문서 요약 시스템을 위한 수사 구조 트리 생성

정준호, 김미진, 이현주, 박미성, 이상조
경북대학교 컴퓨터공학과

Rhetorical Structure Tree Generation for Text Summarization System

Joon-Ho Jung, Mi-Jin Kim, Hyun-Ju Lee, Mi-Sung Park, Sang-Jo Lee
Department of Computer Engineering, Kyungpook National University

요 약

본 논문에서는 수사 정보와 문장간 유사도를 이용하여 문서의 수사 구조 트리를 생성하는 방법을 제안하였다. 말뭉치에서 찾아낸 수사 정보를 종류별로 분류하고, 이를 사용해서 문서 내의 수사 정보를 추출해서 가능한 모든 구조를 생성한다. 다음으로 문장간의 유사도를 사용해서 가중치가 가장 높은 하나의 구조를 선택한다. 생성된 수사 구조를 사용하여 문서를 요약할 수 있는데, 수사 정보는 언어적 특성을 이용하는 것이므로 도메인에 독립적인 요약 시스템을 만들 수 있다.

1. 서론

검색시스템으로 인터넷에 있는 문서들을 검색할 때, 너무나 많은 문서가 검색되기 때문에 검색된 문서가 적확한 것인지를 확인하기가 쉽지 않다. 만약 이러한 문서들에 대한 요약문을 가지고 있다면, 사용자가 원하는 문서를 찾는데 도움을 줄 수 있다.

이러한 요약문을 생성하는 방법은 여러 가지가 있다. 먼저, 국내 논문으로는 문장 유사도와 말뭉치에서 찾아낸 통계적인 정보를 사용한 것과[1], 요약문에 나타날 수 있는 특성과 중요 단어를 학습하여 사용한 것이 있다[2]. 또 Chin-Yew Lin과 Eduard Hovy[3]는 문서 내의 위치로 문장의 중요도를 계산하고, Daniel Marcu[4]는 문서의 수사적인 구조를 사용한다. 이와 같은 요약 시스템은 문서 내의 중요한 문장을 추출(extraction)하는 것으로 요약문에 포함될 수 있는 가능성을 판단하여, 문장을 추출하는 시스템이다.

본 논문에서는 수사 정보를 사용하여 수사 구조를 생성하는 방법을 제안한다. 수사 정보를 사용하여 요약문을 추출하는 방법은 문서에서 수사 구조를 추출하는 과정과 이를 이용해서 요약하는 과정으로 나뉘는데, 본 논문은 그 중 앞부분만을 대상으로 한다. 문장들 간에는 병렬 관계, 부연 관계 등 여러 가지 관계가 있는데, 문서에서 추출한 수사 정보와 문장과 문단간의 유사도를 이용하여 이러한 관계를 추출한다. 통계적인 방법을 사용하는 경우 도메인에 종속된 결과를 가져오지만, 수사 정보는 언어적인 특성을 이용하는 것이므로 이러한 방법을 사용하면 도메인에 독립적인 시스템을 만들 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 수사 구조란 어떤 것인지 살펴보고, 3장에서는 수사 구조를 생성하는 방법을 제안하고, 마지막으로 4장에서 결론을 내린다.

2. 수사 구조

수사 구조는 문서 내에서 각 문장 사이의 관계를 나타내는 이진 트리이다. 트리의 leaf node는 문장을 가리키고, 중간 node는 문장들 사이의 관계를 나타낸다. 이러한 관계는 문서에서 추출한 수사학적인 정보와 문장간의 유사도를 가지고 생성한다.

수사학적인 정보는 말뭉치에서 추출하고, 이러한 정보를 바탕으로 비슷한 역할을 하는 것끼리 분류하여 문장사이에 어떠한 관계가 존재하는지 찾아낸다. 이 논문에서는 20개 정도의 수사 관계로 분류하였다. 다음의 표는 이러한 관계의 예를 보여준다.

