The KLE's Subtopic Mining System for the NTCIR-11 IMine Task

Se-Jong Kim
Pohang University of Science and Technology (POSTECH)
sejong@postech.ac.kr

Jaehun Shin
Pohang University of Science and Technology (POSTECH)
rave0206@postech.ac.kr

Jong-Hyeok Lee
Pohang University of Science and Technology (POSTECH)
jhlee@postech.ac.kr

ABSTRACT

This paper describes our subtopic mining system for the NTCIR-11 IMine task. We propose a method that mines second-level subtopics using simple patterns and a hierarchical structure of subtopic candidates based on sets of relevant documents, and combine the provided resources considering their characteristics. Our system generates first-level subtopics using keywords in second-level subtopics, and groups the results by word correlation.

Team Name

KLE (Knowledge and Language Engineering Laboratory)

Subtasks

Subtopic Mining (English, Japanese, Chinese)

Keywords

query intent, diversity, pattern, hierarchical structure, result combining

1. INTRODUCTION

Many web queries are ambiguous and broad. In the case of ambiguous queries, users may get results quite different from their information needs; as for broad queries, results may not be as specific as users expect.

As one of the solutions for these problems, subtopic mining is proposed, which can find possible subtopics (subtopic strings) for a given query and return a ranked list of them in terms of their relevance, popularity and diversity. According to INTENT and INTENT-2 tasks [1, 2], a subtopic of a given query is a query that disambiguates and specifies the search intent of the original query. For example, if a query is "*windows*," "*windows 8*" and "*windows update*" can be a subtopic. NTCIR-11 IMine task[3] proposes new subtopic mining that the two-level hierarchy of subtopics consists of at most five first-level subtopics and at most ten second-level subtopics for each first-level subtopic. Each type of subtopics has a own rank (Figure 1).

This paper describes our subtopic mining system for the NTCIR-11 IMine task. To get second-level subtopics, we mine subtopics using simple patterns and a hierarchical structure of subtopic candidates based on sets of relevant documents [4], and combine the provided resources considering their characteristics. To get first-level subtopics, our system generates them using keywords in second-level subtopics, and groups the results by word correlation. The provided resources are as follows:

Query : *windows*

1st-level Subtopic	Rank1	2nd-level Subtopic	Rank2
Microsoft Windows	1	*Windows 8*	1
Microsoft Windows	1	*Windows phone 8*	4
Microsoft Windows	1	*Windows update*	5
House Windows	2	*Windows repairing*	2
House Windows	2	*Window glass sale*	3

Figure 1: The two-level hierarchy of subtopics and rank.

- Suggested queries: ranked lists of suggested queries from major web search engines (Bing, Google, Sogou, Yahoo!, Baidu) for English, Japanese, and Chinese.
- Query dimensions [5]: groups of items extracted from the style of lists in top retrieved documents for English, Japanese, and Chinese.
- Related queries [6]: a ranked list of Chinese related queries using Sogou query log.
- Web documents: English document collection ClueWeb12-B13, Japanese document collection ClueWeb09-JA (mentioned by INTENT-2), top 200 Chinese documents for each query from SogouT.

A description of the proposed method is given in Section 2 and 3. In Section 4, our results are presented, and in the final section, we give the discussion and conclusion.

2. SECOND-LEVEL SUBTOPIC MINING

2.1 Second-level Subtopic Extraction

A subtopic is assumed to consist of an original query and one or more noun phrases that make the original query more specific. In general, a word can be specified by its other co-occurring words, and we can also find subtopic candidates using several words that co-occur with the query in documents. From the assumption, we create a simple pattern to extract appropriate subtopic candidates:

- *Pattern 1*: $((\text{adjective})^{?}(\text{noun})^{+}(\text{non-noun})^{*})^{?}(\text{query})$ $((\text{non-noun})^{*}(\text{adjective})^{?}(\text{noun})^{+})^{?}$

where the ? operator means "zero or one"; the + operator "one or more"; and the * operator "zero or more."

Pattern 1 is applied to the top 1,000 relevant documents for the query. Since this pattern covers real phrases that consist of the whole query and noun phrases in the documents, the subtopic candidates we find are truly relevant.

