TTF와 ITTF의 차를 이용한 자동 문서 분류

이상철[0], 하진영
강원대학교 컴퓨터•정보통신공학과
bo99007@mirae.kangwon.ac.kr, jyha@kangwon.ac.kr

Automatic Text Categorization using difference TTF and ITTF

Sang-Choul Lee[0], Jin-Young Ha
Dept. of Computer Engineering, Kangwon National University

요 약

본 논문에서는 일반적으로 Word Based Matching 방법에서 많이 쓰이는 TFIDF 방법대신에 TTF(Total Term Frequency)와 ITTF(Inverse Total Term Frequecy) 에 가중치를 주어 문서분류의 정확도를 높이는 방법을 제안하고자 한다. TFIDF방법에서 IDF는 역문헌빈도를 나타내는데 Term에 대한 빈도비율의 공정성이 떨어져 문서 분류의 정확도에 한계가 있다. 본 논문에서 제시하는 문서 분류방법은 TTF와 ITTF에 각각의 가중치를 준 후에 차연산 이용하여 문서를 분류하는 것이다. 이러한 방법의 특징은 IDF를 사용할 때 보다 각 카테고리에 있는 term, 즉 단어의 중요도에 대한 가중치를 좀 더 공평하게 줌으로써 문서의 분류를 높일 수 있다. 본 논문에서는 조선일보의 카테고리를 사용하였으며 조선일보의 기사를 대상으로 문서 자동 분류 실험을 수행하였다. 실험 결과 TFIDF보다 본 논문에서 제안한 방법이 문서 분류에 높은 정확도를 나타냄을 보였다.

1. 서론

인터넷 보편화에 따른 온라인 문서의 양과 그 종류가 폭발적인 증가로 인해 일반 사용자들은 웹으로부터 다양한 정보를 접하게 되면서 문서의 자동 분류에 대한 필요성이 널리 인식되어지고 있다. 일반적으로 자동 문서 분류 방법에는 크게 문서 내에 나타나는 단어의 빈도를 이용하는 Word-Based-Matching 분류방법과 인간의 경험적인 지식을 바탕으로 문서를 분류하는 통계적인 분류방법과 자연어 처리를 통하여 문서 내에 있는 문장의 의미를 분석하는 의미 분석 방법이다.[1,2,4,7,8]

정확한 문서를 분류를 위해서는 자연어 처리를 통하여 문서 내용의 의미를 파악하고, 전체적인 문서의 내용의 흐름을 파악하여야 한다. 그러나 자연어 자체의 중이적인 뜻이나 모호성 때문에 문장의 의미 분석은 매우 어려우며, 이러한 문제점 때문에 일련의 복잡한 과정을 거치고도 Word- Based-Matching이나 통계적인 분류방법보다 월등한 성능을 가지지 못한 경우가 많다. Word-Based-Matching 분류에서는 문장의 내용에 대한 분석 없이 단어의 출현 빈도만을 이용하여 문서를 분류함으로써 분류 방법이 간단하고 수행속도가 빠르지만 분류의 정확도에 한계가 있다.[3,6]

본 논문에서는 Word-Based-Matching방법에서 가장 많이 사용되는 TFIDF에 의한 분류방법 대신에 TTF와 ITTF의 차를 이용한 방법은(이후부터는 TTF-ITTF라 정의하겠다.) 각각의 가중치를 주어 연산한 분류방법으로 문서 분류의 정확도를 높이는 방법을 제안한다. TTF-ITTF 방법은 term에 대한 가중치를 TTF와 ITTF에 각각 경험적으로 준 후에 차연산을 통하여 문서분류를 한다. 그리고 후처리 방법을 적용하여 문서분류의 정확도를 높였다.

2. 관련 연구

Word-Based-Matching 방법은 수작업에 의해 분류된 실험 집단(training set) 문서에서의 단어 출현 빈도를 이용하여 새로운 문서를 가장 가능성이 높은 카테고리를 찾아 분류하는 방법이다. 일반적으로 Word-Based-Matching$_i$방법이 색인어를 추출하는 단계는 핵심어로서 활용될 수 없는 불용어를 제거하고, 어휘의 기본형으로 변환하는 스테밍 처리를 거쳐 TFIDF 방법을 이용해 각 문서에 해당하는 대표 단어를 추출하는 단계를 거친다.

