MATHEMATICAL MODEL OF INFORMATIN RETRIEVAL SYSTEM WITH FEEDBACK PROCESS

Sakai, Hiroshi
Department of Information Systems, Interdisciplinary Graduate School of Engineering Sciences, Kyushu University

Haraguchi, Makoto
Research Institute of Fundamental Information Science, Kyushu University | Department of Information Systems, Interdisciplinary Graduate School of Engineering Sciences, Kyushu University

Takeya, Shunichi
Research Institute of Fundamental Information Science, Kyushu University | Department of Information Systems, Interdisciplinary Graduate School of Engineering Sciences, Kyushu University

Kano, Seigo
Research Institute of Fundamental Information Science, Kyushu University | Department of Information Systems, Interdisciplinary Graduate School of Engineering Sciences, Kyushu University

https://doi.org/10.5109/13357

出版情報：Bulletin of informatics and cybernetics. 21 (1/2), pp.65-76, 1984-03. 統計科学研究会
バージョン：
権利関係：

Bulletin of Informatics and Cybernetics Vol. 21, No. 1~2, 1984

MATHEMATICAL MODEL OF INFORMATION RETRIEVAL SYSTEM WITH FEEDBACK PROCESS

By

Hiroshi SAKAI*, Makoto HARAGUCHI*,**
Shun-ichi TAKEYA*, and Seigo KANŌ*,****

Abstract

A mathematical model of information retrieval system with feedback process is described. The model is characterized by learning process in which the system gradually guesses the user's interest of retrieval in terms of keywords. Precise formulation of the total system based on probability theory and several results are presented.

1. Introduction

There are generally two types of document retrieval system when classified by the retrieval method. The one is Boolean type which is common in most commercial systems, and the other is document ranking type using inner product value as in SMART system [1]. In both types, a user usually retrieves documents through trials and errors, and finally gets an optimum result. From user's point of view, the fundamental problems of retrieval are how to find the appropriate keywords for his interest of retrieval, and, if it is possible, how to make system understand his interest.

In this paper, we deal with the ranking type method, and formulate a feedback system including users. We try to express the retrieval process uniformly on the basis of probability theory. In practice, the content of this paper is a system's learning process of keywords which appropriately reflect the user's interst of retrieval.

In the field of information retrieval, such query reforming problem is generally considered as query adjustment problem [1, 2, 3, 4]. There are several methods for it, for example, relevance feedback [1], but its formula includes indefinite parameters which may be decided by experience. Chow's feedback query [3] is another example, but it is only one time feedback and has some assumptions about parametric distribution on terms. We describe the total system's flow including users without using indefinite parameters or special parametric distributions.

* Department of Information Systems, Interdisciplinary Graduate School of Engineering Sciences, Kyushu University 39, Kasuga, Fukuoka, 816 Japan.

** Research Institute of Fundamental Information Science, Kyushu University 33, Fukuoka, 812 Japan.

2. Necessary Factors for Feedback Process

First, we define several factors for our feedback system, where D and T stand for the document set and the index term (keyword) set, respectively.

DEFINITION 2.1 (*User participation* I). Assume that a user can mark $R_\omega(d)$-value for documents, where "ω" is one of the user's interest of retrieval. If d_i is more relevant to the user's interest than d_j, then $R_\omega(d_i)$ is greater than $R_\omega(d_j)$.

$$R_\omega : D \longrightarrow [0, 1]. \tag{2.1}$$

The relation between documents and terms used in the retrieval system is given beforehand as Boolean type matrix Γ. But each term for a document doesn't seem to describe it uniformly. For this reason, we have the next definition.

DEFINITION 2.2 (*Contribution value*). Each document is assumed to have initially a probability distribution of terms. That is, for $\forall d \in D$,

$$M_d : supp(d) \longrightarrow [0, 1], \tag{2.2}$$

$$\sum_{t \in supp(d)} M_d(t) = 1, \qquad supp(d) = \{t \in T \mid t \text{ can retrieve } d\}.$$

We call $M_d(t)$ a contribution value of term t for document d. By (2.2), a contribution matrix Γ' is made, and it is not known well to user.

DEFINITION 2.3 (*Entropy of term over d*). $M_d(t)$ has probability over a document d. The entropy $H(d)$ of term is,

$$H(d) = \sum_{t \in supp(d)} -M_d(t) \times \log M_d(t). \tag{2.3}$$

Since an entropy expresses disorder, the less $H(d)$ becomes, the better the document d will be recognized.