However, if a query consists of two or more words, we cannot thoroughly extract various subtopic candidates using only *Pattern 1* from the retrieved documents, because the number of subtopic candidates that fully match the original query decreases. Therefore, we make alternative partial-queries q_{left} and q_{right} from the original query to extract various subtopic candidates by meaningful partial matching. For the left phrases of the original query, which are the remaining words after consecutively removing the right words of the query, we retrieve the top 200 relevant documents for each, and compare these documents with the top 200 relevant documents for the query. If the relevant documents for one of the phrases cover more than 100 documents in the relevant documents for the original query, we regard this phrase as an alternative partial-query candidate. Among alternative partial-query candidates that cover the most documents, we select the shortest candidate as q_{left}. If none of the phrases satisfies this condition, we select the longest phrase as q_{left}. For the right phrases of the original query, which are the remaining words after consecutively removing the left words of the query, we select q_{right} by applying the same process. Using q_{left} and q_{right} instead of the query, we create new simple patterns:

- *Pattern 2*: ((adjective)$^?$(noun)$^+$(non-noun)*)$^?$($\underline{q_{left}}$) ($\underline{\text{(word)}^*(q_{right})}$)((non-noun)*(adjective)$^?$(noun)$^+$)$^?$
- *Pattern 3*: ($\underline{q_{right}}$)(non-noun)*(adjective)$^?$(noun)$^+$
- *Pattern 4*: (adjective)$^?$(noun)$^+$(non-noun)*($\underline{q_{left}}$) .

Using these new patterns, we find various phrases from the retrieved documents, and replace the parts of these phrases corresponding to the underlined patterns with the original query. These found and replaced phrases are subtopic candidates. Even if these subtopic candidates are not real phrases in the documents, we can reduce data sparseness and improve diversity.

We filter similar subtopic candidates to reduce their redundancy. Let s_{np} be a set of lemmas of noun phrases "(adjective)$^?$(noun)$^+$" at the start or end of each subtopic candidate. If two or more subtopic candidates have the identical s_{np}, they are regarded as similar because s_{np} includes important keywords that decide the meaning of each subtopic candidate. Therefore, we merge the frequency information of similar subtopic candidates, and select the most frequent and short subtopic candidate among them.

2.2 Second-level Subtopic Ranking

To rank subtopic candidates, we first propose a three-level hierarchical structure of subtopics. The root is a given query, its child node "primary subtopic," and each leaf node "secondary subtopic," respectively. The primary subtopics are disambiguated and initially-specified search intents, and a group of which may be chosen to satisfy global diversity. The secondary subtopics are more specified to narrow down the search intent of primary subtopics, which affect an improvement of local diversity.

To construct the proposed hierarchical structure, top 200 relevant documents for a given query are assumed to represent the whole search intents of the query anyhow, and the appearance of subtopic candidates in documents is interpreted as covering some search intents. Based on this insight, first we select a relatively small group of subtopic candidates as primary subtopics in order that they may appear in as many relevant documents as possible. Since the primary subtopics should show clear distinction among search intents, each relevant document generally includes one of the primary subtopics. Therefore, a set of relevant documents containing a primary subtopic seldom overlaps with the other sets of relevant documents containing the other primary subtopics. Furthermore, a set of relevant documents containing a primary subtopic generally includes some subsets of relevant documents that contain its secondary or other subtopics. In other words, a primary subtopic is quite distinct from its secondary or other primary subtopics in terms of the overlapping of their corresponding document sets. Thus, the Distinctness Entropy (DE) [7] for document sets can be used to select top n ($n \leq 200$) primary subtopic candidates:

$$DE(st) = -\sum_{st' \in ST, st' \neq st} \frac{|D(st) \cap D(st')|}{|D(st)|} log \frac{|D(st) \cap D(st')|}{|D(st)|} \quad , \tag{1}$$

where st is a subtopic candidate; ST is the set of all subtopic candidates that were extracted by the previous step (Section 2.1) from the top 200 relevant documents for a given query; and $D(st)$ is the set of relevant documents containing st.

If a document set has a high value of DE, its corresponding subtopic candidate is selected as a primary subtopic candidate. Meanwhile, to increase the number of documents covered by a primary subtopic, several document sets for primary subtopic candidates can be merged into a single large one if they are similar to each other in terms of cosine similarity. The primary subtopic candidate with the highest popularity is selected as the name of the merged set of documents. To estimate the popularities, we use the Sum of the values of *TF-IDF* ($STFIDF$):

$$STFIDF(st) = \sum_{doc \in R_q} freq(st, doc) log \frac{|R_q|}{|D(st, R_q)|} \quad , \tag{2}$$

where R_q is the set of the top 1,000 relevant documents for the query; $freq(st, doc)$ is the frequency of st in a document doc; and $D(st, R_q)$ is the set of documents in R_q containing st.

