Electronic Preprint for Journal of Information Processing Vol.24 No.4

Regular Paper

Paraphrasing Sentential Queries by Incorporating Coordinate Relationship

Meng Zhao[1,a)] Hiroaki Ohshima[1,b)] Katsumi Tanaka[1,c)]

Received: December 20, 2015, Accepted: April 13, 2016

Abstract: Although long queries are still a small part of the queries submitted to Web search engines, their usage tends to gradually increase. However, the effectiveness of the retrieval decreases with the increase of query length. Long queries are very likely to have few Web pages returned. We target at sentential queries, a type of long queries, and propose a method called *sentential query paraphrasing* for improving their retrieval performance, especially on recall. We are motivated by the assumption that a sentence is an indivisible whole, which means that removing terms or phrases from a sentence would lead to the missing of some information or query drift. In this paper, we paraphrase sentential queries to avoid missing information and consequently ensure the completeness of the information. Take the sentential query "apples pop a powerful pectin punch," for example. Its meaning will be changed if one or more terms are removed, and few Web pages are returned by conventional search engines. In contrast, querying by its paraphrases, such as "apples contain a lot of pectin" or "apples are rich in pectin," can retrieve more Web pages. The experimental results show that our method can acquire more paraphrases from the noisy Web. Besides, with the help of paraphrases, more Web pages can be retrieved, especially for those sentential queries that could not find any answers with its original expression.

Keywords: sentential query paraphrasing, Web mining, mutual reinforcement, coordinate relationship

1. Introduction

Search engines, such as Google [*1] and Bing [*2], provide convenience for users to obtain useful information by issuing queries based on their information needs. Therefore, they have become the major gateways to the huge amount of information on the Web. Bendersky and Croft [4] stated that Web search queries are mostly short queries whose length is less than four words on average. However, it has been reported that queries of length five words or more are becoming more common, with a year-over-year rate of 10% growth, while shorter queries, averaging those one to four words in length, are becoming less common, with a 2% decrease [10]. Several studies have proved that compared to short queries, long queries can provide more information in the form of context, consequently providing a better way for conveying complex and sophisticated information needs [12], [13], [14], [18]. Therefore, long queries are used in many different applications, such as question answering (QA) search [22] and judgement of fact trustworthiness [24]. However, the effectiveness of retrieval for long queries is generally lower than that for short queries [10].

The expression rarity of long queries would be a conceivable reason why long queries, especially sentential queries, fail in retrieving any useful information. Take a Web search for example. Suppose users want to find more information about pectin in apples and think of a sentential query such as "apples pop a powerful pectin punch". None of the two aforementioned search engines return any matches for such a query (at the time of writing this paper).

Several studies [2], [3], [13], [15] have concentrated on improving the retrieval effectiveness of long queries. All are based on the assumption that long queries always contain extraneous terms. Besides, they can be broadly grouped into two categories: query reduction approach and query re-weighting approach. Query reduction is aimed at improving the performance of long queries by eliminating redundancy. Therefore, a long query is reduced to a concise version by removing one or more terms. Query re-weighting is focused on identifying important or verbose terms in long queries and assigning different weights to them.

In general, long natural language queries can be divided into two categories: sentential queries and joint phrase queries. Obviously, the former are sentences in form, such as "What is the highest mountain in Africa?", while the latter are sequences generated by several separate phrases, such as "2015 Uefa Super Cup FC Barcelona Sevilla FC Pedro score", which is joint by "2015 Uefa Super Cup", "FC Barcelona", "Sevilla FC" and "Pedro score". In this study, we target at sentential queries and focus on improving their poor performance by using other queries that convey the same meaning. We call it ***sentential query paraphrasing***. Contrary to previous assumption, we argue that separate terms or phrases from a long query may lead to the missing of some information or query drift. Neither query reduction approach nor query re-weighting can exhibit an non-disappointing

[1] Graduate School of Informatics, Kyoto University, Kyoto 606–8501, Japan
[a)] zhao@dl.kuis.kyoto-u.ac.jp
[b)] ohshima@dl.kuis.kyoto-u.ac.jp
[c)] tanaka@dl.kuis.kyoto-u.ac.jp

[*1] http://www.google.com
[*2] http://www.bing.com

performance. This is because the query itself "apples pop a powerful pectin punch" is an indivisible whole. Its meaning cannot be completely expressed by any portion of it. In this case, we rewrite the original query by its paraphrases, such as "apples contain a lot of pectin" and "apples are rich in pectin". We can obtain enough Web pages with detailed information by submitting those paraphrases to the Web and aggregating their search results.

Sentential query paraphrasing is also effective in estimating the credibility of facts. Here, we define *fact* as an item of knowledge or a piece of information. It has a variety of different expressions in the surface form. Correspondingly, a certain expression of a fact is defined as a *fact statement*. For example, there is a fact about high level of pectin contained in apples. This fact can be represented in, but not limited to, the following ways:

- *Apples are rich in pectin.*
- *Apples are a great source of pectin.*
- *Apples contain a high amount of pectin.*
- *Apples are packed with pectin.*
- *Apples are abundant in pectin.*
- *Apples have high pectin content.*

Each different way that represented the fact is a fact statement. Hence, the sentence "apples are rich in pectin" is a fact statement. We assume the credibility of a fact is high if people often mention it on the Web. Based on this assumption, a naive way to judge fact credibility is to check its occurrence on the Web. However, this trial always fails. The reason is that although there is a variety of different expressions for a fact, it might be difficult to think of these expressions as many as possible. In the most extreme case, we may only think of one expression, which leads to failure of fact credibility judgement. For example, suppose we want to estimate whether a fact is credible, but can only think of a statement like "apples are abundant in pectin". Actually, this statement is seldom used on the Web. If we judge the fact credibility only based on this statement, we would draw an erroneous conclusion that apples do not have a high amount of pectin. So it is likely to draw erroneous conclusion by only observing the occurrence of a certain fact statement. However, if we also take other statements of the fact into consideration, the ones that convey the same meaning as the given statement, it is more likely for us to come to the right conclusion. For example, other fact statements, such as "apples are rich in pectin" or "apples are a great source of pectin", are widely used on the Web. If we estimate the credibility of the fact also based on these statements, we could draw the correct conclusion that apples are a high pectin fruit.

Based on the intensional-extensional relation representation, our method finds sentential paraphrases from the noisy Web instead of domain-specific corpora. Bollegala et al. [6] stated that a relation can be defined intensionally by listing all the paraphrase templates of that relation. It can be also expressed extensionally by enumerating all the instances of it. Take the **highConcentration** relation [*3] for example. An intensional definition of **highConcentration** is described with templates, including but not limited to ***X are rich in Y*** and ***X are an excellent source of Y***. An extensional definition of **highConcentration** is a set of all pairs of a food and a certain nutrient in which the food is a rich source of the nutrient, including but not limited to *(lemons, vitamin c)* and *(apples, pectin)*. Given a sentential query, our method first extracts templates and entity tuples from the Web, respectively. During the extractions, several filters and limitations are added to eliminate partial inappropriate templates and entity tuples. Finally, a mutually reinforcing approach is used to identify different templates that convey the same meaning with the given template.

