그래프 분할을 이용한 문장 클러스터링 기반 문서요약

이 일 주[†] · 김 민 구[††]

요 약

문서요약은 여러 개의 하위 주제로 구성되어 있는 문서에 대해 문서의 복잡도를 줄이면서 하위 주제를 모두 포함하는 요약문을 생성하는 것이 목적이다. 본 논문은 그래프 분할을 이용하여 하위 주제별로 중요 문장을 추출하는 요약시스템을 제안한다. 문장별 공기정보에 의한 단어의 연관성 분석을 통해 선정된 대표어를 이용하여 문서를 그래프로 표현한다. 그래프는 연결정보에 의해 하위 주제를 의미하는 부분 그래프로 분할되며 부분 그래프는 긴밀한 관계를 갖는 문장들이 클러스터링된 형태이다. 부분 그래프별로 중요 문장을 추출하면 하위 주제별 핵심 내용들로만 요약문을 구성하게 되어 요약 성능이 향상된다.

키워드 : 문서요약, 단어 연관성, 문장 클러스터링, 그래프 분할

Document Summarization Based on Sentence Clustering Using Graph Division

Iljoo Lee[†] · Minkoo Kim[††]

ABSTRACT

The main purpose of document summarization is to reduce the complexity of documents that are consisted of sub-themes. Also it is to create summarization which includes the sub-themes. This paper proposes a summarization system which could extract any salient sentences in accordance with sub-themes by using graph division. A document can be represented in graphs by using chosen representative terms through term relativity analysis based on co-occurrence information. This graph, then, is subdivided to represent sub-themes through connected information. The divided graphs are types of sentence clustering which shows a close relationship. When salient sentences are extracted from the divided graphs, summarization consisted of core elements of sentences from the sub-themes can be produced. As a result, the summarization quality will be improved.

Key Words : Document Summarization, Term Relation, Sentence Clustering, Graph Division

1. 서 론

인터넷 등의 발전으로 정보량이 대량으로 증가하고 있는 상황에서 사용자들은 정보를 효율적으로 관리하고 검색하는 문제가 중요하게 되었다. 이러한 문제에 대해 효과적인 해결책을 제시해 줄 수 있는 방법으로 문서요약이 요구된다. 문서요약이란 문서의 기본적인 내용을 유지하면서 원문으로부터 가장 의미 있는 내용만을 추려내어 문서의 길이를 줄이는 작업을 의미한다[1]. 문서요약은 요약방법에 따라 생성요약과 추출요약으로 나눌 수 있다[2]. 생성요약은 전체 문서의 내용을 압축하여 새로운 문서를 작성하는 작업으로서 문서를 이해하는 분석단계와 변형단계, 그리고 자연언어로 문장을 생성하는 통합단계를 거치는 전문적인 요약이다. 반면에 추출요약은 문서에 존재하는 단어나 구, 문장, 단락 등을 그대로 추출하여 요약으로 제시한다. 따라서 요약문의 가독성은 다소 부족하지만, 새로운 텍스트를 생성하는 일련의 과정을 포함하지 않기 때문에 생성요약에 비해 비교적 쉽게 요약기술에 접근할 수 있다는 특징을 가진다. 추출요약에 있어 중요한 문제는 만약 어떠한 문서가 여러 개의 하위 주제로 구성되어 있는 경우 이러한 하위 주제를 모두 포함하는 중요 문장들을 어떻게 선정할 것이냐 하는 점이다. 이를 위해 문장간의 유사도를 기반으로 하는 문장 클러스터링에 의한 대표 문장 추출이 많이 이용되었다. 그러나 문장 클러스터링에 의한 요약방법은 유사도 계산의 오류로 인하여 같은 하위 주제를 나타내는 문장들이 중복해서 선정될 가능성과 중요 문장의 탈락과 같은 잘못된 클러스터링 결과를 보일 수 있다.

본 논문에서는 기존의 문장 클러스터링에 의한 요약방법이 가지고 있는 문제점을 줄이고 문서의 개념을 요약문에 최대한 반영하고자 한다. 이를 위해 그래프 분할 기법에 의한 하위 주제별

† 정 회 원 : 동원대학 모바일컨텐츠과 부교수
†† 정 회 원 : 아주대학교 정보 및 컴퓨터공학부 교수
논문접수 : 2006년 1월 23일, 심사완료 : 2006년 3월 6일