From the refined primary subtopic candidates, we select the best primary subtopic with the maximum value of our proposed measure, the search Intent Coverage (IC):

$$IC(st, US) = \frac{|D(st) \cap US^c|}{|\bigcup_{sc \in SC} D(sc)|} DE(st) \quad , \tag{3}$$

where US is the union of sets of relevant documents containing the previously selected primary subtopics; and SC is the set of all primary subtopic candidates.

IC checks how many of the top 200 relevant documents are covered by a primary subtopic candidate, and whether the set of documents for this candidate satisfies high distinctness. Primary subtopics are continuously selected using IC. If $|US|$ is equal to $|\bigcup_{sc \in SC} D(sc)|$, the selection process stops

because the selected primary subtopics can cover all relevant documents, which were covered by all primary subtopic candidates. For each of the selected primary subtopics, its secondary subtopics can be selected in the same way recursively as the primary subtopic except that we only use the relevant documents containing the primary subtopic.

The groups of primary and secondary subtopics are relatively small groups satisfying the global and local diversity of search intents, and there is an inheritance of popularity between a primary subtopic and its secondary subtopics. Furthermore, users generally want to know about the subtopics with high popularity in detail. From these characteristics, we propose a ranking order of subtopics along the three-level hierarchical structure so as to keep a balance between popularity and diversity (Figure 2). Primary subtopics are ranked first to achieve the global diversity using few subtopics, then secondary subtopics of the first-ranked primary subtopic are ranked to consider the inherited high popularity from the parent node and the local diversity, and then secondary subtopics of the next-ranked primary subtopic are ranked sequentially, by *STFIDF*.

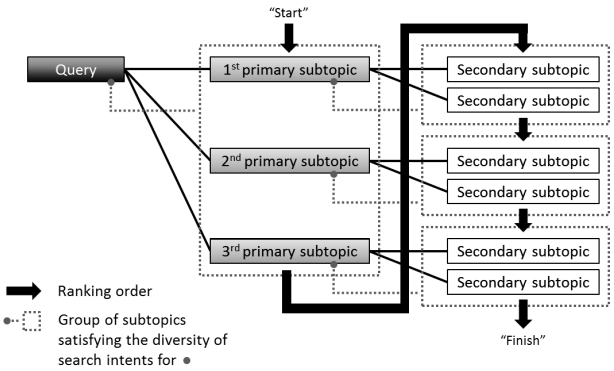

Figure 2: A ranking order and the groups of subtopics satisfying the diversity of search intents for parent nodes.

2.3 Result Combining and Re-ranking

Our system combines and re-ranks the ranked lists of subtopics using the provided resources. As mentioned earlier, the given suggested queries are consisted of several ranked lists of them from major web search engines. These ranked lists are merged into one ranked list by the Global Score (GS):

$$GS(sq) = \sum_{ws \in WS} \frac{N(ws) - (rank(sq, ws) - 1)}{N(ws)} \quad , \qquad (4)$$

where sq is a suggested query; WS is the set of web search engines; $N(ws)$ is the total number of suggested queries from ws; and $rank(sq, ws)$ is the rank of sq among all suggested queries from ws.

Each of the given query dimensions and its items are ranked by [5]. However, each item may not be relevant to a query because they are just extracted from the style of lists in the retrieved documents. Therefore, if two or more items in a query dimension are directly relevant to the query, all items in this dimension are regarded as subtopics. The condition of directly relevant item is that a primary, secondary subtopic or suggested query contains the item, or it includes the query. For these results, we assume that:

- *Assumption 1*: Subtopics (items) in query dimensions satisfy the high diversity.
- *Assumption 2*: Suggested queries are good subtopics which satisfy the high popularity.

To improve the diversity of second-level subtopics by *Assumption 1*, we insert results of query dimensions to the ranked list of primary and secondary subtopics (Figure 3 (a)). First of all, if a primary subtopic contains one of subtopics in a query dimension, the corresponding subtopic in the dimension is replaced with the primary subtopic, and the original place of the primary subtopic is also replaced with the ranked list of subtopics in the dimension. Note that the replaced subtopics are regarded as primary subtopics. If any primary subtopic does not contain one of subtopics in a query dimension, the top subtopic and others in the dimension are regarded as an additional primary subtopic and its secondary subtopics, respectively. The additional primary subtopic is inserted to the next of the last primary subtopic in the ranked list of subtopics, and its secondary subtopics are added to the end of the list.

