문서 내용의 계층화를 이용한 문서 비교 방법

황명권* · 배용근** · 김판구**

Document Clustering Methods using Hierarchy of Document Contents

Myung Gwon Hwang* · Yong Geun Bae** · Pan Koo Kim**

이 논문은 2004년도 조선대학교 연구 보조비 지원에 의하여 연구되었음.

요 약

웹의 비약적인 성장으로 웹에는 무수한 정보를 축적하고 있으며, 특히 텍스트 문서는 인간에 의해 가장 쉽게 그리고 많이 이용되는 형식이라 하겠다. 텍스트 문서의 효율적 검색을 위해 많은 연구가 이루어졌으며, 확률을 이용한 방법, 통계적인 기법을 이용한 방법, 벡터 유사도를 이용한 방법, 베이지안 자동문서 분류 방법 등이 제안되었다. 그러나 이러한 기존의 방법들은 문서의 특징을 정확하게 반영할 수 없고, 의미적 검색이 이루어지지 않는 단점을 가지고 있다. 이에 본 논문은 문서를 미리 분류하는 기존의 방법을 개선하기 위해, 유사한 문서를 의미적으로 찾아내기 위한 새로운 문서 분류의 척도를 제안하며 이를 적용하는 방법을 제시한다. 본 방법은 문서의 내용을 의미적인 계층으로 표현하고 중요 도메인에 가중치를 두며, 문서들간의 도메인 가중치와 도메인 내의 개념 일치도를 이용하여 유사도를 구한다.

ABSTRACT

The current web is accumulating abundant information. In particular, text based documents are a type used very easily and frequently by human. So, numerous researches are progressed to retrieve the text documents using many methods, such as probability, statistics, vector similarity, Bayesian, and so on. These researches however, could not consider both subject and semantic of documents. So, to overcome the previous problems, we propose the document similarity method for semantic retrieval of document users want. This is the core method of document clustering. This method firstly, expresses a hierarchy semantically of document content and, gives the important hierarchy domain of document to weight. With this, we could measure the similarity between documents using both the domain weight and concepts coincidence in the domain hierarchies.

키워드

문서 분류, 계층 비교, 클러스터링, 명사추출

I. 서 론

웹의 비약적인 성장으로 현재의 웹은 인간이 필요로 하는 모든 정보를 텍스트, 이미지, 비디오, 사운드 등의 다양한 데이터 형식으로 담고 있으며, 매일 새로운 정보들이 생성되고 있다. 특히, 텍스트 문서의 검색을 위해 일반

* 조선대학교 컴퓨터공학부 박사과정
** 교신저자 : 조선대학교 컴퓨터공학부 교수

접수일자 : 2006. 11. 1

적으로 문서의 핵심 키워드를 이용한 키워드 매칭을 이용하고 있다. 하지만, 이러한 방법을 통해 나타나는 검색 결과는 단순히 특정 키워드의 출현빈도와 무의미한 단어 매칭을 하여 사용자는 원하는 정보를 찾기 위해 재 검색을 수행해야 하는 번거로움이 있다.

웹 문서의 효율적인 검색을 위해, 문서를 자동으로 분류하는 방법으로 확률을 이용한 방법[1,2], 통계적인 기법을 이용한 방법[3,4], 벡터 공간을 이용하는 방법[2,12], 베이지안 확률을 사용한 방법[5,6] 그리고 퍼지확률을 이용하는 방법[11] 등이 연구되었다. 이 연구들은 사용자가 검색을 수행할 때 문서 안에 출현한 단어들 또는 미리 학습된 규칙을 이용하여 문서들을 분류하며, 질의어에 해당하는 문서그룹을 보여주는 방식이다. 이들 방법들은 효율적인 분류 결과를 도출했으나 여전히 검색결과가 의미적이지 못하다는 한계점을 갖고 있다. 효과적이고 의미적인 웹 문서 검색 기술은 급증하는 웹 문서의 관리를 위해 가장 중요하지만, 문서 내용의 의미를 추출하는 것은 매우 어렵다.[13] 이에 사용자는 원하는 질의어에 대한 검색된 결과들을 이용하여 직접 재검색을 수행해야하는 문제점이 발생한다. 이러한 문제점을 극복하기 위해, 본 논문에서는 문서 분류의 핵심이 되는 문서비교 방법을 제안한다. 제안하는 방법은 문서에 포함된 명사들을 미리 구축된 대형의 온톨로지인 워드넷과 매칭하여 계층으로 구성하고, 계층을 구성하는 도메인들을 이용하여 효율적이고 의미적인 분류를 시도하였다. 제안하는 방법을 실험하기 위해 신문기사들을 이용하였으며, 실험을 통하여 의미적이고 정확한 결과를 확인하였다.

