수사구조를 이용한 텍스트 자동요약

이유리, 최기선
한국과학기술원 전산학과, 전문용어언어공학연구센터
대전시 유성구 구성동 373-1, 우:305-701
{yuri, kschoi}@world.kaist.ac.kr

Text Summarisation with Rhetorical Structure

Yuri Lee, Key-Sun Choi
Department of Computer Science, KORTERM
Korea Advanced Institute of Science and Technology
373-1 Kusong-dong, Yusong-gu, Taejon, 305-701

요 약

텍스트 요약이란 중요정보만을 추출하여 본래 텍스트의 의미를 전달하는 축약 과정이다. 인터넷을 통한 온라인 정보가 급증함에 따라 정보에 대한 처리와 신속한 내용 파악을 위한 효율적인 자동 텍스트 요약 방법이 필요하다.

기존의 통계적 방법으로는 전체 텍스트의 구조적인 특징을 고려할 수가 없기 때문에, 생성된 요약문의 의미적 흐름이 부자연스럽고, 문장간 응집도가 떨어지게 된다. 수사학적 방법은 요약문을 생성하기 위해서 문장간의 접속관계를 이용한다. 수사 구조란 텍스트를 이루는 문장들간의 논리적인 결합관계로, 수사학적 방법은 이러한 결합관계를 파악하여 요약문을 생성하는 방법이다.

본 논문에서는 표지들이 나타내는 접속 관계정보를 사용하여, 텍스트의 수사구조를 분석한 후 요약문을 생성하는 시스템을 구현한다. 수사구조 파싱 과정은 문장간의 수사구조 파싱과 문단간의 수사구조 파싱, 두 단계로 이루어진다. 파싱은 차트파싱 방법을 사용하여 상향식으로 진행된다. 입력된 문장들로부터 두 단계 파싱에 의해 전체 텍스트의 수사구조 트리를 생성하며, 생성된 트리에서 가중치를 계산하여 중요 문장들을 요약문으로 추출한다.

1 서론

인터넷을 통한 온라인 정보의 급증으로 우리가 하루에 접하는 정보의 양도 지속적으로 증가하고 있다. 이러한 정보에 대한 처리와 내용 파악을 위해서 효율적인 요약방법이 필요하다.

본 논문에서는 수사구조를 이용한 요약시스템을 제안한다. 수사구조란 텍스트를 이루는 문장들간의 논리적인 결합관계이다. 이러한 결합 관계는 주문장과 부문장(nucleus-satellite)의 관계로 표현된다. 텍스트의 수사구조는 중심이 되는 문장과 이에 대한 부가적인 문장의 관계를 파악하므로서 분석할 수 있다. 본 논문에서 제시한 요약 시스템은 결합관계를 나타내는 수사어구들에 대한 접속관계 정보를 사용하여, 원문 텍스트로부터 수사구조를 분석하고, 계층적인 수사 트리 구조를 생성한다. 수사구조를 분석하는 과정은 문장간 수사구조 파싱과 문단간 수사구조 파싱, 두 단계로 진행되며, 생성된 수사구조 트리에서 가중치를 계산하여, 사용자가 원하는 수준까지의 요약문을 생성한다.

본 논문은 다음과 같이 구성된다. 2장에서는 기존의 요약시스템들이 중요문장을 추출하기 위해 사용된 가정들에 대해서 설명한다. 3장에서는 수사구조 파싱을 사용한 요약문 생성에 대해서 설명하고 4장에서는 실험과 결과에 대해서 설명한다. 그리고 5장에서는 결론과 향후 연구방향에 대해 설명한다.

2 관련 연구

기존의 요약 시스템들이 텍스트로부터 중요문장을 결정하기 위해 사용된 가정들은 다음과 같다.

- 중요한 문장은 텍스트 내에서 가장 빈번히 사용되는 단어들을 포함한다 (Luhn ,1958).
- 중요문장은 타이틀이나 헤딩에서 사용되는 단어를 포함한다 (Edmundson, 1968).

- 중요문장은 문단의 시작이나 끝부분에 위치한다 (Baxendale, 1958).
- 중요문장은 텍스트에서 장르 의존적인 부분에 위치한다. 이러한 위치는 학습에 의해 자동으로 결정된다 (Lin and Hovy, 1997).
- 중요문장은 단서 단어(cue word)를 갖는다 (Edmundson, 1968).

