가중치가 부여된 연관 규칙을 이용한 문서 분류

김홍남° 이기성 조근식
인하대학교 컴퓨터공학부
nami4596°@eslab.inha.ac.kr, funvirus@eslab.inha.ac.kr, gsjo@inha.ac.kr

Document Classification using Weighted Associative Classifier

Heung-Nam Kim° Kee-Sung Lee Geun-Sik Jo
School of Computer Science & Engineering, Inha University

요 약

인터넷의 급속한 성장과 더불어 많은 정보와 데이터들을 인터넷을 통하여 얻을 수 있게 되었으며 많은 단체들이 문서들을 웹을 통하여 이용 가능하게 만들고 있다. 이에 따라 다양한 정보와 데이터를 효과적으로 분류하고 검색하는 문서 분류 (Document Classification)에 대한 알고리즘이 다양한 분야에서 널리 연구되어 왔으며 본 논문에서 초점을 두고 있는 전자 도서관 (Digital Library) 분야에서도 활발히 연구되어지고 있다. 하지만 기존의 전자 도서관의 문서 분류 알고리즘들은 문서들의 각 단락의 비중을 고려하지 않은 채 단어들의 발생 빈도에 초점을 두어 많은 잡음 단어 (Noise Term)를 포함하고 그로 인하여 분류 성능이 떨어졌다. 본 논문에서는 문서 단락의 중요도에 따라 다른 가중치를 부여하여 단어 지지도 (Term Support)가 높은 단어들을 추출하고 그 단어들로 연관 규칙 (Association Rules)을 이용하여 분류 규칙을 생성하는 방법을 제안한다. 제안된 방법의 성능평가를 위해 문서 분류에 널리 쓰이는 나이브 베이지안 분류자 (Naïve Bayesian Classifier) 및 기존의 단순 연관 규칙 분류자 (Associative Classifier)와 비교 평가 하였다. 그 결과, 각 가중치가 부여된 연관 규칙 분류 방법이 나이브 베이지안 분류 방법과 단순 연관 규칙 분류 방법보다 높은 성능을 보였다.

1. 서 론

인터넷의 급속한 성장과 더불어 우리는 다양한 많은 정보와 데이터들을 인터넷을 통하여 얻을 수 있다. 이에 따라, 다양한 정보와 데이터를 효과적으로 분류하고 검색하는 것이 오늘날의 정보 사회의 중요한 이슈가 되었다. 문서 분류 (Document Classification)는 새로운 문서의 내용에 따라 미리 정해진 적당한 카테고리로 결정하는 과정으로 웹 문서 분류 [1], 스팸 메일 필터링 [2], 뉴스 기사의 카테고리 분류 그리고 본 논문에서 초점을 두고 있는 디지털 도서관 (Digital Library)에서 논문 분류 [3]등 다양한 분야에서 응용되어지고 있다. 우리가 디지털 도서관이나 인터넷에서 수집할 수 있는 연구 논문들은 제목 (Title), 저자 (Author), 요약 (Abstract), 키워드 (Keyword), 서론 (Introduction) 관련연구 (Related Work), 실험 (Experiment), 감사의 글(Acknowledgment), 그리고 참고문헌 (Reference) 등 일정한 형식을 가지고 있다. 제목이나 키워드 단락에서 사용된 단어는 그 단어가 서론이나 감사의 글에서 사용되어질 때보다 더 중요한 의미를 내포하고 있다. 그러나 기존의 문서 분류 알고리즘은 이런 연구 논문의 특성을 반영하고 있지 않기 때문에 문서의 특징을 정확히 추출하기 어렵고, 많은 잡음 단어들이 발생하게 된다. 이를 개선하기 위해 본 논문에서는 각각의 단락에서 발생하는 단어들에 대해서 다른 가중치 값을 설정함으로써 그 문서에 대한 단어들의 중요도가 높은 단어들을 특징으로 추출한다. 그리고 그 추출된 단어들로 연관 규칙을 이용하여 분류 규칙을 생성하고 문서를 분류에 이용한다.