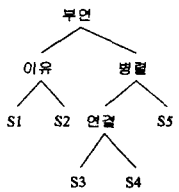
관계	표현
병렬	그리고, 또한
반대	그러나, 하지만
결론	따라서
⋮	⋮
추가	그뿐 아니라, 게다가

[표 1] 수사 관계의 예

다음의 예는 실제로 문서에서 수사구조가 어떻게 구성되어 있는지를 보여준다.

[예제 1]

정보(information)에 대한 개념적 정의는 일괄적이지 않다.(s1) 정보는 그 용어가 쓰여지는 상황(context)에 따라 각기 다른 의미로 해석될 수 있기 때문이다.(s2) 정보의 원어는 중세 라틴어인 informatio에서 출발한다.(s3) 당시의 의미는 주어진 어떤 '형상''구성' 또한 '교시' 등을 뜻했던 것으로 알려지고 있다.(s4) 또한 프랑스에서는 고전적 원어로서 주로 법적인 차원에서 '어떤 진상에 대한 수집 및 처리'의 의미로 사용되었다.(s5)

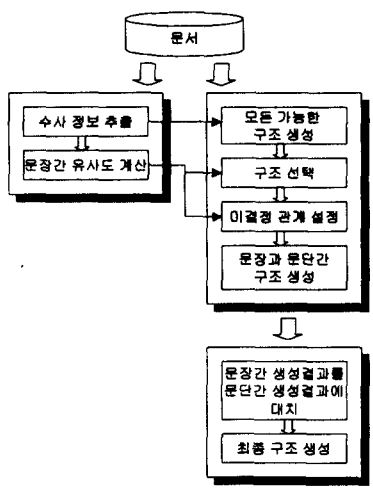


[그림 1] [예제 1]의 수사 구조

위의 그림에서 각 leaf node는 위 예제(1)의 각 문장을 가리키고 중간노드는 관계를 나타내는데, "~ 때문이다", "당시의", "또한" 등의 수사 정보와 5개 문장의 유사도를 사용해서 [그림 1]과 같은 수사 구조를 생성한다.

3. 수사 구조의 생성

전체 시스템 구성도는 [그림 2]와 같다. 본 시스템은 문서에서 수사 정보의 추출과 문장간 유사도 계산, 문장간 구조 생성과 문단간 구조 생성, 그리고 최종 구조 생성 과정으로 이루어져 있다.

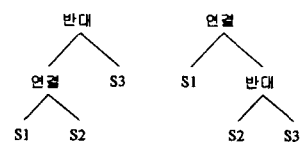


[그림 2] 전체 시스템 구성도

앞서 기술한 바와 같이, 문서 전체의 수사 구조를 만들기 위해서는 문장간, 문단간 수사 구조 생성의 두가지 과정을 거친다. 여기서 문장간 수사 구조는 한 문단에 포함된 문장간의 구조이다. 각각의 구조를 생성한 후 문단간 수사 구조의 leaf node에 문장간 수사 구조를 대치시켜서, 전체 구조를 생성한다. 문장이나 문단간 구조를 만드는 과정은 매우 유사하므로, 이후 문장간 구조를 만드는 과정을 설명한다.

문장간 수사 구조 생성에는 수사정보와 문장간 유사도를 사용한다. 그 과정은 다음과 같다.

먼저, 수사 정보를 이용해서 가능한 모든 수사 구조를 생성한다. n개의 문장이 있을 때, 생성 가능한 트리의 개수는 $2^{(n-2)}$ 이 된다. 하지만 수사 관계의 제약 조건에 의해 수사 구조의 생성에 제약이 가해진다. 이러한 제약은 말뭉치에서 얻어진 수사 관계별로 관계가 포함하는 문장의 범위에 따라 가중치를 부여해서 얻어진다. 즉, 수사구조는 가중치가 작은 것이 가중치가 큰 것의 안에서 관계를 생성하도록 한다. 예를 들어, [S1 그것은 S2 그러나 S3] 라는 문장구성이 있으면, 아래 [그림 3]과 같이 두 가지 트리가 생성 가능하다. 그러나 '그것은' 의 가중치가 '그러나' 보다 작으므로 뒤의 구조는 제거되고 앞의 구조만이 생성되게 된다. 그리고, 수사 정보가 없는 문장에 대해서는 제약 조건을 만족하는 모든 구조를 생성하고, 수사 관계의 종류는 구조 선택 후 결정한다.



