A Model of Implicit Term Relationship for Information Retrieval

Tetsuya Yoshida
Graduate School of Information Science and Technology
Hokkaido University
N-14, W-9, Sapporo, Hokkaido 060-0814
JAPAN
yoshida@meme.hokudai.ac.jp

Abstract: This paper proposes a model for dealing with implicit term relationship among terms toward information retrieval, in the context of information retrieval over the Web. Until now various keyword based search engines have been developed to facilitate information retrieval over the Web. However, it can still be difficult to specify appropriate keywords (terms), which are to be provided to the engines to conduct the retrieval. We hypothesize that, although it is not explicitly represented or specified from the user, there can be some (hidden) relationship among the specified terms. Such relationship can be useful to facilitate effective retrieval, since it can work as "between the terms", as in the between the lines in effective reading. Based on this hypothesis, we propose a model for representing the implicit relationship among the specified terms. Our model tries to capture the implicit relationship in terms of semantic aspect, and represents it as a concrete tree structure so that it can be utilized for further processing. Experiments were conducted to investigate the effectiveness of the proposed model in the context of retrieval, and the results are reported.

Key–Words: Information Retrieval, Term Relationship, Tree Structure,b Thesaurus, World Wide Web

1 Introduction

The rapid development of the network such as the Internet has enabled easy access to the huge quantity of intellectual assets over the network such as the Internet. However, this can be a double-edged sword, since it can be very difficult to find the appropriate one among such huge volume of information. Although various keyword based search engines have been developed and published over the network, still it can be difficult to specify the appropriate keywords [1, 8]. Most search engines show many URLs to the user in response to the specification of the query keywords. However, it often happens that many interactions are required afterward with the user. For instance, it might be necessary to further specify other keywords, based on the provided output of the engines.

The interaction with the user based on the search result can be conceived as follows. The originally specified keywords are still insufficient, and in order to focus the search, the user himself/herself try to capture the relationship among the terms based on the output. The relationship, which was not originally represented as keywords, is then represented as some auxiliary keywords, and given to the engine in conjunction with the original ones. As described above, we hypothesize that there can be some (hidden) relationship among the terms, despite it might not be explicitly represented or specified from the user. We further assume that such information would be valuable if it can be exploited, for instance, in information retrieval.

In order to capture such (implicit) information and to utilize it, we propose a model for representing the implicit relationship among the specified terms in this paper. Our model tries to capture it in terms of semantic aspect and represent it as a tree structure. The semantic information among the terms is conceived using a thesaurus. In our model, the implicit relationship among the terms is represented as a concrete tree structure so that it can be utilized computationally.

This paper is organized as follows. Section 2 gives a brief survey of the related work to this paper. Section 3 describes an overview of our approach, followed by our model. The details of our model and utilized techniques are detailed in Subsection 3.3. Its evaluation is described in Section 4, especially with respect to its utilization for information retrieval. The results are discussed in Section 5. Section 6 briefly gives the conclusion of this paper.

2 Related Work

Various researches have been conducted on supporting information retrieval [7]. Techniques of information filtering [11] are often used to reduce the number of retrieved material by filtering out irrelevant ones. Broadly speaking, information filtering can be categorized into cognitive filtering, social filtering, and economic filtering [9].

Some approach utilizes a user profile to increase the accuracy of filtering. However, it tends to require time and effort for a user to describe his/her profile and it may also difficult to adapt to the change of user's interest [13]. [2] proposes a method for estimating relevant documents based on the stored documents which were assessed as relevant by a user. In their method the similarity of the newly searched material is calculated for all the stored ones in order to estimate its relevance. Since all the stored information can be utilized, the accuracy of filtering is reported as sufficiently high. However, the time complexity for calculating the similarity is quite large and thus is not suitable for interactive support. Furthermore, since the accuracy depends on the quality and quantity of the stored ones, the necessary pre-process for setting up enough material imposes a heavy overload on a user.

As for the problem of specifying appropriate query terms, usually a user him/herself learns what terms are to be used through the interaction with a search engine and manually specifies another term in addition to the initially specified terms to further narrow down the search result [6, 14]. Some approaches aim at supporting retrieval by extracting terms which are related with the initially specified terms. For instance, several terms are extracted among all the terms in the retrieved documents based on statistics in several search engines such as InfoNavigator [1] and Excite Japan [2].

