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 a list of ranked retrieval results which most search engines adopt to present the retrieval results. The proposed method clusters the retrieval results dynamically according to the following 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. The evaluation results showed that the proposed method in formal run achieved better retrieval effectiveness compared to 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 widely used to access necessary information on the Web. However, we cannot always get satisfied with search results returned by them. For example, Yahoo! Japan and Google, one of the most popular search engines in Japan and in the world respectively, have following features.

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

Ranked retrieval results help users locate a specific piece of information indeed, however, categorizing search results into clusters with appropriate labels is anticipated to help those at a loss with myriad and numerous search results. Such users might thereby 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 which 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 OCMULGEE (Overlapping Clustering Method Using Local and Global importance of fEature tErms), which is expected to achieve a high recall rate of 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 the results of both dry and formal runs obtained through the experiments

in which OCMULGEE with various parameters was applied 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.

As 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 pages. Although link based clustering has several advantages such as language independence, original Web documents must be referred to 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 content-based approach and its clustering is based on feature term analysis.

Moreover, there are many (meta) search engines with clustering function similar to OCMULGEE: vivisimo[1], EZ2Find[2], meta crawler[3], WebCrawler[4], Turbo10[5], etc. Many of them, however, do not reveal their technical details especially with commercial sites.

3 OCMULGEE

3.1 Overview

This section describes the proposed overlapping clustering method OCMULGEE supposed to cluster 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.

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 methods comprise the following 5 steps, each of which is explained in more detail in subsequent subsections.

1. Feature term extraction
2. Calculation of LI
3. Calculation of GI
4. Creation of top-level clusters
5. Creation of subclusters

3.2 Feature term extraction

The 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 POS (Part of Speech) 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 LI (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 considered as a sum of weighted occurrence frequency of a

[1] Vivisimo http://vivisimo.com/
[2] EZ2Find http://ez2find.com/
[3] meta crawler http://www.metacrawler.com/
[4] WebCrawler http://www.webcrawler.com/
[5] Turbo10 http://turbo10.com/
[6] http://chasen.aist-nara.ac.jp/

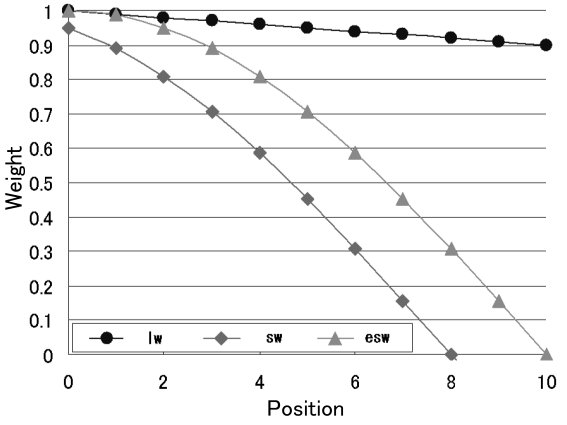

Figure 1. LI weight

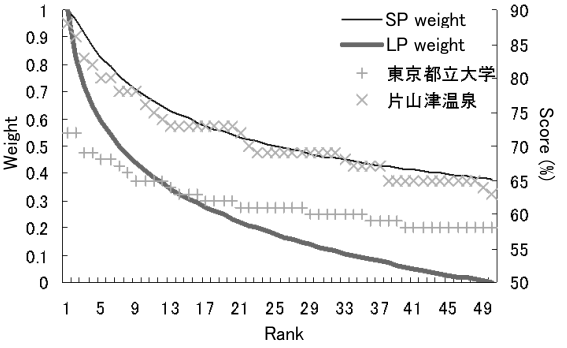

Figure 2. Score of Infoseek and SP,LP weight

feature term f_j within a document r_i. Based on a 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 the preliminary experiments. As shown in Figure 1, 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 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, $LI(r_i, f_j)$ calculated with lw is denoted by LWLI, that with sw by SWLI, and that with a combination of esw for the title and sw for the summary by TESWLI.

3.4 Calculation of GI (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. Based on $GI(f_j)$, clusters with the label of f_j are generated.

As $GI(f_j)$, OCMULGEE adopts $DF(f_j)$ and $TF(f_j) \times IDF(f_j)$, one of major term weighting measures 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; and $IDF(f_j)$, 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. In addition, 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, i.e., r_i is the i-th item in the whole search results R.

