Overlapping Clustering Method Using Local and Global Importance of Feature Terms at NTCIR-4 WEB Task

Manabu Ohta† Hirokazu Narita‡ Shigeyoshi Ohno†
†Graduate School of Engineering, Tokyo Metropolitan University
1-1 Minami-Osawa, Hachioji-shi, Tokyo 192-0397, Japan
ohta@eei.metro-u.ac.jp ohno@cs.uitec.ac.jp

‡Sony Communication Network Corporation
6F Gotenyama hills, 4-7-35 Kitashinagawa, Shinagawa-ku, Tokyo 140-0001, Japan
h-narita@jcom.home.ne.jp

Abstract

In NTCIR-4 WEB Task D (Topical Classification Task), we present an overlapping clustering method for a Japanese meta search engine as an alternative to listing of ranked retrieved results, which most search engines adopt to present the retrieval results. The proposed method clusters the retrieved results dynamically according to two steps: (1) cluster labels consisting of the most important feature terms extracted from the retrieval results are generated first; then (2) each document is classified into one or more (i.e., overlapping) generated clusters based on its relevance to the feature term. Evaluation results showed that the proposed method achieved better retrieval effectiveness in a formal run than the average of all the participants in Task D.
Keywords: *NTCIR, Web Document Clustering, Content Mining, Evaluation Method, Meta Search Engine.*

1 Introduction

Search engines are powerful tools that are widely used to access necessary information on the Web. However, users are not always satisfied with search results that they return. For example, Yahoo!™ Japan and Google™ search engines, which are popular in Japan and the world, respectively, have the following features.

- Keyword search is provided to express users' query intentions.
- Search results are always given as a list of items ranked by relevance to query terms.

Ranked retrieval results are indeed helpful for users to locate a specific piece of information. Notwithstanding, categorizing search results into clusters with appropriate labels is anticipated to help those users who become frustrated with myriad search results. Using such categorization, users might grasp an overview of results and gain access to Web pages that are related directly to their interests.

On the other hand, Web page clustering is a kind of document clustering. Such document clustering divides documents exclusively and sometimes produces a hierarchy of clusters. We have already proposed a Japanese meta search engine that categorizes search results into hierarchical clusters exclusively [1]. A major drawback of this engine is engendered in its exclusiveness: each search result (Web document) that can reasonably be included in two or more clusters is assigned to a single cluster. Because of this exclusiveness, experiments for this engine have shown that the recall rate of clusters tends to be lower than that of another search engine with a clustering function.

For that reason, this paper proposes an overlapping clustering method named Overlapping Clustering Method Using Local and Global importance of fEature tErms (OCMULGEE). It is expected to achieve a high recall rate for each cluster and to categorize more retrieved documents into meaningful clusters. The proposed method offers the following remarkable features:

- dynamic clustering executed each time search results are obtained;
- overlapping (non-exclusive) clustering; and
- appropriately extracted cluster labels.

This paper is structured as follows. We first explain prior related works on document clustering in Section 2. Section 3 describes the proposed clustering method, OCMULGEE, and shows an example of created clusters. Section 4 describes results of both dry and formal runs that were obtained through experiments in which OCMULGEE was applied with various parameters to clustering Japanese Web documents given by the NTCIR-4 organizer. Conclusions are presented in Section 5.

2 Related works

Clustering techniques for Web search results can be divided broadly into two categories: those based on structure mining and content mining.

Regarding structure mining, Wang *et al.* [2] proposed link-based clustering methods by which co-citation and bibliographic coupling were used to characterize the degree and type of similarity between two Web documents. Although link-based clustering offers several advantages including language independence, original Web documents must be referred in order to extract URL sequences, which is not necessary for OCMULGEE.

On the other hand, Scatter/Gather [3] employs an automatic content-based clustering algorithm, named fractionation [4], to organize a set of documents into a given number of topic-coherent groups. Experiments using that system have indicated that the best cluster had more documents relevant to the query than an equivalent number of top-ranked documents of the original search results. Although Scatter/Gather was effective for analyzing relatively long documents such as newspaper articles, OCMULGEE's target is the clustering of snippets of Web documents comprising a title, a summary, and a URL. Eguchi *et al.* [5] also proposed content-based clustering methods in which feature vectors are defined using statistical information of terms such as TFIDF and a certain inter-document similarity measure is introduced for clustering. OCMULGEE, proposed in this paper, is categorized into a content-based approach; its clustering is based on feature term analysis.

