HUKB at NTCIR-12 IMine-2 task: Utilization of Query Analysis Results and Wikipedia Data for Subtopic Mining

Masaharu Yoshioka
Graduate School of Information Science and Technology, Hokkaido University
N14 W9, Kita-ku, Sapporo-shi,Hokkaido Japan
yoshioka@ist.hokudai.ac.jp

ABSTRACT

Query understandings is a task to identify the important subtopics of a given query with vertical intent. In this task, characteristic keywords extracted from query analysis results and Wikipedia are used as candidates for the subtopics. From these candidates, topic-model based on the web documents retrieved by an original query is used for selecting appropriate subtopics, Vertical intent is judged mainly by the typical keyword list used for the particular vertical intent. For the Image, News and Shopping, the system checks type of retrieved documents that are estimated by using ALT value of IMG tag, anchor text and site list for URLs for vertical intent estimation.

Team Name

HUKB

Subtasks

Query Understanding Subtask (Japanese)

1. INTRODUCTION

In order to support novice web search users, it is helpful to find out subtopics of a given query and use such information too find out more focused retrieval results that satisfy users' need. Query understandings task in NTCIR-12 IMine-2 [2] is a task to identify the important subtopics of a given query. In addition to identity the subtopics, it is also required to classify each topic into vertical intent such as Image, News, Shopping.

Query suggestion is one of a useful method that navigates the user to find out such subtopics. However, since suggested queries are not well diversified, it is not so appropriate to find out important subtopics that are not frequently used. In this participation, we propose a method to diversify query suggestion results based on the analysis of queried documents by using topic model[1]. In addition, we also used information extracted from Wikipedia[1] and characteristic keywords for each topic extracted by the topic model as subtopic candidates. For estimating vertical intent, we use typical keyword list used for the particular vertical intent. In addition tot the list, the system estimates retrieved page types for given subtopics by using contents information. Contents information such as ALT value of IMG tag, anchor text, contents text is used for Image, News and Shopping. Site list constructed by using Open Directory Project [2] is also used for estimating page types of News and Shopping.

2. IMPLEMENTATION

2.1 Subtopic candidates generation

In this framework, we use following three main information resource for selecting subtopic candidates. One is information based on the query analysis and the other is Wikipedia.

Followings are summary about information resource.

- Query analysis results
  - Query suggestion: Query suggestion data for Bing related API, Google completion, and Yahoo! suggestion provided by organizers are used.
  - Related queries extracted from Yahoo search logs: IMine co-click and co-topic queries are used. We use top 30 from the list.
- Wikipedia
  - Ambiguity Checker: When corresponding Wikipedia page of the given query term has information about ambiguity, the system selects keywords for disambiguation as subtopic candidates.
  - Yahoo! abstract: When there is a Wikipedia page for the given query, "目次"(contents) titles of Wikipedia pages are used as subtopic candidates.
- Characteristic keywords for each topic extracted by the topic model[1].
  Top 20 characteristic keywords for each topic in the constructed topic model are used as subtopic candidates.

[1] We used Japanese Wikipedia dumped data 2015/10/20 version.

[2] https://www.dmoz.org/World/Japanese/

For the query analysis results, there are several cases that query analysis results contains different expression of the same word (e.g., for the original query IMINE-J-042 "ねこ"(cat in Hiragana), "猫"(cat in Chinese character) and "ネコ"(cat in Katakana) are included in co-click). Therefore, we use Juman [3] to normalize suggested subtopic candidates into normalized their expression and exclude candidates whose normalized terms are included in the original query.

For the Wikipedia, ambiguity of the terms are checked by using category information of corresponding Wikipedia page. When the page have a category that contains "曖昧"(Ambiguous), we extract ambiguous candidates of the given query from Wikipedia page links in the page. Followings are list of Wikipedia page links extracted from the page.

- Since in most of the case, title of the ambiguous page are described as a combination of original keyword + disambiguation information (such as "音楽グループ" (musicians)), pages that contains original keywords are selected for candidates. In these cases, disambiguation information parts are used for subtopic candidates.

- For the abbreviation case (e.g., "cvs"), most of the pages title are original name before the abbreviation. In order to deal with such ambiguous cases, we check the contents of the linked page and check whether original keywords exist or not.

