제33회 한국정보처리학회 춘계학술발표대회 논문집 제17권 제1호 (2010. 4)

논문의 참조 정보를 이용한 새로운 논문 유사도

윤석호*, 김상욱*, 박선주**
*한양대학교 전자컴퓨터통신공학과
**연세대학교 경영학과
e-mail: bogely@agape.hanyang.ac.kr

A New Similarity Measure Using Reference Information for Scientific Literature

Seok-Ho Yoon*, Sang-Wook Kim*, Sunju Park**
*Department of Electronics and Computer Engineering, Hanyang University
**School of Business, Yonsei University

요 약

본 논문에서는 논문들 간의 참조 정보를 이용한 새로운 논문 유사도 계산 방안을 제안한다. 본 논문에서 제안하는 방안은 in-link와 out-link를 undirected-link로 간주함으로써 in-link와 out-link를 동시에 이용하여 논문들 간의 유사도를 적절하게 계산한다. 사례 분석을 통해서 제안하는 방안의 우수성을 검증한다.

1. 서론

최근 들어, 학술정보에 대한 사용자들의 관심이 증가하면서 논문 검색 서비스를 이용하는 사용자가 증가하고 있다. 대표적인 논문 검색 서비스로는 사용자가 관심 있는 논문과 유사한 논문들을 검색해주는 것이다. 이러한 서비스를 제공하기 위해서는 논문들 간의 유사도를 계산하는 방안이 필요하다. 따라서 본 논문에서는 논문들 간의 유사도를 계산하는 기존 방안의 문제점을 설명하고 기존 방안의 문제점을 해결하는 새로운 논문 유사도 계산 방안을 제안한다.

2. 관련 연구

논문들 간의 유사도를 계산하는 기존 방안들은 주로 논문에 포함되어 있는 참조 정보를 링크로 변환하고 링크 기반 유사도 계산 방안을 이용하여 논문들 간의 유사도를 계산하였다. 대표적인 방안들로는 Coupling[1], Co-citation[2], Amsler[3] 등이 있다. Coupling은 논문 들 간의 유사도를 두 논문이 공통적으로 참조하는 논문들의 수를 이용하여 계산한다[1]. Co-citation은 Coupling과 반대로 두 논문을 공통적으로 참조하는 논문들의 수를 이용하여 두 논문의 유사도를 계산한다[2]. Amsler는 Coupling과 Co-citation으로 계산한 각각의 유사도를 가중치 합하여 논문들 간의 유사도를 계산한다[3]. 즉, Coupling의 가중치가 0.5이고 Co-citation의 가중치가 0.5이면 Coupling과 Co-citation으로 계산된 유사도를 더한 후에 2로 나누어준 값이 Amsler로 계산한 유사도가 된다.

논문 데이터베이스의 논문들은 해당 논문이 발행되기 전에 발행된 논문들만을 대상으로 참조 정보를 생성하는 특성을 가지고 있기 때문에 기존 방안으로 논문들 간의 유사도를 계산하면 세 가지 경우에서 문제가 발생한다.

(P1) 오래된 논문들 간에 유사도를 계산하는 경우
(P2) 최근 논문들 간에 유사도를 계산하는 경우
(P3) 최근 논문들과 오래된 논문들 간에 유사도를 계산하는 경우

(P1)에서 발행연도가 오래된 논문은 참조하는 논문들이 해당 논문 데이터베이스 내에 적거나 없기 때문에 서로 유사한 주제의 논문들이라도 Coupling으로 유사도를 계산하면 유사도가 0으로 계산 될 수 있다. (P2)에서 최근 논문은 사람들에게 충분히 알려지지 않아서 해당 논문을 참조하는 논문들이 적거나 없기 때문에 Co-citation으로 유사도를 계산하면 0으로 계산 될 수 있다. 또한, (P3)에서 오래된 논문은 참조하는 논문이 논문데이터베이스 내에 적거나 없고, 최근 논문은 해당 논문을 참조하는 논문들이 적거나 없기 때문에 Coupling과 Co-citation 어느 방안으로 계산하더라도 모두 유사도가 0으로 계산 될 수 있다.