Furthermore, many (meta) search engines have a clustering function that is similar to OCMULGEE: Vivisimo[1], ez2Find[2], metacrawler[3], WebCrawler[4], Turbo10[5], etc. However, many of them, especially commercial sites, do not reveal their technical details.

3 OCMULGEE

3.1 Overview

This section describes the proposed overlapping clustering method, OCMULGEE, which clusters a few hundred search results dynamically. OCMULGEE extracts feature terms from the search results, calculates two kinds of measures of importance for the terms, such as *local importance* (LI) and *global importance* (GI), and determines clusters and their labels based on both values of importance. If possible, subclusters are subsequently generated by analyzing compound nouns included in titles or summaries of documents in the clusters.

The GI is a measure of importance of terms across the whole search results, whereas LI is a measure of terms within each search result. The proposed method generates categories represented by terms of high GIs. Then each search result is clustered into the categories of terms whose LI is greater than a certain threshold. Cluster size (the number of elements in each cluster) and the retrieval effectiveness of each cluster can be controlled by the LI threshold. The maximum number of clusters can also be controlled to prevent generation of too many clusters.

Therefore, the proposed method comprises the following five steps, each of which is explained in more detail in subsequent subsections.

1. Feature term extraction
2. Calculation of local importance
3. Calculation of global importance
4. Creation of top-level clusters
5. Creation of subclusters

3.2 Feature term extraction

Preprocessing of OCMULGEE generates a set of feature terms F. First, a parser is developed to remove HTML tags from HTML sources of search results and divide them into each retrieved document which constitutes R, a set of divided retrieved documents. Any divided retrieved document $r_i \in R$ has three attributes: a title, a summary, and a URL. Secondly, titles and summaries of retrieved documents are analyzed morphologically by Chasen[6]. All nouns and unknown words are extracted as candidates of feature terms based on the part-of-speech (POS) information given by Chasen. Finally, the candidates of feature terms extracted by morphological analysis are distilled into F by normalization, deletion of stopwords, integration of persons' names, and some heuristics.

3.3 Calculation of local importance

Local importance $LI(r_i, f_j)$ of each feature term, $f_j \in F$, in each retrieved document, $r_i \in R$, is defined for determining whether to categorize r_i into the cluster with the label of f_j. $LI(r_i, f_j)$ can also be

[1] Vivisimo™ http://vivisimo.com/
[2] ez2Find http://ez2find.com/
[3] metacrawler™ http://www.metacrawler.com/
[4] WebCrawler™ http://www.webcrawler.com/
[5] Turbo10 http://turbo10.com/
[6] http://chasen.aist-nara.ac.jp/

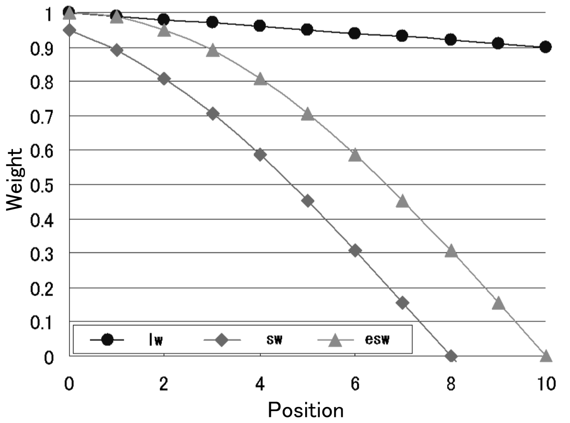

Figure 1. LI weight.