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

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

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" and "東京都立大学" respectively.

3.5 Creation of top-level clusters

OCMULGEE initially generates $c(f_j)$: clusters with the label of feature term f_j whose $GI(f_j)$ values 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 $GI(f_j)$ is under a certain threshold are never

[7] Infoseek http://infoseek.co.jp/

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 provided for excluding weak relevant documents. The r_i with plural f_j whose $LI(r_i, f_j)$ value is greater than a 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 all documents belonging to $c(f_j)$ also belong to another cluster that has been generated previously.

4. Any singleton cluster $c(f_j)$ (having only one member) becomes a child cluster of the "etc." cluster.

5. Clusters generation continues until the number of generated clusters reaches a maximum (threshold) or 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 the adjacent nouns and unknown words containing f_j are regarded as compound nouns and extracted from the titles and summaries of the elements 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 $c(f_j)$, and their string length are deleted where the comparisons are made in this order.

3. Subclusters with the label of the remained compound nouns after the above selection are created.

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 the 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 a part of clusters created by OCMULGEE when searching with 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 Internet Explorer does; documents within the cluster selected by users 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.

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 the 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 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" composed of 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 20. Original ranking information of all the documents were preserved in each cluster after clustering. Table 1 summarizes a system description of OCMULGEE set up for the dry run, which indicates that three runs, METAL-0[123], were submitted with varying parameters[8].METAL-01 adopted LI threshold of 0 which means that any retrieved document, r_i, with a feature term, f_j, is clustered into $c(f_j)$ irrespective of $LI(r_i, f_j)$ values. By contrast, METAL-02 and METAL-03 adopted the threshold of six based on the preliminary experiments.

4.2 Discussion on dry run results

All the 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

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

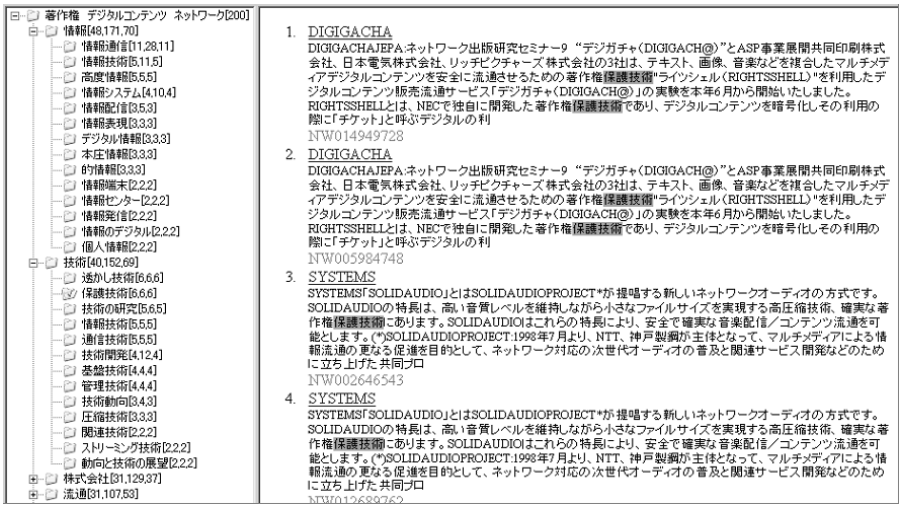

Figure 3. Created clusters when searching on “Copyright”

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

relevant in *rigid* level whereas those judged not irrelevant are considered relevant in *relaxed* level. In the dry run, retrieval effectiveness based on both relevance level were given to each participant.

A summary of the dry run results of OCMULGEE can be seen in Tables 2 and 3 where average precision (AvePrec), precision (P@20) and recall (R@20) at 20 top ranked documents, were calculated after sorting the generated clusters based on the number of relevant documents in the clusters. One of the problems we found was that OCMULGEE assigned not a few documents to the “etc.” cluster irrespective of their relevance and, as a consequence, for some queries the “etc.” cluster ranked first in the number of relevant documents included in it. It was not intended because 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 the participants in Task D although some queries indicated good retrieval effectiveness where the “etc.” cluster did not rank first. METAL-01 showed best results among the three submitted runs, which implies that DF used as GI and the LI threshold of 0 were appropriate.