Another approach proposed a term suggestion method by regarding the sequence of specified query terms in a retrieval session as a kind of "context"[5]. Based on the log analysis of a search engine, correlation among query terms in the log is calculated based on a Q-learning method in reinforcement learning [15]. Then, the highly correlated terms in the sequence of query terms in the same session are suggested for the user.

3 A Model for Implicit Term Relationship

3.1 Overview

We propose a model which try to capture an implicit relationship among the (specified) terms, and represent the relationship in a tree structure.

[1]http://infonavi.infoweb.ne.jp/
[2]http://www.excite.co.jp/

Our approach is based on the following components:

- thesaurus
- a tree structure based model
- algorithms for tree construction
- selection mechanism

A thesaurus is utilized to deal with the semantic aspect in retrieval. Based on the thesaurus, we propose a model, which tries to represent the implicit relationship among terms as a concrete tree structure. Algorithms are proposed to realize this processing. Based on the constructed tree structure, some terms are suggested as auxiliary keywords.

3.2 Reflecting Semantics of Terms based on a Thesaurus

A thesaurus, which defines or represents the relation between words, is utilized to construct the information retrieval system based on the correlation between query keywords. Since the relation of semantic meaning between words can be easily represented as numerical or distance in a thesaurus, it is suitable for computational or symbol processes to utilize as another keyword based on the hidden or implicit correlation between the them. For instance, a thesaurus is often utilized to calculate the degree of similarity between words in Natural Language Processing (NLP) [10, 12]. Since the relation between words defined in a thesaurus can be considered as a graph as in a semantic network [4], it is possible to treat the neighboring words with high similarity and distant words with low similarity.

Based on the above argument, in our current approach we assume that a thesaurus, which defines or describes the relation between terms in their language, is available. We discuss other possible approaches in Section 3.2.1.

3.2.1 Terms considered

Currently we consider only nouns as terms, for which their implicit relationship is considered. This is because nouns are often utilized to represent and specify the so-called "concepts" in many natural languages. Utilization of other terms such as adjectives and verbs are left for future work.

3.2.2 Utilized Thesaurus

As a working example, we intend to apply our model for texts written in Japanese. Thus, the "Modern

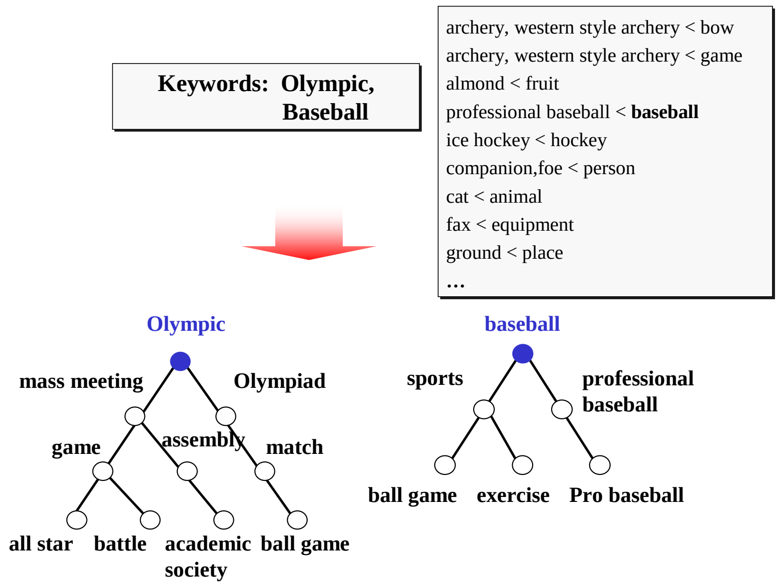

Figure 1: Examples of term trees.

Japanese Noun Thesaurus", which is constructed and distributed by Professor Hagino in Tokyo Metropolitan University [3].

In this thesaurus, the relationship among terms are described with respect to hypernym and hyponym, synonym, and is-a relationship. For instance, "cat $<$ animal" shows that, as for the term "cat", "animal" is its hypernym.