단어빈도수에 기반한 통계적인 방법에서는 텍스트구조 분석이 미흡하여 요약문의 의미적인 흐름이 부자연스럽다. 단서 단어를 이용하거나, 위치정보를 이용한 경우, 문장간의 중요도 비교를 할 수가 없고, 요약문의 길이 조정이 불가능하다.

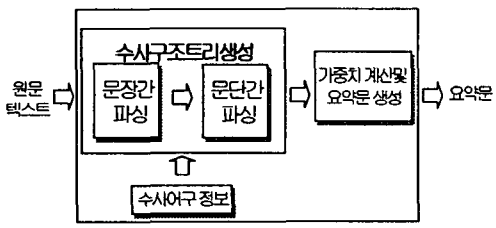
- 중요문장과 개념은 복잡한 의미구조 상에서 많이 연결된 구조를 갖는다 (Barzilay and Elhadad, 1997).
- 중요문장과 그렇지 않은 문장은 텍스트 수사구조 표현에서 추론할 수 있다 (Ono,1994; Marcu,1997).

텍스트 내 단어들의 의미 연결구조를 이용하는 요약시스템의 경우 WordNet과 같은 시소러스에 저장된 의미 관계정보만을 이용하므로 특정 텍스트에서의 단어들의 관계를 파악하는 데는 한계가 있다. Ono(1994)는 일본어 텍스트에 대해서 수사구조를 이용한 요약시스템을 구현하였는데, 수사관계 결합시 정해 놓은 우선순위를 사용하여 결합한 후 수사 구조를 생성하였다.

본 논문에서는 한국어 텍스트에서의 수사어구들을 분석하여, 텍스트의 수사 구조 트리를 생성하였으며, 결합의 우선순위를 정하지 않고, 수사구조들간의 결합을 문법 규칙으로 기술하여 이를 이용한 상향식 차트 파싱방법으로 수사 구조 트리를 생성하였다.

3 수사구조를 이용한 텍스트 자동요약

본 논문에서 제시하는 요약시스템의 구조는 [그림1]과 같다. 수사어구 정보를 이용하여 전체 텍스트의 수사구조 트리를 생성한다. 수사구조 트리를 생성하는 과정은 문장간 수사구조 파싱과 문단간 수사구조 파싱, 두 단계로 이루어진다. 생성된 수사구조 트리로부터 가중치를 계산하여 중요문장들을 요약문으로 추출한다



[그림 1] 요약시스템 구성

3.1 수사어구

수사구조란 텍스트를 이루는 문장들간의 논리적인 결합관계로서 텍스트의 주문장-부문장 (nucleus-satellite) 관계로 표현될 수 있다. 수사구조를 파악하기 위해서는 일반적으로 글을 전개할 때 문장간 연결을 위해 쓰이는 수사어구들에 대한 정보와 그 수사어구들이 갖는 관계들과 그 유형을 설정해야한다.

문장들간의 접속관계를 나타내는 표지들로는 접속어구, 지시어, 반복어, 조응어 등이 있다. 본 논문에서 이용하는 수사어구는 접속어구, 지시어, 숙어적인 표현들로 제한한다.

```
(IP 나아가/pvg+아/ecs )
(IP 다시/mag 말하/pvg+면/ecs )
(IP 다시/mag 말하/pvg+어서/ecs )
(IP 더/mag 나아가/pvg+아/ecs )
(IP 아니/paa+면/ecs )
(IP 그렇/pad+다더라/ecs+도/jxc )
(IP 그리고/maj 나/pvg+아서/ecs )
(IP 그리고/maj 나/pvg+자/ecs )+
(IP 다섯째/nno )+,/sp
(IP 다시/mag 말하/pvg+면/ecs )
(IP 다시/mag 말하/pvg+어/ecs )
(IP 때문/nbn+에/jca )
(IP 말하자면/maj )+,/sp
(IP 반면/ncn )+,/sp
(IP 뿐/nbn+만/jxc 아니/paa+라/ecs )
.....
```

[표 1] 트리부착 말뭉치의 독립구(접속어구만 추출)