본 논문은 2장에서 본 연구에 필요한 요소들을 상세히 설명하고, 3장에서 본 논문의 핵심인 문서의 계층화 방법과 계층 비교 방법을 기술한다. 4장에서 신문기사를 이용한 실험을 통해 본 연구에서 제안한 방법을 평가하고, 5장에서 결론 및 향후 연구방향을 제시한다.

Ⅱ. 관련연구

본 장에서는 기존의 문서 자동 분류 방법들 중 베이지안 자동 문서 분류 방법, 퍼지 집합 이론을 이용한 분류 방법을 소개하고, 문서의 의미적인 분석과 유사도 비교에 필요한 지식베이스인 워드넷에 대해 상세히 설명한다.

2.1. 기존의 문서 분류 방법

2.1.1 베이지안 자동 문서 분류 방법

문서의 자동 분류에 대한 기존 연구방법들의 잡음으로 인한 오분류, 단어 의미 중의성 문제점들을 보완하기 위해 Apriori-Genetic 알고리즘을 이용한 베이지안 분류 방법이 제안되었다. Apriori 알고리즘은 단어간의 의미를 반영하여 연관단어 형태로 추출하고 잡음으로 인한 오류를 줄이기 위해 추출된 연관단어를 이용하여 연관 단어 지식베이스를 구축한 후, 구축된 지식베이스를 유전자 알고리즘을 이용하여 최적화한다. 그리고 베이지안 확률을 이용하여 최적화된 연관 단어 지식베이스를 기반으로 새로운 입력 문서를 클래스별로 분류한다. 그림 1은 Apriori-Genetic 알고리즘의 전체 수행 과정을 나타내고 있다.

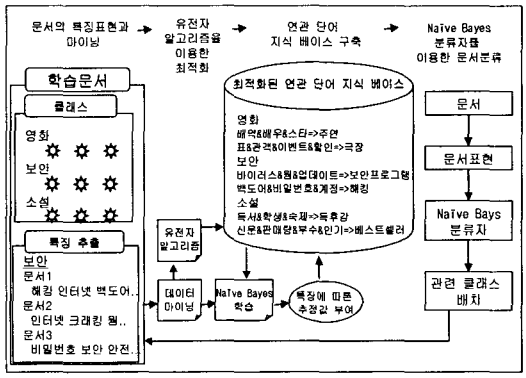

그림 1. Apriori-Genetic 알고리즘 적용 분류
Fig. 1. Clustering Using Apriori-Genetic Algorithm

본 방법은 최적화된 연관단어 지식베이스 구축으로 Naive Bayes 분류자가 빠르고 정확한 문서 분류가 가능하며, 실험문서의 특징을 연관 단어의 형태로 표현함으로써 단어 의미의 중의성 문제를 해결할 수 있다는 장점이 있다.

2.1.2 퍼지 집합 이론을 이용한 문서 분류

웹 문서를 자동으로 분류하기 위한 방법으로 퍼지(Fuzzy)를 이용한 방법이 제안되었다.[11] 본 방법은 다른 주제를 기술하기 위해 동일한 단어들을 사용하여 발생하는 모호성(ambiguity)으로 인해 기존 연구의 한계를 극복하기 위한 것으로, 웹 문서들에 포함된 키워드들과 미리 정의한 인덱스 용어들의 관계를 이용하여 퍼지 집합 이론을 적용하였다.

퍼지 집합 이론은 경계를 확실히 정의할 수 없는 클래스들의 포함 정도를 표현하기 위해 1965년 미국 버클리대학의 자데(L.A. Zadeh)교수에 의해 도입되었으며[14], 현재는 인공지능 분야에서 핵심 기능으로 응용되고 있다. 퍼지 집합 이론에서의 핵심은 멤버함수(Membership Function)인데 결과 값으로 [0,1]사이의 값을 도출한다. 결과 값은 특정 집합에 소속된 정도를 나타내며, 0은 그 정도가 0%, 1은 100% 일치함을 나타내고 있다.