The remainder of the paper is organized as follows. We dedicate Section 2 to the discussion of the previous work on relation extraction and paraphrase acquisition. In Section 3, we define the sentential query paraphrasing problem and introduce the overview of our proposed method. In Section 4, we describe the core of our algorithm. Section 5 gives more details when adapting to the fact credibility judgement application. We explain the evaluation results in Section 6. Finally, we conclude the paper in Section 7.

2. Related Work

2.1 Semantic Relation Extraction

Snowball [1], KnowItAll [8], and TextRunner [25] are well-known information extraction systems. All of them extract valuable information from plain-text documents by using lexical-syntactic patterns.

Given a handful of example tuples, such as an organization-location tuple <*o,l*>, Snowball finds segments of text in the document collection where o and l occur close to each other, and analyzes the text that "connects" o and l to generate patterns. It extracts different relationships from the Web by using the bootstrap method.

KnowItAll is an autonomous, domain-independent system that extracts information from the Web. The primary focus with the system is extracting entities. The input to KnowItAll is a set of entity classes to be extracted, such as "capital", "movie" or "ceo", while the output is a list of entities extracted from the Web. Note that it only uses generic hand-written patterns, such as "including" and "is a".

Compared to these two systems in which relation types are predefined, TextRunner discovers relations automatically. Extractions take the form of a tuple $t = (e_i, r_{i,j}, e_j)$, where e_i and e_j are strings meant to denote entities, and $r_{i,j}$ is a string meant to denote a relationship between them. A deep linguistic parser is deployed to obtain dependency graph representations by parsing thousand of sentences. For each pair of noun phrases (e_i, e_j), TextRunner traverses the dependency graph, especially the part connecting e_i and e_j, to find a sequence of words that comprises a potential relation $r_{i,j}$ in tuple t.

As our method is based on the mutual reinforcement relationship of templates and entity tuples, we can simultaneously identify templates that convey the same meaning and entity tuples that have the same relation. Therefore, it is also possible to use our method to automatically extract entity tuples of user-indicated relations.

[*3] We define highConcentration relation as the relation between a food and a certain nutrient such that the food contains a high amount of the nutrient.

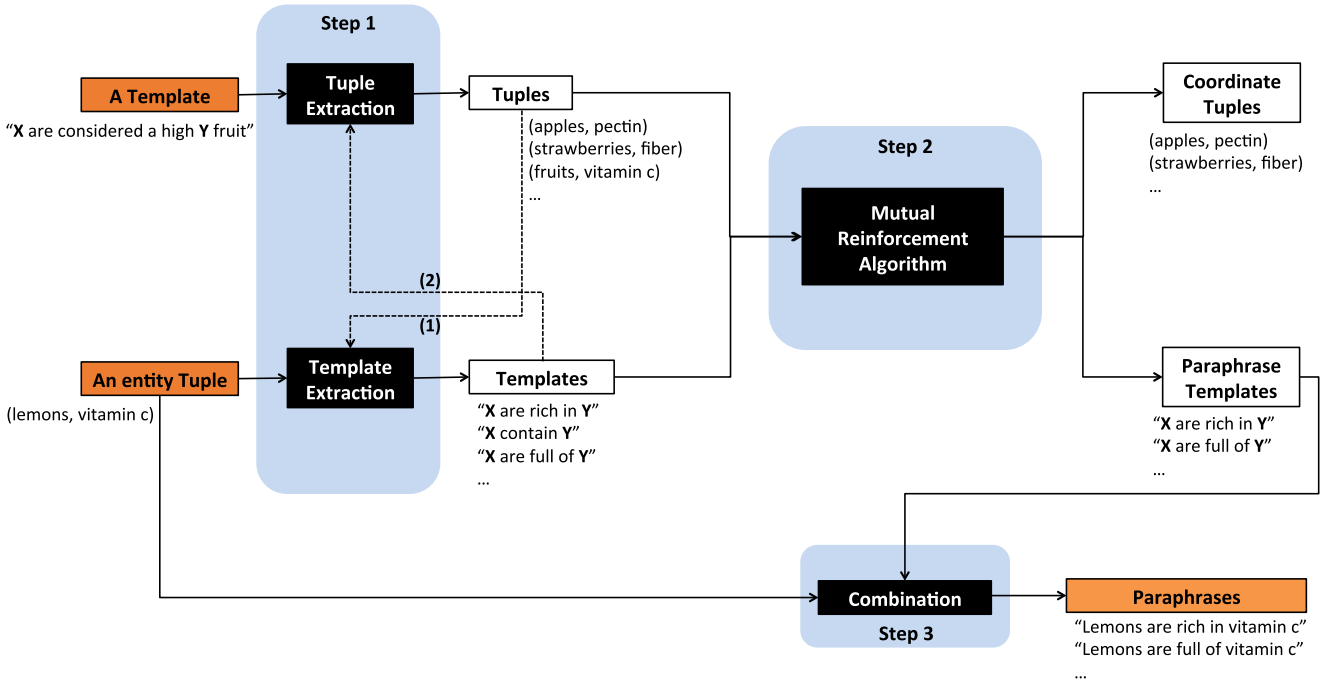


Fig. 1 Overview of the proposed method.

2.2 Paraphrase Acquisition

Paraphrase acquisition is a task of acquiring paraphrases of a given text fragment. Some approaches have been proposed for acquiring paraphrases at word, or phrasal level. However, these techniques are designed to be only suitable for specific types of resources. Shinyama et al. [19] and Wubben et al. [21] acquired paraphrases from news articles. For example, Shinyama et al. [19] argued that news articles by different news agents reporting the same event of the same day can contain paraphrases. Thus, they proposed an automatic paraphrase acquisition approach based on the assumption that named entities are preserved across paraphrases.

Paşca and Dienes proposed a different method [17]. They use inherently noisy, unreliable Web documents rather than clean, formatted documents. They assumed that if two sentence fragments have common word sequences at both extremities, then the variable word sequences in the middle are potential paraphrases of each other. Therefore, their acquired paraphrases are almost word-, or phrase-level ones, while our aim is to obtain sentential paraphrases.

Yamamoto and Tanaka [23] also concentrated on improving search results responded by sentential queries. Unlike our focus on paraphrases, they generally collected several types of sentence substitutions, such as generalized or detailed sentences. They used these substitutions to retrieve more information.

3. Sentential Query Paraphrasing Problem

3.1 Problem Definition

In this section, we give a definition to ***sentential query paraphrasing***. As we discussed in Section 1, our notion is to substitute a sentential query by its frequently used paraphrases to retrieve more answers. Therefore, in the ideal case, the problem can be described as follows:

- Input: A sentence

Table 1 Top 15 paraphrases when given the template *X are considered a high Y fruit* and the entity tuple *(lemons, vitamin c)* as the input.

