# Document Image Retrieval in a Question Answering System for Document Images

Koichi Kise[1], Shota Fukushima[2], and Keinosuke Matsumoto[1]

[1] Department of Computer and Systems Sciences,
Graduate School of Engineering, Osaka Prefecture University
[2] Department of Computer and Systems Sciences,
College of Engineering, Osaka Prefecture University
1-1 Gakuencho, Sakai, Osaka 599-8531, Japan
kise@cs.osakafu-u.ac.jp

**Abstract.** Question answering (QA) is the task of retrieving an answer in response to a question by analyzing documents. Although most of the efforts in developing QA systems are devoted to dealing with electronic text, we consider it is also necessary to develop systems for document images. In this paper, we propose a method of document image retrieval for such QA systems. Since the task is not to retrieve all relevant documents but to find the answer somewhere in documents, retrieval should be precision oriented. The main contribution of this paper is to propose a method of improving precision of document image retrieval by taking into account the co-occurrence of successive terms in a question. The indexing scheme is based on two-dimensional distributions of terms and the weight of co-occurrence is measured by calculating the density distributions of terms. The proposed method was tested by using 1253 pages of documents about the major league baseball with 20 questions and found that it is superior to the baseline method proposed by the authors.

## 1 Introduction

Question answering (QA) is the task of retrieving *answers* rather than documents in response to a question with an emphasis on functioning in unrestricted domains[1]. Since it enables us to realize a more natural mean of "information retrieval" as compared to the keyword-based retrieval of documents, it attracts a great deal of attention in recent years. Much effort has been made including TREC conferences [2], as well as application to the Web [3]. In addition, some research groups have started offering services of QA systems to the public [4, 5].

Question answering has been studied in the field of information retrieval and thus most of the existing QA systems work only on electronic text. But is it enough for us to deal only with electronic text? We consider that it is not sufficient because at least of the following two reasons. First, we have already had a huge amount of document images in various databases and digital libraries. For example, the magazine "Comm. of the ACM" in the ACM digital library [6] consists of 80% of document images and 20% of electronic documents. Another

S. Marinai and A. Dengel (Eds.): DAS 2004, LNCS 3163, pp. 521–532, 2004.

reason is that mobile devices with digital cameras are now coming into common use. Some users have already utilized such devices for taking digital copies of documents, because it is much more convenient than writing memo[1]. This indicates that not only legacy documents but also new documents continue to be stored as document images.

In order to utilize such document images from the viewpoint of question answering, we have started a project of developing a QA system called "IQAS" (document Image Question Answering System)[2]. In this paper, we propose a method of document image retrieval for IQAS, by modifying our previous method [7, 8]. The major contribution of this paper is a way of improving *precision* of spotting parts that include the answer to the question. The previous method, which is called the baseline method in this paper, employs *density distributions* of terms for retrieving appropriate parts of images. In this paper, new density distributions modified by taking into account the co-occurrence of successive terms in the question are introduced and tested by experiments on 1253 pages with 20 questions. The results of experiments show that the proposed method is superior to the baseline method.

## 2 Question Answering for Document Images

### 2.1 Task and Configuration

The task of QA is *precision* oriented in nature. This is because the user is satisfied not by having all documents containing the same correct answer, but by just receiving the correct answer once. In the QA task, the user is allowed to ask questions in natural language. Systems for electronic text developed so far have tackled the questions of seeking simple *facts* by using "who", "what", "which", "when" and "where". Questions using "why" and "how" generally require much longer and complicated answers and thus their processing is still an open problem.

In order to locate facts in documents, QA systems are generally based on the following configuration.

1. Query Processing : The question in natural language is analyzed to obtain both query terms and the type of question. Query terms are employed in the next step of processing. The type of question defines what the question asks about. For example, "location", "time" and "person" are typical types.
2. Document Retrieval : A document retrieval engine is employed to find documents which is likely to contain the answer. Passage retrieval, i.e., to retrieve small portions of text from documents, is often utilized in this step.
3. Answer Extraction : The final step is to locate the answer in the retrieved passages with the help of types. Named entity extraction is applied to the extracted passages so as to locate the terms representing the answer to the question.

[1] In Japan, *digital shoplifting*, i.e., to make pictures of books and magazines in bookstores by mobile phones, has become an object of public concern.