접속사에 포함되지 않는 접속어구들은 트리부착 말뭉치에서 문두에 독립구로 태깅된 구들 중 문장 접속 기능을 하는 것만을 추출하였다 [표1]. 각 수사어구들에 대해서는 수사관계와 관계유형을 결정하였다. 관계유형은 선행문 우선관계, 후행문 우선관계, 선-후행 우선관계로 나누어진다 (김태희,1999). 본 논문에서는 선행문 우선관계와 후행문 우선관계만을 고려하였다. [표2]에서 N: S는 선행문 우선관계를 나타내며, S:N은 후행문 우선관계를 나타낸다.

접속어구	관계유형	관계
그러니까	S:N	인과
그래서	S:N	인과, 결론
왜냐하면	N:S	이유
즉	S: N	결론
뿐만 아니라	S:N	강조
예를 들어	N:S	예시
하지만	S:N	전환
따라서	S:N	결론

[표 2] 수사어구 일부

3.2 수사구조 파싱 문법 규칙

문장간 결합 구조를 분석하기 위해서는 간단하게는 다음과 같은 두가지 문법 규칙만을 사용할 수 있다.

- S → s (s: 문장)
- S → SS

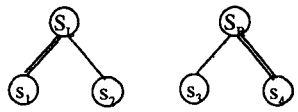
그러나 위와 같은 문법을 사용할 경우, 문장과 문장을 결합할 때 여러 개의 트리가 생성될 수 있다. 이러한 문법의 모호성을 감소시키기 위해서 S를 그 구조의 유형에 따라 다음과 같이 세분화하였다.

- S_R : 우측노드 우선 트리, 후행문 우선 단말노드
- S_L : 좌측노드 우선 트리
- S_l : 선행문 우선 단말노드, 선행문 우선노드로만 이루어진 트리

> s_1[최근 들어 다양한 시각화 기법들이 도입되고 있다.] s_2[*예를 들어*, 구면체에 개체를 늘어 놓거나, 가상 공간 안에 개체들을 표현하는 등의 방법들이 소개되고 있다.] s_3[3차원 공간을 사용한 이러한 방법들은 사용자가 효과적으로 문서의 적합성을 판단할 수 있도록 도와 준다.] s_4[*따라서* 다양한 형태의 시각화 기법이 기존의 단순한 형태의 검색결과를 대체하게 될 것이다.]

[예문 1]

[예문 1]에서 s_1과 s_2의 결합 관계는 s_2 의 수사어구 "예를 들어" 에 의해 예시 관계임을 알 수 있으며 이는 선행문 우선관계이다. 이를 트리로 나타내면 [그림2-(1)]과 같은 형태가 된다. 이때 이중선으로 표시된 노드가 우선함을 나타낸다. 선행문 우선관계를 트리구조로 나타냈을 때, 우선하는 노드가 좌측에 있게 되므로 이를 좌측노드 우선 트리 S_L이라고 한다. [예문1]에서 s_3와 s_4가 결합했을 때는 s_4의 수사어구 "따라서"에 의해 후행문 우선관계임을 알 수 있으며, 이는 [그림2-(2)]와 같은 트리구조를 갖는다. 이때 우선하는 노드가 우측에 있게 되므로 이를 우측노드 우선 트리 S_R이라고 한다.



(1) (2)

[그림 2] 좌측노드 우선트리와 우측 노드 우선트리

S_R의 경우는 우측 노드 우선 트리나 후행문 우선 단말노드를 나타내지만, S_L의 경우는 좌측 노드 우선 트리만을 나타낸다. S_L이 선행문 우선 단말노드를 나타낼 수 없는 이유는 선행문 우선 단말노드의 경우, 후행문 우선 단말노드와 달리, 우선하는 노드를 자신이 포함하고 있지 않고, 단지 선행하는 문장이나 트리가 자신의 노드보다는 우선한다는 것만을 나타내기 때문이다. 세분화된 문장 구조들의 정의를 이용하여 문법 규칙은 9개로 확장된다. 그러나 S_l의 경우 후행문이 S_l일 때를 제외하고는 후행문과의 결합력이 S_l의 선행문과의 결합력보다 약하므로 규칙에서 제외한다. 본 논문에서 사용하는 규칙은 다음과 같다.