Our feedback system Ψ is roughly sketched in Fig. 2.1. In each step, a document set presented by the system will be evaluated by the user. We call the set the system request.

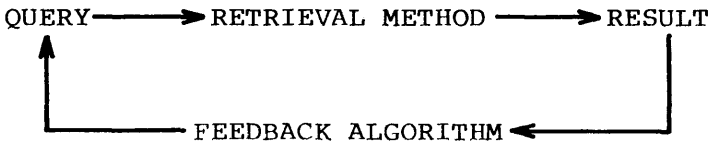

Fig. 2.1. Feedback process

Except the definition of the convergence of feedback process, our total system is denoted as follows;

$$\Psi = (T, D, \Gamma', G, F, \Lambda, R_\omega(\cdot)), \tag{2.4}$$

G: feedback algorithm,
F: inner product retrieval method,
Λ: system request series,
$R_\omega(\cdot)$: user participation.

We regard Ψ as a retrieval process.

3. Definitions of the Feedback System

DEFINITION 3.1 (*Query*). The query of n-step is,

$$Q_n=(q_1{}^n, \cdots, q_{\sharp T}{}^n), \tag{3.1}$$

$q_i{}^n$: weight of term t_i, $\quad \sharp T$: number of terms.

DEFINITION 3.2 (*Occupancy of term*). Occupancy of term t_i in n-step query is,

$$P_n(t_i)=q_i{}^n \Big/ \sum_{j=1}^{\sharp T} q_j{}^n . \tag{3.2}$$

DEFINITION 3.3 (*Efficiency of term*). After a document is evaluated by user, the efficiency of term t_i is given as follows;

$$M(t_i \mid d)=R_\omega(d)\times M_d(t_i), \tag{3.3}$$

$$M(t_i \mid X_n)=(1/\sharp X_{ni})\times \sum_{d\in X_{ni}} M(t_i \mid d), \tag{3.4}$$

X_n: system request, $X_{ni}=\{d\in X_n \mid M_d(t_i)>0\}\ (\neq\phi)$.

$M(t_i \mid d)$ doesn't depend on query and is stationary with respect to steps.

DEFINITION 3.4. (*Value of retrieval*). N-step query Q_n gives documents those values of retrieval;

$$P_{Qn}(d)=\sum_{t\in supp(d)} M_d(t)\times P_n(t), \tag{3.5}$$

$$supp(d)=\{t\in T \mid M_d(t)>0\}\quad (\neq\phi), \tag{3.6}$$

$$\text{if } supp(d)=\phi \text{ then } P_{Qn}(d)=0.$$

Now, we will use (3.5) as F of the system Ψ.

DEFINITION 3.5 (*Term probability*). When document d_j is evaluated by query Q_n, the post probability of term t_i over d_j is,

$$P_{Qn}(t_i \mid d_j)=M_{dj}(t_i)\times P_n(t_i)/P_{Qn}(d_j), \qquad (P_{Qn}(d_j)\neq 0). \tag{3.7}$$

These definitions are all given as probabilities and expected values.

PROPOSITION 3.1.

$$1)\quad 0\leqq P_{Qn}(d)\leqq \max_{t\in supp(d)} M_d(t). \tag{3.8}$$

$$2)\quad R_\omega(d_i)>R_\omega(d_j) \Leftrightarrow \sum_{t\in supp(d_i)} M(t \mid d_i)>\sum_{t\in supp(d_j)} M(t \mid d_j). \tag{3.9}$$

PROOF.

1) $$\begin{aligned} P_{Qn}(d)&=\sum_{t\in supp(d)} M_d(t)\times P_n(t) \\ &\leqq \sum_{t\in supp(d)} \{\max_{t\in supp(d)} M_d(t)\}\times P_n(t) \\ &=\max_{t\in supp(d)} M_d(t)\times \sum_{t\in supp(d)} P_n(t) \end{aligned}$$

$$\leqq \max_{t\in supp(d)} M_d(t).$$

2) $$\sum_{t\in supp(d)} M(t|d)=\sum_{t\in supp(d)} R_\omega(d)\times M_d(t)$$
$$=R_\omega(d)\times \sum_{t\in supp(d)} M_d(t)$$
$$=R_\omega(d).$$

PROPOSITION 3.2.