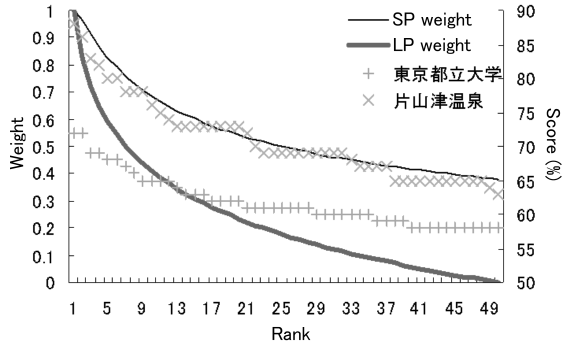

Figure 2. Scores of infoseek and SP, LP weights.

considered as a sum of weighted occurrence frequencies of a feature term f_j within a document r_i. Based on the simple assumption that terms appearing at the beginning of a text are more important than others, OCMULGEE calculates the weighted occurrence frequency in three ways, i.e., lw (linear weight) in Eq. (1), sw (sine weight) in Eq. (2) and esw (enhanced sine weight) in Eq. (3). In these equations, p is the number of morphemes between the head of the title or summary and the appearance of f_j and T is the number of all morphemes in r_i.

$$lw(r_i, f_j) = \begin{cases} 1 - \frac{b}{a}p & a \geq bp \\ 0 & otherwise \end{cases}, \quad (1)$$

$$sw(r_i, f_j) = \sin\left(\frac{T - (p+1) - 1}{2 \times T}\pi\right), \quad (2)$$

$$esw(r_i, f_j) = \sin\left(\frac{T + (p+1) - 1}{2 \times T}\pi\right). \quad (3)$$

Figure 1 shows these weights when $T = 10$. OCMULGEE uses $a = 100, b = 1$ in Eq. (1) based on preliminary experiments. Figure 1 shows that esw is slightly greater than sw.

Based on these equations, $LI(r_i, f_j)$ is calculated as follows:

$$LI(r_i, f_j) = \sum_P weight. \quad (4)$$

In Eq. (4), $weight$ represents any of the three defined weights and $P = \{p_1, p_2, ..., p_n\}$ is a set of positions p, where n is the number of occurrences f_j in r_i. For the rest of this manuscript, $LI(r_i, f_j)$ that is calculated with lw is denoted by LWLI, that with sw is denoted as SWLI, and that with a combination of esw for the title and sw for the summary is denoted as TESWLI.

3.4 Calculation of global importance

Global importance $GI(f_j)$ of each feature term $f_j \in F$ in all retrieved documents R is defined for determining what clusters should be generated. Clusters with the label of f_j are generated based on $GI(f_j)$.

As $GI(f_j)$, OCMULGEE adopts $DF(f_j)$ and $TF(f_j) \times IDF(f_j)$, which are major term weighting measures that are widely used in IR. Here, $DF(f_j)$, the document frequency of f_j, represents the number of documents including f_j in R; $TF(f_j)$, the term frequency of f_j, represents the number of times f_j appears in R. In addition, $IDF(f_j)$, the inverse document frequency of f_j, is calculated as $IDF(f_j) = \log\frac{N}{DF(f_j)}$, where N is the number of documents constituting R. OCMULGEE also proposes $SP(f_j)$ in Eq. (5) and $LP(f_j)$ in Eq. (6) as $GI(f_j)$, both of which represent $TF(f_j) \times IDF(f_j)$ weighted by ranking information of r_i in R. That is, r_i stands for the i-th item in the overall search results R in these equations.

$$SP(f_j) = \sum_{i=1}^{N}\left[TF(r_i, f_j) \times \boxed{\sin\left(\frac{\pi}{1+\sqrt{i}}\right)}\right] \times IDF(f_j). \quad (5)$$

$$LP(f_j) = \sum_{i=1}^{N}\left\{TF(r_i, f_j) \times \boxed{\log_N \frac{N}{i}}\right\} \times IDF(f_j). \quad (6)$$

Figure 2 shows that sine and logarithm weights in these equations are similar in shape to the relationship between the rank and score given by infoseek[7] when searching with “片山津温泉 (Katayamazu spa)” and “東京都立大学 (Tokyo Metropolitan University)”, respectively.

3.5 Creation of top-level clusters

OCMULGEE initially generates $c(f_j)$: clusters with the label of feature term f_j whose $GI(f_j)$ val-

[7]infoseek™ http://infoseek.co.jp/

ues are greater than a given threshold. It then determines whether to categorize each document, $r_i \in R$, into the clusters $c(f_j)$. However, those clusters of f_j whose $GI(f_j)$ are under the threshold are never generated. This characteristic prevents OCMULGEE from increasing unnecessary clusters. The clustering process proposed in OCMULGEE is summarized as follows.