문장 클러스터링을 수행한 후 중요 문장을 추출하는 문서요약을 제안한다. 우선 문서의 개념을 표현하는 대표어를 선정하기 위해 문장별로 동시에 출현한 단어의 공기정보를 이용한다. 문서의 단어들은 문서가 작성된 의도에 따라 서로 의미적으로 연관된 형태로 형성된다는 점에서, 단어의 공기정보를 이용한다면 문서의 의미를 나타내는 핵심 단어들을 대표어로서 선정할 수 있다. 선정된 대표어를 기반으로 문장 벡터를 구성하며, 문장 벡터 사이의 공통 대표어를 이용하여 문장들의 연결성을 측정한 후 문장들을 노드(node)로 하는 그래프를 형성한다. 즉, 문장 간 단어의 연결정보에 의해 모든 문장들은 상호 독립적인 성분에 의해 하위 주제별의 단절 그래프(Disconnected Graph) 형태로 구성된다. 문장들을 보다 정확한 하위 주제로 구분하기 위해 그래프는 몇 개의 부분 그래프(subgraph)로 다시 분할된다. 이렇게 부분 그래프로 분할하는 이유는, 많은 단어들로 이루어진 긴 문장인 경우 여러 개의 서로 다른 하위 주제의 부분 그래프를 서로 연결하는 역할을 할 수 있기 때문이다. 따라서 이러한 연결점을 찾아서 하위 주제별로 구분되는 부분 그래프로 나누는 작업이 매우 중요하다. 그래프 분할은 결과적으로 유사한 내용을 표현하고 있는 문장들이 클러스터 단위로 분류가 되게 하며, 최종적으로 각 부분 그래프의 중요 문장을 추출함으로써 요약문을 작성하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서 관련 연구에 대해 기술하며, 3장에서는 그래프 분할에 의한 문서 요약을 설명한다. 4장에서 제안된 시스템의 성능을 평가하고 5장에서 결론을 맺는다.

2. 관련 연구

초기의 추출요약은 주로 문장들에게 가중치를 부여하고 가중치가 높은 문장을 중요 문장으로 추출하는 방법이었다. 대표적인 연구는 에드먼슨이 문서의 4가지 자질(실마리어, 제목, 키워드, 위치)에 대한 가중치를 이용해 각 문장에 점수를 부여하고 이를 통해 문장을 추출한 방법이다[3]. 그러나 이러한 방법들은 단순히 문장의 가중치만 고려했기 때문에 비슷한 내용의 문장들이 중복 추출되기도 하고 전체적인 문서의 흐름을 고려하지 않아 결합력이 약한 요약문을 완성한다는 문제점이 있다.

최근의 연구로서는 문서를 몇 개의 문단으로 구분한 후 중요 문장을 추출하는 방법이 있다. 작은 하위 주제들이 어디에서 시작되고 어디까지 이어지는 지에 대한 경계정보를 알아낸 후, 하위 주제의 이동에 따라 문서를 여러 개의 문단으로 구분하는 방법으로서 토픽분할의 개념을 응용한 것이라 할 수 있다. 잡지와 같은 긴 문서를 대상으로 하위 주제의 구분에 따라 문서를 여러 개의 단락으로 분할하는 Hearst의 TextTiling[4]이 토픽분할의 대표적 연구이다. 반면에 Salton[5]은 문단별 유사도 관계 그래프를 이용하여 요약문을 완성하였다. 문서를 여러 개의 문단으로 구분한 후 Bush path, Depth-first path, Segmented Bushy path를 이용하여 중요 문장을 추출하는 방법이다. 문단 단위의 중요 문장 추출은 문서의 크기가 큰 경우 효율적일 수 있으나, 문서의 크기가 작은 경우 그 성능이 저하되며 문단에 포함된 중요 문장뿐만 아니라 불필요한 문장도 함께 추출하는 단점이 있다[6, 7].

지금까지의 추출요약에서는 문장 클러스터링을 이용하여 중요 문장을 추출하는 방법이 널리 사용되고 있다. 이것은 문장간의 유사도 계산에 의해 문서의 문장들을 몇 개의 하위 주제별 문장 클러스터로 구분한 후 클러스터별로 가중치가 높은 문장을 추출하는 방법이다. 정영미[8]는 문장 클러스터로부터 대표 문장을 선정하여 요약문을 생성하는 자동요약모델을 제시하고 학습과정에서의 문장 클러스터링 방법으로 7개의 기법들을 비교하였다. 그러나 현재의 유사도 계산에 의한 문장 클러스터링 방법은 잘못된 대표어로 인해 문장간 유사도 계산 시 오류가 발생할 수 있다는 문제점을 가지고 있다. 즉, 주제와 무관한 단어들의 중복만으로도 문장간의 유사도 값이 높아지게 된다. 본 논문에서는 문장 클러스터링 개념은 이용하되, 문장의 유사도 계산에 의한 방법 대신 그래프 분할이라는 개념을 이용하여 클러스터링을 수행한 후 중요 문장을 추출하고자 한다.

3. 문서요약

3.1 단어의 공기정보 연관성 분석을 통한 대표어 추출

단순히 통계정보에 의한 대표어를 이용하여 문서를 표현하면 각 단어들은 서로 독립적으로 존재하게 되므로 원래 문서가 가지고 있는 정보의 의미를 손실한다는 단점이 있다[9]. 그러므로 통계정보의 직접적인 적용은 일반적으로 요약 시스템의 성능을 오히려 저하시킬 수 있다[10]. 또한 문장의 유사도 계산 시 문서내의 모든 단어들을 대상으로 할 경우 대표어가 일치하지 않더라도 비주제어들로 인하여 오류가 발생할 가능성이 높다. 따라서 대표어를 추출하는 기법은 문서요약시스템을 구현하기 위한 전 단계로서 매우 중요한 과정이다.