- 규칙 1 : $S_L \rightarrow S_L S_L$, $S_R \rightarrow S_L S_L$
- 규칙 2 : $S_L \rightarrow S_L S_R$, $S_R \rightarrow S_L S_R$
- 규칙 3 : $S_L \rightarrow S_R S_L$, $S_R \rightarrow S_R S_L$
- 규칙 4 : $S_L \rightarrow S_R S_R$, $S_R \rightarrow S_R S_R$
- 규칙 5 : $S_L \rightarrow S_R S_l$
- 규칙 6 : $S_L \rightarrow S_L S_l$
- 규칙 7 : $S_l \rightarrow S_l S_l$

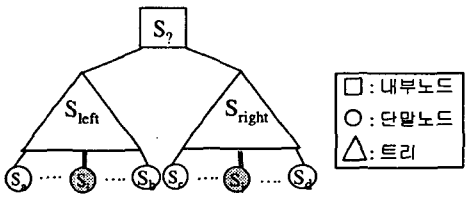
규칙 1~4에서는 S_R과 S_L 이 결합하는 모든 경우에 대해서 수사구조를 결정하는데 모호하게 되므로 좌측 트리 S_{left} 와 우측 트리 S_{right}에서 중요한 문장간에 가중치를 비교하게 된다. 각각의 경우는 다음과 같이 결정된다.

$S_{left}, S_{right} \in \{S_L, S_R\}$
$S_{left} \Rightarrow^* s_a s_{a+1} \cdots s_b$
$S_{right} \Rightarrow^* s_c, s_{c+1}, \ldots, s_d \quad (c=b+1)$
s_i : S_{left}에서 가장 중요한 문장 ($a \le i \le b$)
s_j : S_{right}에서 가장 중요한 문장 ($c \le j \le d$)

$$W(s_i) = Avg \sum_{noun_i \in s_i} w_{noun_i}$$

$$w_{noun_i} = \frac{frequency_{noun_i}}{\sum_{noun_k \in Text} frequency_{noun_k}} \quad \text{식(1)}$$

S_{left}와 S_{right} 는 S_L 또는 S_R의 구조를 가지며, S_{left}는 문장 s_a부터 문장 s_b까지로 이루어지는 수사구조이며, S_{right}는 문장 s_c 부터 s_d까지로 이루어지는 수사구조이다. s_i와 s_j는 S_{left}와 S_{right}의 각 구간에서 가장 중요한 문장이다 [그림3]. S_{left}와 S_{right} 의 결합시에 수사구조를 결정할 수 없을 경우에는 식(1)을 사용하여 s_i와 s_j의 문장의 중요도를 계산하여 $W(s_i) > W(s_j)$이면 결합 결과는 좌측노드 우선 트리가 되고, 반대의 경우 우측 노드 우선 트리가 된다.



[그림 3] 좌측트리와 우측트리의 결합

S_L-S_L' 결합이나 S_L-S_R 결합의 경우 인접한 단말노드 s_b-s_c의 수사관계에 의해서 결합결과가 결정될 수 없으므로 $W(s_i)$와 $W(s_j)$를 비교하여 $W(s_i) > W(s_j)$ 이면 결과가 S_L이고 반대의 경우 S_R이다. S_R-S_L 결합 또는 S_R-S_R' 결합의 경우 i=b 일때, 즉 마지막 문장이 가장 중요한 문장일 경우, 단말노드 s_b-s_c의 수사관계에 의해서 결합 결과가 결정되므로 결합 결과는 S_R이고 i≠ b 일때는 $W(s_i)$와 $W(s_j)$를 비교하여 결정한다.

규칙 5~7은 우측 노드가 S_i인 경우로 선행문이 우선함을 나타내므로 결합의 결과는 S_L 또는 S_i이 된다.

문단간 파싱의 경우 첫 문장의 수사어구가 그 문단의 수사 관계를 나타낸다고 가정하고 문장간 파싱을 위한 문법과 동일한 문법을 사용하며, 기본 단위는 문장이 아닌 문단이 된다. 즉, 문단 파싱의 경우 하나의 문단을 문장으로 보고 위 문법을 똑같이 적용하는 것과 같다.

