns [7]. A clue phrase is a phrase in a sentence that represents a causal relation such as “tame (because)” and “wo haikei ni (behind)”. We search articles for clue phrases, and sentences that include clue phrases are judged to include causal relations. Using Cabocha [6], which is a Japanese dependency structure analyzer, we classify extracted sentences into four kinds of syntax patterns.

Pattern A: Both the predicate and subject are effect phrases in a sentence.

“[cause phrase] no tame, [subject of effect phrase] ga
[predicate of effect phrase]sita. (Because of [cause phrase], [subject of effect phrase] has [predicate of effect phrase].)”
Pattern B: An effect phrase appears after a cause phrase in a sentence.

“[cause phrase] no tame, [effect phrase] shita. (Because of [cause phrase], it has [effect phrase].)”

Pattern C: An effect phrase is the sentence just before a sentence including a clue phrase.

“[effect phrase] shita. [cause phrase] no tame da. (It has [cause phrase]. This is because [cause phrase].)”

Pattern D: A result phrase appears before a cause phrase in a sentence.

“[effect phrase] ha, [cause phrase] tame da. ([effect phrase] was because of [cause phrase].)”

Based on the grammatical features of each syntax pattern, we extract the cause phrase and result phrase from the original sentence. The main ‘clue phrases’ used in our current work are shown in Table 1.

In the example in Figure 4, the system finds the phrase representing a causal relation using the clue phrase “wo ukete(under)”. Using Cabocha, the structure specified by Pattern A, and “Beikoku no kinyukiki wo hottan to suru seikaitekina hanbaimusen(Global sluggish sales by banking crisis in U.S.) ” as the phrase representing the cause event and “Oyagaisya no Toyota jidosya(Aichi) ga 09 nenn 3 gatuki no renketsu eikyo sonkei yoso wo sengohyakuokuon no akaji ni kaho syusei sitakoto ni hure(“Parent company Toyota Motor (Aichi) revised downward connection operating-profit-or-loss anticipation of the term ended March, 09 to a 150 billion yen deficit” is spoke)” as the phrase representing the cause event can be found.

4.2.2 Constructing Event Vertices and Edges from Phrases Representing Causal Relations

As we mentioned before, a causal relation is represented by using a directed edge. The source and destination vertices represent the cause and result events, respectively. When the edge is created, the default importance score 1 of the edge is assigned and keywords for cause or result events are inserted in each vertex.

(a) Extracting event keywords $k_e$

Event keywords are extracted from the body of the article. Using Chasen [1], which is a Japanese morphological analyzer, we analyze the body of an article and extracted nouns and unknown-words as event keyword $k_e$ for further processing.

(b) Extracting topic keywords $k_t$

By using Chasen [1], we analyze the title of an article and extracted nouns and unknown-words as topic keyword $k_t$.

In the example of the event vertex of the result shown in Figure 4, “Toyota jidosya(Toyota Motor)”, “akaji(deficit)” and “kaho syusei(downward revision)”, etc. are extracted as the event keywords from the phrase representing the cause event. “See Table 1 for a full list of clue phrases.”
4.2.3 Calculating the Importance Score of Keywords

We calculate the importance score of both event keywords and topic keywords. The importance score of the keyword is used as the weight when the system merges similar vertices. We also use the importance score to remove unimportant words. In other words, if the importance score of a keyword is too low, it is deleted from the vertex. We assume that a word that appears frequently in an article is important. If the keyword appears frequently, it has a high importance score. Importance score weight is calculated using Equation 5.

\[
\text{weight}(k) = \text{ntf}(k) + \text{ttf}(k) \tag{5}
\]

where, \( \text{ntf}(k) \), which is the normalized term frequency of \( k \) in articles, is calculated by using Equation 6.

\[
\text{ntf}(k) = \frac{\text{freq}(k)}{\sqrt{\sum_{t \in T} \text{freq}(t)^2}} \tag{6}
\]

where, \( \text{freq}(k) \) expresses the frequency of the keyword in an article, i.e., the number of times the word appears. \( T \) is the word set extracted from the article. where, \( \text{ttf}(k) \) is the normalized term frequency of \( k \) in the title of articles and is calculated by using Equation 7.

\[
\text{ttf}(k) = \frac{\text{freq}'(k)}{\sqrt{\sum_{t \in T'} \text{freq}'(t)^2}} \tag{7}
\]

where, \( \text{freq}'(k) \) expresses the frequency of the keyword in the title of articles, i.e., the number of times the word appears in the article. \( T' \) is the word set extracted from the title of the article.