대표어와 비주제어의 구분을 보다 정확하게 하기 위해서는 단어의 의미적 정보를 필요로 한다. 단어의 의미정보를 파악하기 위해 문서의 수사구조나 단어들의 사전적 관계, 파싱트리를 기반으로 하는 구문론적 관계를 이용할 수 있다. 수사구조나 구문론적 방법을 이용하기 위해서는 별도의 복잡한 작업을 필요로 하는 여분의 오버헤드(overhead)가 존재한다. Barzilay와 Elhadat는 WordNet과 파서를 이용하여 텍스트상의 어휘사슬(Lexical Chain)을 식별하였다[11]. 그러나 WordNet과 같은 사전적 관계를 이용하기 위해서는 별도의 어휘사전이나 시소러스사전이 필요하며, 또한 사전에 등록되지 않은 단어에 대해서는 정의를 할 수 없다는 문제점이 있다. 별도의 사전을 사용하지 않고 단어의 의미적 정보를 추출하기 위해 단어의 공기정보(Co-occurrence)를 이용하는 방법이 있다. 공기정보란 두 단어가 동일문서, 문장, 구 등에 같이 발생하는 현상을 말하며 더 자주 발생 할수록 두 단어가 밀접한 관계를 가지고 있다는 전제에 기반하고 있다[12]. 본 논문에서는 단어의 공기정보를 기반으로 단어의 연관성 분석을 통해 문서의 대표어를 추출한다.

단어간의 연관성을 분석해서 단어와 단어 사이의 의미적 관계를 찾아내는 문제는 두 단어 T_i와 T_j가 주어졌을 때 두 단어 사이의 성립 가능한 관계를 찾는 작업으로 정의 할 수 있다. 이러한 동시출현 단어의 연관도는 공기 빈도를 직접 사용할 수도 있고, 코사인 유사도(Cosine Similarity)나 자카드계수(Jaccard Coefficient) 등을 이용하여 정규화 할 수도 있다. 본 논문에서는 지지도(Support),

신뢰도(Confidence), 향상도(Lift/Improvement)를 이용하여 의미 관계의 성립여부를 정의한다.

지지도는 유효한 통계적 추론을 위해 필요한 실질적인 인스턴스의 수로서 전체 문장 중 두 단어의 동시출현 문장의 수를 나타낸다. 또한 적어도 어느 정도 이상의 출현수가 일어난 경우들만을 고려대상으로 하기 위해 최소 지지도를 사용한다. 따라서 빈발하지 않은 단어 집합들은 연관성 분석에서 제외된다. 이 경우 고정된 문장수를 이용하게 되면 최소 지지도의 적용의미가 문서의 길이에 따라 달라질 수 있기 때문에 본 논문에서는 동시출현 비율을 사용한다. 식 (1)과 같이 전체 문장 수 대비 두 단어의 동시출현 문장수의 비율을 0.03으로 하였다. 이 수치는 실험을 통해 얻어낸 결과이다. 동시출현 문장수의 비율이 최소 지지도 미만인 경우에는 우연히 공기한 경우로서 의미가 없다고 판단한다.

• 최소 지지도

$$MinSupport(T_i, T_j) = \frac{\text{단어 } T_i\text{와 } T_j\text{가 동시에 포함된 문장의 수}}{\text{전체 문장의 수}} = 0.03 \quad (1)$$

신뢰도는 단어 T_i를 포함하는 전체 문장 중에서 단어 T_j가 포함된 문장의 비율을 말하며 조건부 확률 $P(T_i \mid T_j)$를 이용하여 신뢰도의 정도를 계산한다. $Confidence(T_i \mid T_j)$는 단어 T_j가 단어 T_i와의 공기 빈도수가 많으면 값이 커지므로 단어 T_j가 단어 T_i에 얼마나 종속적인가를 나타낸다. 반대로 $Confidence(T_j \mid T_i)$는 단어 T_i의 단어 T_j에 대한 의존도를 나타낸다. 식 (2)와 식 (3)의 두 가지 조건 확률은 두 단어 간의 공기 빈도수 크기에 따라 신뢰도가 서로 다르다는 점을 의미한다.

• 신뢰도

$$Confidence(T_i \mid T_j) = P(T_i \mid T_j) = \frac{P(T_i \cap T_j)}{P(T_i)} = \frac{\text{단어 } T_i\text{와 } T_j\text{가 동시에 포함된 문장의 수}}{\text{단어 } T_i\text{가 포함된 문장의 수}} \quad (2)$$

$$Confidence(T_j \mid T_i) = P(T_j \mid T_i) = \frac{P(T_j \cap T_i)}{P(T_j)} = \frac{\text{단어 } T_j\text{와 } T_i\text{가 동시에 포함된 문장의 수}}{\text{단어 } T_j\text{가 포함된 문장의 수}} \quad (3)$$