1. The $c(f_j)$ with the highest $GI(f_j)$ in F is generated; then the f_j is removed from F.
2. Each document r_i belongs to $c(f_j)$ if $LI(r_i, f_j)$ is greater than a certain threshold that serves to exclude weakly relevant documents. The r_i with plural f_j whose $LI(r_i, f_j)$ value is greater than the threshold can belong to a plural number of $c(f_j)$.
3. The $c(f_j)$ is deleted if no document belongs to $c(f_j)$ or if all documents belonging to $c(f_j)$ also belong to another previously generated cluster.
4. Any singleton cluster $c(f_j)$ (a cluster with only one member) is deleted.
5. Such cluster generation continues until the number of generated clusters reaches a maximum (threshold), or until there exists no $f_j \in F$ whose $GI(f_j)$ is greater than a threshold. All Web documents belonging to no clusters after assigning documents to each cluster belong to the "etc." cluster.

3.6 Creation of subclusters

OCMULGEE generates subclusters after creating top-level clusters as follows.

1. All adjacent nouns and unknown words that contain f_j are regarded as target compound nouns. They are extracted from titles and summaries of the documents in top-level clusters $c(f_j)$.
2. If one of the extracted compound nouns is a substring of another, those with smaller values of DF in $c(f_j)$, TF in $c(f_j)$, or their string length are deleted where the comparisons are made in this order.
3. Subclusters are created with the label of the remained compound nouns after the above selection.
4. Each document included in $c(f_j)$ belongs to the created subcluster if its title or summary contains the label of the subcluster.
5. Any singleton subcluster (having only one member) is deleted.
6. If all elements of any subcluster are identical to those of its parent cluster $c(f_j)$, the label of $c(f_j)$, i.e. f_j, is replaced with that of the subcluster.

3.7 An example of created clusters

Figure 3 shows some clusters created by OCMULGEE when searching with the queries "著作権 (Copyright)", " デジタルコンテンツ (Digital contents)", and "ネットワーク (Network)": the left pane displays whole clusters in a tree-view, as Windows Explorer (Microsoft Corp.) does; documents within the cluster that is selected by a user are presented in the right pane. Each folder icon in the left pane stands for a created cluster. Character strings "保護技術 (Protection technology)" are highlighted in the right pane because the cluster "保護技術", a subcluster of the cluster " 技術 (Technology)", is selected in the left pane. In addition, "[200]" beside the label of the root cluster represents the number of all the Web documents for clustering, i.e., 200 items were categorized in this example. Top-level clusters have similar information of "[SIZE, TF, DF]" beside their cluster labels f_j, where SIZE is the number of Web documents in $c(f_j)$, TF is $TF(f_j)$, and DF is $DF(f_j)$. The subclusters also have "[SIZE, TF, DF]" beside their labels, but TF and DF here stand for those in their parent (top-level) cluster, not in all the documents R.

Figure 3 also shows that the cluster "情報 (Information)" has 14 subclusters with labels containing the character string "情報" and the cluster "技術" has 13 subclusters. Almost all labels of subclusters make sense in Japanese.

4 Evaluation

4.1 Dry run

For the dry run at NTCIR-4 WEB Task D, we were given a 100-gigabyte Web document set "NW100G-01" constructed at the NTCIR-3 WEB [6], and search result lists of 12 topics, i.e., "target data set" comprising about 200 or more documents per topic.

For the dry run, OCMULGEE extracted, at most, 50 characters before and behind the query terms in each document as its summary after removing HTML tags from its HTML source (KWIC). The maximum number of generated clusters was set to 20. Original ranking information of all documents was preserved in each cluster after clustering. Table 1 summarizes a system description of OCMULGEE as it was set up for the dry run: three runs, METAL-0[123], were submitted with varying parameters[8]. METAL-01 adopted an LI threshold of zero, meaning that any retrieved

[8]"METAL" stands for OCMULGEE here. METAL is the original name of our *exclusive* clustering system.