3.2.3 Other possible approaches

It might be possible to utilize the so called user profile, as in the recommendation system approach. Since the user profile of a user describes the detailed information on the user, the performance of information retrieval can be improved for that user [13]. However, describing the detailed information on each user is actually a hard work. In addition, it would be difficult to adapt to the change (drift) of user's interest with a fixed profile.

It might be possible to conceive the semantic aspect in terms of the "intention" of the user. For instance, it is widely known that so called design rationale plays an important role in design, and several approaches have been proposed to utilize it [16]. There are some approach which tries to deal with the conceptual aspect in terms of the extension of the data [18].

Admittedly the utilization of thesaurus to reflect the semantic aspects of terms has been widely utilized. However, our contribution in this paper is the proposal of a model based on the thesaurus and the method to construct the model.

3.3 A Model

Our model is based on the following hypothesis: although it is not explicitly represented or specified from the user, there can be some (hidden) relationship among the terms. Our model is based on the following two ideas:

- a thesaurus will reflect the semantic aspect of the terms
- the implicit relationship can be represented as a concrete tree structure

First, a tree structure is constructed for each specified term based on the thesaurus. These are called "term trees" in our approach. For instance, suppose two terms "Olympic" and "baseball" are specified. Then, the trees in Figure 1 are to be constructed based on the utilized thesaurus in Section 3.2.2.

Next, based on the constructed tree structures, the implicit relationship is again represented as a tree, which is called a "relational term tree". This is conceived as relational in the sense that the relationship among the terms would be represented. This structure is to be constructed such that:

- all the specified terms are represented in the tree
- their relationship is represented based on the thesaurus

Terms which are included in all the constructed trees have some relations with all the specified terms. In order to satisfy the above two properties, the commonly shared terms are searched among the trees, and utilized for the construction of the relational term tree.

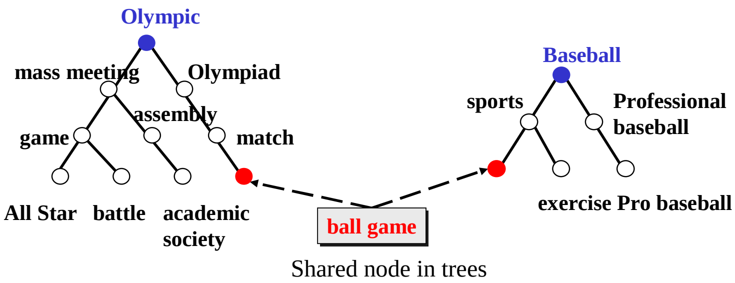

Figure 2: A shared term in the trees.

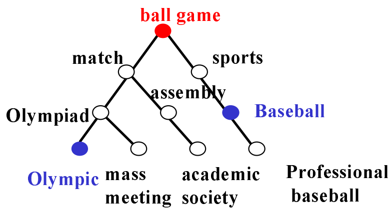

Figure 3: An example of relational term tree.

For instance, as for the trees in Figure 1, the term "ball game" is found as the commonly shared term is searched from the trees. Then, by treating the term as its root, a relational term tree is constructed such that the tree include the specified terms. Figure 3 shows the tree structure to be constructed.

3.4 Algorithms

3.4.1 An Algorithm for a specified Term

The algorithm to construct a tree structure for a specified term is summarized in Algorithm 1.

Currently the termination condition at line 4 in Algorithm 1 is set as when the height of the tree exceeds h. The value h controls the search space and time complexity of tree construction. Thus, a tree is constructed as "level-wise", until its height reaches the specified value.

3.4.2 An Algorithm for the Shared Term

Based on the tree structures constructed by Algorithm 2 for each specified term, we construct another tree structure (relational term tree), which is to represent the implicit term relationship. As described in Section 3.3, the construction of the relational term tree is based on the commonly shared terms in the trees constructed by Algorithm 2.

The algorithm to construct a tree structure based on the constructed tree structures by Algorithm 1 is summarized in Algorithm 2. Algorithm 2 is also based on Algorithm 1.

Since the smaller the tree is, the better it is in oder to reduce the time complexity for the construction. Thus, it is constructed as level-wise in the while loop at lines 4 to 8.