(3.10) 1) $$\sum_{t\in supp(d)} P_{Qn}(t|d)=1.$$

2) *If* $M_d(t_i)=M_d(t_j)$ *for* $\forall t_i, t_j\in supp(d)$ *and* $\sum_{t\in supp(d)} P_n(t)=1$ *then,*

(3.11) $$P_{Qn}(t_i|d)=P_n(t_i).$$

PROOF.

1) $$\sum_{t\in supp(d)} P_{Qn}(t|d)=\sum_{t\in supp(d)} \{M_d(t)\times P_n(t)/P_{Qn}(d)\}$$
$$=1.$$

2) $$P_{Qn}(t_i|d)=M_d(t_i)\times P_n(t_i)/\sum_{t\in supp(d)} M_d(t)\times P_n(t),$$
$$\sum_{t\in supp(d)} M_d(t)\times P_n(t)=M_d(t_i)\times \sum_{t\in supp(d)} P_n(t)$$
$$=M_d(t_i),$$
i.e. $$P_{Qn}(t_i|d)=M_d(t_i)\times P_n(t_i)/M_d(t_i)$$
$$=P_n(t_i).$$

$P_{Qn}(t|d)$ has a probability over a document d, and $P_n(t)$ has a probability over a term set. After the document is evaluated, probability $P_{Qn}(t|d)$ is equal to the prior term probability, if Condition 2) is satisfied.

PROPOSITION 3.3 (*Shannon's inequality*). *If* $P_1=(p_{11}, \cdots, p_{1n})$ *and* $P_2=(p_{21}, \cdots, p_{2n})$ *are two discrete probability distributions and* $0\times\log 0=0$ *then*

(3.12) $$\sum_{i=1}^{n} -p_{1i}\times\log p_{1i}\leqq\sum_{i=1}^{n} -p_{1i}\times\log p_{2i}.$$

Equality is given when two distributions are the same.

PROOF. In general, $\log x\leqq x-1$ $(x>0)$, so

$$\sum_{i=1}^{n} p_{1i}\times\log(p_{2i}/p_{1i})\leqq\sum_{i=1}^{n} p_{1i}\times\{(p_{2i}/p_{1i})-1\}$$
$$=\sum_{i=1}^{n}(p_{2i}-p_{1i})$$
$$=0.$$
$$\sum_{i=1}^{n} p_{1i}\times\log(p_{2i}/p_{1i})=\sum_{i=1}^{n} -p_{1i}\times\log p_{1i}+\sum_{i=1}^{n} p_{1i}\times\log p_{2i}$$
$$\leqq 0.$$

PROPOSITION 3.4.

$$0 \leqq H(d) \leqq \log(\#supp(d)). \tag{3.13}$$

PROOF. By Proposition 3.3,

$$\begin{aligned} H(d) &= \sum_{t \in supp(d)} -M_d(t) \times \log M_d(t) \\ &\leqq \sum_{t \in supp(d)} -M_d(t) \times \log(1/\#supp(d)) \\ &= \log(\#supp(d)). \end{aligned}$$

Next, $-M_d(t) \times \log M_d(t) > 0$ in $(0, 1)$, so if $\#supp(d) \geqq 2$ then $H(d) > 0$. If $\#supp(d) = 1$ then $H(d) = 0$.

THEOREM 3.1. *$P_{Qn}(t \mid d_i)$ has a probability over d_i by* (3.10), *so we define an entropy of terms, and write it $H_{Qn}(d_i)$. Then,*

$$H_{Qn}(d_i) \times P_{Qn}(d_i) < H(d_i), \qquad (H(d_i) > 0 \text{ only}). \tag{3.14}$$

PROOF.

$$\begin{aligned} H_{Qn}(d_i) &= \sum_{t \in supp(d_i)} -P_{Qn}(t \mid d_i) \times \log P_{Qn}(t \mid d_i) \\ &\leqq \sum_{t \in supp(d_i)} -P_{Qn}(t \mid d_i) \times \log M_{di}(t) \quad \text{by (3.12)} \\ &= (1/P_{Qn}(d_i)) \times \{ \sum_{t \in supp(d_i)} -M_{di}(t) \times P_n(t) \times \log M_{di}(t) \} \end{aligned}$$

$0 \leqq P_n(t) \leqq 1$ and $H(d) > 0$, so $\#supp(d) \geqq 2$ (by Proposition 3.4)

$$\begin{aligned} &< (1/P_{Qn}(d_i)) \times \{ \sum_{t \in supp(d_i)} -M_{di}(t) \times \log M_{di}(t) \} \\ &= H(d_i)/P_{Qn}(d_i). \end{aligned}$$

As $H(d_i)$ is fixed, according to (3.14) an increment of retrieval value causes the concentration of term (precisely speaking, term probability). The relation among the factors defined above is shown in Fig. 3.1.