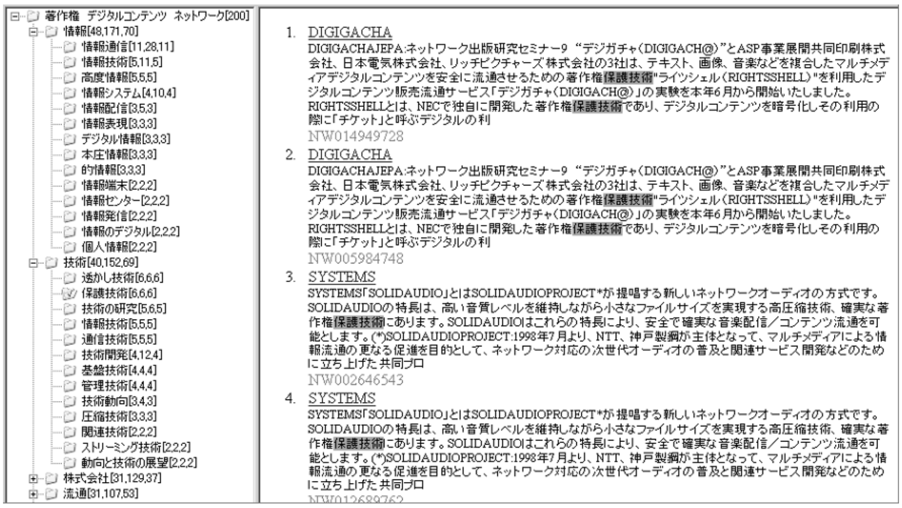

Figure 3. Created clusters when searching on "Copyright", "Digital contents", and "Network".

Table 1. System description for dry run

SystemID	GI	LI	LI threshold
METAL-01	DF	LWLI	0
METAL-02	DF	LWLI	6
METAL-03	SP	LWLI	6

document r_i with a feature term f_j is clustered into $c(f_j)$, irrespective of $LI(r_i, f_j)$ values. In contrast, METAL-02 and METAL-03 both adopted thresholds of six based on preliminary experiments.

4.2 Discussion of dry run results

All Web documents for clustering were given multi-grade relevance to each query, such as highly relevant, fairly relevant, partially relevant, or irrelevant. Highly and fairly relevant documents are considered to be relevant at the *rigid* level, whereas those judged not irrelevant are considered to be relevant at the *relaxed* level. In the dry run, retrieval effectiveness based on both relevance levels were given to each participant.

A summary of dry run results of OCMULGEE is shown in Tables 2 and 3, where average precision (AvePrec), precision (P@20), and recall (R@20) of the 20 top-ranked documents were calculated after sorting the generated clusters based on the number of relevant documents in the clusters. One problem we found was that OCMULGEE assigned not a few documents to the "etc." cluster irrespective of their relevance. Consequently, for some queries, the "etc." cluster ranked first in the number of relevant documents included in it. That result was unintended. Apparently, OCMULGEE regards the "etc." cluster as a set of rather useless documents. The figures in these tables are not good compared to the averages of all participants in Task D, but some queries indicated good retrieval effectiveness where the "etc." cluster did not rank first. METAL-01 showed the best results among the three submitted runs, which implies that the use of DF as GI, and the LI threshold of zero were both appropriate.

However, these evaluation measures used in the dry run become largest when singleton clusters of the same number of given documents are created: such extreme clustering is possible because the number of created clusters is not restricted. Therefore, we tried to eval-

Table 2. Dry run results based on rigid relevance judgment (%)

SystemID	AvePrec	P@20	R@20
METAL-01	5.9	20.8	17.7
METAL-02	5.5	16.7	15.9
METAL-03	5.5	16.3	14.5
Average	5.7	17.9	16.1

Table 3. Dry run results based on relaxed relevance judgment (%)

SystemID	AvePrec	P@20	R@20
METAL-01	6.2	32.1	14.3
METAL-02	5.0	23.8	11.1
METAL-03	4.9	23.3	10.6
Average	5.4	26.4	12.0

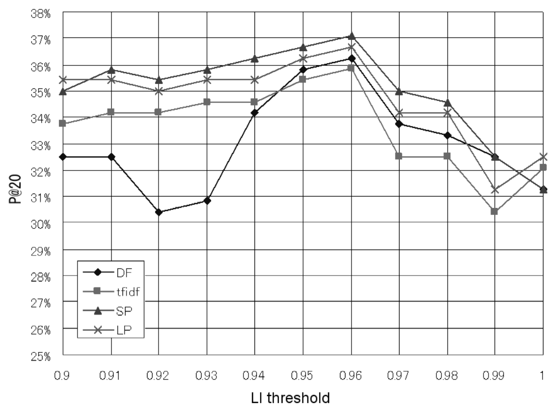

Figure 4. LI threshold vs. P@20 (rigid).