From *Assumption 2*, we reflect the high popularity of suggested queries to second-level subtopics (Figure 3 (b)). The details are as follows:

- *Method 1*: If a primary subtopic contains the i-ranked suggested query, this primary subtopic is re-ranked as the i-ranked primary subtopic. The non-matched suggested query is inserted to the next of the last primary subtopic and deleted from the ranked list of suggested queries.
- *Method 2*: *Method 1* + The secondary subtopics of the re-ranked primary subtopic are re-ranked by the ranking order (Section 2.2).
- *Method 3*: If a primary subtopic contains the i-ranked suggested query, this primary subtopic is re-ranked as the i-ranked primary subtopic. The non-matched j-ranked suggested query is inserted to the front of the j-ranked primary subtopic.
- *Method 4*: *Method 3* + The secondary subtopics of the re-ranked primary subtopic are re-ranked by the ranking order (Section 2.2).

3. FIRST-LEVEL SUBTOPIC MINING

3.1 First-level Subtopic Generating

To generate first-level subtopic candidates, we select keywords such as front terms, back terms, last words, and relevant items of query dimensions in second-level subtopics. Each of the selected keywords is attached to the appropriate position of the query, and these expanded phrases are first-level subtopic candidates of the corresponding second-level subtopic (Figure 4).

3.2 Result Grouping and Ranking

To maintain the quality of second-level subtopics, our system does not choose any strong criteria about constructing the two-level hierarchy of subtopics. We just consider

