韓國 OA學會 論文誌
第6卷 第1號, 2001. 3.

2001-6-1-3-7

확률기법을 이용한 자동 문서 분할에 관한 연구

음 호 식* 이 명 호**

A Study on the Automatic Document Segmentation using Stochastic Method

Ho-Sik Eum* Myung-Ho Lee**

요 약

문서분할이란 내용별로 문서의 경계를 정하는 일로써 정확하고 효율적인 정보검색에 필수적이다.

본 논문에서는 단어간의 상호 정보를 이용하는 확률적인 분석 방법을 이용한 자동 문서 분할 시스템을 구현하고자 한다. 시스템은 윈도우의 경계를 이동시키면서 두 윈도우의 유사도를 계산해 내며 공유하는 단어들이 많을수록 그리고 공유하는 단어들의 중요도가 높을수록 두 윈도우의 응집도는 올라간다. 문서 분할 실험결과 블록의 단위가 달라지더라도 분할하고자 한곳이 정상적으로 분할 됨을 보였다.

Abstract

It is a document segmentation to set a boundary in the documents by the contents. It is essential for the accurate and efficient information search. In this paper, we want to make an automatic document segmentation system with the method of probability analysis which uses the mutual information between the words. Proposed system can move the boundary of window and compute the similarity of the two window. In this system the more words are shared and the more important the words are, the higher the cohesive force of the two window systems goes. The result of experience with the document segmentation is that despite the differences of block unit the division point at which we expected to divide was normally divided.

* 공주영상정보대학 소프트웨어개발 전공 조교수
** 청주대학교 전정반공학부 교수
논문 접수 : 2001년 1월 17일 심사 완료 : 2001년 3월 7일

Ⅰ. 서론

인터넷(internet)은 세상의 모든 것을 변화시키고 있으며, 더 많은 정보를 빠른 시간에 많은 사람에게 전달할 수 있게 하였다. 이러한 정보혁명은 정보의 홍수현상, 정보의 과부하(information overload)상태를 불러왔다. 이로 인해 정보검색 분야가 중요하게 부각하게 되었으며, 전문기술 분야에서 정보검색의 개념은 보다 구체적이고 제한적으로 사용된다. 30여년의 역사를 가지고 있는 이 분야에서 정보검색시스템의 검색대상은 비정형 정보 특히 텍스트 형태의 정보로 국한이 되어 왔고, 검색의 결과는 사용자가 원하는 문헌 혹은 문서의 형태를 가지고 있으므로 문서검색이라고 불리기도 한다. 현재까지 개발된 정보검색 시스템 대부분은 문서가 검색의 기본단위이기 때문에 커다란 문제점을 갖고 있다. 사용자의 관심은 필요한 정보에만 있는 것일 뿐이며, 과잉 생산된 정보는 사용자의 흥미를 유발시키는 것이 아니라 오히려 정신적인 압박감을 주기 때문이다.

이제는 know-how 보다 know-where, 즉 쏟아지는 정보 중 어디에서 필요한 정보를 찾는가 하는 문제에 중요성이 더해지고 있다. 다시 말해서 문서가 많아지고 방대해질수록 단순히 문서만을 검색하는 것이 아니라 문서 중에서도 일부분 필요한 부분을 찾아내고자 하는 필요성이 대두되고 있다. 이와 같이 문서를 문서 단위로 검색하는 것이 아니라, 특정 규칙이나 의미 있는 단위로 나누어 검색하는 분할 검색(passage retrieval)이 그 필요성과 중요성에서 날로 증가하고 있는 상황이므로, 본 연구에서는 검색하기 이전의 확률적인 자동 문서분할 기법에 관한 연구를 하고자 한다. 다양한 주제를 담고 있는 문서를 효율적으로 이용하기 위해서는 문서의 내용을 주제별로 구분하고 이로부터 사용자가 필요로 하는 정보를 선별하는 과정이 필요하다. 여러 가지 주제를 가진 문서의 내용을 선별하는 것은 결국 문서의 경계를 결정하는 문제와 같다. 본 논문에서는 분할되어 있지 않은 연속적인 글의 흐름을 확률적인 분석방법을 이용하여 문서의 경계가 되는 지점을 결정하는 문서분할 시스템을 제안한다.