[2] The pronunciation of the term IQAS is close to "Ikasu" in Japanese which have the two meanings: "to exploit" and "cool".

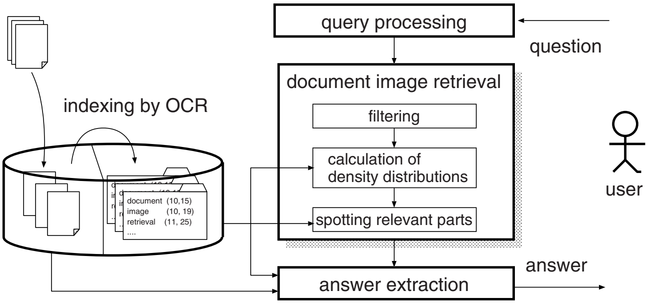

**Fig. 1.** System configuration.

Our system IQAS also follows the above configuration. Figure 1 illustrates the system configuration of IQAS. In this paper we focus only on the second step, which is "document image retrieval" in our case.

### 2.2 Related Work

Document image retrieval has studied in both fields of information retrieval [9] and document image analysis [10]. One of the central issues has been how to cope with OCR errors. Other errors in higher level analyses such as layout analysis and logical labeling cause less influence to the retrieval results if retrieval systems are based on the well-known "bag of words" (BOW) model. This is because only the frequency of terms is utilized in the BOW model.

However, this does not hold for passage retrieval and question answering, since these are to segment parts defined based on the results of higher level analyses. Thus methods for dealing with errors in higher level analyses are required. Although a straightforward way is to improve the accuracy of high level analyses, we have taken an indirect way by proposing a different retrieval method [7, 8], which is an extension of the original work on electronic text [11] to the two dimensional space. The characteristic point of the method is that it relies only on positions of terms in original pages. Parts are segmented not on the recognized text but on the two dimensional space of page regions. Density distributions of terms in the query are employed for locating parts relevant to it. This enables us to retrieve parts of document images independently of the results of higher level analyses.

In this paper, we improve the above method of density distributions to be better suited for precision oriented retrieval.

## 3 Document Image Retrieval

### 3.1 Overview

The basic concept of the proposed method is to find parts of documents which densely contain terms in a query. The processing consists of the three steps shown in Fig. 1. Taking as input a set of index terms or *a query* extracted by the query processing, filtering is first applied to select pages which are likely to contain an answer to the question. Then the density distributions are calculated to find the parts which densely contain terms in the query. Finally, relevant parts are found based on the density distributions.

In the following, the details of each step are explained after a brief introduction of indexing and query processing.

### 3.2 Indexing

The process of indexing is basically the same as in our previous method. First, all words and their bounding boxes are extracted from page images with the help of OCR. Second, stemming and stopword elimination are applied to the extracted words[3]. The resultant words are called index terms (or simply terms) and stored with the centers of their bounding boxes. In other words, each page image is viewed as a two dimensional distribution of terms in it.

A new functionality introduced to the proposed method is the normalization of image size. In general, page images have various layouts. Some documents such as newspapers and technical journals may have multi-column layouts and thus densely contain a lot of terms in one page. On the other hand, others may have single-column layouts with a wider interline spacing and thus contain less terms. Documents with multi-column layouts would, therefore, be unevenly promoted if we simply computed the density of terms.

To avoid this harmful effect, it is necessary to normalize the size of page images. As the normalization constant $C$, we employ $C = H_m/5$ where $H_m$ is the *mode* of textline height included in each document.

### 3.3 Query Processing

The task of query processing is both to identify the type of question as well as to extract index terms from the question. Suppose we have a query “Where is the Baseball Hall of Fame?”. The query type is “location” from what it is asking and the index terms are “baseball”, “hall” and “fame”. Note that only the extraction of index terms is relevant to the task of document image retrieval. In the following, the sequence of extracted index terms is called the query and represented as $q = (q_1, ..., q_u)$ where $q_i$ is called a query term and $i$ indicates the order of occurrence in the question. For the above example, $q =$(baseball, hall, fame).

### 3.4 Filtering

Filtering is applied to ease the burden of the next step which is relatively time-consuming. The task here is to select $N_v$ pages that are likely to include the answer to the query.