3.5 Similarity of Terms

To enable the computational process easier, it is better to represent the semantic relationship among the terms as some numerical value. Until now, several similarity measures have been proposed based on the thesauri in the field of Natural Language Processing (NLP) [10, 12]. Basically, most similarity measures are based on the assumption that neighboring terms are highly similar and distant ones are not.

Following the above approach, we also regard that the distance in the tree structure reflects the degree of "semantic" similarity of the terms in the tree. In addition, we assume that the depth of the "common" upper node for two terms plays an important role to measure their similarity.

Suppose the depth of two terms are d_i,d_j and the depth of the common upper node for these terms is d_c. Then, the degree of similarity is defined as [12]:

$$similarity = \frac{2 \times d_c}{d_i + d_j} \tag{1}$$

In most thesauri the depth of the root node is treated as 0 in the calculation. This is because the root node with no semantic meaning is often introduced arbitrary in order to make the network (graph) of terms as a tree structure. Thus, when the common upper node for two terms is the root of the tree, it is

Algorithm 1 termTree(t,h)

input: a term t, the height h

1: create a queue $\boldsymbol{Q}$ of nodes //*each node stores a term*
2: create a node for t and insert it into $\boldsymbol{Q}$
3: set the node as a tree T
4: **while** termination condition not reached,
5:   **if** $\boldsymbol{Q} = \phi$, go to **line 21:**
6:   n_c := the first node of $\boldsymbol{Q}$ //n_c *: current node*
7:   t_c := the term in n_c
8:   search the thesaurus for the descriptions which include t_c. If not found, go to **line 4:**.
9:   **foreach** descriptions founds, **case**
10:     t_c is in the body:
11:       **if** the index word t_i of the description is not represented as a node label in T
12:         create a node n_i with node label t_i
13:         connect n_i to n_c as a child node
14:         set the edge label between n_i and n_c as the annotation of t_c.
15:     t_c is the head:
16:       **foreach** term t_b in the body which is not represented as a node label in T
17:         create a node n_b with node label t_b
18:         connect n_b to n_c as a child node
19:         Set the edge label between n_i and n_b as the annotation of t_b
20:     Insert the created nodes into $\boldsymbol{Q}$
21: **return** a tree T

possible to interpret that there is no relation between these two terms. On the other hand, since the relational term tree is constructed by treating the common noun in keyword trees as its root, it is possible to consider all the terms represented in the tree have some relationship.

As described above, there can be two different interpretation of the depth of the root node. Currently, the depth of root node is treated as 1, not 0, in the calculation of the degree of similarity in (1). This is because we believe that the relational term tree will reflect some semantic information among the terms and thus its root should play some role in the similarity measure.

3.5.1 A Working Example

Figure 4 shows the degree of similarity for each node with respect to the profile tree in Figure 3. The degree of similarity is calculated by treating the depth of "ball game", which is the root in the profile tree, as 1.

Each value in Figure 3 is calculated based on the equation (1). The left hand side in the figure shows the similarity with respect to the term "Olympic". On the other hand, the right hand side is with respect to the term "Baseball".

For instance, as for the left-hand side tree in Figure 3, since $d_c = 2$, $d_i = 3$, and $d_j = 4$, the degree of similarity of term "assembly" is calculated as:

$$similarity = \frac{2 \times 2}{3 + 4} = 0.57$$

Likewise, the similarity of the same term "assembly" is calculated as

$$similarity = \frac{2 \times 1}{3 + 3} = 0.33$$

in the right hand side of Figure 3.

After measuring the similarity to the terms in the relational term tree with respect to each specified query term, its average is calculated. Then the terms with the largest value is selected. In this case, the term "Olympiad" corresponds to the term with the largest value.

4 Evaluations

Experiments were conducted to investigate the effectiveness of our proposal in Section 3. The scenario of its utilization is illustrated in Figure 5. As shown in Figure 5, our model is utilized to select some terms as auxiliary keywords.