2. 관련 연구

2.1 문서 분류

문서 분류에 대한 기존의 연구는 확률을 이용한 방법, 통계적인 기법을 이용한 방법, 그리고 벡터 유사도를 이용하는 방법 등이 있다 [8]. 그리고 최근에는 보다 정확하고 효과적인 분류를 위해 데이터 마이닝의 기법인 연관 규칙 (Association Rules)을 이용한 분류 방법들이 연구되어 왔다 [4, 6].

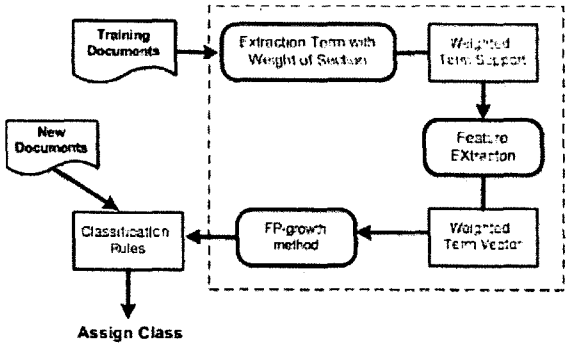
2.2 FP-growth 방법

연관 규칙 마이닝 알고리즘의 하나인 FP-growth 방법은 *Apriori* 알고리즘 [5]과 달리 후보 항목 집합을 생성하지 않고 빈발 항목을 생성하는 알고리즘이다. 이 방법은 자주 발생하는 패턴을 FP-tree라 불리는 트리 구조에 저장한 후 그 트리로부터 연관 규칙을 발견함으로써 큰 데이터나 다양한 패턴을 마이닝하는데 Apriori 알고리즘보다 빠르고 효과적이다 [7]. CBA (Classification Based on Associations) [4]와 같이 이전에 연구된 연관규칙 분류자 대부분은 Apriori 알고리즘을 이용하여 빈발 항목 집합을 생성하고 분류 규칙을 만들었지만 이 방법은 큰 데이터에는 많은 비용이 드는 단점이 있으며 다양한 패턴에는 적합하지 않았다. 이를 보안하기 위해 CMAR (Classification based on Multiple Association Rules)은 빈발 단어 집합을 생성하고 분류 규칙을 생성하는데 FP-tree 방법 [7]을 변형한 확장된 FP-tree 방법 (Extended FP-tree Method)을 이용하였다 [6].

3. 가중치가 부여된 연관 규칙 분류자

일반적으로 "제목" 이나 "키워드"의 단락에서 추출된

단어는 "서론" 이나 "감사의 글"의 단락에서 추출된 단어보다 더 중요도를 가지고 있다. 본 논문에서는 단어의 빈도수 (Term Frequency)와 각각의 단락에 부여된 가중치 (Weight)로 문서에 대한 단어의 지지도 (Term Support)를 측정하여 문서에 대한 단어들의 중요도가 높은 단어들로 연관 규칙을 적용한다. [그림 1]은 본 논문에서 제안하는 가중치가 부여된 연관 규칙 분류자 구성도를 나타낸다.



[그림 1] 가중치가 부여된 연관 규칙 분류자의 구성도

3.1 가중치가 부여된 단어 지지도

문서의 각 단락에서 추출된 단어 t_i 의 지지도 값 $sup_{t,P}$ 은 식(1)과 식(2)와 같이 단어의 빈도수와 단락의 가중치로 측정된다.

$$Sup'_{ti,P} = \sum_{Sj} tf_{ij} \cdot w_{Sj} \qquad (1)$$

여기서 tf_{ij} 는 단락 S_j 에 있는 t_i의 단어 빈도수를 나타내며, W_{Sj} 는 단락 S_j 의 가중치 값을 나타낸다.

$$Sup_{ti,P} = \frac{sup'_{ti,P}}{MAX\{sup'_{ti,P}\}} \qquad (2)$$

단어 t_i 에 대한 지지도 값은 [0, 1]의 범위를 가지며 문서 P에서의 단어 t_i의 중요성을 의미한다. 즉, 단어 지지도 값이 클수록 문서에 대한 단어의 중요성이 높음을 나타낸다.