4.3 Vertices Merging

Here, we describe our method of merging vertices that represent similar events. Using this method, we merge similar event vertices and can thus represent a chain of causal relations. In the example of merged vertices shown in Figure 5, event vertices with the event keywords “ji-dosya(automobiles)” and “gansan(decreased production)” are similar. We merge the vertices and get a chain of causal relations.

4.3.1 Calculating Similarity of Vertices

To merge vertices, we calculate the similarity between each pair of vertices in a network. Of course, we do not calculate the similarity between the source and destination vertices of a causal relation. If the similarity is greater than a threshold value, we merge that pair of vertices. When calculating the similarity, a vertex is represented as a vector on a vector space model. The similarity is the cosine similarity. The similarity between event vertex \( v_a \) and \( v_b \) is \( \text{sim}(v_a, v_b) \) as

\[
\text{sim}(v_a, v_b) = \frac{w_1 a1 \cdot w_1 b1 + w_2 a2 \cdot w_2 b2 + \cdots + w_n an \cdot w_n bn}{\sqrt{w_1 a1^2 + \cdots + w_n an^2} \sqrt{w_1 b1^2 + \cdots + w_n bn^2}} \tag{8}
\]

\( w_a \) and \( w_b \) are the weight (importance scores of keywords calculated by using method described in Sec. 4.1.) of the keyword of \( v_a \) and \( v_b \).

4.3.2 Merging Similar Vertices

In the pair of vertices \( v_a \) and \( v_b \), when the similarity of the pair is above a pre-defined threshold value, we create vertex \( v_{a+b} \), and delete \( v_a \) and \( v_b \). \( v_{a+b} \) consists of the set of the event keywords \( K_e \) and the set of the topic keywords \( K_t \) as

\[
K_{e+a+b} = K_e \cup K_{e+b} \tag{9}
\]

\[
K_{t+a+b} = K_{t+a} \cup K_{b} \tag{10}
\]

The event keyword \( v_{a+b} \) is the union of the event keywords \( v_a \) and \( v_b \). The topic keywords of \( v_{a+b} \) is the union of the topic keywords \( v_a \) and \( v_b \). The weight of the keyword is obtained using Equation 11.

\[
\text{weight}_{a+b}(t) = \frac{\text{weight}_a(t) \ast (i + 1) + \text{weight}_b(t) \ast (j + 1)}{i + j + 2} \tag{11}
\]

where, \( i \) represents the number of original vertices that involve \( v_a \), and \( j \) represents the number of original vertices that involve \( v_b \). Using Equation 11, we compute the weight of the keyword as the average of the original weights at the time of extracting the keywords from the article.

4.4 Reducing the Causal Network

We reduce the size of the causal network to create a network that is easy for users to understand. In the reduction method, the sum of the importance scores of duplicative edges is computed and they are reduced accordingly. Based on the importance score of the edge, the system deletes causal relations with low importance and decides whether or not to show causal relations to users. We call the edges that have similar source and destination vertices duplicative edges.

\[
\begin{align*}
\mathcal{E}_1 &= \{(v_1, v_2, h_{e_1})\} \\
\mathcal{E}_2 &= \{(v_1, v_2, h_{e_2})\} \\
&\vdots \\
\mathcal{E}_n &= \{(v_1, v_2, h_{e_n})\}
\end{align*}
\]
In carrying out reduction, we first look for duplicated edges in the network. If such edges exist, we create a new edge and give the sum of the scores of the duplicated edges to the label of the new edge. The system then deletes the original edges. In the case of Equation 12, the system forms the edge $e_{k_1} + e_{k_2} + \cdots + e_{k_n}$ as in Equation 13, and reduces the edges $e_{k_1}, e_{k_2}, e_{k_n}$.

$$e_{k_1} + e_{k_2} + \cdots + e_{k_n} = (v_1, v_2, h_{e_{k_1}} + h_{e_{k_2}} + \cdots + h_{e_{k_n}})$$ (13)

### 4.5 Incremental Construction of Network

Since news items are incoming every day, we need to update the causal network incrementally. In our current work, a causal network is firstly constructed from articles reported each day. The causal network for each day is added to the network constructed the day before, and we carry out vertex merging and edge reduction for the new network. Figure 1(a) shows how we extract causal relations from new articles and makes a causal network every day ((1) (2)). Secondly, the system constructs the causal network every day by performing merging and reduction on the extracted causal network ((3) (4)). Since related news articles are organized by topic, vertex merging is performed inside of a topic.