[3] Stemming is the process of normalizing words by keeping only *word stems*, e.g., from “processes” to “process”. Stopwords are words that convey little meaning such as “a” and “the”.

For this purpose we utilize the simple vector space model (VSM) [12]. In the VSM, both a page $p_j$ and a query $q$ are represented as $m$-dimensional vectors:

$$\boldsymbol{p}_j = (w_{1j}, ..., w_{mj})^T \enspace , \tag{1}$$

$$\boldsymbol{q} = (w_{1q}, ..., w_{mq})^T \enspace , \tag{2}$$

where $T$ indicates the transpose, $w_{ij}$ is a weight of a term $t_i$ in a page $p_j$, and $w_{iq}$ is a weight of a term $t_i$ in a query $q$. In this paper, we employ a standard scheme called "tf-idf" defined as follows:

$$w_{ij} = \mathrm{tf}_{ij} \cdot \mathrm{idf}_i \enspace , \tag{3}$$

where $\mathrm{tf}_{ij}$ is the weight calculated using the term frequency $f_{ij}$ (the number of occurrences of a term $t_i$ in a page $p_j$), and $\mathrm{idf}_i$ is the weight calculated using the inverse of the page frequency $n_i$ (the number of pages containing a term $t_i$). In computing $\mathrm{tf}_{ij}$ and $\mathrm{idf}_i$, the raw frequency is usually dampened by a function. We utilize $\mathrm{tf}_{ij} = \log(f_{ij} + 1)$ and $\mathrm{idf}_i = \log(n/n_i)$ where $n$ is the total number of pages. The weight $w_{iq}$ is similarly defined as $w_{iq} = \log(f_{iq} + 1)$ where $\mathrm{f}_{iq}$ is the frequency of a term $t_i$ in a query $q$.

The similarity between a page $p_j$ and a query $q$ is measured by the cosine of the angle between $\boldsymbol{p}_j$ and $\boldsymbol{q}$:

$$\mathrm{sim}(\boldsymbol{p}_j, \boldsymbol{q}) = \frac{\boldsymbol{p}_j^T \boldsymbol{q}}{\|\boldsymbol{p}_j\| \; \|\boldsymbol{q}\|} \enspace . \tag{4}$$

where $\| \cdot \|$ is the Euclidean norm of a vector. Pages are sorted according to the similarity and top $N_v$ pages are selected and delivered to the next step.

### 3.5 Calculation of Density Distributions

This step is to calculate density distributions of the query $q$ for each selected page. Density distributions of the query is defined based on those of each query term $q_i$. Let $T_i^{(p)}(x, y)$ be a weighted distribution of a term $q_i(= t_k)$ in a selected page $p$ defined by:

$$T_i^{(p)}(x, y) = \begin{cases} \mathrm{idf}_k & \text{if} \quad q_i(= t_k) \quad \text{occurs at} \quad (x, y), \\ 0 & \text{otherwise} \enspace , \end{cases} \tag{5}$$

where $(x, y)$ is the center of the bounding box of a term. A density distribution $D_i^{(p)}(x, y)$ is a weighted distribution of $q_i$ smoothed by a window $W(x, y)$:

$$D_i^{(p)}(x, y) = \sum_{u=-M_x/2}^{M_x/2} \sum_{v=-M_y/2}^{M_y/2} W(x - u, y - v) T_i^{(p)}(u, v) \enspace . \tag{6}$$

As a window function, we utilize a pyramidal function with the window widths $M_x$ (the horizontal width) and $M_y$ (the vertical width) shown in Fig. 2.

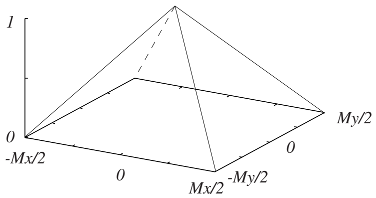

**Fig. 2.** Window function.

As discussed in 2.1, document image retrieval for the QA task should be precision oriented. An easy way of making retrieval precision oriented is to find parts which densely contain *all* the query terms. This is achieved by the point-wise multiplication of corresponding density distributions:

$$C_u^{(p)}(x,y) = \prod_{i=1}^{u} D_i^{(p)}(x,y) \ , \tag{7}$$

where $u$ is the number of query terms.