이 분야에서 가장 중요한 부분은 분할자체가 성공적으로 이루어지도록 실험을 통해서 그 효용성을 보이고자 한다. 문서분할에 적용하고자 하는 기법은 확률적인 모델을 기반으로 하고 있는데 이는 정보검색 분야가 갖는 불확실성의 특성과 그 맥락을 같이 한다고 할 수 있다.

Ⅱ. 분할단위 관련연구

본 장에서는 분할의 의미를 살펴보고 지금까지 제안되고 개발된 분할 시스템을 분류하고 각각의 장단점을 분석하고자 한다. 본 연구에서 수행하고자 하는 연구와 가장 밀접하게 관련이 있는 부분이 분할의 단위이다. 실제로 기존의 연구를 살펴보면 분할의 단위에 따라서 검색성능의 차이를 보이고 있으며, 일반적으로 예상되는 분할의 단위가 아닌 다른 방법이 더 우수한 성능을 보였다는 것이 연구결과로 보고된 바 있다.

정확하게 지정된 분할의 단위는 존재하지 않는다. 즉, 분할의 단위는 확정된 것이 아니라 연구방향에 맞게 설정된 단위라고 생각하면 된다. 예를 들어, 문서를 작성하는 저자가 문서의 구조를 임의로 지정한 것이 분할의 단위가 될 수도 있고, 극단적으로 볼 때 문장 자체를 분할단위로 생각할 수도 있으며, XML과 같이 구조화된 문서의 경우 엘리먼트를 기본단위로 취급할 수도 있는 것이다. 지금까지 개발된 분할 시스템을 분류하면 다음과 같이 크게 세 가지 형태로 나눌 수 있다.

첫째, 문서에 이미 정의 되어 있는 문장, 혹은 단락 등을 분할의 단위로 이용하는 방법이다. 이는 문서자체가 아주 정형화되어 있지 않으면 잘못된 결과를 얻을 확률이 높다는 단점이 있다. 즉, 단락자체는 문서의 저자가 문서를 작성하면서 임의로 지정한 것이 되는데 같은 내용의 문서라도 저자에 따라서 단락을 다르게 구분하게 되는 등 주관성 개입이 너무 뚜렷하다는 것이다.

또한, 문장을 분할의 단위로 취급하는 경우에도 문장단위 자체를 인식하는데 많은 어려움이 있다는 것이다. 예를 들어, 문장의 마침표를 생략한 문장이 온라인 상의 많은 정보에 산재해 있는데 이 경우 문장인식 자체가 하나의 커다란 문제가 된다. 또한, 문장 내의 생략형을 나

타내기 위해 표현한 마침표를 잘못해서 문장의 끝으로 인식하게 되는 오류도 범하게 된다. 실제 연구 결과를 살펴보더라도 원시 문서의 분할자체의 오류와 문장 인식 오류로 인해 우수한 성능을 보일 수 없음이 보고되고 있다. 즉, 이러한 접근 기법은 일정한 규칙을 벗어난 문서에 대해서는 원치 않는 결과 값이 나오는 것을 배제할 수 없는 것이다.

둘째, 문서 내의 주제나 사건(subject, topic, event,…)을 분할의 기준으로 삼는 접근 방법이다. 이러한 접근은 문서의 의미를 파악해서 문서의 내용에 따라서 문장을 분류하는 방법으로 원시문서에 나타난 구조는 무시하고 문서 내용의 흐름에 따라 구분하는 것이다. 이 방법은 타당성이 있어 보이지만 내용을 파악한다는 자체가 어려운 문제로 아직까지는 만족할 만한 성능을 기대하기 어렵다. 그러나, 사건이나 주제가 확연히 나타나는 일련의 신문기사 문서집합을 대상으로 하는 주제인식(topic detection) 연구는 활발히 진행되고 있는 상황이다.