1.	lemons are an excellent source of vitamin c
2.	lemons are rich in vitamin c
3.	lemons are high in vitamin c
4.	lemons are packed with vitamin c
5.	vitamin c obtained from lemons
6.	lemons have a very high vitamin c content
7.	boosts the immune system lemons are high in vitamin c
8.	lemons contain a high amount of vitamin c
9.	lemons are a rich source of vitamin c
10.	the best know natural sources of vitamin c are the citrus fruit such as lemons
11.	lemons are also sources of vitamins and minerals other than vitamin c
12.	lemons and limes help keep your skin looking its best because they're rich in vitamin c
13.	lemons are vitamin c rich citrus fruits
14.	it is no longer news that we all need to use lemons every day because of the high amounts of vitamin c
15.	lemons contain vitamin c

- Output: Sentences that convey the same meaning

However, since a sentence can be mapped by a template and an entity tuple, such as the template ***X** are considered a high **Y** fruit* and the entity tuple *(lemons, vitamin c)* can generate a sentence "lemons are considered a high vitamin c fruit", in the actual case, the problem slightly changes as follows:

- Input: A template and an entity tuple

- Output: Sentences that convey the same meaning with the sentence mapped by the input template and the input entity tuple

Table 1 lists the top 15 running results of our method, given the template ***X** are considered a high **Y** fruit* and the entity tuple *(lemons, vitamin c)* as the input.

3.2 Overview of the Proposed Method

There are three steps in our proposed method. **Figure 1** shows its overview. Black box indicates a processing, while white box

indicates the input and output for each processing. Take as an example the input of the template ***X** are considered a high **Y** fruit* and the entity tuple *(lemons, vitamin c)*.

In step 1, our method extracts candidate entity tuples from the Web through tuple extraction according to the input template. Similarly, our method extracts candidate templates from the Web through template extraction according to the input entity tuple. In some cases, we could not obtain enough candidate entity tuples or templates. Hence, several frequently appeared tuples are used to extract more candidate templates, corresponding to mark (1) in Fig. 1. In the same way, several frequently appeared templates are used to extract more candidate entity tuples, corresponding to mark (2) in Fig. 1. Note that currently, such processing is taken place only once. Finally, we obtain candidate templates, such as ***X** are rich in **Y***, ***X** contain **Y***, and candidate entity tuples, such as *(apples, pectin)*, *(strawberries, fiber)*.

In step 2, we take candidate templates and candidate entity tuples as the input for the mutual reinforcement algorithm to identify paraphrase templates and coordinate tuples at the same time. For example, ***X** are rich in **Y***, ***X** are full of **Y*** are judged as paraphrase templates of the original template ***X** are considered a high **Y** fruit*.

Finally, in step 3, we combine paraphrase templates with the input entity tuple to obtain paraphrases. Hence, we have "Lemons are rich in vitamin c" and "Lemons are full of vitamin c" as the paraphrases of "Lemons are considered a high vitamin c fruit".

4. Mutual Reinforcement between Templates and Entity Tuples

In this section, we describe the core of our algorithm, referred to the step 2 in Fig. 1.

4.1 Intensional-extensional Representation for a Relation

Bollegala et al. [6] stated that a relation can be defined intensionally by listing all the paraphrase templates of that relation. It can be also expressed extensionally by enumerating all the instances of that relation. Take the **highConcentration** relation for example. An intensional definition of **highConcentration** is described by templates, including but not limited to

- ***X** are rich in **Y***
- ***X** are an excellent source of **Y***
- ***X** are full of **Y***

An extensional definition of **highConcentration** is a set of all pairs of a food and a certain nutrient in which the food is a rich source of the nutrient, including but not limited to

- *(lemons, vitamin c)*
- *(apples, pectin)*
- *(strawberries, potassium)*

Entity tuples holding the same relation are defined as "coordinated" to each other. Therefore, *(apples, pectin)* is a coordinate entity tuple of *(lemons, vitamin c)*. Some of the terminology used in this paper is listed in **Table 2**.

It should be noted that relations are limited to binary relations. In other words, the number of entities in an entity tuple is fixed at 2.

Table 2 Terminology.

Template	**X** are considered a high **Y** fruit
Entity tuple	*(lemons, vitamin c)*
Substitution	**X**=*lemons*, **Y**=*vitamin c*
Sentence	Lemons are considered a high vitamin c fruit.
Paraphrase templates	**X** are rich in **Y**
	X are an excellent source of **Y**
	X are full of **Y**
Paraphrases	Lemons are rich in vitamin c.
	Lemons are an excellent source of vitamin c.
	Lemons are full of vitamin c.
Coordinate entity tuples	*(apples, pectin)*
	(strawberries, potassium)

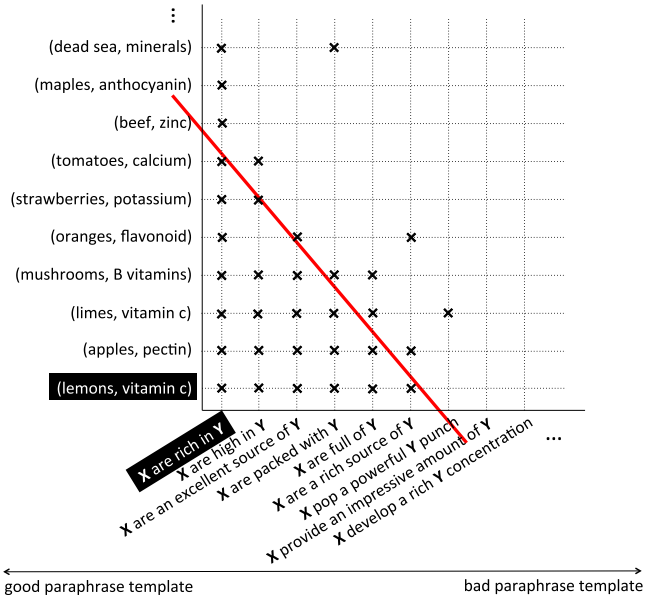


Fig. 2 Ideal case of the mutual reinforcement between paraphrase templates and coordinate tuples, with the input "Lemons are rich in vitamin c". The mark "×" indicates that people use the expression generated by corresponding template and corresponding tuple.

4.2 Relationship between Templates and Entity Tuples

We use **Fig. 2** to illustrate an ideal case of the mutual reinforcement between paraphrase templates and coordinate tuples. The input sentential query is "Lemons are rich in vitamin c", which is mapped by the template ***X** are rich in **Y*** and the entity tuple *(lemons, vitamin c)*. Suppose we have already obtained some paraphrase templates and coordinate tuples. They are plotted on the horizontal and vertical axes, respectively. The intersection of two dotted lines in the figure indicates a combination of the corresponding template and tuple. Moreover, a "×" signifies that people use the expression by this combination. For example, the meeting-point of the dotted lines of template ***X** are packed with **Y*** and tuple *(apples, pectin)* represents a possible expression. That is "Apples are packed with pectin". Since there is a "×" attached, we know people use this expression in daily life.