These evaluation measures used in the dry run, however, become largest when singleton clusters of the same number of given documents are created, that is, each cluster has only one document, because the number of created clusters are not restricted. Therefore, we tried to evaluate the clustering by taking the number of created clusters into consideration [7, 8].

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	5.6	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

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, the following modifications to OCMULGEE were introduced.

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

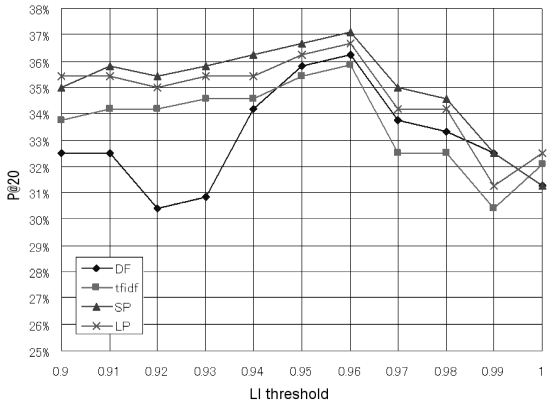

Figure 4. LI threshold vs. P@20 (rigid)

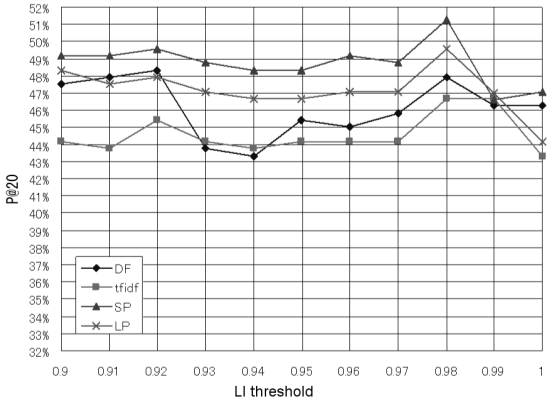

Figure 5. LI threshold vs. P@20 (relaxed)

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

In order to determine appropriate LI thresholds, we examined the relationships between LI thresholds and P@20 explained in Section 4.2. 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 were used with various GI measures. Based on this experiment, SP defined in Eq. (5) was selected as GI for the formal run. Table 4 also gives CR (clustering ratio) denoting 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. GI measurements when 0.96 was used as 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

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

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 by 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 the experiments described in Section 4.3, which are summarized in Table 5.

4.5 Discussion on formal run results

Evaluation was done by the NTCIR-4 organizer on 11 topics among 47 submitted ones. A summary of the formal run results can be seen in Tables 6 and 7 which show three kinds of retrieval effectiveness, i.e., AvePrec, P@20, and R@20 explained in Section 4.2. All the three measures with rigid relevance judgment improved considerably compared to those in Table 2: AvePrec increased more than 6 times on average; P@20 approximately 2.5 times; R@20 approximately 4.7 times. Those with relaxed relevance judgment also improved: AvePrec increased approximately 5.6 times on average; P@20 approximately 1.8 times; R@20 approximately 4.4 times. All of the evaluation measurements irrespective of rigid or relaxed judgment were well above the averages of all the participants in Task D. In addition, METAL-04 in Tables 6 and 7 achieved highest retrieval effectiveness among the four runs.

Concerning the number of generated clusters and the ranks of documents in clusters, evaluation measures reflecting them such as CG (Cumulative Gain)、DCG (Discounted Cumulative Gain)、MDCG1 (Modified DCG 1)、and MDCG2 (Modified DCG 2) were

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

introduced in the formal run. Tables 8 and 9 summarize some of the resultant measurements of OCMULGEE in rigid and relaxed judgments, respectively. All of these cumulated gain-based measurements were also well above the averages of all the participants irrespective of relevance judgment level. Among the four runs, METAL-04 also gave the best results in relaxed relevance judgment although the superiority of METAL-04 became slightly smaller in rigid judgment.

5 Conclusions

In this paper, we proposed an overlapping and dynamic clustering method for a Japanese meta search engine and reported the 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 quality of generated clusters by varying thresholds of local importance.

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

Table 8. Cumulated gain-based measurements in 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. Cumulated gain-based measurements in 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	9.46	3.76	3.60	8.90
Average	9.30	3.70	3.57	8.84

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 International 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 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, 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).