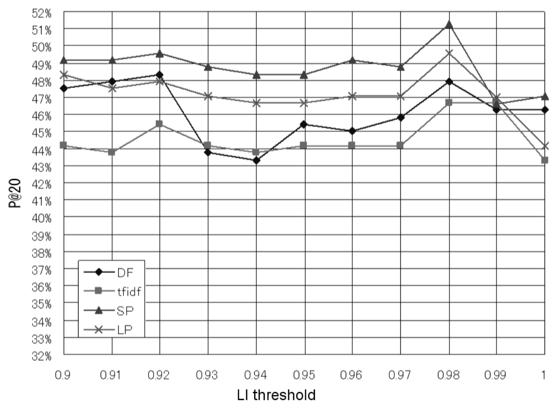

Figure 5. LI threshold vs. P@20 (relaxed).

uate clustering by considering the number of created clusters [7, 8].

4.3 Tuning to formal run

Taking the dry run results into consideration, OCMULGEE was tuned to categorize as few relevant documents as possible into the “etc.” cluster by not restricting the number of created clusters. In addition, we introduced the following modifications to OCMULGEE.

- SWLI and TESWLI, which were defined in Section 3.3, were selected as LI based on the experiments where relevance judgment data of the dry run were used.

- Instead of KWIC described in Section 4.1, 300 characters from the heads of respective Web documents were extracted as their summaries.

We examined the relationships between LI thresholds and P@20 explained in Section 4.2 to determine appropriate LI thresholds. They are shown in Figures 4 and 5 for rigid and relaxed data, respectively, when using TESWLI as LI. The values of P@20 became largest when using the threshold of 0.96 in Figure 4 and 0.98 in Figure 5.

On the other hand, Table 4 shows P@20 with rigid and relaxed relevance judgment when the LI threshold of 0.96 was used with various GI measures. Based on this experiment, SP defined in Eq. (5) was selected as the GI for the formal run. Table 4 also gives the clustering ratio (CR), which expresses the ratio of all elements categorized into clusters except “etc.” to all search results, which indicates that almost all of the Web documents were clustered into meaningful clusters.

Table 4. Influences of GI measures when 0.96 was used as the TESWLI threshold (%)

	DF	tfidf	SP	LP
P@20(rigid)	36.3	35.8	37.1	36.7
P@20(relaxed)	45.0	44.2	49.2	47.1
CR	96.6	96.6	96.6	96.6

4.4 Formal run

In the formal run, each participant in NTCIR-4 WEB Task D was given 47 topics derived from the topic data of Task A. Each meta search was done using only one query term, whereas a few query terms were used in the dry run. We submitted four runs to the formal run with varying parameters based on experiments described in Section 4.3, which are summarized in Table 5.

4.5 Discussion on formal run results

The NTCIR-4 organizer gave evaluation results for 11 topics among the 47 topics that were submitted.

Table 5. System description for formal run

SystemID	GI	LI	LI threshold
METAL-01	SP	TESWLI	0.965
METAL-02	SP	TESWLI	0.955
METAL-03	SP	SWLI	0.915
METAL-04	SP	SWLI	0.985

Table 6. Formal run results based on rigid relevance judgment (%)

SystemID	AvePrec	P@20	R@20
METAL-01	36.0	44.5	75.0
METAL-02	35.8	45.0	75.0
METAL-03	36.0	44.5	75.0
METAL-04	36.2	45.5	76.8
Average	36.0	44.9	75.4

Table 7. Formal run results based on relaxed relevance judgment (%)

SystemID	AvePrec	P@20	R@20
METAL-01	30.1	47.7	51.9
METAL-02	29.9	48.2	53.8
METAL-03	30.1	47.7	51.9
METAL-04	30.0	48.2	55.4
Average	30.0	48.0	53.2

Table 8. Cumulative gain-based measurements in the formal run (rigid)

SystemID	CG	DCG	MDCG1	MDCG2
METAL-01	8.64	3.33	3.18	8.15
METAL-02	8.73	3.28	3.12	8.16
METAL-03	8.64	3.33	3.18	8.15
METAL-04	8.82	3.30	3.11	8.16
Average	8.70	3.31	3.15	8.15