Algorithm 2 relTermTree($\boldsymbol{t}$)

input: $\boldsymbol{t}$: a set of terms $\{t_1, \ldots t_n\}$

$d = 0$ *//set the depth of the trees*
foreach t_i, T_i := termTree(t_i, d)
foreach T_i, $\boldsymbol{S_i}$:= ϕ *//a set of terms for* T_i
foreach $\boldsymbol{S_i}$, $\boldsymbol{S_i}$:= $\boldsymbol{S_i} \cup$ {the terms in T_i at depth d}
$\boldsymbol{P} := \cap\, \boldsymbol{S_i}$
while $\boldsymbol{P} = \phi$
  d := d+1
  foreach T_i, T_i := termTree(t_i, d)
    // as efficient implementation, just expand T_i *without reconstruction*
  foreach $\boldsymbol{S_i}$, $\boldsymbol{S_i}$:= $\boldsymbol{S_i} \cup$ {the terms in T_i at depth d}
  $\boldsymbol{P} := \cap\, \boldsymbol{S_i}$
t_s := the term t_s with the minimum depth in $\boldsymbol{P}$
RT = termTree(t_s)
return RT

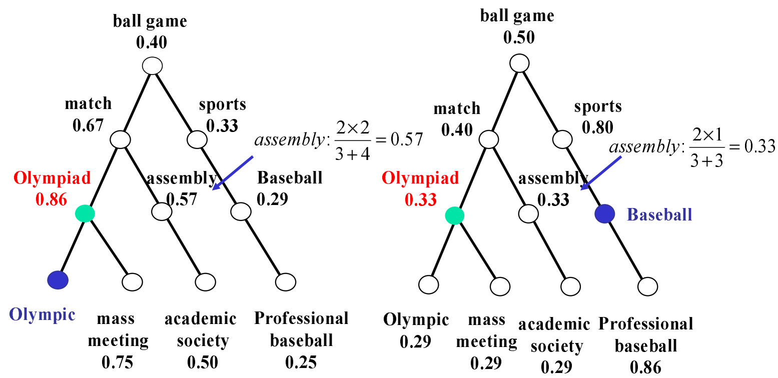

Figure 4: An example of similarity calculation.

4.1 Evaluation Settings

The measure "precision" and "recall" are widely utilized in the field of Information Retrieval [7]. The former is to measure to what extent the selected term from the tree is effective to focus the retrieval. On the other hand, the latter is to measure to what extent necessary information can be retrieved.

Suppose the set of correct documents is $\boldsymbol{A}$ and that of actually retrieved documents is $\boldsymbol{B}$. Then, precision and recall are calculated as:

$$Precision = \frac{|\boldsymbol{A} \cap \boldsymbol{B}|}{|\boldsymbol{B}|} \qquad (2)$$

$$Recall = \frac{|\boldsymbol{A} \cap \boldsymbol{B}|}{|\boldsymbol{A}|} \qquad (3)$$

where $|\cdot|$ represents the cardinality of a set.

The relationship between Precision and Recall is illustrated in Figure 6.

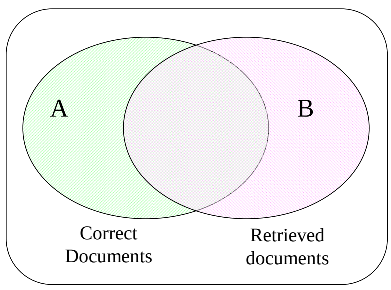

Figure 6: The relationship between Precision and Recall.

Recall becomes smaller when the user specifies inappropriate terms as query keywords. On the other hand, if the user specifies the terms which are shared (included) in many material, the recall becomes larger but the precision becomes smaller. Thus, the effec-

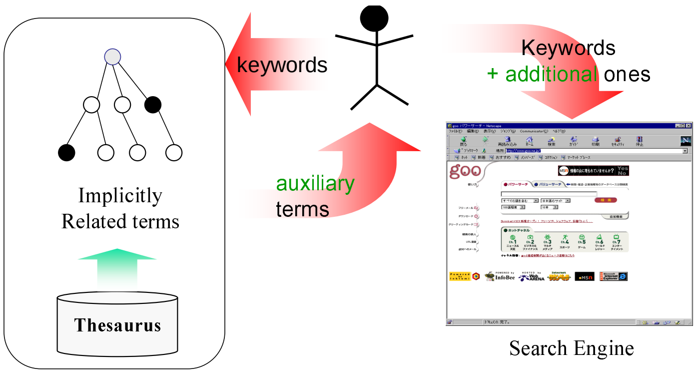

Figure 5: A scenario of auxiliary term.

tiveness of the auxiliary terms is measured with respect the increase or decrease of these measures.