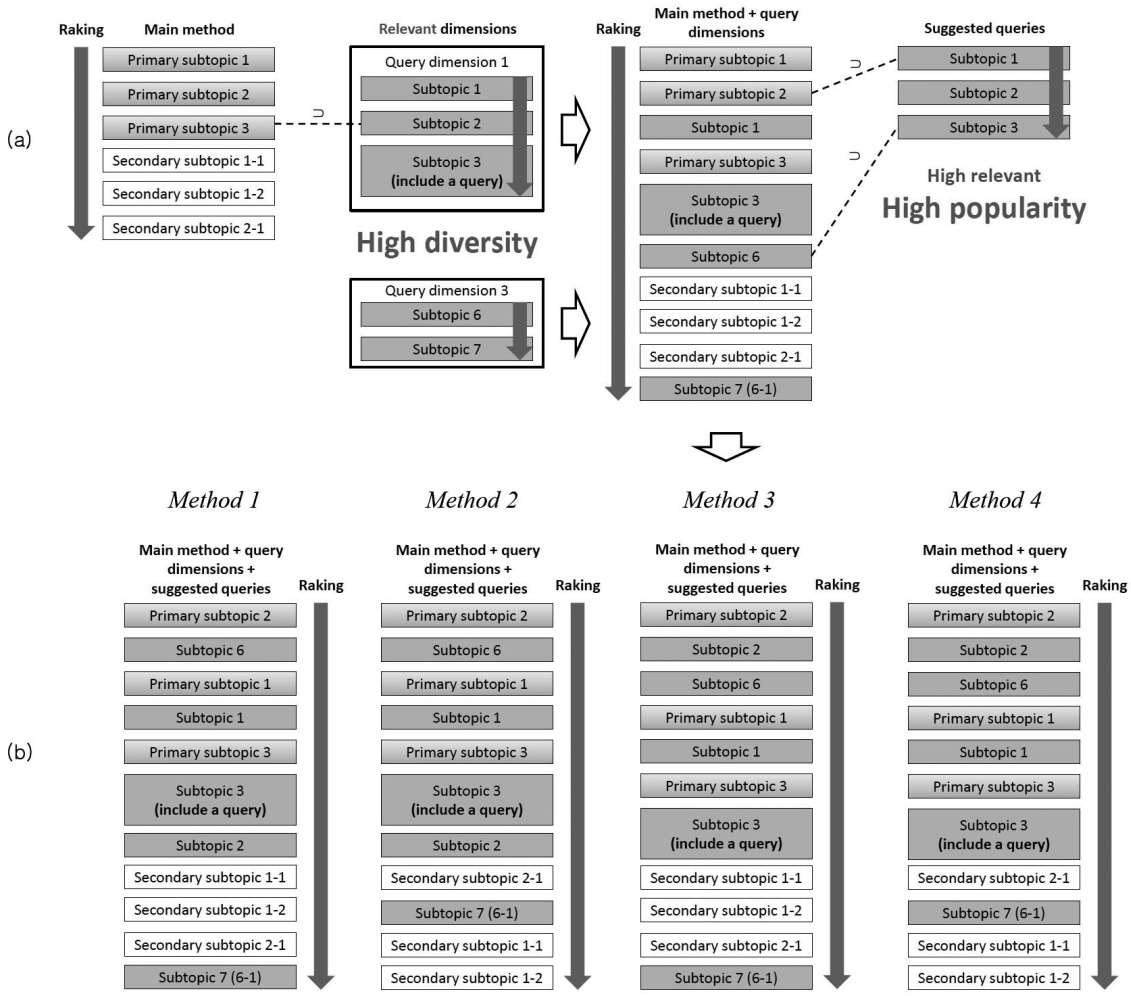

Figure 3: A process of result combining and re-ranking for each *Method*.

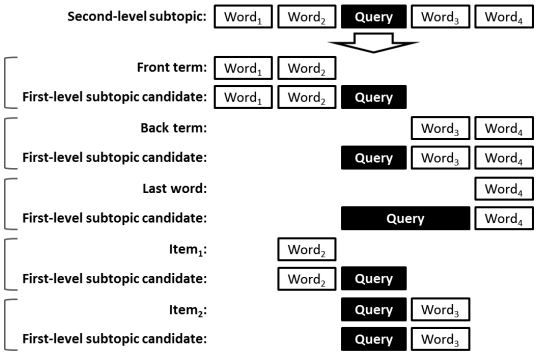

Figure 4: Keywords and generated first-level subtopic candidates.

the Pointwise Mutual Information (PMI) of keywords which were selected by the previous step (Section 3.1):

$$PMI(x, y) = log \frac{p(x, y)}{p(x)p(y)} \quad , \tag{5}$$

where x and y are keywords; $p(x, y)$ is the number of documents containing x and y divided by the number of relevant documents for a query; and $p(x)$ is the number of documents containing x divided by the number of the relevant documents.

We assume that if some second-level subtopics contain the identical keyword, the first-level subtopic based on the keyword has a relationship with these second-level subtopics. For keywords that appear in top N second-level subtopics, top 5 distinct groups of two keywords (keyword pairs) are selected from a ranked list of them by PMI, and the sets of second-level subtopics for each keyword in the same group are merged into a large one. We continuously increase N and expand each of the selected groups of keywords and its set of second-level subtopics by setting x and y in PMI to one keyword in the selected group and a new keyword, respectively (Figure 5).

In each group of keywords, if the sum of PMIs of a fixed keyword and unfixed one of others has the maximum value, the first-level subtopic candidate based on the fixed keyword is selected as the first-level subtopic of the corresponding second-level subtopics. The first-level subtopics are ranked