3.3 수사구조 파싱

수사구조 파싱은 위에서 기술한 문법을 사용하여 문장간 수사구조 파싱, 문단간 수사구조 파싱, 두 단계로 이루어진다. 원문 텍스트는 문단 구분이 되어 있는 것을 전제로 하며, 문장간 파싱이 끝난후 문단간 파싱을 진행한다. 문단간 수사구조 파싱은 문장간 파싱과 동일하게 진행되며 기본 단위는 문단이 된다. 수사구조 파싱은 차트파싱 방법을 사용하여 상향식으로 진행되며 파싱결과로서 이진트리가 생성된다.

파싱 알고리즘은 일반적으로 사용되는 차트파싱 방법과 같으나, 차트의 같은 범위 내에서는 하나의 구조만을 기록하는 것을 원칙으로 한다. 이를 위해서는 차트에 완성된 구조를 기록할 때 중복된 범위 내에 이미 완성된 구조가 있을 때 좌측 트리와 우측 트리의 인접 부분에서의 선-후행문의 유사도를 계산하여 유사도가 작은 쪽의 구조를 선택한다. 이는 일련의 문장들을 두 부분으로 나눌 때 몇 번째 문장에서 분할을 할 것인가를 결정하는 것인데, 두 가지 후보 중에서 경계부분의 문장간의 유사도가 작은 쪽에서 분할되는 것이 문장의 흐름을 구조에 반영하는데 더 적합하기 때문이다. 문장간 유사도를 계산하기 위해서 dice coeffeicent를 사용한다.

차트 파싱 알고리즘은 자료구조로서 차트와 키 리스트를 갖는다. 차트는 입력 문장들로부터 규칙을 적용하여 구해진 모든 완성, 미완성의 구조들에 대한 기록을 갖게 된다. 적용된 규칙이 $C \rightarrow C_l C_r$ 일 때, 완성된 구조의 경우 $<C, C_l, C_r, P_i, P_j, P_k, s_i>$ 의 구조를 갖는다. 이때, P_i, P_j는 완성된 구조의 구간 범위이다. P_k는 결합관계가 이루어진 위치를 나타내며 0일때는 미완성 구조임을 나타낸다. s_i는 완성된 구간에서 가장 중요한 문장번호를 나타낸다.

키 리스트는 차트에 기록되었으나, 아직 파서에 의해 처리되지 않은 모든 완성구조의 문법 기호를 $<C, P_i, P_j, s_i>$ 형태로 가지고 있다. C는 완성구조에 최종적으로 적용된 문법 규칙 좌변기호이며, P_i, P_j는 C가 해당하는 구간의 범위이며, s_i는 이 구간에서 가장 중요한 문장 번호이다. 단말 노드일 경우는 자신의 문장번호를 갖는다. 문장간 파싱은 다음과 같은 알고리즘에 의해 수행된다.

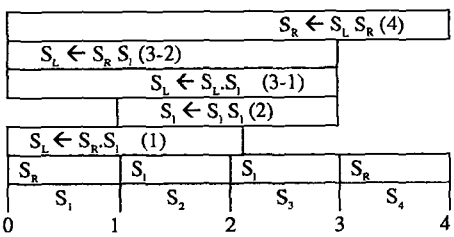
1 키 리스트가 비어있으면 다음 문장을 읽고 새 항목을 키 리스트에 넣는다.
2 차트 파서는 키 리스트의 한 항목을 얻는다.
  2.1 문법규칙 중($C \rightarrow C_l, Cr_r$) 좌측의 C_l과 항목의 기호가 일치하는 모든 규칙에 대해 미완성 구조를 차트에 추가한다.
  2.2 차트의 모든 미완성 구조중에서 C_r이 항목의 기호와 일치하여 완성되는 구조가 있으면, 이를 완성한다.

2.2.1 같은 범위에 대해 이미 완성된 구조가 존재하지 않으면 새로 완성된 구조를 차트에 기록하고 키 리스트에 이 항목를 추가한다.

2.2.2 이미 완성된 구조가 있을 때, 결합부분에서 인접한 문장끼리의 유사도가 존재하고 있는 구조의 것보다 작다면, 이미 존재하는 구조를 새로운 구조로 대체하고 키 리스트에서도 이를 대체한다.