향상도는 단어 T_i에 상관없는 단어 T_j의 확률 대비, 단어 T_i가 주어졌을 때 단어 T_j의 확률의 증가 비율로 이 값이 클수록 단어 T_i의 출현여부가 단어 T_j의 출현여부에 큰 영향을 미친다. 단어 T_i와 단어 T_j의 출현여부가 상호 관련이 없다면 $P(T_j \mid T_i)$와 $P(T_j)$가 같게 되어 향상도가 1이 된다. 향상도가 1보다 크다면 단어의 연관성을 예측하는 데 우연적 기회보다 우수하고, 향상도가 1보다 작다면 연관성을 예측하는 데 우연적 기회보다 나쁘다는 것을 의미한다. 반대로 단어 T_j에 상관없는 단어 T_i의 확률 대비, 단어 T_j가 주어졌을 때 단어 T_i의 확률의 증가 비율 역시 값이 클수록 단어 T_j의 출현여부가 단어 T_i에 대해 큰 영향력을 나타낸다. 식 (4)와 식 (5)의 두 가지 조건부 확률은 두 단어의 빈도수 크기에 따라 향상도가 다르다는 점을 의미하며 두 가지 조건부 확률이 모두 1보다 커야 두 단어의 공기정보가 의미 있다고 판단한다.

$$Lift(T_i \mid T_j) = \frac{P(T_i \cap T_j)}{P(T_i) \cdot P(T_j)} = \frac{P(T_i \mid T_j)}{P(T_i)} = \frac{Confidence(T_i \mid T_j)}{P(T_i)} \quad (4)$$

$$Lift(T_j \mid T_i) = \frac{P(T_j \cap T_i)}{P(T_j) \cdot P(T_i)} = \frac{P(T_j \mid T_i)}{P(T_j)} = \frac{Confidence(T_j \mid T_i)}{P(T_j)} \quad (5)$$

3.2 대표어 가중치

대표어별 가중치는 각 노드에 인접한 노드들과의 의미유사도의 합인 도합유사도[13]로 표현한다. 본 논문에서 노드는 대표어를 의미하며 연관성이 있다고 판단된 다른 대표어와 많은 연결선을 가진 단어를 중요한 단어로 인식한다. 대표어의 가중치는 그 문장을 특징짓는 문장 벡터를 구성한다. 대표어별 가중치 값은 두 대표어 사이의 연결 개수를 식 (6)과 같이 의미유사도로 계산한 후 유사도의 합인 식 (7)의 도합유사도 값으로 표시한다. 단어의 연결 개수를 그대로 사용하면 낮은 빈도의 연결수를 갖는 단어와 높은 연결수를 갖는 단어사이의 가중치가 과도한 차이를 보이며 이 현상을 막기 위해 log를 취한 값을 사용한다. 예를 들어, "attack" 이라는 단어가 "government"와 2개, "country"와 3개, "kingdom"과 2개의 연결 개수를 갖는다면, 이 단어의 가중치는 0.69와 1.09, 0.69의 합인 2.47이 된다.

• 두 단어 T_i와 T_j의 의미유사도

$$Sim(T_i, T_j) = \log(Frequency(T_i, T_j)) \quad (6)$$

$Frequency(T_i, T_j)$: 두 단어 T_i와 T_j의 연결 개수(동시 출현 문장 수)

• i번째 단어의 도합유사도

$$asim(T_i) = \sum_{\substack{j=1 \\ j \neq i}}^{n} Sim(T_i, T_j) \quad (n\text{: 연관단어의 개수}) \quad (7)$$

식 (7)에 의해 계산된 대표어의 가중치를 이용하여 <표 1>과 같이 문서D는 문장 벡터의 집합으로 표현하며 문장 벡터 S_i (i=1..12)는 대표어와 가중치로 표현한다.

<표 1> 문서 D의 벡터표현 예

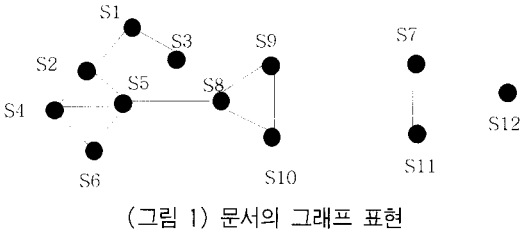
D = { S_1, S_2, S_3, S_4, S_5, S_6, S_7, S_8, S_9, S_{10}, S_{11}, S_{12} }
S_1 = { (Cambodian, 1.38), (HunSen, 1.78) }
S_2 = { (Cambodian, 1.38), (government, 3.85) }
S_3 = { (HunSen, 1.78), (law, 1.09 }
S_4 = { (government, 3.85), (Kingdom, 3.69), (attack, 2.47) }
S_5 = { (government, 3.85), (Kingdom, 3.69), (country, 1.38), (policy, 0.69)}
S_6 = { (attack, 2.47), (country, 1.38) }
S_7 = { (opposition, 1.09), (nation, 1.09) }
S_8 = { (policy, 0.69), (negotiation, 1.78) }
S_9 = { (negotiation, 1.78) }
S_{10} = { (negotiation, 1.78) }
S_{11} = { (nation, 1.09), (opposition, 1.09) }
S_{12} = { (abroad, 0.69), (medicine, 0.69), (doctor, 0.69) }