Amsler는 Coupling과 Co-citation으로 계산된 각각의 유사도를 가중치 합하여 유사도를 계산한다. 따라서 (P1)과 (P2)에서 Amsler로 유사도를 계산하면 유사도가 0이 되지는 않는다. 그러나 만약 Amsler에서 Coupling과 Co-citation의 가중치가 둘 다 0.5라고 하면 (P1)과 (P2)에서 Coupling과 Co-citation으로 계산된 유사도는 각 방법의 가중치만큼 유사도를 잃어버리기 때문에 최대값이 0.5가 될 수 있다. 즉, 유사도를 계산하고자 하는 두 논문들 중 하나의 논문이라도 참조하는 논문들이 없거나 해당 논문들을 참조하는 논문들이 없다면 Coupling과 Co-citation으로 계산한 유사도가 둘 중 하나라도 0이 될 수 있기 때문에 Amsler로 계산한 유사도는 0.5이하의 값이 된다. 따라서 Amsler로 계산한 유사도는 적절하지 못하다 또한, (P3)에서는 Coupling과 Co-citation 모두 유사도가 0으로 계산 될 수 있기 때문에 Amsler 역시 0으로 계산된 될 수 있다.

3. 제안하는 방안

두 논문 A와 B가 있을 때 논문 A와 B가 동시에 가리키는 논문들과 논문 A와 B를 동시에 가리키는 논문들이 많으면 논문 A와 B가 서로 유사하다고 할 수 있다. 마찬가지로 논문 A가 가리키는 많은 논문들이 논문 B를 가리킨다면 논문 A와 B가 서로 유사하다고 할 수 있다. 두 논

문이 유사하다고 말할 수 있는 이러한 3가지 경우는 결국 두 논문과 공통적으로 연결되어 있는 논문들의 수가 많으면 두 논문은 유사하다고 설명할 수 있다.

본 논문에서는 이러한 아이디어를 기반으로 두 논문들 간의 유사도를 계산하고자 한다. 따라서 제안하는 방안은 논문들 간의 참조 방향을 무시해서 모두 양방향로 변환하고 공통적으로 연결되어 있는 논문들의 수를 이용해서 유사도를 계산한다. 이러한 방안은 Coupling, Co-citation 그리고 두 논문을 이어주는 논문들을 이용해서 유사도를 계산하는 방법들 모두를 적절하게 통합해서 논문들 간의 유사도를 계산한 것이 된다.

기존 방안들은 논문을 정점으로 논문들 사이에 존재하는 참조 정보를 간선으로 정의해서 그래프로 표현했을 때 in-link만을 이용해서 유사도를 계산하거나 out-link만을 이용해서 유사도를 계산하였다. Amsler의 경우 in-link와 out-link를 모두 이용한다고 주장하지만 in-link와 out-link를 각각 이용하기 때문에 각 방법의 가중치만큼 유사도를 잃어버리는 문제가 발생한다. 본 논문에서 제안한 방안은 in-link와 out-link를 undirected-link로 동일하게 변환하였기 때문에 두 논문이 공통적으로 가리키는 논문들 그리고 두 논문을 공통적으로 가리키는 논문들 그리고 두 논문을 이어주는 논문들을 모두 이용하여 유사도를 계산한다.

4. 실험

제안하는 방안을 검증하기 위해서 본 논문에서는 DBLP[1] 에 있는 논문들을 사용했으며 논문들 간의 참조 정보는 Libra[2]에서 크롤링해서 사용하였다. 실험 방법은 본 논문에서 언급한 3가지 문제점들이 실제 논문 데이터베이스에서 논문들 간의 유사도를 계산할 때 발생하는지 살펴보고 이러한 문제점들을 제안하는 방안이 해결하는지 실제 사례를 통해서 알아본다. 실험 대상은 Coupling, Co-citation, Amsler, 그리고 제안하는 방안이다. 제안하는 방안의 최종 유사도는 자카드 계수(Jaccard's coefficient)[4]와 같은 방식으로 0과 1사이의 값으로 정규화한다.