다음은 [예문1]에 대한 수사구조 파싱의 진행과정이다. 문장s_1~s_4의 수사관계는 수사어구에 의해 결정된다. 문장 s_i에 수사어구가 존재하지 않을 때는 식(1)에 의해서 $W(s_{i-1}) > W(s_i)$ 이면 S_I이 되고 반대의 경우 S_R이 된다. 첫번째 항목은 선행문장이 없으므로 항상 S_R로 한다.

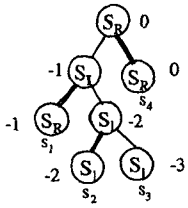
키리스트의 첫 항목 <S_R,0,1,1>에 대해서 우변의 첫번째가 S_R인 규칙에 의해서 만들어지는 모든 미완성 구조를 차트에 기록한다. 키 리스트의 다음 항목인 <S_I,1,2,2> 에 대해서 적용될 수 있는 규칙을 기록하고, 이미 차트에 기록된 항목 중에서 완성될 수 있는 항목, 즉, 우변에 S_I를 필요로 하는 미완성 구조에 대해서 이를 완성하고 차트에 기록한다. 차트항목(1) 은 규칙5 를 적용해서 완성된 구조이다. [그림4]에서는 완성된 구조들만을 나타내고 있으며, 각 번호는 생성된 순서이다. <0,3>의 범위에서 (3-1), (3-2) 의 완성구조가 생성되나, 좌측 트리와 우측 트리의 인접문장에서의 유사도가 (3-2)구조일 때 더 낮기 때문에 구간 <0,3>의 완성구조로 (3-2)가 선택된다. [예문1]로부터 만들어진 수사구조 트리는 [그림5]와 같다.



[그림 4] 차트파싱 진행과 차트

3.4 가중치 계산 및 요약문 생성

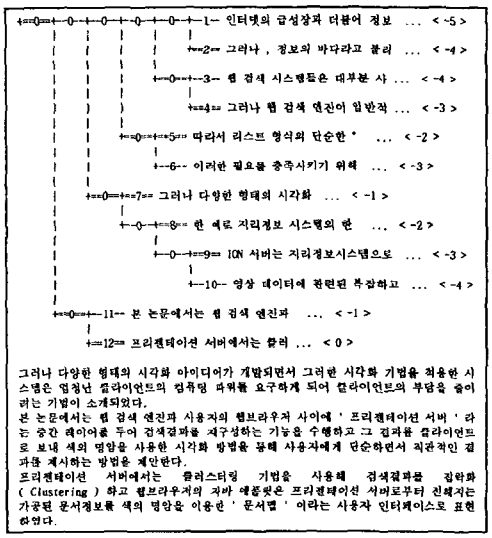
원문 텍스트로부터 생성된 수사구조로부터 중요문장 추출을 위해서 가중치를 계산한다. 가중치는 벌점을 부여하는 방법을 사용한다. 수사구조 트리의 루트로부터 시작하여 가중치를 계산하는데, 루트를 0으로해서 해당 노드의 수사관계 유형에 따라 벌점을 부가한다. 좌측 노드 우선 트리의 경우 우측 노드에 벌점이 부가되며, 우측 노드 우선 트리의 경우 좌측에 벌점이 부가된다. 자식노드의 가중치는 부모노드의 가중치와 자신의 벌점을 더한 값이 된다 [Ono,1994]. [그림5]는 앞에서 생성된 트리에 가중치가 계산된 결과이다. 사용자가 -1수준까지의 요약문을 요구한다면, 요약문으로 추출되는 문장은 s_1, s_4 이다.



[그림 5] 가중치가 부가된 수사구조 트리

4 실험 및 결과

실험에 사용한 데이터는 정보과학회와 인지과학회 논문 30건으로 요약수준은 사용자가 설정할 수 있다. 요약의 결과로서 [그림6]과 같은 수사구조 트리와 요약문을 생성한다. [그림6]에서 이중선으로 표시된 부분이 우선되는 노드이며, 단말노드에 표시된 점수가 각 문장의 가중치이다.