However, this causes a problem in many cases because it is relatively rare that all the query terms co-occur within a small region defined by the window function. In other words, $C_u^{(p)}(x,y)$ is zero if at least one of the density distributions $D_i^{(p)}(x,y)$ has the value of zero.

A way to avoid this undesirable situation is to relax the requirement. In this paper, we consider the smaller number of successive query terms. For example, the density distribution obtained by $u-1$ successive query terms is defined by

$$C_{u-1}^{(p)}(x,y) = \prod_{i=1}^{u-1} D_i^{(p)}(x,y) + \prod_{i=2}^{u} D_i^{(p)}(x,y) \ . \tag{8}$$

The reason for taking account of only the successive terms is that they are more relevant as compared to those randomly selected. For the general case of $k$ successive query terms, the density distribution is defined by

$$C_k^{(p)}(x,y) = \sum_{j=0}^{u-k} \prod_{i=j+1}^{j+k} D_i^{(p)}(x,y) \ . \tag{9}$$

In the proposed method, the density distribution of the whole query for a page $p$ is defined as the weighted sum of the combinations from all the $u$ terms down to $s$ successive terms:

$$D^{(p)}(x,y) = \sum_{k=s}^{u} \alpha_k C_k^{(p)}(x,y) \ , \tag{10}$$

where the parameter $s$ and the weight $\alpha_k$ are experimentally determined.

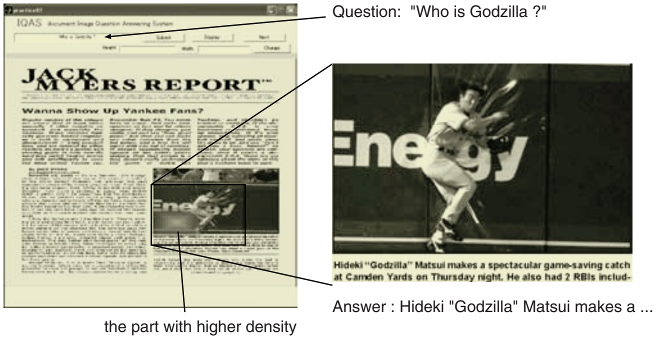
Question: "Who is Godzilla ?"
IQAS
JACK MYERS REPORT
Wanna Show Up Yankee Fans?
Hideki "Godzilla" Matsui makes a spectacular game-saving catch at Camden Yards on Thursday night. He also had 2 RBIs includ-
Answer : Hideki "Godzilla" Matsui makes a ...
the part with higher density

**Fig. 3.** Graphical user interface.

### 3.6 Spotting Relevant Parts

Based on the density distribution of Eq. (10), parts which are likely to include the answer are located on page images. First, page images are ranked according to their score of the maximum density:

$$s^{(p)} = \max_{x,y} D^{(p)}(x,y) \quad . \tag{11}$$

Then, the top-ranked page is presented to the user through the GUI shown in Fig. 3. In this figure, the part with high density is highlighted. The user can magnify the retrieved part in the page. If it does not contain the answer, the user can retrieve the next page.

## 4 Experimental Results

### 4.1 Data and Parameters

The proposed method of document image retrieval was tested using PDF documents about the major league baseball. The number of documents and the total number of pages are 197 and 1253, respectively. We employed PDF documents because it is the easiest way for us to obtain terms and their coordinates with no OCR errors. We consider that such clean data would be appropriate for evaluating the method as the first trial.

For the above documents, we prepared the queries shown in Table 1. Some queries are associated with several possible answers delimited by commas; we regarded an output of the method as correct if at least one of them is included. Some answers consist of several terms like "setup man" for the query 11. In such cases, an output must include all of them to be regarded as correct. The parentheses in Table 1 indicate stopwords in the answers; these were not checked for marking.