The intensional-extensional representation for a relation suggests the use of suitable tuples to represent the context of a template, and accordingly, the use of suitable templates to represent the context of a tuple. Intuitively, therefore, for template ***X** are packed with **Y***, tuples with "×"s generate its context, such as *(lemons, vitamin c)*, *(dead sea, minerals)* and so forth. On the other hand, for tuple *(apples,pectin)*, templates with "×"s generate its context, such as ***X** are packed with **Y***, ***X** are a rich source*

of ***Y*** and so forth. The distributional hypothesis [9] has been the basis for statistical semantics. It states that words that occur in the same contexts tend to have similar meanings. We are motivated by its extended version:

> - If two templates share more common coordinate tuples, they are more likely to be paraphrased to each other.
>
> - If two tuples share more common paraphrase templates, they are more likely to be coordinated to each other.

Thus, paraphrase templates and coordinate tuples are in a mutually reinforcing relationship.

With the aid of "×"s, Fig. 2 can be divided into two areas: dense and sparse. We use a red oblique line to symbolically separate these two areas. The closer a paraphrase template to the dense area, the better it is, and vice versa. The closer to the left, the better it is as a paraphrase template. Accordingly, the closer to the right, the worse it is as a paraphrase template. A good paraphrase template means it is more semantically similar to the original template. As a result, paraphrase templates that belong to the dense area are regarded as good paraphrase templates for ***X*** *are rich in* ***Y***, such as ***X*** *are high in* ***Y***, ***X*** *are an excellent source of* ***Y*** and ***X*** *are packed with* ***Y***. In the same way, we can identify whether templates are paraphrase templates.

4.3 Mutual Reinforcement Algorithm

In this section, we introduce the essential feature of our method. We start with a template set T and a tuple set E. The details of how to extract them from the Web according to a sentential query are addressed in Section 5. Suppose there are m templates in T and n tuples in E. At the beginning, a bipartite graph is constructed. Let $W^{TE} \in \mathbb{R}^{m\times n}$ denote the transition matrix from T to E and $W^{ET} \in \mathbb{R}^{n\times m}$ the transition matrix from E to T. The meanings of w_{ij}^{te} and w_{ij}^{et} depend on different applications.

We define the following two functions and regard them as the weight of edges between templates and the weight of edges between tuples, respectively.

- **Para**$(\mathbf{t_i}, \mathbf{t_j})$: paraphrase degree between two templates t_i and t_j, which returns a value between 0 and 1. A high value will be returned when t_i and t_j are more likely to be paraphrased to each other.
- **Coord**$(\mathbf{e_i}, \mathbf{e_j})$: coordinate degree between two tuples e_i and e_j, which returns a value between 0 and 1. A high value will be returned when e_i and e_j are more likely to be coordinated to each other.

There are two different situations when considering the paraphrase degree between t_i and t_j. One is exact equivalence of t_i's and t_j's suitable tuples, such as e_k in **Fig. 3** (a). In other words, if we can find many tuples that are shared by two templates t_i and t_j, the paraphrase degree between them is high. This situation has been often considered in previous studies of paraphrase acquisition. To improve retrieval performance, some studies involved taking semantic similarity between terms into account. This motivates us to consider semantic similarity between both templates and tuples. For example, ***X*** *are rich in* ***Y*** is semantically similar to

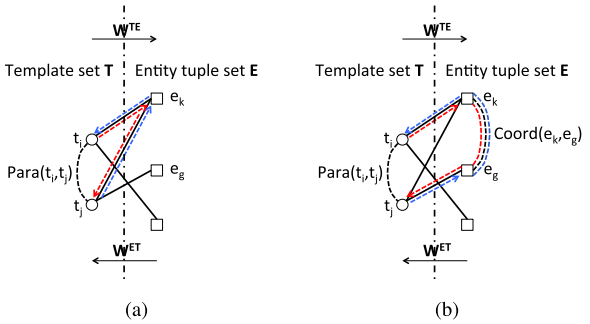


Fig. 3 Paraphrase degree calculation. Two disjoint sets T and E (vertical chain line is used to separate them) comprise a bipartite graph, where T is composed of candidate paraphrase templates and E of candidate coordinate tuples.

X *are an excellent source of* ***Y***, and *(lemons, vitamin c)* is semantically similar to *(apples, pectin)*. Therefore, in addition to the former situation, we also consider the coordinate degree between e_k and e_g, which is $Coord(e_k, e_g)$ shown in Fig. 3 (b). In other words, if we can find many coordinate tuple pairs that are shared by t_i and t_j, the paraphrase degree between them is high. As a result, the value of $Coord(e_k, e_g)$ is propagated to $Para(t_i, t_j)$ according to the transition probability. Similarly, additional values are propagated from other pairs of coordinate tuples in E to $Para(t_i, t_j)$, then the value of $Para(t_i, t_j)$ is updated. The new value is propagated to $Coord(e_k, e_g)$ in two similar situations. Since the edges between T and E are directional, we use different colors for distinguishing in Fig. 3. For example, when considering the paraphrase degree between t_i and t_j, we consider both the routes from t_i to t_j (shown in red in Fig. 3) and the routes from t_j to t_i (shown in blue in Fig. 3).

Formally, the mutually reinforcing calculations are written as:

$$Para(t_i, t_j) = \frac{1}{2}\left(\sum_{e_k, e_g \in E} w_{ik}^{te} w_{gj}^{et} Coord(e_k, e_g) + \sum_{e_k, e_g \in E} w_{jg}^{te} w_{ki}^{et} Coord(e_k, e_g) \right) \tag{1}$$

$$Coord(e_k, e_g) = \frac{1}{2}\left(\sum_{t_i, t_j \in T} w_{ki}^{et} w_{jg}^{te} Para(t_i, t_j) + \sum_{t_i, t_j \in T} w_{gj}^{et} w_{ik}^{te} Para(t_i, t_j) \right) \tag{2}$$

where $i, j \in [1, m]$ and $k, g \in [1, n]$. When $i = j$, $Para(t_i, t_j) = 1$, which indicates the exactly equal case. Similarly, when $k = g$, $Coord(e_k, e_g) = 1$. After values for all pairs of templates are updated, normalization takes place. This is the same for all pairs of entity tuples. Update continues until the difference between each new value and old value is smaller than a threshold θ. Since the calculations of transition matrices W^{TE} and W^{ET} depend on applications, we discuss the details in Section 5.

Finally, as a result, the paraphrase degree between two templates will be high if they share many common tuples, or have many coordinate tuple pairs; the coordinate degree between two entity tuples will be high if they share many common templates, or have many paraphrase template pairs.

5. Application: Judgement of Fact Credibility

In this section, we give details of our paraphrasing method when handling a certain application: judgement of fact credibility, especially how to obtain candidate templates and candidate tuples from the Web (referred to the step 1 in Fig. 1).