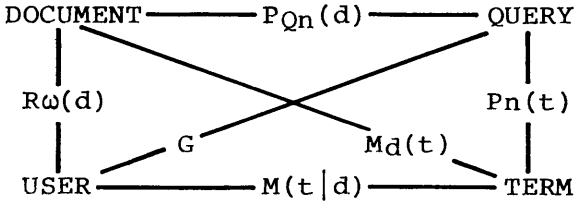

Fig. 3.1. Factors of the system Ψ

Next we define G of the system Ψ.

DEFINITION 3.6. After the user participation for n-step system request X_n, the weight of term t_i in n-step is defined as follows;

$$G : q_i^n(X_n) = (1/\#X_{ni}) \{ \sum_{d \in X_{ni}} R_\omega(d) \times P_{Qn-1}(t_i \mid d) \} \qquad (X_{ni} \neq \phi), \tag{3.15}$$

$$G : q_i^n(X_n) = 0 \qquad (X_{ni} = \phi).$$

For the formulation of the transition of the system Ψ, (3.15) will be the most important formula. It shows that term t_i will possess the relevancy of a document d with the probability $P_{Qn-1}(t_i|d)$ that is the probability of term t_i over d after evaluated by Q_{n-1}. Thus we use the term weights as to express the relevancy of documents. Therefore an evaluation of a document will eventually gives the same effect to documents that have similar terms. This property comes from the special condition of document-term matrix Γ'.

4. Definition of Classes by Document-Term Matrix

Before retrieval, we (the system) could know the relation between documents and terms according to Γ'. We define several classes on documents and terms using Γ'.

DEFINITION 4.1.

$$supp(d_i)=\{t\in T\,|\,M_{di}(t)>0\}\,, \tag{4.1}$$

$$supp(D_i)=\bigcup_{d\in D_i} supp(d)\,. \tag{4.2}$$

DEFINITION 4.2.

$$C^1(d_i)=\bigcup_{t\in supp(d_i)}\{d\in D\,|\,M_d(t)>0\}\,, \tag{4.3}$$

$$C^n(d_i)=\bigcup_{t\in supp(C^{n-1}(d_i))}\{d\in D\,|\,M_d(t)>0\}\,. \tag{4.4}$$

$C^n(d_i)$ is not monotone decreasing and $C^n(d_i)\subset D$, so $C^n(d_i)$ converges to $C(d_i)$. These classes are called as follows;

$C(d_i)$: document class of d_i-analogy,

$supp(C(d_i))$: term class of d_i-analogy.

PROPOSITION 4.1. *Any one document belonging to one class analogizes with documents in the same class and specifically if* $C(d_i)=D$ *then any one document analogizes with the total document set.*

PROOF. Let d be an arbitrary document in $C(d_i)$. Then d_i analogizes with d, so inversely d analogizes with d_i. Therefore $C(d)\supset C(d_i)$ and $d_i\in C(d)$. Equally, $C(d)\subset C(d_i)$ will be derived. So $C(d)=C(d_i)$.

PROPOSITION 4.2. *If* $d\notin C(d_i)$ *then* $supp(d)\cap supp(C(d_i))=\phi$.

PROOF. Suppose $supp(d)\cap supp(C(d_i))\neq\phi$, then there is at least one term $t(\in T)$ and it satisfies the following;

$$M_d(t)>0 \quad\text{and}\quad M_{d'}(t)>0 \qquad (d'\in C(d_i))\,.$$

The condition shows that d is analogized by d' through term t. It is contradictory to $d\notin C(d_i)$.

PROPOSITION 4.3. $C(d_i)\cap C(d_j)=\phi$ *or* $C(d_i)=C(d_j)$.

PROOF. If $d_i\in C(d_j)$ then the proof is equal to the one of Proposition 4.1. Therefore, suppose as follows;

$$d_i\notin C(d_j) \quad\text{and}\quad C(d_i)\cap C(d_j)\neq\phi\,,$$

then $\exists d\in C(d_i)$ and $\exists d\in C(d_j)$. That is, d_i is analogized by d_j through d, which contradicts to $d_i\notin C(d_j)$.