In Wikipedia page, "目次"(contents) represents characteristic subtopics of the page. Therefore, we collect contents keywords from Yahoo's active abstract project data[4]. However, following Wikipedia related contents keywords are excluded from the candidates "外部リンク"(External links), "参考文献"(References), "注釈"(Notes), "出典"(Sources) "概要"(Abstract), "関連項目"(Related pages) , "その他"(Other), "脚注"(footnote), "曖昧さ回避"(Disambiguation).

2.2 Subtopic mining

In order to keep the diversity of subtopics, we use topic model [1] for analyzing topics in the relevant documents and select subtopics based on their representative topics.

When we check the retrieved results provided by organizers, there are several cases that includes many irrelevant documents especially for queries with two or more terms. Since quality of the relevant documents may affect the quality of topic model, we use two strategy for selecting relevant documents D_{Rel} . One is using all retrieved results provided by organizers as relevant documents. The other is selecting retrieved documents that have query keywords closer in the documents. Snippet function is a function to extract a part of document that contains given keywords in given text window size. When there is a part of documents that contains all given keywords in the window size, the system can extract that part as a candidate for the snippet. We create function $Snipselect(D, keywords)$ to select documents form a document set D by using this snippet function to check whether given $keywords$ are closely located or not in the retrieved documents. In this case, $Snipselect(D_{Rel}, original\ query)$ is used for topic model construction.

In order to extract 10 subtopics for each query, it is necessary to set the number of topics n_t for the topic model larger than 10. In this experiment, we use $n_t = 30$ for all experiments.

In this framework, we evaluate the appropriateness of the subtopic candidates by using size of documents that contains a particular topic and representativeness of the keywords for the topic. Followings are procedures to calculate the score.

1. Calculate representative topics for each retrieved document
   As a result of topic model construction, the variational Dirichlet parameters for each document is calculated. Since this value corresponds to representativeness of the topic for each document, we use this value $dtr_{tid,did}$ for analyzing representative topics, where tid represents topic number and did represents document id of a retrieved document. In this model, even though there is no corresponding keywords for a particular topic in the document, non-zero value is assigned to those topics for smoothing. However, those value is no meaning for our analysis, we set 0 instead of the lowest common positive values assigned for those topics.

2. Calculate representative topics for each subtopic candidate.
   Topic model inference is used for the subtopic candidates for estimating the representative topics. Same procedure for calculating value of documents are used for this value $qtr_{tid,sid}$, where tid represents topic number and sid represents subtopic candidates id.

3. Calculation of representativeness of the candidates for each topic.
   In order to calculate representativeness of candidates for the topic, we calculate Jacakard coefficient between documents related to the topic ($Dt_{tid} = \{did|dtr_{tid,did} > 0\}$) and documents retrieved from the original retrieved documents that contains subtopic candidates of sid (Dr_{sid}).

   $$rep_{tid,sid} = |Dt_i \cap Dr_{sid}|/|Dt_i \cup Dr_{sid}| \qquad (1)$$

4. Calculate score based on the representativeness of the topic and retrieved document size of topic.
   For each candidate, the representativeness of topic $Frep_{tid,sid}$ are calculated for all representative topic and documents by using following equation and select highest one Rt_{sid} as representative topics.

   $$\begin{aligned} Frep_{tid,sid} &= rep_{tid,sid} * |Dt_i| && (2)\\ Rt_{sid} &= \operatorname*{argmax}_{tid \in \{tid|qtr_{tid,sid}>0\}} Frep_{tid,sid} && (3)\\ Rtrep_{sid} &= Frep_{Rt_{sid},sid} && (4)\\ & && (5) \end{aligned}$$

Subtopic candidates are categorized into three groups.

1. Subtopic candidates extracted from Wikipedia and query analysis results.
   It is preferable to have certain amount of documents in which original query and subtopic keywords exists in a given text window size. Snippet ratio Sr_{sid} is calculated by using following equation.

   $$Sr_{sid} = \frac{|Snipselect(Dr_{sid}, all\ keywords)|}{|Dr_{sid}|} \qquad (6)$$

[3] http://nlp.ist.i.kyoto-u.ac.jp/index.php?JUMAN

[4] https://dumps.wikimedia.org/jawiki/20151020/jawiki-20151020-abstract.xml

where *all keywords* includes all original query keywords and subtopic keywords. Subtopic candidates with $Sr_{sid} \leq Sr_t$ ($Sr_t = 0.3$ is used for all experiments) belong to the first group of subtopic candidates. Candidates with $Sr_{sid} < Sr_t$ belong to second group.