3.3 그래프 분할을 이용한 문장 클러스터링

문서의 구성형태는 일반적으로 작은 하위 주제가 여러 개 모여서 하나의 문서를 이루는 경우가 대부분인데, 비슷한 주제에 대한 언급이 있는 문장들은 구성 단어 또한 유사하므로 이러한 단어의 유사성을 바탕으로 문장들을 클러스터링 하고자 한다. 하위 주제별로 문장들을 클러스터링 하기위해 중요한 점은 각 하위 주제의 특징을 표현하는 주제 범주들을 어떻게 구분할 것이냐 하는 점이다. 이를 위해 본 논문에서는 그래프 분할을 이용하여 주제 범주를 구분한다. 문서를 그래프로 표현하는 방법은 문서의 구조를 단어와 단어들이 연결되는 그래프 구조로 문서를 파악한다는 점이다[14]. 예를 들면, <표 1>의 문장 S_1과 S_2는 공통 단어인 'Cambodian'에 의해 연결되고, 문장 S_1과 S_3는 'HunSen'에 의해 서로 연결된다. 모든 문장들에 대해 이와 같이 공통단어를 이용하여 연결을 하게 되면 문서를 단절 그래프 형태로 표현할 수 있다. 단절 그래프의 연결요소(Connected Component)의 개수는 문서가 포함하고 있는 하위 주제의 개수에 따라 1 개 혹은 여러 개가 될 수 있으며 각각의 주제 범주들은 서로 다른 주제 범주의 단어들은 포함하지 않게 되는 직교(orthogonal)의 형태가 된다. (그림 1)은 <표 1> 문서의 문장들을 그래프로 표현한 모습이다.

결과적으로 각 클러스터들은 관련 문장들에 의한 무방향 연결 그래프 구조(Undirected Connected Graph)를 갖게 된다. 여기서 노드는 문장을 나타내고 노드사이의 에지(edge)는 문장 간에 단어가 중복되어 나타나는 것을 연결한 것이다. 강하게 연결된 그래프는 여러 개의 문장이 하나의 주제에 대한 언급을 하고 있는 것으로 볼 수 있는 반면 일직선을 나타내는 그래프는 문장 사이에 커다란 연관관계는 없다는 것을 나타내 준다. 이 경우 만일 문서가 한 개의 단일 주제로 구성되어 있다면 단지 한 개의 클러스터만 생성될 것이고, 만약 여러 개의 독립적인 하위 주제로 구성되어 있다면 여러 개의 클러스터들이 생성될 것이다. <표 2>는 (그림 1)을 하위 주제별 문장 클러스터로 표현한 결과이다.

그러나 문서의 내용에 따라 차이는 있겠지만 한 클러스터가 독점적으로 커지면 대부분의 중요 문장들이 한 곳에 몰리게 되고 결과적으로 중요 문장들이 대표 문장으로 선정될 가능성이 줄어든다. 또한 여러 개의 하위 주제들을 모두 표현할 수가 없는 경우도 발생하게 된다. 이를 해결하기 위해 공통 대표어에 의해 연결된 그래프를 다시 몇 개의 부분 그래프로 분할한다. 즉 하나의 클러스터는 몇 개의 하위 주제별 클러스터로 다시 한번 분리가 가능하다는 의미이다. 부분 그래프로 분할하기 위해서는 그래프가 가지는 중요한 특성들 가운데 하나인 이음새(Connectivity)를 이용한다. 이음새 란, 이어진 그래프에서 어떤 노드는 떼어내면 그래프가 끊어지고, 어떤 노드는 떼어내도 그래프는 이어진 대로 남는 특성을 말한다[15]. 이와 같이 그래프에서 어떤 노드를 제거하면 그래프가 비 연결로 되는 노드가 있을 수 있는데, 이와 같은 노드를 articulation point, 혹은 cut-vertex 라 부르며, 이 노드들은 부분 그래프들을 연결하는 점으로 전체 그래프 구조에서 중요한 역할을 한다. 서로 다른 하위 주제들로 구성된 각각의 부분 그래프에 속한 문장들은 자체만의 특성에 의해 타 부분 그래프에 속하는 문장들과의 독립성을 유지하게 된다. 그러나 많은 단어들로 이루어진 긴 문장과 같이, 어떤 문장들은 여러 개의 서로 다른 주제영역의 부분 그래프를 서로 연결하는 articulation point 역할을 할 수 있다. 따라서 이러한 articulation point를 이용하여 그래프 분할을 한다면, 보다 효율적인 클러스터링 결과를 기대할 수 있게 된다. 본 논문에서는 그래프를 분할하기 위해 그래프내의 모든 articulation point를 찾아낸 후, 이 노드를 기준으로 인접노드들이 서로 다른 부분 그래프에 속하게 하는 방법으로 그래프를 분할한다. articulation point를 찾는 방법은 깊이 우선 탐색(depth-first search) 알고리즘을 이용한다.