셋째, 문서를 일정한 크기로 분할하는 방법이다. 즉, 문장의 구조를 전혀 고려하지 않고 일정한 단어의 개수를 기준으로 분할하는 것이다. 분할되는 크기를 윈도우 크기(window size)라고 하는데 보통 몇 십개의 단어로부터 몇 백개의 단어가 하나의 윈도우가 된다. 분할되는 단위 사이에는 일반적으로 겹치는 부분을 만들게 되는데, 이는 임의로 크기를 정한 상태로 분할하는 것이기 때문에 잘못 분할되는 것을 조금이나마 완화시키고자 하기 위한 방책이다. 즉, 문서를 {D1, D2, D3, · · · Dn}으로 분할 했을 경우 D1의 뒷부분과 D2의 앞부분은 D1과 D2에 동일하도록 하는 것인데, 실제 검색 시간에는 각각의 우선순위가 부여되기 때문에 부분적으로 중복된 내용이 있더라도 상관이 없다. 세 번 째로 설명한 접근방법은 실제 연구결과에 의하면 앞서 기술한 첫 번 째와 두 번 째 설명의 접근방법보다 우수한 성능을 보인다는 것이 연구결과로 제시되고 있다. 이는 여러 가지로 해석될 수 있는데 먼저 기술한 방법들이 좀 더 과학적이고 체계적인 방법 같지만 완벽하지 않다는 점에서 이 방법에 비해 우수하다고 단정할 수 없다는 것이다. 또한, 마지막 방법이 정보검색 모델에 좀 더 융합이 잘 될 수 있다는 장점이 있다. 왜냐하면, 분할단위의 크기가 일정하기 때문에 검색성능에 많은 영향을 끼치는 요소 중의 하나인 정규화 문제가 전혀 없다는 것이다.

Ⅲ. 확률적인 분할 시스템

본 논문에서는 확률적인 접근 방법을 통해 분할의 문제를 해결하고자 한다. 즉, 인접한 문장 사이의 결합 가능성을 확률적으로 계산하여 일정한 수준 이상의 확률을 갖게 되면 결합을 시키고 그렇지 않으면 분할하는 방법이다. 여기서 두 문장의 유사도를 계산하는 식은 다음과 같다.

$$Sim_{A,B} = \sum_{i=1}^{n} P(A_{Ti}, B_{Ti})$$

이 식은 두 개의 인접한 문장 A, B의 유사도를 계산하는 식으로 문장이 n개의 키워드로 구성되었다고 가정한다. 이 식에서 $P(A_{Ti}, B_{Ti})$는 다음과 같이 구한다.

$$P(A_{Ti}, B_{Ti}) = \frac{P(A_{Ti}, B_{Ti})}{P(A_{Ti}) \cdot P(B_{Ti})}$$

이 식은 두 문장 A, B 각각에 나타난 키워드가 출현빈도를 통계 값으로 표현하고 있다. 즉, 두 개의 키워드 A_{Ti}, B_{Ti}가 동시에 나타날 확률 $P(A_{Ti},B_{Ti})$를 A_{Ti}가 임의로 나타날 확률 $P(A_{Ti})$과 B_{Ti}가 임의로 나타날 확률 $P(B_{Ti})$의 곱으로 나누는 식이다. 결과적으로 두 개의 키워드가 동시에 나타날 확률 값이 높으면 높을수록 두 문장의 유사도는 높게 되고 그렇지 않은 경우 낮기 때문에 분할의 근거가 되는 것이다. 위와 같은 과정에서 한 가지 해결해야 될 것은 유사도를 계산하기 위해서 필요한 각각의 확률 값을 구하는 것이다. 확률 값은 학습 문서 집합을 이용해서 추출하여 이용하는 것이 통계적인 접근 방법에서 취하는 일반적인 방법이다. 즉, 통계 수치를 인정받기에 충분할 정도의 문서집합으로부터 필요한 통계 값을 획득하는 것이다. 본 논문에서도 이러한 접근방법을 적용하는 것을 기본으로 한다. 또한, 분할하고자 하는 문서로부터도 학습문서에 적용한 같은 내용의 통계 수치를 추출