PROPOSITION 4.4. *$supp(C(d_i))\cap supp(C(d_j))=\phi$ or $supp(C(d_i))=supp(C(d_j))$.*

PROOF. If $C(d_i)=C(d_j)$ then $supp(C(d_i))=supp(C(d_j))$. This is trivial. Next, according to the result of Proposition 4.2 $supp(d)\cap supp(C(d_j))=\phi$ for $\forall d\in C(d_i)$, so

$$\bigcup_{d\in C(d_i)} supp(d)\cap supp(C(d_j))=\phi\,.$$

It shows that $supp(C(d_i))\cap supp(C(d_j))=\phi$.

These definitions are similar to those of transitive closure in graph theory. According to these propositions and definitions, we get the decomposition classes of D and T as follows;

$$D=\bigcup_i C(d_i)\,,\qquad T=\bigcup_i supp(C(d_i))\,. \tag{4.5}$$

THEOREM 4.1. *Formula* (3.15) *is well-defined on the only one class of D and T. Namely, the classes are independent with respect to retrieval.*

PROOF. Let $C(d_i)$ and $C(d_j)$ are two classes. By (4.5), $supp(C(d_i))\cap supp(C(d_j))=\phi$, so the system request $X_n\subset C(d_i)$ gives no information about $supp(C(d_j))$. Therefore $P_n(t)=0$ for $\forall t\in supp(C(d_j))$ and $P_{Qn}(d)=0$ for $\forall d\in C(d_j)$.

Theorem 4.1 clarifies the sets which $\mathcal{Y}$ can deal with. By Theorem 4.1 and (3.15) the system $\mathcal{Y}$ has a property of inference in a class. If we (the system) have any information about one class, we (the system) can reach to any document in the class by $R_\omega(d)$-value, but can never reach to documents of other classes. In practice, the classes are decided by matrix Γ, so the indexing of document is especially important in $\mathcal{Y}$.

In general, when we construct a feedback system, we must prepare both ways of concentration and expansion of the retrieved document set. But in $\mathcal{Y}$, the expansion of term set is decided in the initial step according to Theorem 4.1, and its concentration is done in sequence. We call here the process 'feedback'.

5. Some Results and Total System Algorithm

THEOREM 5.1. *The occupancy of term t_i in n-step ($n\geqq 2$) is,*

$$P_n(t_i)=Z_{in}\Big/\sum_{j=1}^{\#T} Z_{jn}\,, \tag{5.1}$$

$$Z_{in}=P_1(t_i)\times\prod_{m=1}^{n-1} E_{Qm}[t_i\,|\,X_{m+1}]\,,$$

$$E_{Qm}[t_i\,|\,X_{m+1}]=(1/\#X_{m+1,i})\times\{\sum_{d\in X_{m+1,i}} M(t_i\,|\,d)/P_{Qm}(d)\}\,.$$

PROOF. By induction,

$$\begin{aligned} q_i^2(X_2)&=(1/\#X_{2i})\times\{\sum_{d\in X_{2i}} R_\omega(d)\times P_{Q1}(t_i\,|\,d)\}\\ &=(1/\#X_{2i})\times\{\sum_{d\in X_{2i}} R_\omega(d)\times(M_d(t_i)\times P_1(t_i)/P_{Q1}(d))\}\\ &=P_1(t_i)\times(1/\#X_{2i})\times\{\sum_{d\in X_{2i}} R_\omega(d)\times M_d(t_i)/P_{Q1}(d)\}\\ &=P_1(t_i)\times E_{Q1}[t_i\,|\,X_2]\,. \end{aligned}$$

$$P_2(t_i)=q_i^2\Big/\sum_{j=1}^{\#T}q_j^2$$

$$=\{P_1(t_i)\times E_{Q1}[t_i\,|\,X_2]\}\Big/\sum_{j=1}^{\#T}\{P_1(t_j)\times E_{Q1}[t_j\,|\,X_2]\}$$

$$=Z_{i2}\Big/\sum_{j=1}^{\#T}Z_{j2}\,.$$

Next,

$$q_i^n(X_n)=(1/\#X_{ni})\times\{\sum_{d\in X_{ni}}R_\omega(d)\times P_{Qn-1}(t_i\,|\,d)\}$$

$$=(1/\#X_{ni})\times\{\sum_{d\in X_{ni}}R_\omega(d)\times(M_d(t_i)\times P_{n-1}(t_i)/P_{Qn-1}(d))\}$$

$$=(1/\#X_{ni})\times\{\sum_{d\in X_{ni}}R_\omega(d)\times M_d(t_i)/P_{Qn-1}(d)\}$$