[그림 3] 수사 구조 트리의 예

다음에는, 위에서 생성한 모든 수사 구조에 가중치를 부여하여, 그중 가장 가중치가 높은 것을 선택한다. 가중치는 문장간의 유사도를 이용한다. 문장간의 유사도가 높은 것을 지역적으로 가깝게 묶인 것이, 가중치가 가장 높게 된다. 만약 가중치가 같거나 비슷하다면, 우측 노드의 깊이가 좀 더 깊은 것을 선택한다. 문장간의 유사도는 코사인 계수를 응용하여 다음과 같은 식으로 계산된다.

$$sim(Si,Sj)=\frac{\sum_{t\in Si\cap Sj} Wi(t)Wj(t)\dfrac{(Wi(t)+Wj(t))}{W(t)}}{\sqrt{\sum_{t\in Si} Wi(t)^2 \sum_{t\in Sj} Wj(t)^2}}$$

여기서 Si, Sj는 비교할 대상이 되는 문장이나 문단이고, Wi(t), Wj(t)는 각각 Si, Sj에서의 명사 t의 빈도이다. W(t)는 Si, Sj가 문장이냐 문단이냐에 따라서, 문단 내에서의 명사 t의 빈도나 문서 전체에서의 빈도가 된다. 여기서 W(t)값으로 나누어 주는 이유는 문서 내에서 일반적으로 많이 나온 명사는 문장간의 유사도 계산에서 가중치를 낮추어 주기 위함이다.

마지막으로, 생성된 수사 구조에서 수사 정보가 없는 문장에 대한 수사 관계를 설정해 준다. 첫째 단계에서 수사 정보가 없는 문장에 대해서는 관계를 설정하지 않았다. 수사 정보가 없는 문장은 앞 문장들과 같은 주제를 이야기하는 경우(부연)와 다른 주제를 이야기하는 경우(전환)의 두가지 관계만을 생성한다. 아직 결정되지 않은 노드의 좌측 문장들과 우측 문장들 사이의 유사도를 구해서 임계치를 넘으면 부연, 넘지 않으면 전환의 관계로 만들어 준다.

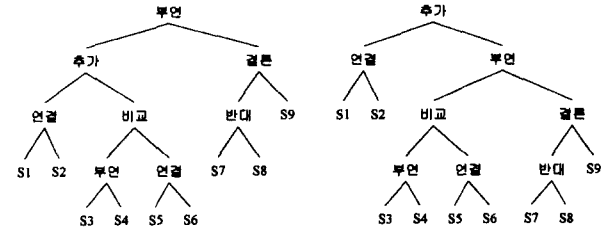
다음의 예는 실제로 이와 같이 수사 구조를 생성한 예인데, [그림 3]의 첫 번째 그림은 수작업으로 구조를 생성해 본 것이고, 두 번째 그림은 위에서 제안한 방법을 이용하여 생성한 구조이다.

흔히 Y2K라고 부르는 '컴퓨터 2000년 문제'에 대비할 시간이 8개월도 남지 않았다.(S1) 선진국에서는 몇년전부터 이에 대처해 오고 있으나 우리는 문제인식도 늦었을 뿐더러 때마침 닥쳐온 경제위기 때문에 대처도 미온적이었다.(S2)

게다가 "설마 무슨 일이 있겠느냐" 식의 불감증까지 겹쳐있는 실정이라 한국이 Y2K 최대 피해국이 될 것이라는 이야기도 있다.(S3) 정부는 뒤늦게 전분야에 걸쳐 Y2K 대책을 추진해 오고 있고 대기업들도 그런대로 대비를 하고 있다.(S4) 더 큰 문제는 중소기업, 학교, 고층건물 등의 경우다.(S5) 이들은 자신들이 어떤 문제를 갖고 있는지조차 모르고 속수무책으로 2000년을 맞을 것이니 한심한 일이다.(S6)