2. Subtopic candidates generated from topic model
   Subtopic candidates generated from topic model is used as third group.

In this experiment, we check the subtopic candidates based on these categories. In order to add one score for the final submitted results, we add $cat_{score} > \max Rtrep_{sid}$ for the first and second categories; i.e., final score of candidates belongs to the first categories, second categories, and third categories are $Score_{sid} = 2 * cat_{score} + Rtrep_{sid}$, $Score_{sid} = cat_{score} + Rtrep_{sid}$, $Score_{sid} = Rtrep_{sid}$ respectively. In this experiment $cat_{score} = 150$ is used.

After calculating scores, all candidates are sorted by using $Score_{sid}$ and check them from the highest score. In order to keep the diversity of the subtopics, following rules are used for selecting the final results.

- Exclude subtopic candidates whose representative topic RT_{sid} is already included in the selected subtopic candidates.
- Exclude subtopic candidates whose retrieved results (Dr_{sid}) is similar to the results of already selected subtopic candidates.

$$\max_{ssid \in selected} |Dr_{ssid}|/|Dr_{sid}| \; > \; dsim \qquad (7)$$

$$\max_{ssid \in selected} |Dr_{sid}|/|Dr_{ssid}| \; > \; dsim \qquad (8)$$

  where $dsim$ is a parameter to control this similarity calculation. $dsim = 0.6$ is used for all experiments.

Finally, at most 10 candidates are selected as final results.

2.3 Vertical intent estimation

In Japanese task, it is required to estimate vertical intent into following categories; i.e., Web, Image, News, QA, Encyclopedia, and Shopping.

The strategy for vertical intent estimation is simple. For all categories, we make the list of subtopic keywords to estimate the vertical intent. Simple keyword matching is used for the estimation. For Shopping and News, we construct site list for the Shopping and News and decide the vertical intent based on the percentage of the shopping or news pages in retrieved results. We also used ALT value of IMG tag for checking keyword related to the Image is used for finding out image or not. Finally the system cannot estimate its vertical intent by using these information, the system returns Web as vertical intent.

Followings are detailed explanation about this estimation process. Generally speakings, there are particular types of subtopic for characterizing the vertical intents. First, we make the list of keywords to characterize those intent. Since Web is a category that is used for others, we make the list for Image, News, QA, Encyclopedia, and Shopping. In addition, Videos is one of the major vertical intent. However, there is no corresponding category and it should be dealt as Web. Therefore, we use keywords for Videos as keywords for Web.

By using this keywords, following procedures are used for estimating vertical intents.

Table 1: Keywords list for estimating vertical intents

Vertical intent	keywords
Image	イラスト (illustration), 画像 (image), 絵 (picture), 似顔絵 (portrait), 写真 (Photo), 壁紙 (Wallpaper)
News	news, ニュース (news), 記事 (article), 速報 (Breaking news)
Shopping	注文 (Order), 価格 (price), 値段 (price), 料金 (price), 相場 (market price), 費用 (Cost), 安い (Cheap), 格安 (Very cheap)
QA	質問 (Question), 方法 (Method)
Encyclopedia	内容 (Contents), 意味 (Meaning), 知識 (Knowledge), 辞書 (Dictionary), Wikipedia
Web	動画 (Video), Youtube

1. Basically, type of extended keywords are decided by comparing terms with keyword list in Table 1. For QA, News, Shopping, Encyclopedia and Web, when there is a keyword for each intent, the system returns corresponding intent as a result. For the image, system checks the existence of corresponding image in the retrieved results by using ALT value of IMG tag in the retrieved HTML pages. In order to show the retrieved results of Image vertical intent, it is preferable to have images for the original query. Therefore, the system check the ALT value of all IMG tags in the corresponding retrieved documents. When there are three or more web page exists, the system returns Image as vertical intent.