$$\times\Big\{Z_{i,n-1}\Big/\sum_{j=1}^{\#T}Z_{j,n-1}\Big\}$$

$$=\Big(1\Big/\sum_{j=1}^{\#T}Z_{j,n-1}\Big)\times(1/\#X_{ni})$$

$$\times\{\sum_{d\in X_{ni}}R_\omega(d)\times M_d(t_i)\times Z_{i,n-1}/P_{Qn-1}(d)\}$$

$$=\Big(1\Big/\sum_{j=1}^{\#T}Z_{j,n-1}\Big)\times E_{Qn-1}[t_i\,|\,X_n]\times Z_{i,n-1}$$

$$=\Big(1\Big/\sum_{j=1}^{\#T}Z_{j,n-1}\Big)\times Z_{in}\,.$$

$$P_n(t_i)=q_i^n\Big/\sum_{j=1}^{\#T}q_j^n$$

$$=Z_{in}\Big/\sum_{j=1}^{\#T}Z_{in}\,.$$

In system Ψ, the occupancy of term t_i for query Q_n is sequentially reformed by (5.1) using system requests and $R_\omega(d)$-values. We want to define the end of retrieval with the convergence of term occupancy, but it is not trivial.

PROPOSITION 5.1. *If* $P_n(t_i)=0$ *then* $P_m(t_i)=0$ $(m\geqq n)$.

PROOF. By (3.7), $P_{Qn}(t_i\,|\,d)=M_d(t_i)\times P_n(t_i)/P_{Qn}(d)$. $P_n(t_i)=0$ is a condition, so $P_{Qn}(t_i\,|\,d)=0$ for $\forall d\in D$. By (3.15), $q_i^{n+1}=0$, so $P_{n+1}(t_i)=0$. We can prove it, sequentially.

According to Proposition 5.1, once the system regards a term as useless, the term doesn't cause any effect, so it can be eliminated from the term set.

PROPOSITION 5.2. *The process such as* $P_n(t_i)>0$ *and* $P_{n+1}(t_i)=0$ *occurs in either one of the following two cases.*

(5.2) 1) $R_\omega(d)=0$ *for* $\forall d\in X_{n+1,i}$ $(\neq\phi)$,

(5.3) 2) $X_{n+1,i}=\phi$.

PROOF. By (3.2), $P_{n+1}(t_i)=q_i^{n+1}\Big/\sum_{j=1}^{\#T}q_j^{n+1}$, and in order to $P_{n+1}(t_i)=0$, q_i^{n+1} must be 0.

According to (3.15), q_i^{n+1} is 0, when 1) or 2) is satisfied.

The case 1) is reasonable, and term's concentration will depend on the degree of elimination of terms by 1). The case 2) expresses a unique condition that system requests must satisfy. In each step, a system request must be decided such that the term set of requested documents covers the term set of 1-step before, and this condition excludes unreasonable elimination of terms. The problem of the system request seems to be solved by using a thesaurus [5, 6] and it is very important and difficult. We will consider the problem next time.

In fact, the term set, the document set and $supp(d)$ depend on the steps. So we denote the substances of them as T_n, D_n and $supp_n(d)$, respectively.

PROPOSITION 5.3. *T_n, D_n and $supp_n(d)$ are not monotone increasing in inclusion relation. A term is eliminated when one of the conditions of Proposition* 5.2 *is satisfied, and a document is eliminated when $supp_n(d)(=supp(d)\cap T_n)=\phi$ is satisfied.*

PROOF. If $q_i^n=0$ then $T_n=T_{n-1}-\{t_i\}$, so $T_n\subset T_{n-1}$ and never become $T_n\supset T_{n-1}$. It shows that T_n is not monotone increasing. By (3.6), if $supp_n(d)=\phi$ then $P_{Qn}(d)=0$, so $D_n\subset D_{n-1}$ and never become $D_n\supset D_{n-1}$. It shows D_n is not monotone increasing, too.

Substantial sets are decided by the next algorithm.