문장번호	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
미국	1		1	1	1		1		1	1				1		1	1	1				1			1	1	1		1	1					
부시	1	1	1		1	2						1		1	1							1	1	1					1			1	1		
정책											1				1	2	1	1											1			1	1	1	
경제					1			2	1			1				2	2		2																
정치					1			1						1													2				1				
문제					1			1								1																			
세계			1						1								1										1			1	1				
행정		2	1			1						1			1																				
아시아																																			1
정부																						1										1	1	1	
변화										1	1									1	1													1	
대통령					1			1														1		1					1						
영향					1					1																				1					
국가				1																				1	1							1			
부가		2	1												1																				
관심					1														1				1												
국정					1		1	1																											
운영					1		1	1																											
국민				1																							1	1							
취임				1																		1		1											
클린턴		1				1		1																											
지도자																										1		1	1						
안정																												1	1						
갑세															1	1	1																		
대외																													1				1	1	

〔그림 1〕 문서 내의 단어 분포도

하여 실시간으로 추가함으로써 성능향상을 꾀하고자 한다. 위에서 제시한 방법은 문장들 사이의 친밀도를 확률적인 방법으로 접근하여 계산하고자 하는 것으로 통계적인 관점에서 보았을 때는 기본적인 이론이다. 또한, 지금까지 키워드의 확률적인 분포를 이용한 분할 문제를 해결한 연구가 없었으며, 학습 문서의 통계 값만을 이용하지 않고 실시간 데이터도 이용한다는 측면에서 의의가 크다고 할 수 있다.

그림1은 신문기사의 일부분을 나타낸 것으로서 문서 내에서의 빈도수가 3이상인 단어에 대한 분포도를 나타낸 것이다. 문서에서 내용의 전개에 따라서 그에 맞는 적당한 단어가 선택되어 사용되는데, 특정 위치에 나타나는 각 단어의 출현빈도가 다를 것이라고 가정하고 단락검색의 필요성을 제기했다. 그림1을 통해 단어의 빈도수가 다양하게 나타나고 있음을 알 수 있으며, 그 단어들의 출현위치 및 분포도를 확인해 볼 수 있다. 행정, 클린턴, 국정과 같은 단어는 문서의 전반부에, 아시아, 정부와 같은 단어는 문서의 후반부에 집중적으로 나타나는 것을 볼 수 있다. 이와 같이 문서의 각 부분이 나타내고 있는 내용에 따라서 단어의 분포가 달라지게 되는 것이다. 따라서, 단어의 분포도가 각 단락의 특징을 나타내므로 이러한 단어를 이용해서 문서 내의 단락을 자동으로 생성할 수 있는 것이다. 위와 같은 방법으로 단락을 유추하기 위해서는 먼저 단락을 대표할 만한 가능성이 있는 단어들을 추출해야 한다. 일반적으로 정보검색에서 단어의 중요도를 나타낼 때 역문헌 빈도수(IDF: Inverse Document Frequency)를 이용한 "문서 내 단어 빈도수*(전체 문헌수/ 문서 빈도수)" 가중치를 적용한다. 그러나, 단어 각각의 중요도만을 이용해서 단락을 인식할 경우 문맥을 전혀 고려하지 않기 때문에 오류가 발생할 가능성이 높지만, 그림1에서 보았듯이 특정 위치에서 발생하는 단어들의 상호연관성을 단락구분의 중요요소로 이용할 필요가 있으며 본 연구에서는 이를 이용하기로 한다.

본 연구를 위해서 개발한 시스템의 전체적인 구성은 그림2 에서 보는 바와 같다.

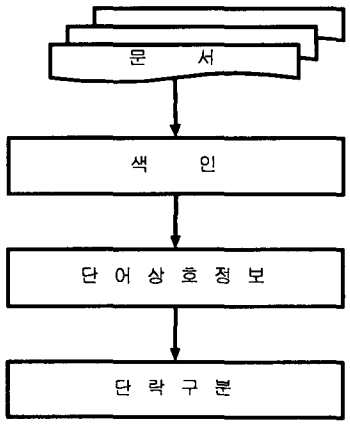

그림 2. 시스템 구성도

그림2의 각 단계별 시스템 흐름을 살펴보면 다음과 같다.