Table 2 lists the ranges of parameters tested in the experiments. As the unit of length for the window size, 1/5 of the mode of textline height in each

**Table 1.** Queries.

| Id | Query | Answer |
|---|---|---|
| 1 | What is the oldest stadium in Japan? | Koshien |
| 2 | Who is Godzilla? | Matsui |
| 3 | Who is the American League Leader in hits? | Ichiro |
| 4 | Who is the American League Leader in batting average? | Ichiro |
| 5 | Who is BRET BOONE? | (All-)star (second) baseman |
| 6 | From what are baseball gloves made? | cowhide |
| 7 | From what are baseball bats made? | wood |
| 8 | What variations are thrown in the major league? | seam fast ball, changeup, curveball, slider, split finger, forkball, knuckleball |
| 9 | Which team uses Koshien as home? | Hanshin |
| 10 | Who is Shigetoshi Hasegawa? | setup man |
| 11 | Where was Ichiro Suzuki born? | Japan, Tokyo, Honshu |
| 12 | Where is the Baseball Hall of Fame? | New York |
| 13 | Who is the world's best-known athletes? | Sosa, Jeter, Piazza, Rordriguez |
| 14 | Who is the most dominant and visible athlete in Japan? | Ichiro |
| 15 | Which stadium known as the House that Ruth Built? | Yankees |
| 16 | What is First Aid Kit Rule? | first aid kit |
| 17 | What team does Mark McGuire play for? | Cardinals |
| 18 | What team did Babe Ruth play for? | New York Yankees |
| 19 | What record is Mark McGwire close to breaking? | (the most) homeruns (in one) season |
| 20 | Which is the most famous stadium in Japan? | Koshien |

document is utilized. In the experiments, the window size varied from the height of 3.6 (=18/5) textlines to 20 (=100/5) textlines. The value of $s$ indicates the minimum number of combined terms. Thus if a query includes five terms and $s = 2$ is applied, the successive combinations of two terms up to five terms are considered in calculating the density distributions. As for the value of $\alpha_k$, we tested "1" (equal weight) and "$k$" (varied weight). Since $k$ corresponds to the number of combined terms, combinations with a larger number of terms are more important in the case of $\alpha_k = k$. The number of pages $N_v$ selected at the filtering was fixed to 10 throughout the experiments.

### 4.2 Evaluation and Experiments

The output of the method is the ranked pages with their density distributions. We regarded a page as correct in case a correct answer listed in Table 1 is found in the $N_t$ nearest terms to the peak of the density distribution in the page. For each query, top *five* pages obtained by the method are verified whether they are correct.

**Table 2.** Parameters and their ranges.

| Parameter | | Range |
|---|---|---|
| width of the window | $M_x$ | 18 ∼ 100 step 4 † |
| height of the window | $M_y$ | 18 ∼ 100 step 4 † |
| min. no. of terms combined in Eq.(10) | $s$ | 1 ∼ total no. of terms in a query |
| the weight for $C_k^{(p)}(x,y)$ | $\alpha_k$ | $\equiv 1$ or $= k$ |

† measured in units of 1/5 of the mode of textline height.

The results were evaluated using the score called "mean reciprocal rank" (MRR) defined as the average of "reciprocal ranks" for all queries. The reciprocal rank of a query is calculated as $1/r$ where $r$ is the rank of the page which first contains the correct answer. For example, if the third-ranked page first contains the correct answer, the reciprocal rank is 1/3.

Experiments were carried out based on the *leave-one-out cross validation*, i.e., values of parameters were selected by training based on the all but one left-out query and the selected values were applied to the processing of the left-out query as a test. MRR was obtained by leaving out every query and averaging resultant reciprocal ranks.

For the purpose of comparison, we applied the simplest variant of our previous method [7, 8] as the *baseline*. In this method, density distributions are calculated based not on Eq.(10) but on the following:

$$D^{(p)}(x,y) = \sum_{i=1}^{u} D_i^{(p)}(x,y) \ . \tag{12}$$

Except for this difference, all processing steps are shared with the proposed method.

### 4.3 Results and Discussion

Let us first show the results of training. Table 3 shows MRR obtained through the training. As the number of nearest terms $N_t$, which is related to the accuracy of results, we utilized 30 and 10; the task is harder with a smaller $N_t$. As shown in this table, the proposed method outperformed the baseline method for both values of $N_t$.