2. The same procedures are conducted for original query. However, due to the bugs of the system, the system didn't return Encyclopedia for this case (i.e., All subtopics for J-095 "生ビールの生"(draft of draft beer) "意味"(meaning) should be dealt as Encyclopedia, but some of the subtopic are categorized as Web in the submitted run.).

3. Vertical intent Shopping is checked by using the retrieved results. We check the simple program to check whether a retrieved page is Shopping page or not. One program checks URL of the retrieved page. We correct a shopping site list from Open Directory Project [5]. All URLs that belongs to "オンラインショップ"(online shop) category or subcategory of "オンラインショップ"(online shop) are candidates for the shopping site. If URL of the retrieved page belongs to those site are classified as shopping pages. The other program compares the anchor text and keywords that are characteristically exist in the shopping page. In this experiment we use following four keywords "注文"(order), "支払"(payment), "送料"(shipping charge), "買い物カゴ"(cart). When the retrieved page have at least one keywords as anchor texts, we classify the page as a shopping page. When the number of shopping page is larger than 5 and half of the retrieved pages are shopping pages, the system returns Shopping as vertical intent.

[5] https://www.dmoz.org/World/Japanese/

4. Vertical intent News is also checked by using the retrieved results. We also construct news site list from Open Directory Project [6]. All URLs that belongs to "ニュース"(news) category or subcategory of "ニュース"(news) are candidates for the news site. In addition, we also correct URLs from Google news from 2015/11/13-2015/11/20 and extract site name for the candidates of the news site. In the news case, there are several cases that retrieved results includes old news or column pages of the news site. Since those pages are not News page based on its definition, we check whether the news page contains year keyword (2015 or "平成 27"(Heisei27)), the system classified the pages as news pages. When the number of news page is larger than 10 and half of the retrieved pages are news pages, the system returns News as vertical intent.

5. For the subtopic candidates that can not be identified its vertical intent, the system returns Web as vertical intent.

3. SUBMIT RUNS

Our system is constructed based on the information provided by the organizers. IMine2-J-WebCorpus is used for retrieved document results and IMine2-QuerySuggestions is for the query suggestion data. In addition, Yahoo! Search query data of imine_coclick and imine_cotopic is also used for query analysis data.

At first, we extract text information from IMine2-J-WebCorpus by using text based web browser w3m. There are several cases that w3m fails to extract appropriate texts from the HTML. First group is failure based on a coding problem. For example, original files are encoded as euc-jp with code specification by using meta tag (e.g., IMINE2-J-001-0001.html), but distributed files are converted as UTF-8. The other group is PDF files (e.g., IMINE2-J-001-0037.html). In both cases, w3m generates inappropriate extracted texts. However, most of such texts are excluded for the snippet based selection settings.

We implement our system by using groonga [7] as full-text search engine and lda-c [8] implemented by Blei for topic model. For the snippet calculation, groonga is used for text window size 500.

We submit 5 runs for the official results. Table 2 represents different options used for each run. For the documents, Snippet means using snippet based selected documents are used for the topic model construction. Query suggestion means usage of query suggestion data provided by the organizers or not. Yahoo! Search logs means usage of related queries extracted from Yahoo! search log. For example, HUKB-Q-J-3Q uses no query analysis results (using only Wikipedia based subtopic candidates and topic model based subtopic candidates) for candidates and topic models are constructed based on the relevant documents whose keywords are closely located in the text window size. And HUKB-Q-J-4Q use all information to generate subtopic candidates and use all query analysis results provided by the organizers.

Table 3 shows evaluation result of each submitted run. The best run of the submitted results is J-4Q and J-1Q.

Table 2: Settings of each submitted run

run name	Documents	Query suggestion	Yahoo! Search Logs
HUKB-Q-J-1Q	Snippet	Yes	Yes
HUKB-Q-J-2Q	Snippet	Yes	No
HUKB-Q-J-3Q	Snippet	No	No
HUKB-Q-J-4Q	All	Yes	Yes
HUKB-Q-J-5Q	All	Yes	No

From the comparison between the result of using all documents for topic model or snippet selected documents for topic model (J-1Q v.s. J-4Q and J-2Q v.s. J-5Q), the difference is small. However, Utilization of Yahoo! Search logs is slightly effective (J-1Q v.s. J-2Q and J-4Q and J-5Q) (not so significant by using Wilcoxon signed rank test). The result without using Yahoo! Search logs and query expansion (J-3Q) is significantly worse than other runs ($p < 0.01$). From this results, we confirm the query analysis results is useful resource to identify the subtopic.