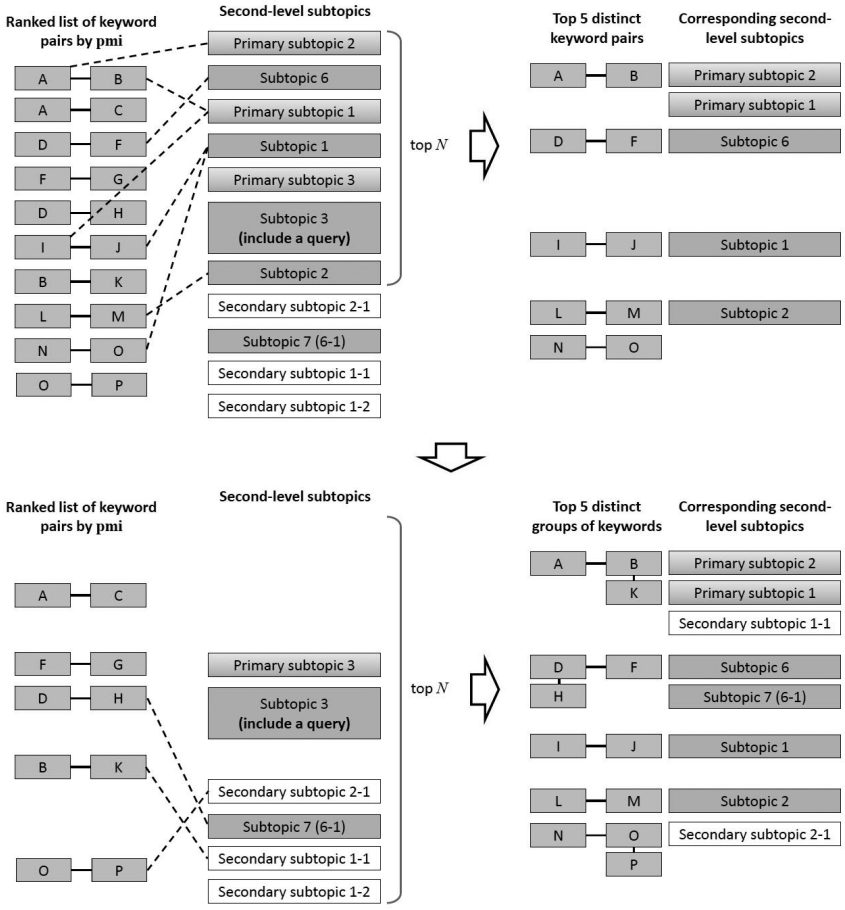

Figure 5: A process of result grouping and expansion.

by only the highest rank of second-level subtopics for each of them.

4. EVALUATION RESULTS

4.1 Overview

We mined subtopics for English, Japanese, and Chinese queries (topics) of the NTCIR-11 subtopic mining subtask. We used only the provided resources. Especially, the given list of Chinese related queries was regarded as the ranked list of primary subtopics (we did not mine additional primary and secondary subtopics using the documents). For English and Japanese document retrieval, the search interface by Lemur project[1] and BM25 model [8] are used, respectively. To perform word segmentation and identify noun phrases, we used the English Stanford POS tagger[2] and the Japanese MeCab POS tagger[3].

Our run names were "KLE-S-E(English)/J(Japanese)/C(Chinese)-1A(*Method 1*)/2A(*Method 2*)/3A(*Method 3*)/4A(*Method 4*)." The results were evaluated using *Hscore* (relationship of first-level subtopic and its second-level subtopics), *Fscore* (quality of first-level subtopics), *Sscore* (quality of second-level subtopics), and *H-measure* (representative measure) [3, 9, 10].

[1]http://lemurproject.org/clueweb12/
[2]http://nlp.stanford.edu/software/tagger.shtml
[3]http://mecab.sourceforge.net

4.2 Official English Subtopic Mining Results

For English, our *H-measure* values of KLE-S-E-1A, KLE-S-E-2A, KLE-S-E-3A, and KLE-S-E-4A were 0.0873, 0.0893, 0.0980, and 0.0938, respectively (Table 1). Our best value of *H-measure* was 0.0980 of KLE-S-E-3A [3].

Table 1: Runs sorted by *H-measure* over 33 unclear topics for English.

Run	*Hscore*	*Fscore*	*Sscore*	*H-measure*
KLE-S-E-3A	**0.1291**	**0.6539**	0.7317	**0.0980**
KLE-S-E-4A	0.1260	0.6511	0.7294	0.0938
KLE-S-E-2A	0.1200	0.5698	**0.7342**	0.0893
KLE-S-E-1A	0.1185	0.5591	0.7298	0.0873

4.3 Official Japanese Subtopic Mining Results

For Japanese, our *H-measure* values of KLE-S-J-1A, KLE-S-J-2A, KLE-S-J-3A, and KLE-S-J-4A were 0.0853, 0.0908,

0.1038, and 0.1008, respectively (Table 2). Our best value of *H-measure* was 0.1038 of KLE-S-J-3A [3].

Table 2: Runs sorted by *H-measure* over 34 unclear topics for Japanese.

Run	*Hscore*	*Fscore*	*Sscore*	*H-measure*
KLE-S-J-3A	**0.2030**	0.4416	**0.5086**	**0.1038**
KLE-S-J-4A	0.2025	0.3920	0.4997	0.1008
KLE-S-J-2A	0.1867	**0.4502**	0.4697	0.0908
KLE-S-J-1A	0.1759	0.4372	0.4509	0.0853

4.4 Official Chinese Subtopic Mining Results

For Chinese, our *H-measure* values of KLE-S-C-1A, KLE-S-C-2A, KLE-S-C-3A, and KLE-S-C-4A were 0.3303, 0.3360, 0.3255, and 0.3279, respectively (Table 3). Our best value of *H-measure* was 0.3360 of KLE-S-C-2A [3].