예를 들어, 어떤 연구 논문 P_1에서 'Learning'란 단어가 "제목"에서 1번, "키워드"에서 1번, "요약"에서 3번, 그리고 "감사의 글"에서 5번 발생하였고, 각각의 "제목", "키워드", "요약", 그리고 "감사의 글"에 대한 단락 가중치 값이 W_{title} = 1.8, $W_{keyword}$ = 2, $W_{abstract}$ = 1.6, $W_{ackowledgement}$ = 1이라 한다면, $Sup'_{Learning,P1}$의 값은 다음과 같이 계산되어 진다.

$$Sup'_{Learning,P1} = (1\times1.8)+(1\times2)+(3\times1.6)+(5\times1) = 13.6$$

그리고 P_1의 연구 논문에서 'Agent'란 단어가 $Sup'_{Agent,p1}$ = 15.4 이고 P_1에서의 최대의 Sup' 값이라 한다면, 'Learning'이란 단어에 대한 지지도 값은 다음과 같다.

$$Sup_{Learning,P1} = \frac{Sup'_{Learning,P1}}{Sup'_{Agent,P1}} = 0.88$$

3.2 특징 추출

문서 분류의 정확성을 향상시키고 데이터의 벡터 크기를 줄임으로서 학습의 복잡도 (Complexity)를 감소시키기 위해 문서에서 덜 중요한 단어들을 버리는 과정이 필요하다. 이때 고려해야 할 사항은 단어 지지도 임계값 (Term Support Threshold) 선택이다. 단어 지지도 임계값이 높으면 많은 중요한 단어들을 걸러내어 문서의 데이터 표현에 충분하지 않으며, 반대로 단어 지지도 임계값이 낮으면 많은 잡음 단어 (Noise Term)들을 포함하게 되어 분류 성능을 저하시킨다. 본 논문에서는 트레이닝 데이터의 실험을 통한 Heuristic 방법에 의해 단어 지지도 임계값을 설정한다. 가중치가 부여된 단어들의 특징 추출 과정 후 각 연구 논문들은 아래와 같은 벡터 집합으로 표현된다.

$$P = \{P_c,(t_1,Sup_{t1,P}),(t_2,Sup_{t2,P}),\ldots,(t_n,Sup_{tn,P})\}$$

여기서 P_c 는 트레이닝 데이터에서 미리 분류된 문서의 분야 (Class)이고 t_n은 문서에서 추출된 단어, $Sup_{tn,P}$ 는 가중치가 부연된 단어 t_n의 지지도 값이다.

3.3 분류 규칙 마이닝

연관 규칙을 이용한 분류에서 첫번째로 중요한 일은 빈번히 발생하는 아이템 집합 (Frequent Itemsets)을 만드는 것이다. 본 논문에서는 빈발 항목 집합을 빈발 단어 집합 (Frequent Termsets)이라 하겠다. 본 논문에서는 빈발 단어 집합을 생성하고 분류 규칙을 생성하는데 FP-tree 방법 [7]을 변형한 확장된 FP-tree 방법 (Extended FP-tree Method)을 이용한다 [6]. 가중치가 부여된 지지도로 구성된 단어들의 데이터 집합은 확장된 FP-tree 알고리즘을 적용하여 EFP-tree를 생성하고 생성된 EFP-tree를 이용하여 분류 규칙을 마이닝한다 [7].

4. 실험 및 결과

4.1 실험 환경 및 데이터 집합

본 논문의 실험을 위해서 Apache 1.3, PHP4 와 MYSQL 4.0 을 사용해서 구현하였으며, 실험 환경은 펜티엄IV 1.7GHz, 256MB RAM의 시스템이었다.