4.2 Similarity, Precision and Recall

A snapshot of the Graphical User Interface of the implemented system is shown in Figure 7.

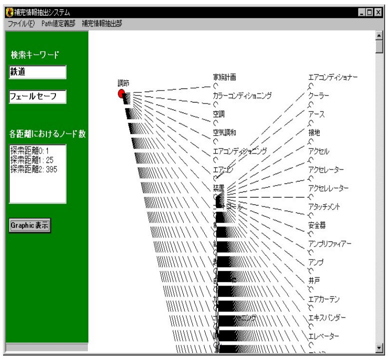

Figure 7: A snapshot of the Graphical User Interface of the implemented system.

The degree of similarity is calculated for all the nodes in the relational term tree, as shown in Figure 4. However, since the degree of similarity is calculated with respect to the utilized thesaurus, it is not necessarily the case that the term with large degree of similarity is the "effective" one as an auxiliary keyword.

Experiments were conducted to investigate to what extent the degree of similarity defined in (1) contributes to the selection of auxiliary keywords. In the experiment the terms "sports" and "gamble" were used as query keywords. By utilizing the Algorithms 1 and 2, the relational term tree was constructed and the degree of similarity was calculated for all the nodes in the tree. The quality of the search result with the auxiliary keyword was judged subjectively, and evaluated as precision and recall using Equations 2 and 3.

Figure 8 shows a scatter plot of precision and recall with respect to each term in the tree [3].

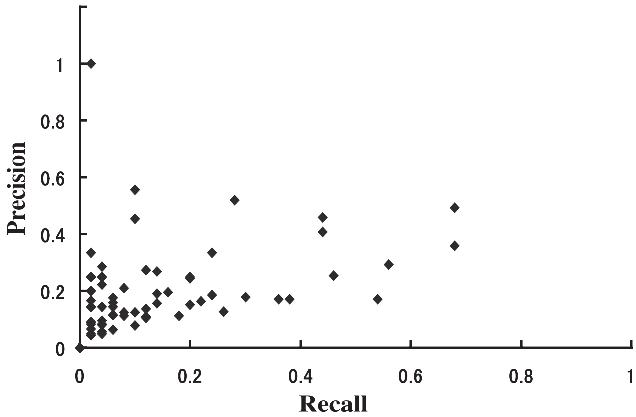

Figure 8: A scatter plot of Precision and Recall in the experiment, where each term in the tree is utilized as an auxiliary keyword.

4.3 Effectiveness of Retrieval with the additional Terms

Other experiments were also conducted to investigate how effective (increase or decrease) the terms in the tree would be as auxiliary keywords with respect to precision and recall. In the experiments, goo and InfoNavigator were used as the search engines.

These search engines are effective when the user is familiar with the appropriate query keyword and all the URLs which include the query keyword should be retrieved. However,it is difficult to conduct retrieval based on such engines when the user is not so familiar with the appropriate query keyword or when the exact description of query keyword is difficult.

Experiments were conducted to evaluate whether the terms with large similarity values could work

[3] The search engine goo was utilized as the search engine in this evaluation.

as effective auxiliary keywords. Figure 9 shows the change of precision and recall the user specified "sports" and 'gamble" as the keywords. Figure 10 shows the result for "new trunk line" and "automatic control".

In both figures, the terms in the circle are the terms with large similarity values. In Figure 9 the term " match " had the largest value. On the other hand, the other terms outside the circle are the promising terms in terms of precision and recall.

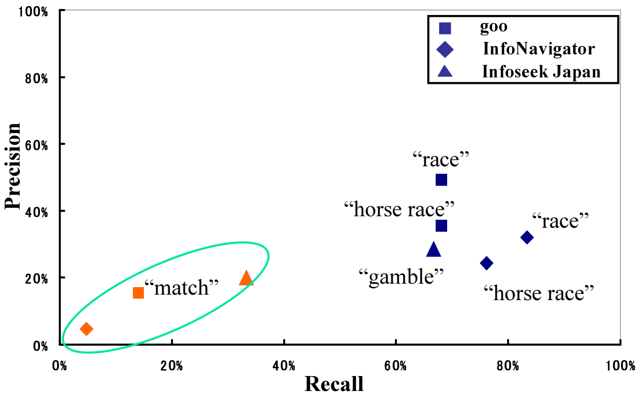

Figure 9: Result:"sports", "gamble".