퍼지 집합 이론을 적용한 분류 방법[11]에서는 분류 카테고리에 포함될 개념들의 모델($K=\{K_1, K_2, \ldots, K_m\}$)과 분류 카테고리에 포함되지 않을 개념들의 모델($CK=\{CK_1, CK_2, \ldots, CK_n\}$)을 구축하고, 각 카테고리에 포함된 개념들과 웹 문서의 개념들의 집합을 자체적으로 정의한 다음과 같은 멤버함수를 이용하여 분류하고 있다.

$$\mu_{i,j} = \max_{\forall k_a \in d_i} [1 - \prod_{\forall k_b \in CK_j} (1 - r_{a,b})]$$

위 멤버함수에서, $\mu_{i,j}$는 문서 d_i가 카테고리 C_j에 포함될 정도를 나타내고, $r_{a,b}$는 카테고리의 개념 k_a가 문서 d_i에 포함된 정도와 k_b가 C_j에 포함되지 않을 개념 모델 CK_j에 포함될 정도를 말한다.

위에서 설명한 자동 문서분류 방법들은 문서 내의 키워드들을 이용하여 효율적이고 합리적인 분류 결과를 도출할 수 있지만 아직 문서 내에 포함된 개념들의 정확한 의미를 파악하지 못하는 한계점이 있다.

2.2. 워드넷(WordNet)

워드넷은 현재까지 가장 널리 사용되는 범용의 대형 온톨로지로써 실제 그 내용은 6개의 데이터베이스 테이블들로 구성되어져 실세계에 존재하는 어휘에 대해서 체계적으로 정의하고 있다. 워드넷에서 중요한 내용은 바로 개념간의 관계를 정의하고 있는 부분이며, 사전과 가장 큰 차이점이 또한 바로 이러한 부분이다.[9]

그림 2는 워드넷 내의 6개의 테이블의 논리적 관계를 잘 나타내고 있다.

워드넷은 크게 4개의 카테고리(명사, 형용사, 부사, 동사)로 분류되고, 그 안에는 다시 45개의 소카테고리로 분류되어져 있다. 그리고 워드넷 내의 모든 개념들은 특정의 심볼들을 사용하여 각 개념들간의 관계를 표현하고 있다. 표 1은 워드넷에서 사용되는 개념들 간의 관계를 정의해 놓은 심볼들과 그 의미를 설명하고 있다.

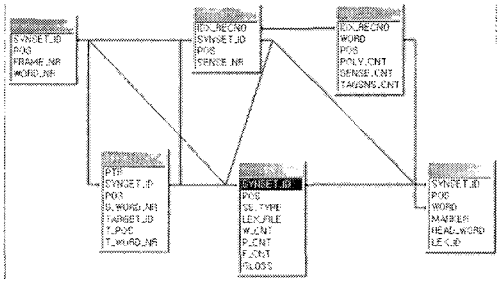

그림 2. WordNet 데이터베이스 관계도
Fig. 2 WordNet Database Relationship

표 1. 심볼과 의미
Table. 1 Symbols and Meanings

심볼	의미
!	반의어
@	상위 개념
~	하위 개념
동일 synset ID	동일 의미(유의어)

이러한 워드넷은 영어 단어 개념들 간의 의미와 관계를 상세하게 정의해 놓은 범용의 대형 온톨로지로서 미국의 프린스턴 대학(Princeton University)에서 10년 이상 개발하고 있다. 또한 자바 워드넷 라이브러리(Java Wordnet Library)가 제공되어 이를 응용한 많은 연구가 진행되고 있다.

III. 문서의 계층화 방법 및 문서 분류 방법

사용자가 원하는 문서와 비슷한 문서를 의미적으로 검색하기 위해 본 연구에서는 각각의 문서를 계층화한 후, 생성된 계층을 서로 비교하는 방법을 제안한다. 본 연구의 전체 구성은 그림 3과 같다.