트레이닝 및 테스트에 사용된 데이터들은 ACM 전자 도서관에서 수집하였으며, ACM 컴퓨터 분류 시스템 (Computing Classification System) [9]중 카테고리 I.2 인공지능 (Artificial intelligence) 분야에 속하는 연구 논문을 선택하였다. 데이터 구성은 [표 1]과 같다.

[표 1] 데이터셋의 구성

	Training Data	Test Data
Documents	583	197
Classes	5	5

4.2 실험 평가 방법

가중치가 부여된 연관 규칙 분류자를 이용하여 문서가 얼마나 정확하게 분류되었는지 성능을 평가하기 위해서 정확도 (Precision), 재현율 (Recall)과 F1-measure 측정식을 이용하였으며 각각의 정의는 다음과 같다 [8].

$$Precision\ of\ P_c = \frac{P_c\text{로 분류된 실제 문서 수}}{P_c\text{로 분류된 문서 수}} \qquad (3)$$

$$Recall\ of\ P_c = \frac{P_c\text{로 분류된 실제 문서 수}}{\text{전체 } P_c\text{에 속하는 문서 수}} \qquad (4)$$

$$F1 = \frac{2 * Recall * Precision}{Recall + Precision} \qquad (5)$$

여기서 P_c는 문서의 분야 (Class)를 의미하며 F1-measure의 값이 클수록 분류가 우수함을 의미한다.

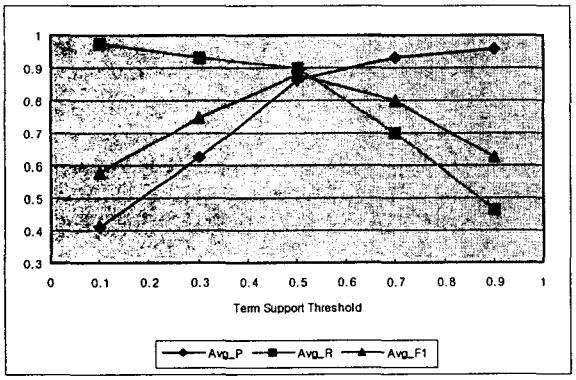
또한 각 단락 S_j 에 대하여 Heuristic 방법을 통하여 다른 가중치 값을 설정해 주었으며 [표 2]와 같다.

[표 2] 가중치 W_{Sj} 값

	제 목	요 약	키워드	서 론	관련연구
가중치	1.9	1.8	2	1.3	1.6
	내 용	실 험	결 론	감사의글	참고문헌
가중치	1.5	1.2	1.2	1	1.4

4.3 실험 결과

[그림 2]는 단어 지지도 임계 값 변화에 따른 평균 정확도 (Avg_P), 평균 재현율 (Avg_R) 그리고 평균 F1-measure의 값 (Avg_F1)의 변화를 나타낸 것이다. 임계값이 높아질수록 평균 정확도는 올라갔으나 재현율은 낮아졌다. 즉, 단어 지지도 임계값이 높으면 적은 수의 단어들로 분류 규칙을 생성했기 때문에 충분한 분류 규칙이 생성되지 않았으며 반대로 단어 지지도 임계값이 낮으면 잡음 단어까지 포함하여 분류 규칙을 생성하였기에 분류 규칙 정확성이 떨어졌다. 실험 결과 단어 지지도 임계값이 0.5일 때 분류 성능이 가장 우수하였다.