(그림 1) 문서의 그래프 표현

〈표 2〉 하위 주제별 문장 클러스터

클러스터	구성 문장
C_1	{ S_1, S_2, S_3, S_4, S_5, S_6, S_8, S_9, S_{10} }
C_2	{ S_7, S_{11} }
C_3	{ S_{12} }

• articulation point 탐색알고리즘

1. 그래프로부터 임의의 node r 선택
2. r 로 시작하는 트리T를 구성하기 위한 depth-first search 수행
3. 트리 T의 r 이 2개 이상의 child node를 갖게 되면 r 은 articulation point
4. 트리 T의 root가 아닌 node v 의 child node u 의 어느 descendant node 로부터 v 의 ancestor node를 가리키는 back edge가 없으면 v 는 articulation point

S_5로 시작한다면 (그림 1)에서 S_1, S_5, S_8이 articulation point이다. articulation point는 여러 개의 부분 그래프에 공통으로 포함되게 되는 데, 각각의 부분 그래프를 구성하고 있는 문장들과의 유사도 계산에 의해 articulation point가 포함 되어야할 부분 그래프를 결정한다. 이 때 (그림 1)의 S_{12}와 같이 무연결점(Missing Connection), 즉 단일노드(단일문장)에 의한 특별한 형태의 그래프가 형성될 수 있으며, 이 경우에는 문서의 내용을 대표하는 중요 문장이라고 보기에는 어렵기 때문에 대표 문장 선정대상에서 제외한다. <표 3>은 그래프 분할을 수행한 후의 최종 문장 클러스터를 나타낸다.

〈표 3〉 최종 문장 클러스터

클러스터	구성 문장
C_1	{ S_1, S_3 }
C_2	{ S_2, S_4, S_5, S_6 }
C_3	{ S_8, S_9, S_{10} }
C_4	{ S_7, S_{11} }
C_5	{ S_{12} }

3.4 클러스터별 대표 문장 추출

그래프를 분할한 각각의 부분 그래프는 하나의 하위 주제를 공통

으로 하는 문장들의 모임이다. 따라서 클러스터에는 유사한 내용의 문장들이 속하게 되고, 각 클러스터로부터 대표 문장을 선정하여 나열하면 전체 문서의 내용을 축약적으로 표현하는 요약문이 된다. 이때 문장의 중복성을 배제하기 위해 각 클러스터별로 가장 중요도가 높은 1개의 문장을 클러스터 대표 문장으로 추출한다. n개의 대표어 $w_1, w_2, \cdots, w_n$으로 구성된 k번째 문장 $Sentence_k$의 스코어 $Score_{sentence}$는 식 (8)과 같으며, 이 계산 값에 의해 클러스터 1개당 k개의 구성문장 중에서 중요도가 가장 높은 1 문장을 선택한다.

$$Score_{sentence}(Sentence_k) = (\sum_{i=1}^{n} asim(w_i)) \quad (8)$$

클러스터의 수는 요약문을 구성하는 문장의 수를 의미하며 문서의 하위 주제 구성에 따라 요약문의 크기는 일정하지가 않다. 즉, 요약문은 문서별 부분 그래프의 개수에 따라 유동적이지만 일정 크기에 의한 최종 요약문 완성을 위해 클러스터 중요도에 의해 전체 문서 길이의 20%에 해당하는 대표 문장을 선정한다. 이것은 일반적인 요약문서의 크기는 원문서의 1%에서 30%에 달하지만[16], 추출요약의 경우에는 약 20% 정도는 되어야 원문서의 의미를 전달할 수 있다[17]는 데에 기반 한 것이다. 또한 아무 의미 없는 문장이 어느 클러스터에도 속하지 않다가 단독으로 클러스터를 형성하게 되어 자동으로 대표 문장으로 추출되는 것을 방지하기 위함이다. 최종 요약문 완성을 위해 대표 문장들은 클러스터내의 문장 개수를 기준으로 한 클러스터 중요도에 의해 순위화 된다. 크기가 큰 클러스터의 주제가 그렇지 않은 클러스터에 비해 반드시 더 중요하다고 보기가 어려운 점도 있지만 많은 수의 문장들이 어떤 주제에 대해 관련이 있다는 것은 그만큼 중요하다고 판단되기 때문에 크기에 의해 클러스터의 중요도를 결정하였다.

4. 실험 및 평가

제안시스템의 성능을 실험하기 위한 데이터는 SUMMAC (TIPSTER Text Summarization Evaluation Conference) [18]에서 제공하는 문서 집합을 이용 하였다. 이 문서 집합은 ACL (Association for Computational Linguistics)의 183건 과학논문으로 이루어져 있으며, 요약문과 본문으로 구성되어 있다. 본문에 대해 제안시스템을 적용하여 자동으로 생성된 요약문과 문서 집합의 요약문을 비교하였다. 평가 방법은 ISI/USC의 Chin-Yew이 요약 성능을 평가하기 위해 개발한 ROUGE (Recall Oriented Understudy for Gisting Evaluation)[19]를 이용하였다. 이 방법은 단어의 재현율(Recall)에 기반 한 평가 시스템으로서 ROUGE-n gram과 ROUGE-L, ROUGE-W 가 있다. ROUGE-n gram은 문자열의 선두에서부터 한 문자씩 옮기면서 n문자 단위(n=1,2,3,4)로 일치 여부를 측정하는 방법이다. ROUGE-L은 두 문장에서 공통으로 나타나는 단어 개수를 순서에 상관없이 측정하며, ROUGE-W는 두 문장에서 공통으로 나타나는 단어 개수에 대해 연속적인 일치여부를 고려하여 측정한다.