**Table 3.** MRR and values of parameters obtained by training.

| | $N_t$ | MRR | $M_x$ | | $M_y$ | | $s$ | $\alpha_k$ |
|---|---|---|---|---|---|---|---|---|
| | | | ave. | mode | ave. | mode | | |
| proposed method | 30 | 0.626 | 67.6 | 70 | 75.8 | 78 | 2 | $k$ |
| | 10 | 0.579 | 58 | 58 | 77.6 | 78 | 2 | $k$ |
| baseline method | 30 | 0.503 | 42.4 | 38 | 38 | 38 | — | — |
| | 10 | 0.490 | 33.8 | 30 | 33.4 | 30 | — | — |

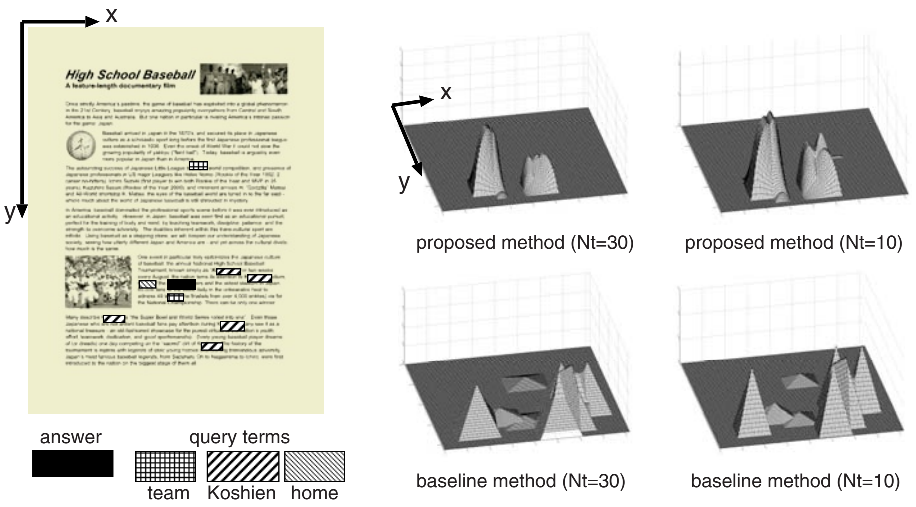

**Fig. 4.** Examples of density distributions.

Values of parameters selected at the training are also listed in Table 3. Let us next discuss the values of $s$ and $\alpha_k$, both of which are only for the proposed method. The proposed method often performed best with $s = 2$ and $\alpha = k$ for both $N_t$'s. The value $s = 2$ indicates that it is better not to take into account the case of $s = 1$, i.e., the distributions of single terms. As stated in Sect. 3.5, $s$ indicates the requirement of "co-occurrence" of successive terms within the window region. The baseline method is, on the other hand, to calculate density distributions by taking into account only the case of single terms (see Eq.(12)). Thus the results indicate that the co-occurrence plays an important role for locating the answer accurately. The selection of $\alpha = k$ means that the "co-occurrence" with a larger number of terms is more important than those with less terms.

Let us turn to the window widths. Table 3 shows that (1) smaller windows are required for smaller $N_t$ for locating the answers more accurately, and (2) the baseline method requires smaller windows as compared to the proposed method. Because smaller windows provide us less capability of smoothing the distributions, they are not desirable from the viewpoint of stability of the processing. For example, the baseline method with $N_t = 10$ uses the window of size $30 \times 30$. Since the typical height of body textlines are normalized to 5, the window is of size 6 lines. On the other hand the proposed method employs windows of size 12 to 16 textlines.

Examples of density distributions are illustrated in Fig. 4. The baseline method yielded some spikes in the distributions. On the other hand, the proposed method generated smooth distributions. In general, larger windows allow us to obtain smoothness, though they spoil the accuracy of locating the answers. The proposed method avoids this side effect by using the combinations of terms.

**Table 4.** Results of test.

| | $N_t$ | query Id | | | | | | | | | | | | | | | | | | | | MRR |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | |
| proposed | 30 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | $\frac{1}{2}$ | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0.575 |
| method | 10 | $\frac{1}{2}$ | 1 | 1 | $\frac{1}{2}$ | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0.450 |
| baseline | 30 | $\frac{1}{3}$ | 1 | 1 | $\frac{1}{3}$ | 0 | $\frac{1}{4}$ | 1 | 0 | 0 | 0 | $\frac{1}{5}$ | 0 | 0 | 1 | 0 | $\frac{1}{2}$ | 0 | 0 | 0 | $\frac{1}{3}$ | 0.298 |
| method | 10 | $\frac{1}{2}$ | 1 | 1 | 1 | 0 | $\frac{1}{4}$ | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | $\frac{1}{2}$ | 0 | 0 | 0 | $\frac{1}{3}$ | 0.379 |

Table 4 shows the results of test for the left-out queries. In this table, "query Id" indicates the left-out query and the numbers for them represent the reciprocal ranks.