It is now intuitive to use the Web as a huge encyclopedia and trust information on the Web. However, the information is not always correct or true. For example, Denning et al. [7] reported that information on Wikipedia, which is regarded as the largest online encyclopedia, is not so credible. Therefore, it is necessary to understand risks of obtaining Web information and distinguish credible ones from it. We assume the credibility of a fact is high if people often mention it on the Web. Based on this assumption, a naive way to judge fact credibility is to check its occurrence on the Web. However, this trial always fails. As we stated in Section 1, the reason is that although there is a variety of different expressions for a fact, might be difficult to think of these expressions as many as possible. In the most extreme case, we may only think of one expression, which leads to failure of fact credibility judgement. For example, suppose we can only think of a statement like "apples are abundant in pectin", and want to know whether its represented fact is credible. We have known that the fact statement itself is rarely used. If we estimate the credibility of the fact only based on this statement, we would draw a wrong conclusion that apples do not contain much pectin. However, in fact, other fact statements, such as "apples are rich in pectin" or "apples are a great source of pectin", are widely used on the Web. These statements convey the same meaning as the given statement. Therefore, if we also take other statements of the fact into consideration, it is more likely for us to come to the right conclusion that apples are a high pectin fruit. To conclude, we judge fact credibility by observing both the given fact statement and other ones on the Web that convey the same meaning (called *paraphrases* for short hereinafter).

To run our mutual reinforcement algorithm, it is necessary to gather templates and entity tuples in advance. We now briefly introduce how to extract candidate templates and tuples from the Web by using a fact statement "Google has purchased Nest Labs", mapped by the template ***X** has purchased **Y*** and the entity tuple *(Google, Nest Labs)*, as the input.

5.1 Template Extraction

Since there might be many relations between entities and the Web is too large, it is necessary to limit our extraction to a certain field. We use ***context terms*** for this purpose. To obtain context terms, we prepare two kinds of queries. One is a wildcard query generated by the input template, i.e., "* has purchased *". The other is an AND query generated by nouns and verbs (excluded ones such as *be* and *has*) extracted from the given fact statement, i.e., "Google AND purchased AND Nest Labs". The context terms are chosen as the highest *tf·idf* scoring terms in the top search results of these two queries. For example, term "company" is chosen as a context term for the input. Correspondingly, we generate an AND query by the input tuple and the context term, i.e., "Google AND Nest Labs AND company". Candidate templates are extracted from the top N search results of each generated AND query. We heuristically eliminate non-essential phrases, such as additional prepositional phrases (e.g. "Google now owns Nest Labs after shelling out…" is analyzed as "Google now owns Nest Labs…"), or individual tokens, such as adverbs (e.g., "previously announced" is reduced to "announced").

Our template extraction is not limited to the text between two entities. For example, we obtain a candidate template such as ***X** announced the **Y** acquisition back in **.* We also assume an overlong template is more likely to contain additional information, while a too-short template is more likely to miss some information. Both situations lead to non-paraphrases. Therefore, we also exclude overlong and too-short templates.

5.2 Entity Tuple Extraction

We first find coordinate terms for both of the entities *Google* and *Nest Labs* using the bi-directional lexico-syntactic pattern-based algorithm [16]. As a result, we obtain *Yahoo* as a coordinate term of *Google*, *Dropcam* as a coordinate term of *Nest Labs*. Substituting each entity by its coordinate terms, we generate wildcard queries for extracting candidate tuples. For example, for *Google*'s coordinate term *Yahoo*, we generate the query "Yahoo has purchased *". For *Nest Labs*'s coordinate term *Dropcam*, we generate the query "* has purchased Dropcam". We then extract entities [*4] from the corresponding asterisk part in the top M search results of the above queries. As a result, we obtain tuples such as *(Yahoo, Tumblr)* from the former query, tuple such as *(Nest, Dropcam)* from the latter.

We use coordinate terms for the following two reasons. First, there is a massive amount of information on the Web. If we only search by "* has purchased *" and extract entity tuples from corresponding portions of sentences, many irrelevant tuples are gathered, such as *(God, freedom)*. Hence, coordinate terms are used to reduce the number of irrelevant tuples. Second, there might be few entity tuples extracted from the Web if the binary relation between two entities is one-to-one. For example, in the sentence "The capital of Japan is Tokyo", the relation between *Japan* and *Tokyo* is one-to-one, since we can only find *Tokyo* as the answer for which city the capital of *Japan* is, and vise versa, we can only find *Japan* as the answer for *Tokyo* is the capital of which country. Thus, it is difficult to obtain other entity tuples from the wildcard query "The capital of * is Tokyo" or "The capital of Japan is *". In this case, coordinate terms are used to increase the number of entity tuples extracted from the Web.

In some cases, we could not obtain enough candidate entity tuples or templates. Hence, several frequently appeared tuples are used to extract more candidate templates. In the same way, several frequently appeared templates are used to extract more candidate entity tuples.

5.3 Calculations of Transition Matrices

Since our objective is to find frequently used paraphrases of the given fact statements, the possibility of combinations of templates and entity tuples is significant. That is, we are concerned about

[*4] We employ the Stanford part-of-speech tagger to extract nouns or noun phrases.

whether people use an expression. For example, even if template ***X are considered a high Y fruit*** conveys the same meaning with ***X pop a powerful Y punch***, we do not obtain any search results with the query "apples are considered a high pectin fruit". Hence, such paraphrases are useless for judging fact credibility. Based on the above discussion, transition matrices W^{TE} and W^{ET}(in Section 4.3) are calculated in the following respective manners. Entry w_{ij}^{te} is the proportion of e_j's occurrence in t_i's top search results, while entry w_{ij}^{et} is the proportion of t_j's occurrence in e_i's top search results.

6. Evaluation

In this section, we discuss the experiments we conducted to validate the main claims of the paper, which is an enhancement of what we did in Ref. [26].

6.1 Experimental Setting

Given a sentential query, it is costly to find all templates and all entity tuples throughout the entire Web. For our experiments, we extracted candidate templates from the top N = 1,000 (mentioned in Section 5.1) search results of each AND query formed by a tuple and an additional context term, using the Bing Search API [*5]. We extracted candidate tuples from the top M = 1,000 (mentioned in Section 5.2) search results of each wildcard query. We fixed the value of threshold θ (mentioned in Section 4.3) to 0.0001 and found values of $Para(t_i, t_j)$ and $Coord(e_k, e_g)$ converging after 20 $\sim$ 25 updates. Note that we empirically set the above values to N, M and θ, respectively.

6.2 Query Data

To our knowledge, there is few widely accepted public dataset for paraphrase acquisition at sentence level. Correspondingly, it is difficult to directly use query data from evaluation part of any previous work. Therefore, we manually create our query data, containing 120 sentential queries, for evaluation. Since our proposed method, the mutual reinforcement algorithm, is based on the intensional-extensional representation for a relation, these 120 sentential queries are actually from the following six semantic relations:

(1) **highConcentration:** *We define this as a food contains a high amount of a certain nutrient.*
(2) **acquisition:** *We define this as the activity between two companies such that one company acquired another.*
(3) **field:** *We define this as the relation between a person and his field of expertise.*
(4) **majorLanguage:** *We define this as the relation between a language and an area such that the language is the major one in the area.*
(5) **manufacture:** *We define this as the relation between a product and its manufacturer.*
(6) **produce:** *We define this as the relation between a mineral and its producing area.*

We select last five relations by referring to some previous works [5], [6], [11] about acquiring paraphrases or detecting paraphrases in a corpus, and questions from TREC-8 Question-Answering Track. In addition to these five relations, we added the **highConcentration** relation, since we believed that it is an important relation and many queries in this relation cannot be broken down into joint phrase queries.