- $ROUGE\text{-}N$ =

$$\frac{\sum_{S \in Reference\ Summaries} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in Reference\ Summaries} \sum_{gram_n \in S} Count(gram_n)} \quad (9)$$

$n : n-gram, gram_n$의 길이

$Count_{match}(gram_n)$: 전문가 요약문과 시스템 생성 요약문에서 동시 발생한 $n-gram$수

- $ROUGE\text{-}L$: Longest Common Subsequence with maximum length
- $ROUGE\text{-}W$: Weighted Longest Common Subsequence

제안시스템의 문서요약 성능개선의 평가를 위해 기존의 유사도 계산에 의한 문장 클러스터별 대표 문장 선정과 문장 가중치에 따른 단순 중요 문장 추출에 의한 요약결과와 비교하였다. 기존의 문장간 유사도 계산에 있어서는 코사인 유사도를 이용하였으며 클러스터링 방법은 계층적 방법을 이용하였다. <표 4>는 제안시스템에 대한 비교 실험 결과이다. 실험 결과 그래프 분할에 의한 중요 문장 추출방법이 유사도 계산에 기반한 문장 클러스터별 대표 문장 추출방법보다 좋은 결과를 보이고 있다. 전체 요약시스템에서 문서의 내용을 의미하는 대표어를 선정하는 과정은 매우 중요하다. 특히 본 논문에서는 대표어를 이용하여 문서를 그래프로 표현하기 때문에 어떻게 대표어를 추출 할 것인가 하는 문제는 그래프 분할을 하기 위한 전 단계로서 요약 성능에 많은 영향을 미친다. 단순한 단어의 통계정보를 이용하는 것이 아니라 단어의 공기정보를 이용하여 대표어를 선정함에 따라 보다 정확한 그래프 표현이 가능하게 되었다.

추출요약에서는 비슷한 내용의 문장들이 중복되어 선정되는 현상과 함께 서로 다른 하위 주제별 문장들이 같은 클러스터를 형성함에 따라 중요 문장이 탈락되는 문제가 발생한다. 따라서 요약의 성능을 향상시키기 위해서는 이 문제점을 반드시 해결해야 하는 데, 본 논문에서는 그래프 분할을 이용하여 서로 다른 클러스터를 연결하는 부분을 찾아낸 후 하위 주제별로 분리하여 결과적으로 전체 요약 성능이 향상된 실험결과를 보였다. 즉, 그래프 분할이 하위 주제별 문장 클러스터의 분리가 정확하여 기존의 클러스터링 방법이 가지고 있는 유사도 계산의 오류에 따른 중요 문장 탈락의 방지와 문장의 중복성 제거에 효율적임을 의미한다. 따라서 전체 문서의 중요의미를 충분히 반영할 수 있어 대표 문장 추출에 의한 문서 요약에서 좋은 결과를 보인 것으로 평가된다. 그러나 문서의 내용에 따라서는 그래프 분할에도 불구하고 여러 개의 부분 그래프로 분리되지 않고 서로 강하게 연결된, 단지 1-2개의 부분 그래프 형태로 남는 경우도 발생한다. 이러한 문서는 단지 1-2개의 하위 주제들만 포함하고 있다고 판단하며, 이 경우 1-2개의 문장에 의해 지나치게 짧은 요약문이 구성되는 것을 막기 위해 1개의 부분 그래프에서 여러 개의 문장을 중요도 순으로 추출하였다.

ROUGE-n gram에 의한 실험결과를 보면, Rouge-1과 Rouge-2가 Rouge-3와 Rouge-4에 비해 상대적으로 더 좋은 성능 향상을 보이고 있다. 즉 연속단어의 일치 여부에서 단어의 길이가 길어질수록 저조한 성능향상을 보이고 있다. 이것은 실험의 방법이 추출요약 결과와 생성요약 결과와의 단순한 단어 재현율 비교이기 때문에 갖는 특성으로 보인다. 즉 생성요약은 문장 중에서 중요한 부분만 응용하여 새로운 문장들을 생성하지만, 추출요약은 문장 자체를 그대로 추출하여 요약문을 생성하게 된다. 따라서 중요하지 않은 부분까지도 원문 그대로 요약에 적용하기 때문에

〈표 4〉 요약시스템의 성능 비교 실험

Method	제안시스템	단순 중요 문장	문장 클러스터링(유사도)
Rouge 1	0.41925	0.27141	0.32928
Rouge 2	0.10795	0.04285	0.08745
Rouge 3	0.04792	0.01043	0.04401
Rouge 4	0.03181	0.00400	0.02777
Rouge L	0.27135	0.18466	0.21473
Rouge W	0.07814	0.05311	0.06402

생기는 추출요약의 한계성으로 평가된다.

5. 결론 및 향후 연구과제

본 논문은 그래프 분할을 이용하여 하위 주제별로 중요 문장을 추출하는 요약시스템을 제안하였다. 중요 문장 추출에 의한 문서 요약에서는 정보의 중복 없이 문서의 하위 주제 내용을 표현하는 것이 매우 중요하다. 이를 위해 그래프 분할을 이용하여 유사 문장들을 클러스터링 함으로써 주제 분리를 하였다. 또한 대표어를 이용하여 문서를 그래프로 표현하기 때문에 대표어를 추출하는 기법도 문서요약시스템을 구현하기 위한 전 단계로서 중요한 과정이다. 대표어를 추출하기 위해 단어의 공기정보를 기반으로 연관성 분석을 실시하였다. 실험결과 제안시스템이 유사도 계산에 의한 문장 클러스터별 요약에 비해 성능이 향상됨을 볼 수 있었다. 이는 기존의 문장 클러스터링에서 사용하는 문장간 유사도 계산방법에 의한, 즉 어떠한 수식을 사용하든 가지고 있는, 근본적인 문제점을 어느 정도 해결했다고 할 수 있다. 하지만 본 논문에서 제안한 방법 외에, 문서요약을 하기 위한 주제 분리에 사용할 수 있는 그래프 분할 방법들에 대한 비교 연구가 더 필요하다.