불용어와 스테밍 처리를 거친 후에 추출된 문서 내의 단어를 가운데에서 문서를 대표할 만한 중요 핵심어를 추출하는 방법에는 문서간의 단어 출현 빈도의 상관관계를 구하는 TFIDF 방법을 사용한다. TFIDF 란 문서 내의 해당 단어에 대한 출현 빈도(Term Frequency)와 출현하는 문서의 개수(Document Frequency)이용하는 것으로, 문서에서의 특정 단어 중요도는 해당 문서 내의 출현 빈도와 비례하고 총 출현 문서의 개수와는 반비례하는 특성을 활용하여 중요 핵심어를 추출한다.
특정 카테고리에서 TFIDF의 weight를 구하는 식은 다음과 같이 계산된다. [3,5,6,7]

$$tf_{ij} = \frac{freq_{ij}}{\max_k freq_{kj}}$$ ($freq_{ij}$: 문서 d_j 안에서 term k_i의 빈도수)

$$idf_i = \log\frac{N}{n_i}$$

(N : 문서의 전체 개수 ,n_i : term k_i가 있는 문서의 개수)

$$W_{ij} = tf_{ij} \times idf_i$$

본 논문에서는 문서와 카테고리의 벡터 표현에서 각 색인어의 가중치로 일반적으로 많이 사용되는 역문헌빈도 IDF 대신에 공통적으로 많이 나오는 단어가 모든 카테고리에서 나오지 않을 경우 단어(Term)에게 가중치를 주는 총역단어 빈도(ITTF)를 사용하였다.

2.1 문서의 정규화와 Word Cut Ratio

특징 추출의 대상이 되는 문서들은 각기 크기가 다르고 핵심어 분포가 상이하기 때문에 정규화 과정이 필요하다. 여기서 문서의 정규화란 핵심어 출현 빈도 수에서 문서 길이에 대한 의존성을 제거하는 것으로, 각 문서별 최다 출현 핵심어를 이용하여 모든 핵심어에 대한 가중치를 정규화 하는 것이다. 문서의 정규화에 사용되는 식은 다음과 같다.

$$tf_i = \alpha \cdot \frac{freq_{ij}}{\max_k freq_{kj}} + (1-\alpha)$$

Word Cut Ratio 는 색인어 파일에 색인어를 저장할 때 색인어의 개수를 결정하는 것으로 Cut Ratio 값이 커짐에 따라서 색인어의 수는 감소하게 된다. Cut Ratio 값이 커지면 색인어 파일의 크기가 감소하여 색인어 파일을 메모리에 적재할 때 메모리를 줄일 수 있다. 그러나 Cut Ratio 값이 너무 크면 핵심어까지 제거되므로 Cut Ratio의 값은 경험에 의해서 적당한 값으로 조정해 주어여 한다.

3. TTF-ITTF 이용한 문서 분류

3.1 TTF-ITTF에서 차연산과 가중치 계산

문서에서 얻어진 Term으로 TTF 값을 계산한 후에 ITTF 값을 계산하여 그 결과값을 색인어 파일에 저장한다.
TTF-ITTF의 값은 다음과 같이 계산되어 진다.

$$ttf_i = \frac{Cfreq_i}{TNum}$$

$$ittf_i = Cfreq_i - Efreq_i,\ W_i = ttf_i - \alpha \cdot ittf_i$$

$Cfreq_j$= 모든 카테고리안에서의 Term k_i의 빈도수
$TNum$ = 모든 카테고리의 Term의 개수(총 단어수)
$ittf_i$ = 자신의 카테고리를 제외한 모든 카테고리에서 Term k_i의 빈도수

3.3 색인어 파일 구조

문서를 분류하기 위해서는 단어의 가중치 값이 계산된 색인어 가중치 파일을 만들어야 한다. 기본적으로 문서에서 불용어를 제거하고 스테밍한 단어가 색인어가 된다. 그 색인어를 training 프로그램에 의해 계산하여 하나의 색인어 파일로 만든다. 색인어 파일의 구조는 다음과 같다.