표 1은 같은 주제의 오래된 논문들 간의 유사도와 최근 논문들 간의 유사도 그리고 최근 논문과 오래된 논문 간의 유사도의 예를 보여준다. 표 1에서 같은 주제의 오래된 논문 [5]와 [6]의 유사도는 Coupling에 의해서만 0으로 계산되었다. 마찬가지로 같은 주제의 최근 논문 [7]과 [8]의 유사도는 Co-citation에 의해서만 0으로 계산되었다. 그러나 같은 주제의 최근 논문 [9]와 오래된 논문 [5]의 유사도는 오직 본 논문에서 제안한 방안만 0으로 계산하지 않았고 기존 방안 모두 0으로 계산하였다.

5. 결론

본 논문에서는 논문들 간의 유사도를 계산하기 위해서 논문들 간의 참조 정보를 이용한 링크 기반 유사도 계산 방안을 제안하였다. 본 논문에서 제안하는 방안은 기존 방안과 다르게 in-link와 out-link를 동시에 이용하기 위해서 in-link와 out-link을 undirected-link로 간주함으로써 논문들 간의 유사도를 적절하게 계산하였다. 사례분석을 통하여 본 논문에서 제안하는 방안이 기존 연구에 비해서 우수하다는 것을 검증하였다.

표 1. 사례 분석 결과

	같은 주제의 오래된 논문들	같은 주제의 최근 논문들	같은 주제의 최근 논문과 오래된 논문
대상 논문들	[5] vs [6]	[7] vs [8]	[5] vs [9]
Coupling	0	0.111	0
Co-citation	0.4674	0	0
Amsler	0.2337	0.0556	0
제안하는 방안	0.4551	0.0954	0.0125

참고문헌

본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업(NIPA-2010-(C1090-1011-0009))의 부분적인 지원과 NHN(주)의 지원을 받았습니다. 그러나 본 논문에서 제시된 의견이나 결론, 또는 권고 등은 온전히 저자(들)의 것이며, 반드시 지원회사의 입장을 대변하는 것은 아닙니다.

참고문헌

[1] M. Kessler, "Bibliographic Coupling Between Scientific Papers," *Journal of the American Documentation*, Vol. 14, No. 1, pp. 10-25, 1963.

[2] H. Small, "Co-citation in the Scientific Literature: A New Measure of the Relationship between Two Documents," *Journal of the American Society for Information Science*, Vol. 24, No. 4, pp. 265-269, 1973.

[3] R. Amsler, "Application of Citation-Based Automatic Classification. Technical report," *The University of Texas at Austin Linguistics Research Center*, 1972.

[4] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2006.

[5] R. Ng and J. Han, "Efficient and Effective Clustering Methods for Spatial Data Mining," In *Proc. Int'l. Conf. on Very Large Databases*, pp. 144-155, 1994.

[6] T. Zhang, R. Ramakrishnam, and M. Livny, "BIRCH: an Efficient Data Clustering Method for Very Large Databases," In *Proc. Int'l. Conf. on Management of Data*, pp. 103-114, 1996.

[7] E. Ng, A. Fu, and R. Wong, "Projective Clustering by Histograms," *IEEE Transactions on Knowledge and Data Engineering*," Vol. 17, No. 3, 2005.

[8] A. Meka and A. Singh, "Distributed Spatial Clustering in Sensor Networks," In *Proc. Int'l. Conf. on Extending Database Technology*, pp. 980-1000, 2006.

[9] H. Kriegel, A.Pryakhin, and M.Schubert, "An EM-Approach for Clustering Multi-Instance Objects," In *Proc. Int'l. Pacific-Asia Conf. on Knowledge Discovery and Data Mining,* pp. 139-148, 2006.

1) http://www.informatic.uni-trier.de/ley/db/

2) http://academic.research.microsoft.com/