As we mentioned in Section 3.1, a sentence can be mapped by a template and an entity tuple. We list all templates and all entity tuples used to generate our sentential queries in **Table 3**, grouped by their semantic relations. For each relation, there are 5 templates and 4 entity tuples. Consequently, there are 20 combinations between templates and entity tuples. Thus, we have 20 sentential queries in each relation. Since each of them can be regarded as a "fact statement", we also analyze the performance for fact credibility judgement in our evaluation. Actually, for entity tuples, we manually selected 2 and also manually created 2 incredible tuples for each relation. Here, an incredible entity tuple indicates one that there does not exist the certain relation between entities in this tuple. Take the **highConcentration** relation for example. *(avocados, pectin)*, *(strawberries, protein)* are incredible tuples, since avocados do not contain much pectin and strawberries are not full of protein. Therefore, among 120 fact statements, there are 60 credible ones and 60 incredible ones. Besides, we also check the occurrence of each fact statement on the Web by Web search. We find there are 47 queries commonly used and correspondingly, 73 queries seldom used on the Web. Here, if the occurrence of a query on the Web is more than 10, we regard it as a commonly-used query, and vice versa.

6.3 Performance of Paraphrase Acquisition

As there is not much work in acquiring sentence-level paraphrases from the Web, it is difficult to directly compare against existing methods. Therefore, we constructed a baseline method for comparison, a variation of method stated in Ref. [5]. In the baseline method, we regard tuples as the context of each template, and use them to construct vector for each template. Then calculating the paraphrase degree between templates turns to be a problem to compute the cosine similarity between two vectors.

Since we mentioned in Section 3.2 that frequently appeared tuples are used to extract more candidate templates, and vice versa, frequently appeared templates are used to extract more candidate tuples, we investigate the employment of tuples and templates. As a result, we have 4 ways to obtain candidates: (1) Simple: extracted tuples and templates are not further used; (2) Template reused: extracted templates are further used to extract more candidate tuples; (3) Tuple reused: extracted tuples are further used to extract more candidate templates; (4) Complete: both extracted templates and extracted tuples are further used to extract more candidate tuples and candidate templates, respectively.

Table 4 shows the performance of our method for paraphrase acquisition, compared with the baseline method. Here we calculated the precision as how many "correct" paraphrases are in the paraphrases obtained by our method. From the table, we can point out that the employment of extracted templates and extracted tuples can make a big increase in precision, about 24.8% increased when compared the complete method with the simple method. Moreover, we can see that our complete method obtains a preci-

[*5] http://datamarket.azure.com/dataset/bing/search

Table 3 Sentential queries for evaluation. Entity tuples shown in bold are credible ones, while the rest are incredible ones.

Template	Relation	Entity Tuple
X are rich in **Y**	**highConcentration**	***(lemons, vitamin c)***
X are a great source of **Y**		***(apples, pectin)***
X are packed with **Y**		*(avocados, pectin)*
X are abundant in **Y**		*(strawberries, protein)*
X are considered a high **Y** fruit		
X has purchased **Y**	**acquisition**	***(Google, Nest Labs)***
X bought **Y**		***(Facebook, WhatsApp)***
X has agreed to acquire **Y**		*(Twitter, Dropbox)*
X has announced plans to buy **Y**		*(Microsoft, Instagram)*
X finished its acquisition of **Y**		
X revolutionized **Y**	**field**	***(Albert Einstein, physics)***
X is popularly known as the father of **Y**		***(Euclid, geometry)***
X is known for his work in **Y**		*(Nietzsche, philosopher)*
X laid much of the foundation for **Y**		*(James Waston, biology)*
X made enormous advances in **Y**		
X is the major language of **Y**	**majorLanguage**	***(Cantonese, Hong Kong)***
X is widely spoken in **Y**		***(French, France)***
X is a **Y**-speaking city		*(French, Spain)*
X is **Y**'s official language		*(English, Taiwan)*
X is the official language of **Y**		
X manufactures **Y**	**manufacture**	***(Toyota, Lexus)***
X is planning to release **Y**		***(Nissan, Infiniti)***
Y was created by **X**		*(Honda, Vovle)*
Y is the luxury brand of **X**		*(Toyota, Mini Cooper)*
Y is a segment of **X**		
X is the biggest producer of **Y**	**produce**	***(China, tungsten)***
X dominates the primary production of **Y**		***(Russia, oil)***
X is the largest supplier of **Y**		*(Russia, aluminium)*
X has the highest **Y** reserves		*(Canada, natural gas)*
X dominates the global **Y** market		

Table 4 Performance of paraphrase acquisition.

Method	Fact Statement Type	# Obtained	# Correct	Precision [*6]
Simple	**All**	13.4	8.4	0.464
	Widely-used	31.2	20.2	0.499
	Seldom-used	4.5	2.5	0.447
Template Reused	**All**	24.3	15.7	0.668
	Widely-used	23.7	17	0.730
	Seldom-used	24.8	14.5	0.614
Tuple Reused	**All**	28.1	19.1	0.688
	Widely-used	35.8	26.2	0.746
	Seldom-used	23.1	14.4	0.649
Complete	**All**	29.1	20.4	0.712
	Widely-used	35.6	26.2	0.751
	Seldom-used	22.9	15.1	0.680
Baseline	**All**	11.4	6.8	0.417
	Widely-used	22.4	14.9	0.498
	Seldom-used	6.2	3.0	0.380

sion of 71.2% over all fact statements, compared to 41.7% with the baseline. Furthermore, for frequently appearing statements, our complete method gets a precision of 75.1%, compared to 49.8% with the baseline. While for infrequent ones, we also get a good result, about 68%, compared to 38% with the baseline. Besides, compared to the baseline, our method makes a significant growth in obtaining correct paraphrases, nearly 3 times.

Take as an example the sentential query generated by the template ***X** is popularly known as the father of modern **Y*** and the entity tuple *(Albert Einstein, physics)*. 5 "correct" paraphrases are shown as below:

- *Albert Einstein is known as the father of physics.*
- *Albert Einstein is a prominent and legendary man acknowledged for his astounding contribution to physics.*
- *Albert Einstein is held up as a rare genius,who changed the field of theoretical physics.*
- *Albert Einstein fundamentally changed the world-view of physics.*
- *Albert Einstein laid much of the foundation for physics.*

Table 5 shows a comparison between the baseline and our complete method for paraphrase acquisition. Our method outperforms the baseline method no matter what value k is. Besides, significant improvement is achieved when k equals to 5. On the other hand, it is difficult to estimate the recall since we do not have a complete set of paraphrases for a certain fact statement. However, following Tague-Sutcliffe [20], we can pool the correct results (corresponding here to correct paraphrases) of each method to form the answer set. The relative recall can be calculated using the following formula:

[*6] Note that this is the macro average (the average of precisions of all fact statements) rather than the proportion of the correct paraphrases in the obtained paraphrases.