첫째, 단락인식의 대상이 되는 문서 집합을 입력으로 받아서 색인을 하게 된다. 이 과정은 문서에서 명사 위주의 중요단어를 추출하는 단계로서 불필요한 단어(stop word)들을 제거하기도 한다. 이 과정을 통해서 각 색인어들의 빈도수를 알 수 있게 되는데 그 빈도수가 3보다 작은 단어에 대해서는 상호정보 계산에서 제외하였다.

둘째, 각 단어 사이의 상호정보를 계산하는 단계이다. 본 연구에서 사용하게 될 상호정보는 두 단어가 특정 부분에서 발생하게 될 확률이므로 위치에 대한 정의가 필요하다. 즉, 여기에서 말하고 있는 "특정 부분"이라는 단어에 대한 정의가 필요한데 보통 이를 윈도우(window), 또는 블록(block)이라 칭한다. 윈도우는 단어의 수가 될 수도 있으며 문장 개수가 될 수도 있다. 예를 들어, 윈도우 크기가 20단어라고 한다면 이 윈도우 내에서 발생하게 되는 단어들 사이의 확률이 상호정보가 되는 것이다. 따라서, 윈도우 크기에 상호정보 값이 다르게 나타나게 되는데 본 연구에서는 윈도우 크기를 다양하게 할 수 있도록 함으로써 이를 이용한 실험을 할 수 있도록 하였으며, 윈도우는 문장을 기준으로 했는데 이는 단락 분할 시 복잡성을 줄이기 위해서 이다. A와 B가 동시에 발생한 확률을 A가 발생할 확률의 값으로 나누어준 값이 상호정보 값이며 계산하는 식은 다음과 같다.

$$P(B/A,A) = \frac{P(B/A)}{P(A)}$$

일반적으로 상호정보는 대용량의 코퍼스(corpus)를 이용하여 계산하지만 본 연구에서는 입력 문서에 대해서 실시간으로 계산한 후 이용하는 방법을 적용하였다. 이를 통해 분야에 의존적인 확률 값을 미리 계산해 놓는 노력을 제거하고자 하였다.

마지막 단계는 상호정보 값을 이용하여 분할하는 것이다. 즉, 윈도우 사이의 상호정보를 이용한 유사도를 계산한 후 그래프를 그려 값의 변화가 심한 계곡을 단락으로 인식하고자 한다. 윈도우 사이의 유사도는 두 윈도우에 속한 단어들의 상호정보를 더한 후 윈도우 크기로 나눈 값이다. 단락분할을 하는 방법은 여러 가지가 있을 수 있으나 본 연구에서는 유사도 값이 계속 감소하다가 증가하는 부분을 단락으로 지정하였다. 다음의 그림3은 2개의 신문기사를 하나의 문서로 합하여 시스템에 입력시킨 후 윈도우 크기를 1로 했을 경우 윈도우 사이의 유사도 값을 계산한 결과를 보여주고 있다. 문장 사이의 단어 상호정보가 적어 유사도가 낮을수록 골짜기가 생성되는데 이곳이 단락의 경계선이 된다. 윈도우의 크기를 다양화한 실험에서도 이러한 형태의 결과를 볼 수 있었는데 윈도우의 크기가 작을수록 골짜기가 많았으며 반대로 윈도우의 크기가 클수록 골짜기의 개수가 감소되었다. 그러나 윈도우 크기의 변화에 따른 골짜기의 개수변화와 상관없이 실제 단락이 분할되어야 할 부분은 공통적으로 골짜기로 인식하기 때문에 문제가 되지 않았다. 단락의 구분을 세밀하게 하느냐 아니냐의 여부에 따라서 블록의 크기조절이 가능하기 때문에 시스템을 그 사용목적에 따라서 크기를 적용시킨다면 시스템의 성능 향상에 도움이 되리라 기대된다.