Next, the system merges vertices and reduces edges between the causal network for each day and the network constructed previously. In the process of merge, we calculate the similarity of the topics at first. If the similarity of the topic $a$ and topic $b$ is lower than the threshold, we don’t perform the process of network merging between these topics. When calculating the similarity between topics, a topic is represented as a vector on a vector space model using topic keywords included in the articles of the topic. In this manner, we can reduce the time complexity. When the process of merging performs for all vertices, the number of the calculation $N$ is represented as Equation 14.

$$N = \frac{(T \cdot a) \cdot (T \cdot a - 1)}{2}$$ (14)

where, $T$ represents the number of topics and $a$ represents the average number of the article per topic. When the merging process performs only for vertices between similar topics, the number of the calculation $N'$ is represented as Equation 15.

$$N' = \frac{(T \cdot a) \cdot (s \cdot T \cdot a - 1)}{2}$$ (15)

where, $s$ represents the average probability that two topics are similar. In most cases, $s$ is very small, therefore $N'$ is much smaller than $N$.

In short, the network is updated by repeating extraction of causal relations, merging and reduction, as shown by the cycle in Figure 1.

### 5 Experiments

We conducted experiments to evaluate the proposed method, in particular the performance of vertex extraction and merging.

#### 5.1 Experimental Conditions

We used 7 topics, 60 articles in Japanese as follows.

1. “Toyota Motor will reduce production.” contains 8 articles
2. “The labor union of Toyota Motor demanded the wage increase.” contains 5 articles
3. “The profit of Toyota Motor fell.” contains 16 articles

5. “The reactivation policy in America is expected to be changed.” contains 2 articles.

6. “President Obama proposed the tax cut policy.” contains 10 articles.


These articles were collected from the economic section of Google News [4] and were collected from January 11, 2009 to January 20, 2009.

5.2 Evaluation of Vertex Extraction

We extracted the causal relations from the sets of articles based on the TEC model. We evaluated the extracted vertices that how well each vertex represented the original event in the article. If a vertex has the keywords which represent the original event well and the keywords which represent the topic of the original event well, we say that vertex is extracted well, and call it a correct vertex. If a vertex doesn’t have the keywords which represent the original event well or the keywords which represent the topic of the original event, the vertex is judged to be an incorrect vertex. The accuracy of the result of extracting vertices is calculated as follows.

\[
\text{accuracy} = \frac{\text{number of correct vertices}}{\text{number of all extracted vertices}}
\]  

(16)

Table 2 shows the result of extracting vertices. In the all topics, the result using both event keywords and topic keywords is the same as using event keyword only, or better than it. Two examples of the correct vertex are shown in Figure 6. In Figure 6 (a), the event keywords of the vertex don’t represent what are stopped. However, adding the word, “kojyo (factory)” in the topic keywords, the vertex represent the factory shutdown. In Figure 6 (b), the event keywords of the vertex don’t represent who demanded and what was demanded. However, adding the word, “syunto (laborer offensive)” and “denryokusouren (the federation of electric power related industry worker’s union)”, chinnage (rise in wages) in the topic keywords, the vertex represent the federation demanded the rise in wages. These results showed that the proposed method improved the accuracy of the event vertex extraction, and easier for understanding.

5.3 Evaluation of Vertex Merging

To evaluate the method of merging vertices, we calculated the similarity of pairs of vertices for 84 event vertices from the experimental article set. In the 84 event vertices, the number of pairs of vertex which are judged to represent similar events was 107 by a human. We call such pair of similar vertices a correct merging. Accuracy was defined as in Equation 17.

\[
\text{accuracy} = \frac{\text{number of correct merging}}{\text{number of system merging}}
\]  

(17)

Recall was defined as in Equation 18.

\[
\text{recall} = \frac{\text{number of correct merging}}{\text{number of all correct merging}}
\]  

(18)

Figure 7 shows the accuracy of merging vertices. Figure 8 shows the recall of merging vertices. In the experiment, we computed three kinds of similarities:

1. using only the event keywords and calculating the similarity for all vertices,
2. using only the event keywords and calculating the similarity for vertices between similar topics and
3. using the event keywords and the topic keywords and calculating the similarity for vertices between similar topics.