표 1. 색인어 파일의 구조

카테고리번호	Term	TTF-ITTF
0	여야	5989
0	국회	3242
0	야당	1121
18	투수	7865
18	홈런	2355
18	타자	1321

표 1를 에서 보면 카테고리번호 0은 정치-국회,여야의 카테고리를 나타내고 카테고리번호 18은 스포츠-야구의 카테고리를 나타낸다. Term은 문서에서 나온 색인어를 나타내며 TTF-ITTF는 모든 카테고리에서 계산해 얻은 가중치 값이다. 색인어 파일은 각각의 카테고리 별로 TTF-ITTF으로 내림차순으로 정렬되어 있다.

3.4 후처리

입력 문서를 처리하면 단어의 후보자가 생기게 된다. 각 문서에서 1순위 후보자가 그 문서의 범주를 결정짓는 결정적인 역할을 한다. 이 1순위 후보자는 대부분 원하는 범주로 문서를 분류 시켜주지만 1순위 후보자가 모든 것을 적절히 문서를 분류 시킬 수 없다. 따라서 결과로 나온 후보자를 가지고 후처리(Post processing)을 해주어야만 한다. 후처리를 해줌으로써 문서 분류의 정확도를 높일 수 있다.

후처리에서 간단하게 사용되는 방법이 두 가지 있는데 첫번째 방법은 후보자 단어가 선택한 카테고리의 개수를 각각 세어서 가장 많이 나온 카테고리를 입력문서의 범주로 결정하는 방법이다. 두 번째 방법은 같은 카테고리를 선택한 후보자 단어의 weight값을 각각 더해서 가장 weight 값이 큰 것을 선택하여 그 단어의 카테고리를 입력문서의 범주로 결정하는 방법이다.

본 논문에서는 첫번째 방법과 두 번째 방법 그리고 그 방법을 조합한 방법을 후처리에 사용하여 문서를 분류하였다.

4. 실험 및 결과

4.1 실험 환경 및 실험 방법

본 논문에서는 실험을 위한 문서 집단으로 조선일보 웹사이트로부터 수집한 1350개의 기사를 이용했으며, 이들 문서 중 1026개(76%)를 실험 집단(training set)으로, 나머지 324개(24%)를 검증 집단(testing set)으로 하였다. 전체 기사는 총 27개의 카테고리를 분류된다. 또한 문장의 형태소를 분석기로 문장을 분석했다. 본 논문에서는 형태소 분석을 위하여 HAM[5] 을 사용하였다. 첫번째 실험은 Word Cut Ratio의 값에 따라 문서

분류 정확도를 구하고 두 번째 실험은 후처리 방법에 따라 문서분류 정확도를 구하였다. 본 실험에서 정확도를 *P*라고 했을 때 정확도를 구하는 식은 다음과 같다.

$$P = \frac{\text{바르게분류된문서수}}{\text{분류된총 문서수}}$$

4.2 실험 결과

표 2. Word Cut Ratio에 의한 분류실험 결과

분류방법	실험집단		검증집단	
	WCR	정확도	WCR	정확도
TTF-ITTF	0	61.11%	0	97.95%
	1	48.46%	1	93.66%
	2	40.43%	2	81.19%
TFIDF	0	55.25%	0	60.23%
	1	50.62%	1	64.33%
	2	48.46%	2	66.67%

조선일보를 대상으로 한 분류 실험 결과를 표 2, 표 3에 나타냈다. 표 2 는 문서를 정규화 한 후에 WCR(Word Cut Ratio)의 값을 변화시키면서 TTF-ITTF와 TFIDF에 적용하여 정확도를 나타냈다. 표 2의 결과를 보면 TTF-ITTF에서 WCR이 0(개)일때 실험집단과 검증집단에서 가장 정확도가 높게 나타났고, WCR의 값이 커질수록 실험집단의 정확도와 검증집단의 정확도가 낮아지는 것을 볼 수 있다. TFIDF에서는 WCR이 0일때 실험집단에서 정확도가 가장 높게 나왔다. 또한 TTF-ITTF와 달리 WCR의 값이 커짐에 따라 검증집단의 정확도는 증가하는 것을 볼 수 있다. 표 2에서 보면 실험집단에서 TTF-ITTF가 WCR이 0일때만 TFIDF보다 정확도가 5~6% 가 높게 나타났다. 검증집단에서는 TTF -ITTF가 TFIDF보다 14~36%가 정확도가 높게 나타났다.