Table 5 A comparison between baseline and our method for paraphrase acquisition.

System	Precision@5	Precision@10	Precision@15	Precision@20	Precision	Relative Recall
Baseline	0.451	0.700	0.619	0.660	0.417	0.315
Complete	0.813	0.793	0.756	0.738	0.712	0.932

Table 6 Performance for judging fact credibility by using top 10 paraphrases.

Fact Statement Type	Average HitCount of Input Statements	Average Increase HitCount	Average Increase Rate
Widely-used	7,318.00	14,709.18	2.01
Seldom-used	0.86	9,831.75	11,432.27
Credible	4,212.35	22,030.59	5.23
Non-credible	0	0.55	–

$$\text{Relative recall} = \frac{\text{\# of correct paraphrases obtained by a method}}{\text{\# of correct paraphrases obtained by our method and baseline}}$$

From Table 5, we can know that our method achieved a big increase in relative recall, which indicates that it is effective to incorporate mutual reinforcement between templates and tuples to identify paraphrases. We also did a *t-test* between precision for our method and the baseline method, which yielded a p-value of 0.00281. As the p-value is smaller than 0.05, we can infer that the experimental results of our method is reliable and there is a statistically significant difference between baseline and our method.

Table 1 lists the top 15 paraphrases of "Lemons are considered a high vitamin c fruit" generated by our method. By observing these top results, we find sentences, such as "lemons contain vitamin c" are misjudged as paraphrases. Such sentences are not paraphrases because we can not infer that lemons have a high amount of vitamin c from them. Our method fails to identify such sentences because templates from such sentences, we call them "general" templates, are more likely to have many suitable entity tuples. Therefore, these templates may share many common tuples with the original template, which leads them obtain high scores of paraphrase degree.

Interestingly, we find that sometimes fact statements that have the opposite meaning are misjudged as paraphrases. For example, for the fact statement "China dominates the global tungsten market", we obtain "China was the world's leading tungsten consumer" in the top 3. Therefore, how to identify the opposite or negative meaning of the original statement is a problem that needs further consideration.

6.4 Performance for Judging Fact Credibility

A fact has a variety of different expressions in the surface form, namely fact statements. For example, all the following statements represent the same fact:

- *Apples are rich in pectin.*
- *Apples are a great source of pectin.*
- *Apples are packed with pectin.*

When we say a fact statement is credible, it basically refers to its represented fact is credible. We assume that a fact is credible if people often mention it on the Web. Hence, when we judge the credibility of a fact, in the ideal case, we should consider all its possible expressions in the surface form. However, since it is difficult to obtain all possible statements, we believe part of fact statements is enough for fact credibility judgement.

Take the fact statement "apples pop a powerful pectin punch" for example. As we already know, this statement never appears on the Web. If we estimate the credibility of the fact only based on the given statement, we would draw an erroneous conclusion that apples do not have a high amount of pectin. However, if we also take other statements of the fact into consideration, the ones that convey the same meaning as the given statement, such as "apples contain a lot of pectin" or "apples are rich in pectin" which are widely used on the Web, we could draw the right conclusion that apples are a high pectin fruit.

As a result, when we judge whether a fact is credible or not, we actually check how many times its fact statements appear on the Web. Accordingly, the hitcount on the Web could be an indicator for fact credibility judgement. Besides, if a fact is credible, all its statements are credible correspondingly.

Table 6 shows the performance for fact credibility judgement. For each fact statement, viz. each sentential query, we also take into consideration the top 10 paraphrases obtained by our method. It means we not only estimate the hitcount of each fact statement on the Web, but also aggregate the hitcount of its paraphrases, and make a comparison between them.

The first column of Table 6 indicates the classification of fact statements. The second column represents the average number of how many times each fact statement occurs on the Web. The third column is the comparison of absolute value: how the occurrence increased by considering paraphrases, while the forth column is a relative value: the average increase rate.

From this table, we know that with the help of paraphrases, we can retrieve more Web pages. Especially, for fact statements that are not frequently used by people, we got a tremendous increase of retrieved Web pages, 11,432.27 times increased. This finding also illustrates the effectiveness of paraphrased queries, since they solve the problem caused by the expression rarity of the original sentential queries. It is much more likely to obtain desired information, because we are able to find much more Web pages hit these paraphrased queries. When considering fact credibility judgement, we found for non-credible fact statements, even we extend them by their paraphrases, there are few Web pages returned, which results in a small increase in hitcount, only 0.55 on average. Hence, if we consider part of statements of a fact, but still cannot find enough appearances, we can figure out that the fact is incredible and its statements are correspondingly incredible. As a result, paraphrases are effective for estimating fact credibility.

6.5 Case Study

In this section, two cases are presented to demonstrate the ef-

fectiveness of paraphrases for fact credibility judgement. In each case, we take one fact statement used in our evaluation for example. Notice that all the information about both the fact and its statements are based on the Bing search engine at the time of writing this paper.

Case 1 Seldom-used & Credible

The fact statement "Facebook has announced plans to buy WhatsApp" never appears on the Web. However, its represented fact is true since Facebook did buy WhatsApp in 2014. The result of our experiment can corroborate this. Specifically, in the beginning, the hitcount of the statement is 0. After we took into consideration the top 10 paraphrases obtained by our method, such as "Facebook has announced that it is acquiring WhatsApp", "Facebook has agreed to buy WhatsApp", or "Facebook will acquire WhatsApp", we found 3,600 more search results in total. As the number is not small, we can conclude that there is an acquisition between Facebook and WhatsApp, and Facebook is the buyer. Therefore, the fact statement is credible since we have already known its represented fact is credible.

Case 2 Seldom-used & Incredible

We cannot find the fact statement "Avocados are packed with pectin" on the Web. Besides, we looked up many other materials, such as Wikipedia, websites of food nutrition. We found there is little pectin contained in avocados. The result of our experiment can support this finding. In more details, in the beginning, the hitcount of the statement is 0. After we took into consideration the top 10 paraphrases obtained by our method, such as "Avocados contain high levels of pectin", "Avocados are a rich source of pectin", or "Avocados are an excellent source of pectin", we found 4 more search results in total. However, since the total occurrence is still very small, we can draw a conclusion that avocados contain little pectin. Therefore, the fact statement is incredible since we have already known its represented fact is incredible.