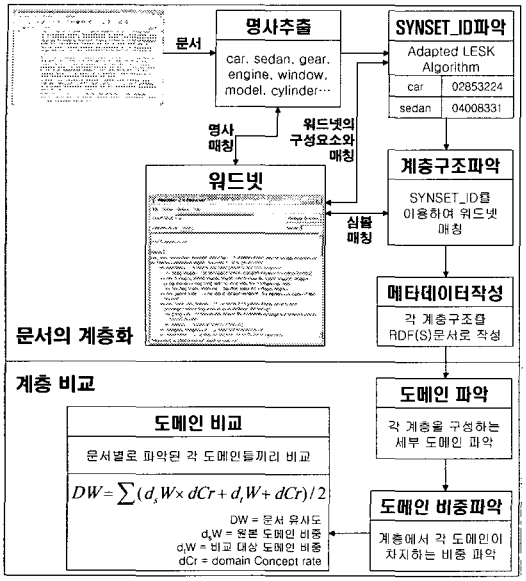

그림 3. 전체 구성
Fig. 3. General Architecture

3.1 문서의 계층화 방법

문서를 의미적으로 분류하기 위해 우선 문서가 포함하는 내용을 계층화 한다. 문서를 계층화하기 위해 문서에서 명사를 추출하고, 각 명사의 정확한 의미를 파악한 후, 파악된 의미를 이용하여 계층구조를 형성하는 단계가 요구된다.

3.1.1 문서 내에서 명사 추출

어떤 특정한 주제를 위해 기술된 문서는 단어들 사이에 의미적 관계가 존재한다. 이러한 문서는 명사, 동사, 형용사 등의 여러 품사들의 단어들로 구성이 될 수 있는데, 특히 명사들은 문서를 구성하는 핵심 기술자라 할 수 있다. 이에 문서를 계층화하기 위해 첫 번째로 문서 내에 포함된 명사들을 추출하는 단계를 수행한다.

앞에서 설명한 워드넷에는 명사, 형용사, 동사, 부사가 기술되어 있다. 이에 본 연구에서는 특정한 문서를 입력받은 후, 문서 내의 각 단어를 워드넷의 명사 데이터베이스와 매칭을 시켜 일치하는 명사의 SYNSET_ID(의미)들을 얻는다. 하나의 명사는 여러개의 의미를 가질 수 있는데, 정확한 의미는 다음 과정에서 파악하고 현 과정에서는 명사가 갖고 있는 모든 의미를 찾는다. 본 내용은 그림 4와 같다.

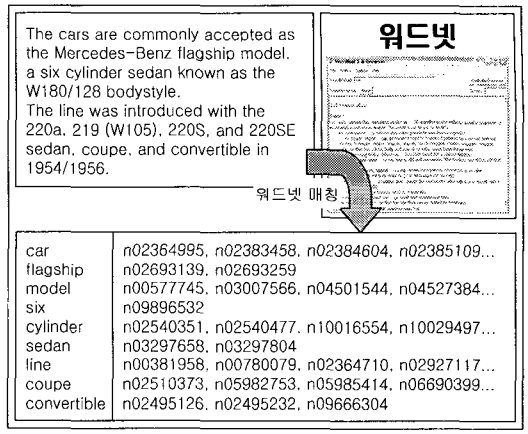

그림 4. 명사 추출
Fig. 4. Noun Extraction

3.1.2 명사의 의미파악

여러 가지 의미를 갖고 있는 단어의 정확한 의미를 파악하는 것은 쉽지 않다. 이를 위해 WSD(Word Sense Disambiguation)라는 한 연구 분야가 생성되어 연구가 진행되고 있다. 특히, 워드넷에 기반한 것으로는 Adapted LESK Algorithm이 있다.[7,10] Adapted LESK Algorithm은 워드넷 내에 기술되어 있는 모든 관계와 개념들의 정의를 최대한 활용하여 단어의 의미를 파악하는 것으로써, 현재까지 가장 신뢰성 있는 연구라 할 수 있다. Adapted LESK Algorithm은 워드넷에서 정의하고 있는 개념의 정의와 상위, 하위, 포함 등의 관계에 포함된 단어들을 매치하고, 의미 내의 단어들의 배열이 일치하면 더욱 가까운 의미일 수 있다는 가정 아래 제안되었다. 표 2는 Adapted LESK Algorithm을 사용하여 'coupe'와 'sedan'을 비교한 간단한 예이다. 'coupe'는 워드넷에서 하나의 의미, 'sedan'은 두 가지의 의미로 정의가 되고 있다.