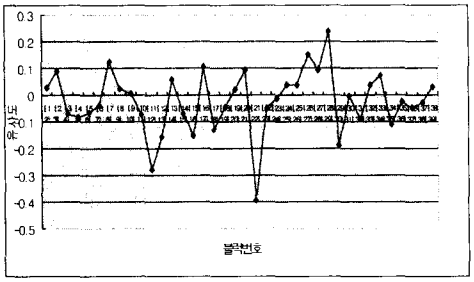

그림 3. 윈도우 크기가 1인 경우

Ⅳ. 텍스트 분할 실험

제안한 분할기법의 검증을 위한 실험으로 분할단위의 크기에 따른 실험결과를 정확도와 재현도로 제시한다.

정확도(precision)와 재현도(recall)는 다음과 같이 계산한다.

$$\text{정확도} = \frac{\text{정확히 분할한 갯수}}{\text{전체 분할한 갯수}}$$

$$\text{재현도} = \frac{\text{정확히 분할한 갯수}}{\text{분할 해야되는 정답 갯수}}$$

실험데이터는 두 가지 다른 종류를 이용하기로 하며, 분할기점에 대한 주관성을 배제하기 위해서 이미 분할되어 있는 데이터를 하나의 문서로 통합된 상태에서 분할하는 실험을 하기로 한다.

첫째, 신문기사 여러 개를 모아놓고 원래 상태로 분할하는지 측정하는 실험이다. 지금까지의 대부분 연구에서 실험에 사용하고 있는 형태로 이를 통해서 분할자체가 제대로 이루어지는지를 알아보도록 한다.

둘째, 원래 문서자체가 장문인 데이터에 대한 실험이다. 이 실험은 학회지에 게재된 논문을 대상으로 하며 논문에 구분되어 있는 단락 인식정도를 알아보는 실험이다. 제안한 방법의 최적결과를 생성하기 위해서 분할단위의 크기를 다양화해서 실험하도록 한다. 문장단위를 최소 분할의 크기로 정하고 문장이 모여 블록(block)이 이루어진다고 가정하고, 블록 크기 1부터 5까지 변화시켜 가면서 시스템의 성능을 측정한다.

표 1. 신문기사를 대상으로 한 실험결과

블록크기 (block size)	정확도 (precision)	재현도 (recall)
1	0.4696	0.7538
2	0.8601	0.8307
3	0.8629	0.7154
4	0.8498	0.6769
5	0.7005	0.5615

표 2. 논문 데이터를 대상으로 한 실험결과

블록 크기 (block size)	정확도 (precision)	재현도 (recall)
1	0.4105	0.7983
2	0.8248	0.8507
3	0.8229	0.8164
4	0.7889	0.7832
5	0.7025	0.6934

V. 고 찰

제안한 방법의 효용성을 평가하기 위한 실험은 다각도에서 이루어질 수 있다. 먼저 실험을 위해 사용되어야 할 문서 집합의 선정은 신중하게 이루어져야 한다. 왜냐하면, 문서집합의 종류에 따라서 평가 결과가 다르게 나올 수 있기 때문이다. 본 논문에서는 테스트를 위한 문서집합이 지정되어 있는 것이 아니기 때문에 논문과 같이 정형화된 문서집합과 길이가 다소 짧은 신문기사로 실험하여 성능의 차이를 분석한다.

1) 블록크기에 따른 성능: 블록의 크기가 성능에 미치는 영향은 결과에서 보듯이 크게 나타난다.
블록의 크기가 크다는 것은 그만큼 이용할 정보가 많다는 것이고 크기가 작다는 것은 그 반대이다. 일반적으로 검색 시스템은 사용자 질의에 포함된 모호하고 적은 정보로 인해 성능저하의 결과를 초래할 수 있지만 본 실험결과는 무조건 블록 크기가 크다고 해서 좋은 결과를 보이지 않음을 제시하고 있다. 또한, 블록크기가 너무 작은 경우에도 성능이 떨어지고 있음을 알 수 있다. 따라서, 적당하게 블록크기를 설정해야 최상의 결과를 얻을 수 있다.