Table 3: Evaluation results of each submitted run

	J-1Q	J-2Q	J-3Q	J-4Q	J-5Q
I-rec@10	0.646	0.632	0.497	**0.653**	0.645
D-nDCG@10	**0.507**	0.475	0.368	0.505	0.470
D#-nDCG@10	0.576	0.553	0.433	**0.579**	0.557
V-Score	**0.535**	0.480	0.384	**0.535**	0.481
QU-Score	0.556	0.517	0.408	**0.557**	0.519

In order to analyze the characteristics of each information resource, we check the information resource used for selecting candidates.

Table 4,5,6 shows number of subtopic candidates from each resource for J-1Q, J-2Q and J-3Q respectively. Numbers for selection represent total numbers of selected candidates, number of selected candidates listed in the IMine2-J-Intents and ratio between these two numbers. Since there are several cases that two or more resources are found for a subtopic candidate, numbers of selection that is from only one resource are shown as unique selection.

Table 4: Number of subtopic candidates from each resource (J-1Q)

	Selection	Unique Selection
Yahoo coclick	260/368 (0.71)	145/218 (0.67)
Yahoo cotopic	318/447 (0.71)	145/217 (0.67)
Query suggestion	220/314 (0.70)	80/124 (0.65)
Wikipedia Abstract	23/41 (0.56)	21/39 (0.54)
Wikipedia Alternatives	5/15 (0.33)	3/10 (0.3)
Topic model	36/115 (0.31)	36/115 (0.31)

From these tables, most of the candidates are from Yahoo! search logs in J-1Q and Query suggestion is a secondary resource. Candidates from Wikipedia are not frequently used in J-1Q and quality of the candidates is not good, especially for the candidates generated by Wikipedia alternatives. When we check those candidates manually, it seems that those keywords may represent small size subtopic

[6] https://www.dmoz.org/World/Japanese/

[7] http://www.groonga.org/

[8] http://www.cs.princeton.edu/ blei/lda-c/index.html

Table 5: Number of subtopic candidates from each resource (J-2Q)

	Selection	Unique Selection
Query suggestion	403/584 (0.69)	401/578 (0.69)
Wikipedia Abstract	50/84 (0.60)	48/81 (0.59)
Wikipedia Alternatives	10/26 (0.38)	10/23 (0.43)
Topic model	102/289 (0.35)	102/289 (0.35)

Table 6: Number of subtopic candidates from each resource (J-3Q)

	Selection	Unique Selection
Wikipedia Abstract	50/84 (0.60)	50/84 (0.60)
Wikipedia Alternatives	10/26 (0.38)	10/26 (0.38)
Topic model	503/867 (0.58)	503/867 (0.58)

extracted from the retrieved results, but those are not representative subtopics. It may be better to reconsider the scores related to the retrieved document size of topic (Equation 2 - 4) for controlling these issues.

4. CONCLUSION

In this paper, we propose a system that estimates important subtopic by using characteristic keywords extracted from Wikipedia and query analysis information and diversified the results based on the topic model. We confirm query analysis information is a good resource to estimate the appropriate subtopics. However, it is necessary to conduct more detailed failure analysis. For example, it is necessary to investigate characteristics of each information resource by using evaluation data (e.g., Which information resource covers wide varieties of correct subtopics most?). It is also necessary to investigate the appropriateness of the parameters used in the experiments.

Acknowledgement

This work was partially supported by JSPS KAKENHI Grant Number 25280035.

5. REFERENCES

[1] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[2] T. Yamamoto1, Y. Liu, M. Zhang, Z. Dou, K. Zhou, I. Markov, M. P. Kato, H. Ohshima, and S. Fujita. Overview of the ntcir-12 imine-2 task. In *Proceedings of the 12th NTCIR Workshop Meeting on Evaluation of Information Access Technologies: Information Retrieval, Quesiton Answering, And Cross-Lingual Information Access*, 2016. (to appear).