<table>
<thead>
<tr>
<th>Event Keyword only</th>
<th>Event Keyword and Topic Keyword</th>
</tr>
</thead>
<tbody>
<tr>
<td>Toyota Motor will reduce production</td>
<td>0.22(=4/18) 0.72(=13/18)</td>
</tr>
<tr>
<td>The labor union of Toyota Motor demanded the wage increase</td>
<td>0.33(=2/6) 1.00(=6/6)</td>
</tr>
<tr>
<td>The profit of Toyota Motor fell</td>
<td>0.19(=3/16) 0.38(=6/16)</td>
</tr>
<tr>
<td>The factory of Toyota Motor in North America suspended operation</td>
<td>0.08(=1/12) 0.50(=6/12)</td>
</tr>
<tr>
<td>The reactivation policy in America is expected to be changed</td>
<td>1.00(=4/4) 1.00(=4/4)</td>
</tr>
<tr>
<td>President Obama proposed the tax cut policy</td>
<td>0.375(=6/16) 0.56(=9/16)</td>
</tr>
<tr>
<td>The Democratic Party in America released the reactivation policy</td>
<td>0.29(=4/14) 0.57(=8/14)</td>
</tr>
<tr>
<td>Total</td>
<td>0.27(=24/86) 0.60(=52/86)</td>
</tr>
</tbody>
</table>
In Figure 7, there is almost no difference in the result of calculating the similarity for all vertices and calculating the similarity for vertices between similar topics. In this experiment, we extracted 84 event vertices. If the process of merging performs for all event vertices, the number of the similarity calculation will be $3486 = \frac{84 \times 83}{2}$. However, if we compute the similarity of topic keywords at first, we can split the event vertices into two groups to reduce time of computing event similarity. One group consists 50 event vertices about the topic of TOYOTA. The other group consists 34 event vertices about the topic of USA politics. When the process of merging performs each group, the number of the similarity calculation will be $1786 = \frac{50 \times 49}{2} + \frac{34 \times 33}{2}$. The result shows using that similarity of topics reduces computational complexity without decreasing accuracy and recall, and using topic keywords to calculating the similarity of the vertices rise the accuracy of merging vertex.

From the results, we can say that the method of merging vertices using event keywords and topic keywords is better than the method of merging vertices using event keywords only.

6 Discussion

We currently propose the method of vertices merging by using the similarity of keywords. However, its method has a disadvantage that the order of the vertices merging effects the result network. For example, as shown in Figure 9, suppose the threshold of merging is 0.65. In Figure 9(1), the
similarity between Vertices A and B is 0.79 and the similarity between Vertices A and C is 0.70. If we merge vertices A and B firstly, the vertex AB’ will be the result. Since the similarity between Vertices AB’ and C is 0.61, we will not merge vertices AB’ and C as shown in Figure 9(2). If the process firstly merges vertices A and C in first, the result vertex will be the vertex AC’ and we will have a further merging between AC’ and b as shown Figure 9(3). It’s obviously that the order of the vertices merging will effects the network construction. We will discuss this issue in near future.

7 Conclusion

We proposed a novel TEC model to represent causal relations extracted from news articles. In our TEC model, causal relations are represented as an edge-labeled directed graph, in which source and destination vertices denote the cause and result of a causal relation. In contrast to conventional causal models, which only contain keywords describing events, topic keywords are also included in our model to support construction of a causal network and facilitate users’ understanding. A network construction method that includes casual relation extraction, vertex merging, and edge reduction was developed. The experimental results showed that the proposed method improved the accuracy of casual relation extraction and vertex merging. In other words, the results demonstrated the usefulness of our TEC model. Further study on the construction of the causal network is necessary.

We will carry out more experiments and improve the merging and reduction methods based on the results. The features of concept of words will be considered in improving construction of the causal network. The time-series features of news articles will be considered to improve construction of the causal network, especially the merge function. The correlation coefficient between cause and result events will also be studied in future work.

Acknowledgment

This research is partly supported by the research for the grant of Scientific Research (No.20700084, 20300042 and 21013026) made available by MEXT, Japan. This work is also partly supported by Kyoto University Global COE Program: Informatics Education and Research Center for Knowledge-Circulating Society.

References