For the queries 8, 12, 13, 15, and 17–19, neither of the methods could find the answers within top five pages. This was partly due to repetitious use of general query terms in pages. For example, the query 8 includes the terms "variation", "throw", "major" and "league" all of which are commonly used in documents on the major league baseball. Another and more important reason is that the methods are without the "filtering" capability based on the type of queries. For instance, the query 12 asks the location but only one page among all top five pages (in total 20 pages) included the name of location. Filtering would allow us to solve the problem as in the systems for electronic text.

For the queries 2, 3 and 14, both of the methods found the answers in the top ranked page. The difference between the methods was caused by the rest.

For the query 20, the proposed method was inferior to the baseline method. This was caused by the erroneous selection of the value of $s$ in the proposed method. In this case, $s = 3$ is selected by the training. Since the query includes three terms, the selection indicates the requirement of co-occurrence of all terms. But unfortunately, no page included all within the size of the window.

For the queries 5, 6, 9, 11 and 16, on the other hand, the proposed method outperformed the baseline method. In most of these cases, the baseline method yielded erroneous results due to the repetitious use of some terms in the query. For example, a page including the term "glove" frequently was erroneously ranked at the top by the baseline method, though the query also includes the terms "baseball" and "made". For these cases, therefore, the proposed method which put additional weights for co-occurrence of terms was successful.

In total, the values of MRR show that the proposed method outperformed the baseline method for both values of $N_t$.

## 5 Conclusion

In this paper we have presented a method of document image retrieval that is modified to be precision oriented for the task of question answering. The characteristic point of the method is that it takes combinations of successive query terms into account when calculating density distributions. This allows us to improve the accuracy of locating answers without increasing the spurious spikes in the distributions. From the experimental results we confirmed that the proposed method outperformed the baseline method.

Future work includes experiments with a larger number of queries and documents, as well as with OCR'ed documents. The implementation of the whole system with the capabilities of "query type identification" and "answer extraction" is also important future work.

## Acknowledgment

This research was supported in part by Grant-in-Aid for Scientific Research (C) from Japan Society for the Promotion of Science (No.14580453).

## References

1. Voorhees, E.M.: Overview of the TREC 2002 Question Answering Track, Proc. of Text REtrieval Conference 2002, http://trec.nist.gov/pubs/trec11/t11_proceedings.html .
2. http://trec.nist.gov/ .
3. Kwok, C. C. T., Etzioni, O. and Weld, D.S.: Scaling Question Answering to the Web, Proc. WWW10, pp.150–161, 2001.
4. http://www.ai.mit.edu/projects/infolab/ .
5. http://labs.nttrd.com/ (in Japanese) .
6. http://www.acm.org/dl/ .
7. Kise, K., Tsujino, M. and Matsumoto, K., Spotting Where to Read on Pages — Retrieval of Relevant Parts from Page Images, in *Proc. DAS'02*, pp.388–399, 2002.
8. Kise, K., Yin. W. and Matsumoto, K., Document Image Retrieval Based on 2D Density Distributions of Terms with Pseudo Relevance Feedback, in *Proc. ICDAR 2003*, pp.488–492, 2003.
9. Information Retrieval and OCR: From Converting Content to Grasping Meaning, A Workshop at SIGIR 2002.
10. Doermann, D.: The Indexing and Retrieval of Document Images: A Survey, *Computer Vision and Image Processing*, Vol. 70, No. 3, pp.287–298, 1998.
11. Kurohashi, S., Shiraki, N., Nagao, M.: A Method for Detecting Important Descriptions of a Word Based on Its Density Distribution in Text, *Trans. Information Processing Society of Japan*, Vol.38, No.4, pp.845–853, 1997 (In Japanese).
12. Baeza-Yates, R., Ribeiro-Neto, B.: *Modern Information Retrieval*, Addison-Wesley Pub. Co., 1999.