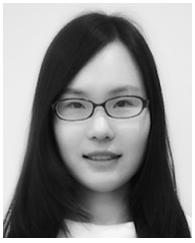
7. Conclusion

We handle with sentential queries and aim at improving its retrieval performance. Different from previous studies, we argue that separate terms or phrases from a sentential query may lead to the missing of some information or query drift. To avoid such problems, we propose query paraphrasing for sentential queries. In sentential query paraphrasing, we use other frequently used queries that convey the same meaning to avoid returning no answers. Furthermore, we incorporate coordinate relationships between entity tuples and take a mutually reinforcing approach to identify paraphrase templates. The experimental results show that our method can acquire more paraphrases from the Web. Besides, with the help of paraphrases, more Web pages can be retrieved, especially for those sentential queries that could not find any answers with its original expression.

Acknowledgments This work was supported in part by the following projects: Grants-in-Aid for Scientific Research (Nos. 15H01718 and 24680008) from MEXT of Japan.

References

[1] Agichtein, E. and Gravano, L.: Snowball: Extracting Relations from Large Plain-text Collections, *Proc. DL*, pp.85–94 (2000).
[2] Balasubramanian, N., Kumaran, G. and Carvalho, V.R.: Exploring Reductions for Long Web Queries, *Proc. SIGIR*, pp.571–578 (2010).
[3] Bendersky, M. and Croft, W.B.: Discovering Key Concepts in Verbose Queries, *Proc. SIGIR*, pp.491–498 (2008).
[4] Bendersky, M. and Croft, W.B.: Analysis of Long Queries in a Large Scale Search Log, *Proc. WSCD*, pp.8–14 (2009).
[5] Bhagat, R. and Ravichandran, D.: Large Scale Acquisition of Paraphrases for Learning Surface Patterns, *Proc. ACL2008:HLT*, pp.674–682 (2008).
[6] Bollegala, D.T., Matsuo, Y. and Ishizuka, M.: Relational Duality: Unsupervised Extraction of Semantic Relations Between Entities on the Web, *Proc. WWW*, pp.151–160 (2010).
[7] Denning, P., Horning, J., Parnas, D. and Weinstein, L.: Wikipedia Risks, *Comm. ACM*, Vol.48, No.12, pp.152–152 (2005).
[8] Etzioni, O., Cafarella, M., Downey, D., Popescu, A.-M., Shaked, T., Soderland, S., Weld, D.S. and Yates, A.: Unsupervised Named-Entity Extraction from the Web: An Experimental Study, *Artificial Intelligence*, Vol.165, pp.91–134 (2005).
[9] Harris, Z.S.: Distributional structure, *Word*, Vol.10, pp.146–162 (1954).
[10] Hitwise Intelligence: Google Received 72 Percent of U.S. Searches in January 2009, available from ⟨http://image.exct.net/lib/fefc1774726706/d/1/SearchEngines_Jan09.pdf⟩ (2009).
[11] Idan, I.S., Tanev, H. and Dagan, I.: Scaling Web-based Acquisition of Entailment Relations, *Proc. EMNLP*, pp.41–48 (2004).
[12] Kraft, R., Chang, C.C., Maghoul, F. and Kumar, R.: Searching with Context, *Proc. WWW*, pp.477–486 (2006).
[13] Kumaran, G. and Allan, J.: A Case for Shorter Queries, and Helping Users Create Them, *Proc. HLT*, pp.220–227 (2007).
[14] Lau, T. and Horvitz, E.: Patterns of Search: Analyzing and Modeling Web Query Refinement, *Proc. UM*, pp.119–128 (1999).
[15] Mei, Q., Fang, H. and Zhai, C.: A Study of Poisson Query Generation Model for Information Retrieval, *Proc. SIGIR*, pp.319–326 (2007).
[16] Ohshima, H., Oyama, S. and Tanaka, K.: Searching Coordinate Terms with Their Context from the Web, *Proc. WISE*, pp.40–47 (2006).
[17] Paşca, M. and Dienes, P.: Aligning Needles in a Haystack: Paraphrase Acquisition Across the Web, *Proc. IJCNLP*, pp.119–130 (2005).
[18] Phan, N., Bailey, P. and Wilkinson, R.: Understanding the Relationship of Information Need Specificity to Search Query Length, *Proc. SIGIR*, pp.709–710 (2007).
[19] Shinyama, Y., Sekine, S. and Sudo, K.: Automatic Paraphrase Acquisition from News Articles, *Proc. HLT*, pp.313–318 (2002).
[20] Tague-Sutcliffe, J.: The Pragmatics of Information Retrieval Experimentation, Revisited, *Inf. Process. Manage.*, Vol.28, No.4, pp.467–490 (1992).
[21] Wubben, S., van den Bosch, A., Krahmer, E. and Marsi, E.: Clustering and Matching Headlines for Automatic Paraphrase Acquisition, *Proc. ENLG*, pp.122–125 (2009).
[22] Xue, X., Jeon, J. and Croft, W.B.: Retrieval Models for Question and Answer Archives, *Proc. SIGIR*, pp.475–482 (2008).
[23] Yamamoto, Y. and Tanaka, K.: Towards Web Search by Sentence Queries: Asking the Web for Query Substitutions, *Proc. DASFAA*, pp.83–92 (2011).
[24] Yamamoto, Y., Tezuka, T., Jatowt, A. and Tanaka, K.: Supporting Judgement of Fact Trustworthiness by considering Temporal and Sentimental Aspects, *Proc. WISE*, pp.206–220 (2008).
[25] Yates, A., Cafarella, M., Banko, M., Etzioni, O., Broadhead, M. and Soderland, S.: TextRunner: Open Information Extraction on the Web, *Proc. NAACL*, pp.25–26 (2007).
[26] Zhao, M., Ohshima, H. and Tanaka, K.: Sentential Query Rewriting via Mutual Reinforcement of Paraphrase-Coordinate Relationships, *Proc. iiWAS* (2015).



Meng Zhao is a Ph.D. student of Kyoto University. She received her B.S. degree in Management from University of International Business and Economics in 2009, and the M.S. degree in Informatics from Kyoto University in 2012. Her research interests include information retrieval, Web search and mining. She is a student member of the Database Society of Japan (DBSJ).

Hiroaki Ohshima has been a program-specific associate professor of the Graduate School of Informatics, Kyoto University since 2013. He received his B.S. and M.S. degrees in Engineering from Kobe University, in 2000 and 2003, respectively. In 2006, he received his Ph.D. degree in Informatics from Kyoto University. His research interests include Web search and mining, information retrieval, and database systems. He is a member of the ACM, the Information Processing Society of Japan (IPSJ), the Institute of Electronics, Information and Communication Engineers (IEICE), and the Database Society of Japan (DBSJ).

Katsumi Tanaka received his B.S., M.S. and Ph.D. degrees in Information Science from Kyoto University, in 1974, 1976 and 1981, respectively. Since 2001, he has been a professor of the Graduate School of Informatics, Kyoto University. His research interests include database theory and systems, Web information retrieval, Web data mining and multimedia retrieval. Dr. Tanaka is a member of the ACM, IEEE, the Database Society of Japan (DBSJ) and the Information Processing Society of Japan (IPSJ), and IPSJ fellow.

(Editor in Charge: *Terunao Hochin*)