[그림 6] 생성된 수사구조 트리와 요약문의 예

요약실험은 논문의 초록을 제외한 전문을 대상으로 하였으며, 자동으로 생성된 요약문과 초록의 유사도를 계산하여 평가하였다. 비교실험으로서 같은 양의 문장을 무작위 추출하였을 때와 단어 빈도수만을 사용하였을 때의 요약문을 생성하였다.

유사도는 dice coeffecient 를 이용하여 계산한 값으로 0에서 1사이의 값을 갖는다. 실험데이타의 평균 문장수는 88 문장이며 초록의 평균 길이는 4.7 문장이다. 자동으로 생성된 요약문의 길이는 벌점 -1 수준에서 평균 6.8 문장이다. [표3] 은 벌점 수준 -1, -2, -3 수준에 대해 같은 수의 문장을 무작위로 추출했을 경우와 단어빈도수에 따른 문장 가중치에 의해 추출했을 경우의 유사도 비교 결과이다.

벌점수준 (문장수)	-1 (6.8)	-2 (12.9)	-3 (19.1)
빈도수 이용	0.21	0.20	0.21
무작위 추출	0.14	0.15	0.15
수사구조이용	0.24 +15% +69%	0.23 +16% +58%	0.18 -12% +24%

[표 3] 유사도 비교

실험결과 같은 양의 요약문을 생성할 경우 벌점이 작을수록 유사도가 증가했으며, 요약문의 문장수가 증가할 경우는 무작위로 문장을 선택하여 요약문으로 추출했을 때와 유사도 차이가 감소하는 것을 알 수 있다. 요약문으로 추출하는 문장이 많아질 경우 벌점 수준-3 에서는 빈도수를 이용했을 때보다 유사도가 낮게 된다.

5 결론 및 향후계획

본 논문에서는 수사구조를 이용한 자동 요약 시스템을 구현하였다. 텍스트의 수사구조 파싱을 하기 위해 문장간 결합에 대한 문법 규칙을 정의한 후 이를 이용하여, 차트파싱 방법으로 전체 텍스트의 수사구조 트리를 생성하였다. 생성된 트리로부터 가중치를 부여하여 주어진 수준에 따라 중요 문장들을 요약문으로 추출하였다. 생성된 요약문은 무작위 추출된 요약문보다는 원문의 초록과 높은 유사도를 보이고 있으나, 요약문의 문장수가 많아질 경우 빈도수를 이용한 요약문 생성보다 낮은 유사도를 보이고 있다.

본 논문에서는 수사어구를 접속어구, 지시어에 한정했으나, 다양한 텍스트에서 쓰이는 수사어구들과 조응어등을 사용하여 수사구조 트리를 생성한다면, 결합관계의 애매성이 감소하여, 보다 정확한 수사구조를 생성할 수 있을 것이다.

6 참고문헌

[1] 김태희, 박혁로, 신중호, 송인석, "자동 텍스트 요약에 있어서 인지적 요소에 대한 고찰", 한국인지과학회,1999

[2] 강상배, 조혁규, 권혁철, 박재득, 박동인, "한국어 문서의 통계적 정보를 이용한 문서 요약 시스템 구현", 제 9 회 한글 및 한국어 정보 처리,pp28-33,1999

[3] 장동현, 맹성현, "문서 구조 정보를 이용한 확률 모델 기반 자동 요약 시스템", pp15-22, 1997

[4] 장재성, "문장 표현 사전",박문각,1998

[5] Barzilay, Regina and Michel Elhadad, "Using Lexical Chains for Text Summarization",In Proceedings of the ACL/EACL'97 Workshop on Intelligent Scalable Text Summarization,1997

[6] Ono, Kenji, Kazuo Sumita, Seiji Miike, "Abstract Generation bsaed on Rhetorical Structure Extraction, In Proceedings of the International Conference on Computational Linguistics, pp344-348,1994

[7] Lin, Chin-Yew and Eduard Hovy, "Identifying topics by position", In Proceedings of the 5th. Conference on Applied Natural Language Processing,pp283-290, 1997

[8] Marcu, Daniel. "Building up Rhetorical Structure trees", In Proceedings of the 13th National Conference on Artifical Intelligence, pp1069-1074, 1996

[9] Marcu, Daniel. "From discourse structure to text summaries", In Proceedings of ACL/EACL,1997