Table 3: Runs sorted by *H-measure* over 33 unclear topics for Chinese.

Run	*Hscore*	*Fscore*	*Sscore*	*H-measure*
KLE-S-C-2A	**0.5413**	**0.5736**	0.6339	**0.3360**
KLE-S-C-1A	0.5306	0.5666	0.6360	0.3303
KLE-S-C-4A	0.5148	0.4986	0.6640	0.3279
KLE-S-C-3A	0.5072	0.4817	**0.6718**	0.3255

5. DISCUSSION AND CONCLUSION

Our proposed methods achieved high performances for second-level subtopics. For each language, the performance differences of *Sscore* between our methods and others were statistically significant [3]. The results for first-level subtopics were also good. However, each *Hscore* of our results was too low. Especially, for English and Chinese, although each first-level subtopic and its second-level subtopic belonged to the same search intent, a large portion of this case did not satisfy the hierarchical relation (Figure 6).

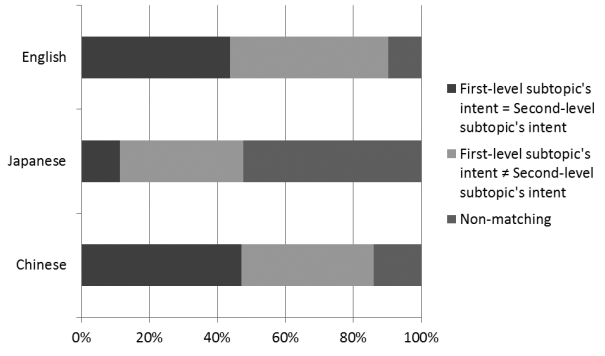

Figure 6: Error case ratios of *Hscore* for each language.

The main reason of these low performances is that we consider only the word correlation to check the hierarchical relation of subtopics. This criterion is too weak to solve this problem. Therefore, we have to find some appropriate criteria to improve *Hscore*, and research the relationship of first-level subtopic and corresponding second-level ones.

6. ACKNOWLEDGMENTS

This work was partly supported by the IT R&D program of MSIP/KEIT (10041807), the SYSTRAN International corporation, the BK 21+ Project, and the National Korea Science and Engineering Foundation (KOSEF) (NRF-2010-0012662).

7. REFERENCES

[1] R. Song, M. Zhang, T. Sakai, M. P. Kato, Y. Liu, M. Sugimoto, Q. Wang, and N. Orii. Overview of the NTCIR-9 INTENT task. In *Proceedings of NTCIR-9 Workshop Meeting*, pages 82–105, 2011.

[2] T. Sakai, Z. Dou, T. Yamamoto, Y. Liu, M. Zhang, and R. Song. Overview of the NTCIR-10 INTENT-2 task. In *Proceedings of NTCIR-10 Workshop Meeting*, pages 94–123, 2013.

[3] Y. Liu, R. Song, M. Zhang, Z. Dou, T. Yamamoto, M. Kato, H. Ohshima, and K. Zhou. Overview of the NTCIR-11 IMine task. In *Proceedings of NTCIR-11 Workshop Meeting*, 2014.

[4] S.-J. Kim and J.-H. Lee. Subtopic mining using simple patterns and hierarchical structure of subtopic candidates from web documents. *Information Processing & Management*, 2014 (submitted).

[5] Z. Dou, S. Hu, Y. Luo, R. Song, and J.-R. Wen. Finding dimensions for queries. In *Proceedings of ACM CIKM 2011*, pages 1311–1320, 2011.

[6] Y. Liu, J. Miao, M. Zhang, S. Ma, and L. Ru. How do users describe their information need: Query recommendation based on snippet click model. *Expert Systems With Applications*, 38(11):13847–13856, 2011.

[7] H.-J. Zeng, Q.-C. He, Z. Chen, W.-Y. Ma, and J. Ma. Learning to cluster web search results. In *Proceedings of ACM SIGIR 2004*, pages 210–217, 2004.

[8] S. Robertson and H. Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009.

[9] T. Sakai. NTCIREVAL: A generic toolkit for information access evaluation. In *Proceedings of the Forum on Information Technology 2011*, volume 2, pages 23–30, 2011.

[10] T. Sakai and R. Song. Evaluating diversified search results using per-intent graded relevance. In *Proceedings of ACM SIGIR 2011*, pages 1043–1052, 2011.