Table 9. Cumulative gain-based measurements in the formal run (relaxed)

SystemID	CG	DCG	MDCG1	MDCG2
METAL-01	9.18	3.65	3.54	8.80
METAL-02	9.36	3.72	3.58	8.88
METAL-03	9.18	3.65	3.54	8.80
METAL-04	9.45	3.76	3.60	8.90
Average	9.30	3.70	3.57	8.84

Summaries of the formal run results are shown in Tables 6 and 7. They show three kinds of retrieval effectiveness: AvePrec, P@20, and R@20, as explained in Section 4.2. All three measures with rigid relevance judgment improved considerably compared to those in Table 2: AvePrec increased more than 6 times comparing both averages of all our submitted runs; P@20 approximately 2.5 times; R@20 approximately 4.7 times. Those with relaxed relevance judgment also improved compared to those in Table 3: AvePrec increased approximately 5.6 times on average; P@20 approximately 1.8 times; R@20 approximately 4.4 times. In addition to the remarkable improvement compared to the dry run results, all evaluation measurements, irrespective of rigid or relaxed judgment, were well above averages of all the participants in Task D. Among the four runs, METAL-04 in Tables 6 and 7 achieved the highest retrieval effectiveness.

Concerning the number of generated clusters and the ranks of documents in clusters, evaluation measures reflecting them, such as Cumulative Gain (CG), Discounted Cumulative Gain (DCG), Modified DCG 1 (MDCG1), and Modified DCG 2 (MDCG2) were introduced in the formal run. Tables 8 and 9 summarize some resultant measurements of OCMULGEE in rigid and relaxed judgments, respectively. All of these cumulative gain-based measurements were also well above the averages of all participants irrespective of relevance judgment level. Among the four runs, METAL-04 also gave the best results in relaxed relevance judgment, but the superiority of METAL-04 became slightly smaller in rigid judgment.

5 Conclusions

This paper presented an overlapping and dynamic clustering method for a Japanese meta search engine. We reported results of applying it to NTCIR-4 WEB Task D. The salient feature of OCMULGEE is that cluster labels are first created according to the *global importance* of feature terms. Then each search result is assigned to clusters based on the *local importance* of the terms. OCMULGEE can control the quality of generated clusters by varying the thresholds of local importance.

Formal run results indicated that OCMULGEE achieved not only better retrieval effectiveness, but also better cumulative gain-based measurements than the averages of all participants in Task D. In terms of future work, we intend to use some thesauri to handle synonyms properly.

References

[1] H. Narita, M. Ohta, K. Katayama, and H. Ishikawa. Meta search engine using hierarchical clustering –METAL–. *IPSJ SIG Notes*, 2002-DBS-128-50, 375–382, 2002 (in Japanese).

[2] Y. Wang and M. Kitsuregawa. Link based clustering of Web search results. *Advances in Web-Age Information Management Second Interna-*

tional Conference (WAIM2001), *LNCS*, **2118**, 225–236, 2001.

[3] M. A. Hearst and J. O. Pedersen. Reexamining the cluster hypothesis: Scatter/Gather on retrieval results. *Proc. ACM SIGIR-96 International Conference on Research and Development in Information Retrieval*, 76–84, 1996.

[4] D. R. Cutting, D. R. Karger, J. O. Pedersen, and J. W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. *Proc. ACM SIGIR-92 International Conference on Research and Development in Information Retrieval*, 318–329, 1992.

[5] K. Eguchi, H. Ito, A. Kumamoto, and Y. Kaneda. Adaptive document clustering using incrementally expanded queries. *Systems and Computers in Japan*, Scripta Technica, 32(2):64–74, 2001.

[6] K. Eguchi, K. Oyama, E. Ishida, N. Kando, and K. Kuriyama. Overview of the Web retrieval task at the third NTCIR workshop. *NII Technical Report*, NII-2003-002E, 2003 <http://research.nii.ac.jp/TechReports/03-002E.html>.

[7] H. Narita, M. Ohta, K. Katayama, and H. Ishikawa. OCMULGEE: Overlapping Clustering Method Using Local and Global Importance of Feature Terms. *Proc. DBWeb2003*, 85–92, 2003 (in Japanese).

[8] H. Narita. Study on clustering methods for Japanese search engines. Master's thesis, Tokyo Metropolitan University, 2003 (in Japanese).