ALGORITHM 5.1.

begin
  calculate $q_1^n, \cdots, q_{\#T_{n-1}}^n$;
  $T_n:=T_{n-1}$;
  $D_n:=D_{n-1}$;
  for $i:=1$ **until** $\#T_{n-1}$ **do**
    if $q_i^n=0$ **then** $T_n:=T_n-\{t_i\}$;
  for $j:=1$ **until** $\#D_{n-1}$ **do**
    begin
      $supp_n(d_j):=supp(d_j)\cap T_n$;
      if $supp_n(d_j)=\phi$ **then** $D_n:=D_n-\{d_j\}$
    end;
end

PROPOSITION 5.4. *When $M_{di}(t_j)=\delta_{ij}$ (complete information) then one document makes one class, and can't analogize with other classes. So we eventually evaluate the total document set. In this case, system Ψ is useless.*

PROOF. Trivial.

We defined classes before retrieval, but in practice, we deal with the subset $D^*(\subset D)$. The total algorithm is shown next.

TOTAL ALGORITHM.

D_1 $(=\bigcup_i D(i) : D(i)$ is a subset of $C(d)$, $d\in D(i))$, (*)

T_1 $(=supp(D_1))$, $P_1(t_i)$ $(i=1, \cdots, \#T_1)$. (**)

(*) and (**) are given initially.

begin

query-type: $Q:=\bigcup_i Q_i$ (Q_i: query for $supp(D(i))$);

$n:=2$;

user: **for** $i:=1$ **until** $\sharp X_n$ **do** decide $R_\omega(d)$-value;
$sum:=0$;
term-set: $T_n:=T_{n-1}$;
query: **for** $j:=1$ **until** $\sharp T_{n-1}$ **do**
begin
calculate $q_j{}^n(X_n)$;
if $q_j{}^n(X_n)=0$ **then** $T_n:=T_n-\{t_j\}$;
$sum:=sum+q_j{}^n(X_n)$
end;
occupancy: **for** $j:=1$ **until** $\sharp T_n$ **do** $P_n(t_j):=q_j{}^n(X_n)/sum$;
value: **for** $i:=1$ **until** $\sharp D_1$ **do** calculate $P_{Qn}(d_i)$;
comment D_n and $supp_n(d)$ are decided by Algorithm 5.1;
if continue **then** $n:=n+1$ **else** stop;
comment System's convergence check is not given;
goto user
end

6. Convergence of Retrieval

It is natural to think that if term occupancies converge under the influence of each step's information, then the final occupancies will be the optimum term occupancies. By (5.1), each step's information is $E_{Qn}[t\,|\,X_{n+1}]$. It is concerned with Q_n, $M_d(t)$, $R_\omega(d)$ and the system requests. The expansion of term set is decided in the initial step, and the system request is given in such a way that the support of documents covers substantial term set, so the process doesn't seem to depend on the system requests too much.

Next, we describe how to mark $R_\omega(d)$-value in order to converge the process.

DEFINITION 6.1. (*User participation* II). Suppose that a user doesn't mark precise $R_\omega(d)$-value for system requests but he only classifies the system request roughly in the order of relevancy.

PROPOSITION 6.1. *If a user only divides a system request into two, namely, he classifies a system request as "$X_n=X_n(1)\cup X_n(2)$, $X_n(1)\cap X_n(2)=\phi$ and $X_n(1)$ is more relevant than $X_n(2)$", then $R_\omega(d)$-values can be decided as to satisfy the following*;

$$P_{Qn}(d_1)>P_{Qn}(d_2) \quad \textit{for} \quad \forall d_1\in X_n(1),\ \forall d_2\in X_n(2). \tag{6.1}$$

PROOF.

$$P_{Qn}(d)=\sum_{t_m\in supp(d)} M_d(t_m)\times P_n(t_m)$$

$$=\sum_{t_m\in supp(d)} M_d(t_m)\times\left\{q_m{}^n\Big/\sum_{j=1}^{\sharp T_n} q_j{}^n\right\}.$$

So, (6.1) becomes

$$\sum_{t_m\in supp(d_1)} M_{d1}(t_m)\times q_m{}^n>\sum_{t_k\in supp(d_2)} M_{d2}(t_k)\times q_k{}^n.$$

By (3.15),

$$\sum_{t_m\in supp(d_1)} (M_{d1}(t_m)/\sharp X_{nm})\times\{\sum_{d\in X_{nm}} R_\omega(d)\times P_{Qn-1}(t_m\,|\,d)\} \tag{6.2}$$

$$> \sum_{t_k \in supp(d_2)} (M_{d_2}(t_k)/\sharp X_{nk}) \times \{ \sum_{d \in K_{nk}} R_\omega(d) \times P_{Qn-1}(t_k \mid d) \} .$$