표 2. Adapted LESK Algorithm 예제
Table. 2 Example of Adapted LESK Algorithm

coupe(1) : (**a car** with **two doors and front seats and** a luggage compartment)
sedan(1) : (**a car** that is closed **and** that has **front and** rear **seats** and **two** or four **doors**)
sedan(2) : **a** closed litter for one passenger
1*(a car)+1*(two)+1*(doors)+2*(and)+1*(front)+1*(seats) =$1*2^2+1*1^2+1*1^2+2*1^2+1*1^2+1*1^2$ = 4+1+1+2+1+1 = 10

표 2와 같이 'coupe'와 'sedan'의 첫 번째 개념과 유사함을 Apdated LESK Algorithm을 통하여 얻을 수 있다. 이처럼 Adapted LESK Algorithm은 단어의 정의를 이용하여 연속하는 단어가 있을 경우에 비중을 더욱 많이 주기 위해, 연속하는 단어에 제곱을 수행함으로써 유사도를 비교한다.

특정한 주제를 설명하는 문서는 관련된 단어가 많이 나올 수 있다. 문서 내에서 단어들의 관련성을 이용하여 정확한 의미를 파악하기 위해, Adapted LESK Algorithm을 문서 내의 명사들의 정확한 SYNSET_ID를 얻는데 적용하였다.

3.1.3 SYNSET_ID 기반 문서 계층구조 파악

워드넷에는 단어들의 관계를 정의하기 위해 각종 심볼(Symbol)을 이용하고 있다. 이러한 심볼들은 표 3과 같이 SYNSET_ID, TARGET_ID를 이용하여 단어들 사이의 관계를 정의하고 있다. 표 3은 워드넷 데이터베이스 내에 작성된 Pointers 테이블에 작성된 내용의 일부를 보여주고 있다.

표 3. 워드넷 데이터베이스의 일부
Table. 3 Part of WordNet Database

PTR	SYNSET_ID			TARGET_ID		
~	n02392911	n	0	n02311742	n	0
~	n02392911	n	0	n02946569	n	0
~	n02392911	n	0	n02946676	n	0
~	n02392911	n	0	n03346295	n	0
@	n02393107	n	0	n03569523	n	0
@	n02393264	n	0	n03055972	n	0
@	n02393349	n	0	n02236345	n	0
...	...	...	...	...	...	...

앞의 과정에서 얻어진 모든 SYNSET_ID들의 의미적인 계층구조를 파악하기 위하여, 워드넷 데이터베이스에 접근한 후, 워드넷 내에 정의된 SYNSET_ID들의 관계와 매칭을 이용하여 그림 5와 같이 SYNSET_ID들의 계층구조를 생성할 수 있다.

3.1.4 계층 구조를 RDF(S) 문서로 작성

작성된 계층 구조는 추출된 SYNSET_ID의 보존과 문서비교의 용이성을 위해 W3C에 의해 2004년 Semantic Web 표준으로 작성된 RDF(Resource Description Framework) 문서로 작성한다. 이렇게 작성된 RDF(S) 문서는 웹 문서의 메타데이터 역할을 하고, 실제 사용자가 선택한 문서와 유사한 문서를 검색하고자 할 때 유사도 측정을 위한 데이터로 사용된다. 표 4는 위의 과정에서 추출된 명사들의 SYNSET_ID들을 RDF(S) 문서로 작성한 것이다.

표 4. SYNSET_ID들의 RDF(S) 문서
Table. 4 Document of using RDF(S)

```
<?xml version="1.0"?>
<rdf:RDF
 xmlns:rdf=
   "http://www.w3.org/1999/02/22-rdf-syntax-ns#"
 xmlns:rdfs=
   "http://www.w3.org/2000/01/rdf-schema#"
 xml:base="http://vector.chosun.ac.kr/doc/vehicles">
 <rdfs:Class rdf:ID="n04348422"/>
 <rdfs:Class rdf:ID="n02853224">
  <rdfs:subClassOf rdf:resource="#n04348422"/>
 </rdfs:Class>
 <rdfs:Class rdf:ID="n04008331">
  <rdfs:subClassOf rdf:resource="#n02853224"/>
 </rdfs:Class>
 <rdfs:Class rdf:ID="n03006338">
  <rdfs:subClassOf rdf:resource="#n02853224"/>
 </rdfs:Class>
 ...
```

n04348422 : vehicle	n02853224 : car
n04008331 : sedan	n03006338 : coupe

3.2 계층 비교 방법

문서의 의미적인 유사도 측정을 위해, Ⅲ. I 의 과정에서 작성한 계층구조를 비교하기 위한 알고리즘을 작성하였다. 구축된 각 문서의 계층구조는 그림 5와 같이 몇 개의 도메인으로 구성된다.