앞으로 반드시 개선해야 할 연구과제는 이형동의어와 동형이의어 같은 단어 의미의 다양성에 관한 것과 추출 문장들을 대상으로 문서 흐름에 맞도록 문장을 다시 정렬(sentence ordering)하는 방법이다. 향후 이 문제를 보완하여 요약시스템에 적용하면 더 정확한 결과가 나올 것으로 기대된다. 또한 본 논문에서는 요약시스템의 성능평가를 위해 재현율에 기반한 ROUGE 평가 방법을 사용 했지만 요약문의 의미측정과 같은 새로운 평가 방법과 정확한 성능측정을 위한 보다 객관적인 평가 방법에 대한 연구가 요구된다.

참 고 문 헌

[1] Inderjeet Mani, Automatic Summarization, John Benjamins Publishing Co., 2001.

[2] Mary McKenna, Elizabeth D.Liddy, "Evaluation of Automatic Text Summarization Across Multiple Documents," AAAI Symposium, 1998.

[3] H.P.Edmundson, "New Methods in Automatic Extracting," Journal of the ACM, 16(2), 1969.

[4] Marti A. Hearst, "Multi-paragraph segmentation of expository text," In Proceedings of the 32nd Annual Meeting of the ACL, June, 1994.

[5] Salton.G., Singhal.A., Mitra.M. and Buckly.C., "Automatic text structuring and summarization," Information Processing and Management, Vol.33, No.2, 1997.

[6] 류동원, 이종혁, "단어공기정보를 이용한 자동화 문서요약," 한국정보과학회 학술논문발표지 27권 1호, pp.345-347, 2000.

[7] 류제, "단어의 공기 관계 그래프를 이용한 문서의 핵심 문장 추출에 관한 연구", 호서대학교 벤처전문대학원 석사학위논문, 2000.

[8] 정영미, 최상희, "문장 클러스터링에 기반한 자동요약 모형," 한국정보관리학회지, 제18권 3호, pp.159-178, 2001.

[9] 박성배, 장병탁, "Co-Trained Support Vector Machines을 이용한 문서분류," 한국정보과학회 봄 학술발표 논문집 (B), 제29권 1호, pp. 259-261, 2002.

[10] Julian Kupiec, Jan Pedersen, and Francine Chen, "A Trainable Document Summarizer," In Proceedings of ACM-SIGIR'95, pp.68-73,1995.

[11] Barzilay, Regina and Michael Elhadad, "Lexical Chains for Text Summarization", Master's thesis, Ben-Gurion University, 1997.

[12] C.J.van Rijsbergen., "A Theoritical Basis for the Use of Co-occurrence Data in Information Retrieval," Journal of Documentation,Vol.33:106-119,1977.

[13] 김재훈, 김준홍, "도합유사도를 이용한 한국어 문서요약 시스템," 한국 인지과학회 논문지 제12권 제1·2호, pp.35-42, 2001.

[14] Skorochodko,E.F., "Adaptive method of automatic abstracting and indexing," Information Processing 71: Processing of the IFIP Congress 71, ed. by Freiman, pp.1179-1182, North-Holland Publishing Company, 1972.

[15] 김철언, 그래프론과 알고리듬, POSTEC PRESS, 1997.

[16] Sparck Jones, K., "Automatic summarizing:factors and directions," Advances in Automatic Text Summarization, pp.1-12, The MIT Press. 1999.

[17] Morris. A.H.. Kasper and G.M, Adams. D.A., "The effects and limitations of automated text condensing on reading comprehension performance," Information systems Research, 3(1), pp.17-35, 1992.

[18] http://www.itl.nist.gov/iaui/894.02/related_projects/tipster_summac

[19] http://www.isi.edu/~cyl/ROUGE/

이 일 주

e-mail : ijlee@tongwon.ac.kr
1988년 아주대학교 전자계산학과(공학사)
1994년 한양대학교 전자계산학과(공학석사)
2002년 아주대학교 컴퓨터공학과 박사수료
1989년~1998년 현대미디어시스템/현대정보기술 책임
1998년~현재 동원대학 모바일컨텐츠과 부교수
관심분야 : 정보검색 시스템, 모바일프로그램

김 민 구

e-mail : minkoo@ajou.ac.kr
1977년 서울대학교 계산통계학과(이학사)
1979년 KAIST 전산학과(공학석사)
1989년 펜실베니아 주립대학교 전산학과(박사)
1999년~2000년 루지애나 대학 연구과학자
1981년~현재 아주대학교 정보 및 컴퓨터공학부 교수
관심분야 : 지능형 정보검색 시스템, 인공지능(지식표현 추론), 온톨로지 자동구축