표 3. 후처리에 의한 분류실험 결과

분류방법	실험집단		검증집단	
	후처리	정확도	후처리	정확도
TTF-ITTF	하지않음	61.11%	하지않음	97.95%
	방법①	63.27%	방법①	99.71%
	방법②	62.35%	방법②	98.54%
	방법③	64.81%	방법③	98.74%
TFIDF	하지않음	55.25%	하지않음	60.23%
	방법①	59.88%	방법①	82.26%
	방법②	56.17%	방법②	73.88%

표 3 은 문서를 정규화 한 후에 후처리의 방법에 따라 TTF-ITTF와 TFIDF에 적용하여 정확도를 나타냈다. ①,② 방법은 3.4 장에서 제시한 후보자 단어 카테고리를 세는 방법과 후보자 단어의 weight 값을 더하는 방법이고 방법③은 두 방법을 조합한 방법이다. 위 실험에서 후보자의 수는 5 개로 제한하였다. 또한TFIDF에서는 ③의 방법을 측정할 수 없기 때문에 실험에서 제외시켰다. 표 3 의 결과를 보면 TTF-ITTF와 검증집단에서 모두 정확도가 가장 낮게 나타났다. 또한 TTF-ITTF에서는 ①,② 방법보다 ③의 방법을 적용했을 경우 좀 더 높은 정확도를 나타내었고 TFIDF에서는 ①의 방법을 적용했을 경우 가장 정확도가 높게 나타났다.

본 논문에서 제안된 TTF-ITTF방법을 사용했을 경우 TFIDF를 사용 했을 경우와 비교 했을 때 실험집단에서는 TTF-ITTF가 TFIDF보다 1.2~ 5% 가 높게 나타났다. 또한 검증집단에서는 TTF-ITTF가 16~ 37% 가 높게 나타났다.

5. 결론

본 논문에서는 Word-Based-Matching에서 많이 쓰이는 TFIDF 방법 대신에 TTF 와 ITTF 의 차연산 이용하여 문서의 분류의 정확도를 높이는 방법을 제안하였다. 실험 결과 TTF-ITTF에 의한 분류 방법이 일반적으로 사용되는 TFIDF 보다 Word Cut Ratio 이 0 일때 높은 정확도를 나타내었다. 또한 후처리를 했을 경우 두 가지 방법 모두 하지 않았을 때 보다 높은 정확도를 나타내었다. TTF-ITTF에서는 후보자 카테고리를 세는 방법과 후보자 weight를 더하는 방법을 조합했을 때 좀 더 높은 장확도가 나타났다..

향후과제로는 두 개 이상의 카테고리에서 나타나는 핵심어의 문자가 뜻이 같을 경우 어떤 카테고리로 분류할 것인가를 결정하는 방법, TTF-ITTF에 통계적인 방법을 접목시켜 분류하는 방법등에 대한 연구가 필요하다.

참고문헌

[1] Yiming Yang, John Wilbur, " Using Corpus Statistics to Remove Redundant Words In Text Categorization," *Journal Of The American Society For Information Science*, 1996.

[2] W. Frakes and R. Baeza -Yates, *Information Retrieval*, Prentice Hall, 1992

[3] R. Hoch, " Using IR technique for text classification in document analysis, " *SIGIR94*, 1994.

[4] 강승식, 이하규, " 한국어 형태소 분석기 HAM 의 형태소 분석 및 철자 검사 기능," *한글 및 한국어 정보처리 학술 발표논문집*, 1996.

[5] 한정기, 박민규, 김준태, " 구문 패턴과 키워드 집합을 이용한 지동 문서 분류의 성능 향상," pp.70-73, *HCI 98 학술대회*, 1998.

[6] 김형훈, 백혜정, 박영택," 기계학습 기반의 개인 웹 에이전트 구축," pp.65-69, *HCI 98 학술대회*, 1998.

[7] Baeza-Yates, Ribeiro-Neto, *최신정보검색론*, 홍릉과학출판사, 2001.

[8] 정영미, *정보검색론*, 구미무역 출판부, 1993.