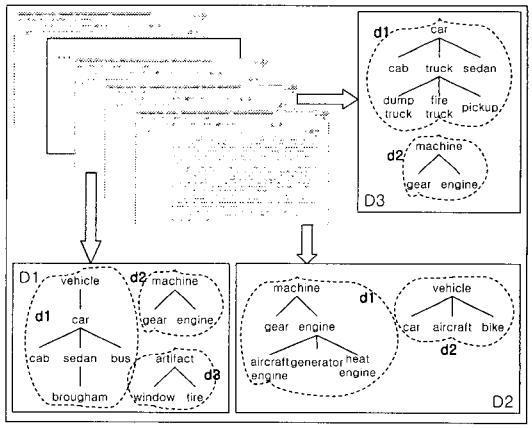

그림 5. 계층 표현
Fig. 5. Hierarchy of Documents Contents

특정한 주제를 가지고 기술된 문서는 주제를 포함하는 도메인에 개념들이 집중되어 있고, 주변의 도메인들은 핵심이 되는 도메인의 설명을 돕는 속성요소로써의 역할을 하고 있다. 이에 본 계층 비교 방법에서는 문서에서 기술하고 있는 각 도메인별 비중을 계산하고, 각 도메인에서 측정한 개념들을 Jaccard-Similarity를 이용해서 일치도를 측정한 후, 문서들 사이의 유사도를 계산한다. 그림 5에서 몇 개의 도메인들이 각 문서에 포함되는데, 각 도메인이 문서에서 차지하는 비중을 구하기 위해 식 (1)을 적용하였다. 식 (1)의 도메인 비중(Domain-Weight)은 전체 계층에 포함된 개념들 중 해당 도메인에 포함된 개념들의 수를 구하는 것이다. 식 (1)을 이용하여 각 문서에 포함된 도메인들의 비중을 구한 후, Jaccard-Similarity를 이용하여 각 문서의 동일 도메인에 포함된 개념들의 일치도를 측정한다. 식 (2)는 Jaccard-Similarity 측정 수식이다. Jaccard 수식의 c_1, c_2는 각 도메인에 포함된 개념 집합을 의미하며, 수식의 결과는 최소 0과 최대 1 사이의 값을 갖으며, 0은 두 도메인이 전혀 연관이 없음을 의미하고, 1은 두 도메인이 일치함을 나타낸다.[8]

위의 두 수식을 적용하여 구해진 값을 이용하여, 문서의 유사도를 측정하는 식은 동일 개념을 기술하는 도메인의 비중과 일치하는 개념 비율의 곱으로 구성되어져 있다. 식 (3)은 문서들 사이의 유사도를 측정하는 수식이다.

식 (3)에서 $D_s W$는 원본 문서의 도메인 비중을 나타내고, $D_t W$는 비교 대상 문서의 도메인 비중이며, dCr은 Jaccard-Similarity로 측정한 도메인 내의 개념 일치도를 의미한다. 식 (3)으로 얻어진 값에서 1은 두 문서가 일치함을 나타내고, 0은 전혀 다른 문서임을 나타낸다.

표 5는 위의 수식들을 이용하여 그림 5에 표현된 각 문서의 계층들 중 D1을 중심으로 유사도를 측정한 결과이다. 표 5에서 문서 D1, D2, D3의 각 도메인의 비중을 식 (1)을 이용하여 구하고, Jaccard-Similarity를 적용하여 개념 일치도를 측정한다. 문서 D1을 중심으로 D2와 D3의 유사한 정도를 파악한 결과 D1과 D3은 0.475의 유사성이 측정되어 0.325의 유사성 값이 측정된 D2보다 좀 더 유사한 문서임을 확인할 수 있다.