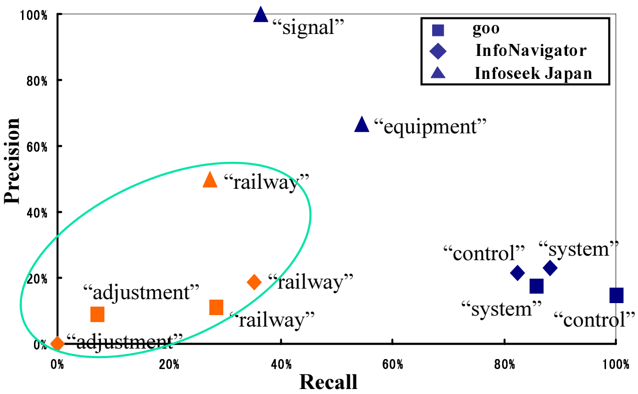

Figure 10: Result:"new trunk line", "automatic control".

5 Discussion

5.1 Similarity and Retrieval

Experiments in Section 4 were conducted to investigate whether the similarity value based on a thesaurus could be effective in the context of information retrieval. Results in Section 4 show that the terms with the largest similarity are not necessarily the ones with the largest precision and recall. However, the similarity based a thesaurus still reflects the semantic information among the terms.

The comparison with other terms as the additional query keyword in Figure 9 showed that, utilizing the auxiliary terms can still contribute to both high degree of precision and recall[4]. However, the results remain at indicating or hinting that there exists some noun with high degree of precision and recall.

An example of possibly better terms for this case is illustrated in Figure 11. In Figure 11, the term "match" has larger similarity value (with 0.65) than that of "race" (with 0.54). However, the latter showed better performance in the result in Figure 9.

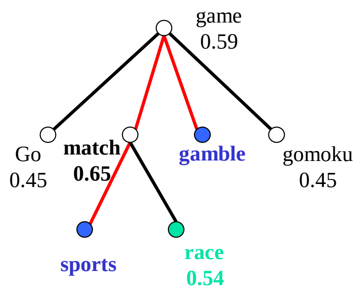

Figure 11: An example of possibly better auxiliary terms.

Likewise, Figure 12 shows another example of possibly better terms for the result in Figure 10. As shown in Figure 10, despite smaller similarity value calculated by equation 1, the term "system" showed better performance in Figure 10.

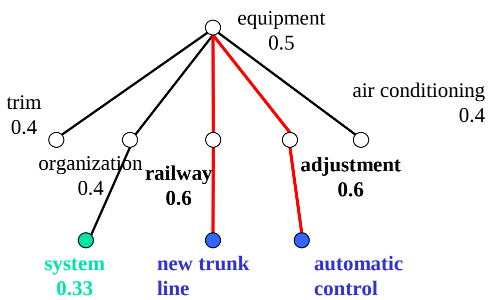

Figure 12: Another example of possibly better auxiliary terms.

From the above results, in addition to the similarity calculation based on the relational term tree, it would be necessary to come up with another mechanism for the selection of better auxiliary terms.

In our future work, we plan to further investigate the relationship between the similarity based on a thesaurus and the evaluation measures (precision and recall) in information retrieval. Based on the analysis, we will propose a modified similarity measure for

[4]Although in some case the keyword selected by the user also showed high degree of recall.

equation (1). Especially, the terms have equal weight in that equation. We plan to give different weights to the terms to reflect the influence in the context of information retrieval.

5.2 As a Support System

In order to increase the quality of search (e.g., precision and recall), some approaches require the user to specify his/her profile and utilize it. A user profile is to describe the detailed information of each user so that some personalization can be enabled [13]. However, describing the detailed information could require lots of work for a user.

In our approach, instead of requiring much work on a user, a thesaurus, which describes the commonly shared semantic information, is utilized. On the other hand, since it is uniformly utilized for all user, it is not sufficient to enable the personalization. As suggested in Section 5.1, it would be necessary to adapt the utilization of the thesaurus for each user, for instance, by modifying the weight of the terms.