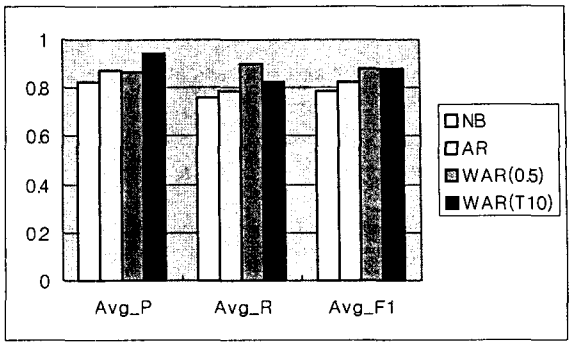


[그림 2] 단어 지지도 임계 값의 변화에 대한 정확도, 재현율 및 F1-measure 값

[그림 3]은 나이브 베이지안 분류자 (NB), 단어에 가중치를 부여하지 않은 연관 규칙 분류자 (AR) 와 본 논문에서 제안한 가중치가 부여된 연관 규칙 분류자 (WAR)를 이용한 분류 성능을 비교한 것이다. WAR(0.5)는 단어 지지도 임계 값을 0.5로 실험한 것이고 WAR(T10)은 단어 지지도 값 상위 10개의 단어로 실험을 했을 때를 나타낸다. 본 논문에서 제안하는 WAR 방법이 NB 방법보다 9.1%, 단순한 AR 방법보다는 5.6% 높은 분류 성능을 보였다. 또 단어 지지도 임계값이 0.5일 때와 단어 지지도 값 상위 10개의 단어로 비교 실험 했을 때 평균 정확도와 평균 재현율은 다소 차이가 있었으나 F1-measure 값을 비교했을 때 분류 성능이 거의 비슷함을 알 수 있었다.

5. 결론 및 향후 연구

본 논문에서는 전자 도서관에서 연구 논문의 효과적인 분류를 위해 가중치가 부여된 연관 규칙 분류자를 제안하였다. 본



[그림 3] 분류 성능 비교

논문에서 제안한 가중치가 부여된 연관 규칙 분류자의 성능을 평가하기 위해서 가중치를 주지 않은 연관 규칙 분류자와 기존의 널리 사용되고 있는 나이브 베이지안 분류자와 비교 실험하여 높은 분류 성능을 보였다. 또 단어 지지도 임계치의 변화에 따른 정확도, 재현율 및 F1-measure의 변화를 알아보았으며 F1-measure 값이 최대 값을 가지는 단어 지지도 임계 값과 상위 10개의 단어로 분류 규칙을 생성했을 때와 분류 성능을 비교 하여 비슷한 분류 성능을 보였다.

하지만 본 논문에서 제안된 방법은 모든 텍스트로부터 중요한 단어를 추출하는데 복잡도와 시간 비용이 다소 높았다. 그러므로 향후에는 모든 단락이 아닌 중요한 단락에서만 단어를 추출하여 분류 비용을 축소하는 연구가 필요하겠다.

6. 참고 문헌

[1] S. H. Lin, M. C. Chen, J. M. Ho, Y. M. Huang. "ACIRD : Intelligent Internet Document Organization and Retrieval," IEEE Transactions on Knowledge and Data Engineering, Vol. 14, No. 3, May/June 2002

[2] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz. "A Bayesian approach to filtering junk email," Proc. AAAI'98 Workshop on Learning for Text Categorization, July,1998

[3] Kurt Maly, Mohammad Zubair, and Hesham Anan, "An Automated Classification System and Associated Digital Library Services", Proc. NDDL 2001, July 2001.

[4] B. Liu and W. His and Y. Ma "Integrating Classification and Association Rule Mining," In Pro. Of 1998 Int. Conf. On Knowledge Discovery and Data Mining(KDD'98), August 1998

[5] R. Agrawal and R. Srikant. "Fast Algorithms for Minig Association Rules," Proc. 20th Int'l Conf. Very Large Data Bases(VLDB), Sept. 1994

[6] We. Li, J. Han and J. Pei. "CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules," Proc. ICDM 2001, Dec. 2001

[7] J. Han, J. Pei and Y. Yin. "Mining Frequent Patterns without Candidate Generation," In Proc. 2000 ACM-SIGMOD Int. Conf. Management of Data, May 2000

[8] Y. Yang, X. Liu. "A re-examination of text categorization methods," Proc. of SIGIR-99, 1999

[9] ACM Computer Classification System, http://www.acm.org/class