$$Domain - Weight = \frac{Concepts\,in\,Domain}{Concepts\,in\,Document} \tag{1}$$

$$Jaccard - similarity(c_1, c_2) = \frac{P(c_1 \cap c_2)}{P(c_1 \cup c_2)} \tag{2}$$

$$Document\ Weight = \sum (D_s W \times dCr + D_t W \times dCr)/2 \tag{3}$$

표 5. 그림 5의 유사도 구하는 예
Table. 5 Example of Documents Similarity on the Figure 5

문서	D1			D2		D3	
도메인	d1	d2	d3	d1	d2	d1	d2
도메인 비중	0.5	0.25	0.25	0.6	0.4	0.7	0.3
Jaccard-Similarity							
D1과 D2			D1과 D3				
(D1-d1):(D2-d2)	(D1-d2):(D2-d1)		(D1-d1):(D3-d1)		(D1-d2:D3-d2)		
2/8=0.25	3/6=0.5		3/10=0.3		3/3=1		
문서 유사도							
D1과 D2			D1과 D3				
(0.5*0.25+0.4*0.25+0.25*0.5+0.6*0.5)/2=0.325			(0.5*0.3+0.7*0.3+0.25*1+0.3*1)/2=0.475				

Ⅳ. 실험 및 평가

본 논문에서 제안하는 문서 비교 방법을 평가하기 위해, 국외 뉴스 전문 사이트들(reuters.com, cnn.com, abcnews.com)에서 제공하는 신문 기사들에서 각각 100개씩 총 300개의 뉴스기사들을 모았다. 선정기준은 각 사이트에 모두 기술되고 있는 기사들이며, 선택된 기사들에서 적어도 3건은 같은 주제에 대해 기술되어 있다. 순수 기사 외에 실험의 정확성을 저해시키는 요소들을 제거하기 위해, 데이터 셋 300개의 타이틀과 본문을 순수 txt 파일로 저장하여 실험을 하였다.

실험방법은 데이터 셋 300개에 대해 본 연구에서 제안하는 계층화 방법을 통해 RDF(S) 문서로 생성한 후, reuters.com의 각 기사들을 기준으로 하여 본 논문에서 제안하는 방법을 통해 299개의 기사를 비교하여 유사도 측정하였다.

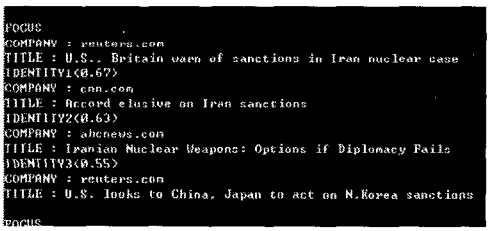

그림 6. 실험
Fig. 6 Experiment

표 6. 실험 결과
Table. 6 Experiment Result

결과확인범위	동일 기사 검색률(%)
5	75.7
7	81.0
10	91.4

그림 6과 표 6은 실험과정과 실험에 대한 결과를 보이고 있다. 그림 6에서 reuters.com의 특정 기사를 기준으로 검색한 결과 cnn.com, abcnews.com, reuters.com의 기사가 각각 0.67, 0.63, 0.55의 유사도로 순위대로 검색됨을 확인할 수 있다. 본 방법을 통해 reuters.com에서 발췌한 100개의 기사와 유사한 기사 검색의 정확도는 순위의 범위가 5일 때, 표 6과 같이 75.7%가 측정되었으며, 도출되는 순위의 범위를 7로 확대했을 때의 정확도는 81.0%, 10으로 확대했을 때 91.4%의 정확도를 얻을 수 있었다. 또한 이 결과는 정확히 일치하다고 판단되는 각 뉴스 전문 사이트들에서 제공되는 뉴스를 포함하는지 여부이며, 순위의 범위에 포함된 다른 기사들 또한 의미적으로 유사한 문서가 대부분임을 확인할 수 있었다. 실험 결과를 볼 때, 본 논문에서 제안하는 문서의 계층화를 통한 문서 비교 방법은 정확하며 의미적임을 확인할 수 있었다.

Ⅴ. 결 론

본 논문은 사용자가 원하는 문서와 유사한 문서를 찾아주기 위해 문서들을 의미적으로 비교하여 유사도를 구하기 위한 연구이다. 문서들의 의미적인 비교를 위해 각 문서들을 워드넷과 매칭하여 명사를 추출, Adapted-LESK 알고리즘을 적용한 SYNSET_ID를 파악, 파악된 SYNSET_ID를 통해 계층으로 표현하였다. 이러한 과정으로 추출된 각 문서들의 계층들은 도메인 비중 구하는 식 (1), 동일 도메인 내의 개념 일치도를 구하는 Jaccard-Similarity 수식 (2), 그리고 도메인 비중과 개념 일치도를 이용하여 문서의 유사도를 구하는 식 (3)을 통해 최종적으로 문서들 사이의 유사도를 측정할 수 있었다. 본 연구는 문서의 특징을 정확하게 반영할 수 없고 의미적이지 않은 단점이 있는 기존연구에 비해, 문서의 내용을 의미적인 계층으로 표현하고 중요 도메인에 가중치를 부여함으로써 좀더 의미적인 문서검색을 할 수 있다.