Here, the same $R_\omega(d)$-value is given for documents in the same class, and it must satisfy $R_\omega(d_1) > R_\omega(d_2)$, so there are two different values of $R_\omega(d)$. (6.2) is changed to the next.

$$R_\omega(d_1) \times \{ \sum_{t_m \in supp(d_1)} (M_{d_1}(t_m)/\sharp X_{nm}) \times \sum_{d \in X_{nm}} P_{Qn-1}(t_m \mid d) \}$$

$$> R_\omega(d_2) \times \{ \sum_{t_k \in supp(d_2)} (M_{d_2}(t_k)/\sharp X_{nk}) \times \sum_{d \in X_{nk}} P_{Qn-1}(t_k \mid d) \} .$$

Let $h_{ni} = \sum_{t_j \in supp(d_i)} (M_{di}(t_j)/\sharp X_{nj}) \times \sum_{d \in X_{nj}} P_{Qn-1}(t_j \mid d)$. Then,

$$R_\omega(d_1)/R_\omega(d_2) > h_{n2}/h_{n1} . \tag{6.3}$$

In (6.3), if X_n is given then the right side is fixed for d_1 and d_2. Next we substitute the maximum of numerator and minimum of denominator for each of the right side. We write it $K_n(X_n(1), X_n(2))$, and finally get the next formula.

$$R_\omega(d_1)/R_\omega(d_2) > \max\{1, K_n(X_n(1), X_n(2))\} . \tag{6.4}$$

If the system marks $R_\omega(d)$-values according to (6.4), then (6.1) is concluded.

THEOREM 6.1. *If a user classifies* X_n *as* $\bigcup_i X_n(i)$ ($X_n(i)$ *more relevant than* $X_n(i+1)$), *then* $R_\omega(d)$*-values can be decided to satisfy the following*;

$$P_{Qn}(d_i) > P_{Qn}(d_j) \quad \text{for} \quad \forall d_i \in X_n(i), \quad \forall d_j \in X_n(j) \quad (i<j) . \tag{6.5}$$

PROOF. Sequentially, system can decide conditions of $X_n(i)$ and $X_n(i+1)$ by Proposition 6.1. The following conditions are decided.

$$R_\omega(d_1)/R_\omega(d_2) > \max\{1, K_n(X_n(1), X_n(2))\} ,$$

$$R_\omega(d_2)/R_\omega(d_3) > \max\{1, K_n(X_n(2), X_n(3))\} ,$$

$$\vdots \quad \vdots \quad \vdots$$

$$R_\omega(d_{m-1})/R_\omega(d_m) > \max\{1, K_n(X_n(m-1), X_n(m))\} .$$

If $R_\omega(d_1)$ is fixed then the other $R_\omega(d)$-values are decided by these conditions, and they cause (6.5).

Theorem 6.1 needs enormous calculation. But the ranking of X_n influences the ranking of total document set. We finally find that a user need not give precise $R_\omega(d)$-values for a system request but just classify it. The method has merits of Boolean type and ranking type. Moreover if we sequentially adopt documents that are ranked as upper middle, the process converges steadily.

7. Conclusion

We defined several factors necessary for information retrieval system with feedback and formulated the retrieval process in contrast with Salton's relevance feedback formula. For the sake of the classes defined by matrix Γ', we cleared substantial sets that are dealt with by the system $\mathcal{Y}$ in the present step. (3.15) is the most important formula, and expresses the analogy of document in one class.

The user participation I and II are defined. Both of them will give the direction of retrieval, but the effect of II is much more steady, clear and realistic.

The authors hope that these discussions will clear the total retrieval process and give vivid influence to formulate it more in detail. The practical implementation of our system is not considered at present because of enormous calculation.

References

[1] SALTON, G. and MCGILL, M. J.: Introduction to modern information retrieval, Mcgraw-Hill, New York (1983).
[2] VERNIMB, C.: *Automatic query adjustment in document retrieval*, IPM, **13** (1977), 339-353.
[3] CHOW, D. and YU, C. T.: *On the construction of feedback query*, JACM, **29** (1982), 127-151.
[4] DRIYANSKII, V. M.: *Retrieval models in on-line documentary information systems: An analytic review*, Cybernetics, **17** (1981), 269-287.
[5] RAS, Z. W.: *An algebraic approach to information retrieval systems*, IJCIS, **11** (1982), 275-293.
[6] MAZUR, Z.: *Inverted organization in the information retrieval system based on thesaurus with weights*, IPM, **15** (1979), 227-234.

Received September 20, 1983