Y2K 문제는 기본적으로 시스템이나 기기를 구입한 당사자가 책임지고 해결할 사안이다.(S7) 하지만 온갖 네트워크로 거미줄같이 얽혀있는 것이 현대 사회라서 한 기업의 Y2K 문제는 곧 사회 전체의 Y2K 문제인 것이다.(S8) 따라서 정부와 경제단체가 보다 적극적으로 중소기업을 상대로 홍보에 나서고 관련기술과 정보를 제공하는 등 각종 대책에 앞장서야 할 것이다.(S9)

[예제 2] (조선일보 사설 중에서)



(a) 수작업으로 생성한 결과 (b) 제안한 방법을 사용한 결과

[그림 4] [예제 2]의 수사 구조

위의 결과에서 문단간 구조 생성결과가 직접 생성한 것과 제안한 방법으로 생성한 것에 차이가 나는데, 이는 둘째 문단과 셋째 문단의 유사도가 첫째 문단과의 유사도보다 높기 때문이다. 위의 예에서 '2000년' 아니 'Y2K' 는 같은 의미이지만, 실제로는 다른 명사로 추출되었다. 이와 같이 형태는 다르지만 의미는 같은 동의어와 연어정보 등을 사용하면 좀더 정확한 결과를 생성할 수 있다.

4. 결론

본 논문에서는 문서의 수사 정보와 문장간 유사도를 이용해서 수사 구조 트리를 생성하는 방법을 제시하였다. 말뭉치에서 얻어낸 수사 정보를 종류별로 분류하고, 이를 이용해서 문서의 가능한 모든 수사 구조를 생성한 후, 유사도를 이용해서 하나의 구조를 선택하였다.

본 시스템에서 생성한 수사 구조는 문서를 요약하는데 쓰일 수 있다. 또한 제안한 방법을 요약 시스템에 적용할 경우에는 각 관계에 따라 문장들의 상대적인 중요도가 다른 점을 이용해서 각 문장의 중요도를 계산한 후, 중요한 순서대로 문장을 추출해야 한다.

향후 연구 과제는 수사 정보의 종류를 좀 더 세밀히 분류하고, 문장간 유사도 이외에 동의어와 연어정보 등의 정보를 사용하여 수사 구조의 생성의 정확도를 높이는 연구가 필요하다.

참고문헌

[1] 강상배, 조혁규, 권혁철, 박재득, 박동인, "한국어 문서의 통계적 정보를 이용한 문서 요약 시스템 구현", 제9회 한글 및 한국어정보처리 학술대회, 1997

[2] 장동현, 맹성현, "문서 구조 정보를 이용한 확률 모델 기반 자동요약 시스템", 제9회 한글 및 한국어정보처리 학술대회, 1997

[3] Lin, C.Y. and E.H. Hovy. 1997. "Identify Topic by Position", In Proceedings of the 5th Conference on Applied Natural Language Processing, ANLP. Washington, DC.

[4] Daniel Marcu, "The Rhetorical Parsing, Summarization, and Generation of Natural Language Texts", Ph.D dissertation, University of Toronto, Canada, 1997

[5] Kenji Ono, Kazuo Sumita, Seiji Miike, "Abstract Generation based on Rhetorical Structure Extraction" in Proceedings of the 15th International Conference on Computational Linguistics (COLING-94), Vol 1 pp 344-348, Kyoto, Japan. 1994.

[6] Daniel Marcu, "Building Up Rhetorical Structure Trees", American Association for Artificial Intelligence, 1996.

[7] Eduard Hovy and Chin-Yew Lin. Automated text summarization in SUMMARIST. IN ACL/EACL97 Workshop on Intelligent Scalable Text Summarization, pages 18-24, 1997.