본 연구에서 계층으로 구성할 때 단어들의 빈도와 중요성은 고려되지 않았다. 이는 앞으로 진행될 연구이며, 단어의 빈도와 중요성까지 고려하게 될 때, 정확도는 더욱 향상될 것으로 보이며, 본 연구를 이용하여 특정 문서 그룹을 모델로 두어 의미적인 문서 분류(Clustering)의 핵심으로 수행될 수 있을 것을 기대한다.

참고문헌

[1] D.D.Lewis, "Naive(Bayes) at forty : The Independence Assumption in Information Retrieval," In European Conference on Machine Learning, 1998

[2] J. McMahon and F. Smith, "Improving statistical language model performance with automatically

generated word hierarchies," Computational Linguistics, Vol.22, No.2, 1995.

[3] T.Joachims, "A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization," ICML-97, 1997.

[4] 한광록, 선복근, 한상태, 임기욱, "인터넷 문서 자동 분류 시스템 개발에 관한 연구", 제9회 한국정보처리 학회 논문집, 제7권 제9호, pp.2867-2875, 2000

[5] A. McCallum and K. Nigram, "A Comparsion of Event Models for Naive Bayes Text Classification," AAAI-98 Workshop on Learning for Text Categorization, 1998

[6] 고수정, 이정현, "Apriori-Genetic 알고리즘을 이용한 베이지안 자동 문서 분류", 정보처리학회 논문지 B, Vol.01, No.01, p.001~012, 2001년 6월

[7] Satanjeev Banerjee, Ted Pedersen, "An Adapted Lesk Algorithm for Word Sense Disambiguation Using WordNet", Computational Linguistics and Intelligent Text Processing: Third International Conference, p.136-147, Vol.2276, February 17-23, 2002.

[8] Hyunjang Kong, M.G. Hwang, P.K. Kim, "A New Methodology for Merging the Heterogeneous Domain Ontologies based on the WordNet", International Conference on Next Generation Web Services Practices, 2005. 08.

[9] http://wordnet.princeton.edu/

[10] S. Banerjee, T. Pedersen, "An adapted Lesk algorithm for word sense disambiguation using WordNet," In Proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics, Mexico City, pp. 136 - 145, 2002

[11] Haruechaiyasak, C. Mei-Ling, Shyu Shu-Ching Chen, "Web Document Classification Based on Fuzzy Association", Computer Software and Applications Conference, 2002. COMPSAC 2002. Proceedings. 26th Annual International, p.487- 492

[12] "The Classic Vector Space Model", http://www.miislita.com/term-vector/term-vector-3.html

[13] D.L. Lee, H. Chuang, K. Seamons., "Document Ranking and the Vector-Space Model", IEEE Software, p.67-75, 1997.

[14] L.A. Zadeh, "Fuzzy Sets", in D.Dubois, H.Prade, and R.R.Yager, editors, Readings in Fuzzy Sets for Intelligent Systems, Morgan Kaufmann Publishers, 1993.

저자소개

황 명 권(Myung-Gwon, Hwang)

2004 조선대학교 컴퓨터공학부 (공학사)
2006 조선대학교 대학원 전자계산학과 (이학석사)
2006.3-현재 조선대학교 대학원 컴퓨터공학부 박사과정
※관심분야 : 문서분류, 시맨틱웹, 멀티미디어검색

배 용 근(Yong-Geun, Bae)

1984 조선대학교 컴퓨터공학과 (공학사)
1987 조선대학교 대학원 전자공학과 (공학석사)
2001 원광대학교 대학원 전자공학과 (공학박사)
※관심분야 : 프로그래밍 언어, 마이크로 프로세서

김 판 구(Pan-Koo, Kim)

1988 조선대학교 컴퓨터공학과 (공학사)
1990 서울대학교 대학원 컴퓨터공학과 (공학석사)
1994 서울대학교 대학원 컴퓨터공학과 (공학박사)
※관심분야 : 시맨틱웹, 온톨로지, 멀티미디어 정보검색, 감성정보처리