Currently no feedback from the user is utilized for further processing. It would be important to utilize such feedback, known as relevance feedback, in future work. In addition, the interaction between the system and the user would be beneficial to improve the performance [17].

6 Conclusion

This paper has proposed a model for representing the implicit relationship among the specified terms in this paper. Our model tries to capture it based on their semantic information and represent it as a tree structure. The semantic information among the terms is conceived using a thesaurus, and the degree of similarity is calculated based on the thesaurus. Evaluation was conducted to investigate the effectiveness of our proposal, especially within the context of information retrieval. The result suggested the effectiveness of our approach. Currently it is not yet realized the fully automatic selection of the “best” auxiliary term based on our current similarity measure. We plan to improve the measure and the algorithms to realize this in our future work.

Acknowledgments

This research is partially supported by the grant-in-aid for scientific research (No. 20500123) funded by MEXT, Japan. The author expresses gratitude to Mr. Daiki Shinkai for his contribution to the computational support.

References:

[1] D. Cabeza, M. Hermenegildo, and S. Varma. The PiLLoW/CIAO Library for INTERNET and WWW Programming using Computational Logic Systems. http://www.clip.dia.fi.upm.es/miscdocs/pillow /article/pillow.html, 1996.

[2] P.W. Foltz and S.T. Dumais. Personalized information delivery: An analysis of information filtering methods. *Communications of the ACM*, 35(12):51–60, 1992.

[3] T. Hagino. A Classification System of Meaning based on a Modern Japanese Noun Thesaurus, 1996. (in Japanese).

[4] Philip J. Hayes. On semantic nets, frames and associations. In *Proceedings of the Fifth International Joint Conference on Artificial Intelligence (IJCAI-77)*, pages 99–107, Cambridge, Massachusetts, August 1977.

[5] C.K. Huang, Y.J. Oyang, and L.F. Chien. A contextual term suggestion mechanism for interactive web search. In *Proceedings of Web Intelligence 2001*, pages 272–281, 2001.

[6] B. J. Janse, A. Spink, J. Bateman, and T. Saracevic. Real life information retrieval: A study of user queries on the web. *SIGIR FORUM*, 32(1):5–17, 1998.

[7] K.S. Jones and P. Willett. *Readings in Information Retrieval*. Morgan Kaufmann, 1997.

[8] S. W. Loke and A. Davidson. Logic web:enhancing the web with logic programming. *The Journal of Logic Programming*, 36:195–240, 1998.

[9] T.W. Malone, K.R. Grant, F.A. Turbak, S.A. Brobst, and M.D. Cohen. Intelligent information sharing system. *Communications of the ACM*, 30(5):390–402, 1987.

[10] R. Mitkov and N. Nicolov, editors. *Recent Advances in Natural Language Processing*. Benjamins, 1997.

[11] M. Morita and Y. Shinoda. Information filtering based on user behavior analysis and best match text retrieval. In *Proceedings of SIGIR-94*, pages 272–281, 1994.

[12] M. Nagao. *Natural Language Processing*. Iwanami Publisher, 1996. in Japanese.

[13] U. Shardanand and P. Maes. Social information filtering: Algorithm for automating 'word of mouth'. In *Proceedings of CHI'95*, pages 210–217, 1995.

[14] C. Silverstein, M. Henzinger, H. Marais, and M. Morics. Analysis of a very large altavista query log. Technical Report Technical Report 1998-014, Digital Systems Research Center, 1998.

[15] R.S. Sutton and .G. Barto. *Reinforcement Learning*. MIT Press, 1998.

[16] T. Yoshida. A Comment based Software Design Support Method with Cases. *WSEAS Transactions on Computers*, 5(6):1354–1360, 2006.

[17] T. Yoshida. A Human-Machine Cooperative Design Support System in Multi-Agent System Framework. *WSEAS Transactions on Computers*, 5(4):782–789, 2006.

[18] T. Yoshida. A Term Alignment Method for a Mixed Dataset via Dataset Partitioning. *WSEAS Transactions on Computers*, 5(12):3131–3138, 2006.