2) 데이터 특성에 따른 성능: 결과에서 보듯이 전체적인 정확도면에서는 신문데이터를 대상으로 한 경우가 성능이 좋았으며, 반대로 재현도면에서는 논문을 대상으로 한 경우가 좋은 성능을 보이고 있다. 이는 신문 데이터의 경우 상이한 내용이 분할의 기점이 되는 경우가 많고, 신문데이터는 분할단위에 속한 문장의 평균 개수가 20여 개 정도로 짧은 반면 논문은 120여 개로 데이터의 특성이 다르기 때문에 나타나는 현상이다.

Ⅵ. 결 론

본 논문에서는 단어의 상호 정보를 이용한 확률적인 접근방법을 통해 자동문서 분할 시스템을 구현하였다. 일반적으로 확률정보는 학습 문서 집합을 이용해 미리 생성하기 때문에 학습과정이 필요하지만 본 연구에서는 대상이 되는 문서 집합으로부터 직접 확률정보를 추출하기 때문에 학습과정이 필요 없다는 장점이 있다. 신문기사와 문서를 대상으로 하는 실험을 통해 제안한 방법의 성능을 정확도와 재현도로 보여 주었으며, 윈도우의 크기와 적용 문서에 따라서 실험결과 여러 가지 특징이 나타남을 알 수 있었다. 실험 데이터가 다르므로 다른 연구와의 직접적인 비교는 불가능하지만, 향후 검색시스템을 이용한 실험을 통해서 분할여부에 따른 검색성능의 평가는 반드시 필요하며 이 과정에서 얻게 되는 여러 가지 결과 값 및 분석 내용은 향후 이루어질 연구에 많은 보탬이 되리라 기대한다.

참고문헌

[1] Gerard Salton & Michael J. McGill, Introduction to Modern Information Retrieval, McGraw-Hill Book, 1983.

[2] Gerard Salton, J. Allan & C. Buckley, Approaches to Passage Retrieval in Full Text Information Systems , ACM SIGIR 93, pp.49-58, 1993.

[3] Elke Mittendorf & Peter Schauble, Document and Passage Retrieval Based on Hidden Markov Models , ACM SIGIR 94, pp.318-327, 1994.

[4] Michael Hess, Deduction over Mixed-Level Logic Representations for Text Passage Retrieval , in Proc. Of the 1996 International Conference on Tools with Artificial Intelligence, pp.383-390, 1996.

[5] Gerard Salton & J. Allan, Automatic Text Theme Decomposition and Structuring , in Proc. RIAO 94, pp.6-20, 1994.

[6] Marcin Kaszkiel & Justin Zobel, Passage Retrieval Revisited , ACM SIGIR 97, pp.178-185, 1997.

[7] Ross Wilkinson, Effective Retrieval of Structured Documents , ACM SIGIR 94, pp. 311-317, 1994.

[8] Marti A. Hearst & Christian Plaunt, Subtopic Structuring for Full-Length Document Access , SCM SIGIR 93, pp.59-68.

[9] James P. Callan, Passage-Level Evidence in Document Retrieval , ACM SIGIR 94, 1994.

[10] Amit Singhal, Chris buckley & Mandar Mitra, Pivoted Document Length Normalization , SCM SIGIR 96, pp.21-29, 1996.

[11] William B. frakes, Ricardo Baeza-Yates, Information Retrieval: Data Structures & Algorithms, Prentice Hall, 1992.

[12] James Allen, Natural Language Understanding, 2nd Edition, The Benjamin/ Cummings Publishing Company, 1995.

저 자 소 개

음 호 식

1991년 2월 : 청주대학교 전자계산과(공학석사)
1999년 2월 : 청주대학교 전자공학과(전자계산 및 계산기 전공)박사수료
1993년 3월 ~ 현재 공주영상정보대학 소프트웨어개발 전공 조교수
관심분야 : 네트워크, 에이전트, 멀티미디어통신처리

이 명 호

1981년 : 연세대학교 전자공학과(공학석사)
1991년 : 연세대학교 전자공학과(공학박사)
1984년 ~ 현재 청주대학교 전정반공학부 교수
관심분야 : 정보기술(IT), Communications