

주관식 문제 채점에서의 구문의미트리 비교 시스템에 대한 연구

강원석[†]

요 약

질높은 주관식 문제 채점을 위해서는 답변 속에 들어있는 단어간의 구문의미적 관계를 분석하는 구문의미 분석이 필요하다. 그러나 구문의미 분석의 결과인 구문의미트리는 단어간의 구조적 의미 관계를 내포하고 있어 단어의 나열인 일차원적인 벡터의 유사도 계산을 적용할 수가 없다. 본 연구는 단어의 지식과 함께 단어와 단어간의 구조적 의미 관계를 내포하는 구문의미트리를 비교하는 비교 시스템에 대한 연구를 한다. 본 연구에서는 구문의미트리 비교를 위해 유사성 계산 원칙을 제안하고 실험을 통해 검증하였다. 본 구문의미트리 비교 시스템은 구문의미분석의 결과를 비교할 수 있게 하여 주관식 문제 채점에 도움을 줄 것이고 문서 유사도 영역에도 활용할 수 있을 것이다.

주제어 : 주관식 채점, 구문의미트리, 유사도

Research on Comparing System with Syntactic-Semantic Tree in Subjective-type Grading

WonSeog Kang[†]

ABSTRACT

To upgrade the subjective question grading, we need the syntactic-semantic analysis to analyze syntactic-semantic relation between words in answering. However, since the syntactic-semantic tree has structural and semantic relation between words, we can not apply the method calculating the similarity between vectors. This paper suggests the comparing system with syntactic-semantic tree which has structural and semantic relation between words. In this thesis, we suggest similarity calculation principles for comparing the trees and verify the principles through experiments. This system will help the subjective question grading by comparing the trees and be utilized in distinguishing similar documents.

Keywords : Subjective-type Grading, Syntactic-Semantic Tree, Similarity

[†] 정 회 원: 안동대학교 교수(교신저자)

논문접수: 2017년 7월 10일, 심사완료: 2017년 9월 12일, 게재확정: 2017년 9월 14일

* 이 논문은 2015학년도 안동대학교 연구비에 의하여 연구되었음

1. 서론

주관식 문제를 자동 채점하기 위해서는 답변의 자연어 문장을 해석하고 해석 결과를 비교해야 한다. 많은 연구들이 형태소 해석 방법을 이용하여 단어를 추출하고 이를 비교하고 있다[1][2][3]. 이 방식은 단어들이 답변의 정보를 표현한다고 전제한다. 그렇지만 답변에 들어 있는 자연어 표현은 단어, 단어의 구문적 관계와 의미적 관계 등을 포함한 문장으로 구성되어 있어 단어 정보만 추출하여 비교하기에는 부족하다.

답변 속에 들어있는 단어간의 구문적 관계를 찾아내는 구문의미해석 방법을 적용한다면 보다 나은 채점을 할 수 있다. 그렇지만 구문의미해석을 하는 경우 그 결과인 구문의미트리를 비교하는 문제가 발생한다. 답변을 형태소 해석하여 단어들로 표현할 때는 단어들의 벡터로 표현하여 유사성을 검사하는 함수를 쉽게 구현할 수 있으나 구문의미트리로 표현할 때는 그렇게 할 수가 없다. 구문의미트리를 하나의 벡터값으로 지정한 후 비교한다면 같은지 다른지의 이분법적인 판단만 가능하고 트리가 구조의미적으로 얼마나 유사한지에 대한 판단은 어렵다. 이를 해결하기 위해서는 구문의미트리간의 유사성을 비교하는 비교기가 필요하다.

본 논문은 주관식 문제 채점의 질을 향상하기 위해 구문의미 분석을 적용하는 경우, 결과인 구문의미트리의 유사성을 비교하는 비교시스템을 제안한다. 이를 위해 구문의미트리 비교에 고려할 원칙을 정의하고 실험을 통해 제안된 원칙이 타당함을 검증한다. 2장은 구문의미트리 비교기의 관련 연구를 기술하고 3장은 구문의미트리 비교에 사용한 원칙과 시스템을 기술한다. 4장은 시스템을 실험한 후 그 결과를 분석하고 5장은 결론과 논의사항을 기술한다.

2. 관련 연구

구문의미트리 비교기와 관련되는 연구는 크게 어휘 중심의 연구, 영어 영역의 연구, 구문트리 중심의 연구로 구분할 수 있다.

2.1 어휘 중심의 유사도 연구

장성호와 강승식(2003)은 용어를 선별하여 유사 문서 판별에 사용하였다[1]. 이 연구는 형태소 해석에 기반한 단어 추출에 의존하고 있어 시스템의 성능 향상에 한계가 있다. 김혜숙과 동료들(2003)은 문서간 유사도를 계산하기 위해 신경망을 사용하였으나 신경망의 입력이 되는 요소가 형태소 해석의 결과인 단어에 기반하고 있고 역시 한계가 있다[2].

조우진과 동료들(2005)은 대량의 말뭉치를 이용하여 의미커널을 구축하고 답안을 채점할 때 구축된 의미커널과 한글워드넷을 이용하여 답안을 자동채점하였다[3]. 이 연구는 한글 워드넷을 이용한 의미 정보를 활용하고 있으나 형태소 해석을 통해 단어를 추출하고 추출된 단어의 의미를 이용하지만 단어간의 구문의미적 관계는 사용하지 않아 역시 한계가 있다.

박일남과 동료들(2013)은 단어와 구 수준의 자동채점 시스템을 구축하였다[4]. 이 시스템은 단어와 구 중심의 정답 템플릿을 설계하고 이를 토대로 정답과 일치하는지 여부와 개념 중심의 일치여부의 채점을 실행하였다. 이 연구는 단어와 단어의 유의어를 활용한 실무적인 시스템을 구성하였으나 구문의미분석은 활용하지 않았다. 강승식과 장은서(2016)도 박일남과 동료들의 연구(2013)를 토대로 정답 템플릿을 여러 학생답안으로부터 정답 템플릿을 구성하여 채점을 하도록 개선하였다[5]. 이 연구도 단어와 구 중심의 채점을 적용하여 한계가 있다.

김영철과 최재영(2005)은 자연어 표현에 대한 것이 아닌 프로그램에 대한 유사도 평가를 하였다[6]. 프로그램의 유사도를 비교하기 위해 토큰 중심의 비교를 하는 방법을 적용하였다. 이 연구는 프로그램에 대한 연구라 자연어 표현에 대한 적용은 어렵다.

2.2 영어 구문트리 중심의 유사도 연구

Attali와 Burstein(2006)의 연구는 영어 논술의 자동채점을 위한 것으로 자연어 분석의 결과를 정형화되어진 속성집합으로 표현하고 이를 벡터

화하여 표준화 샘플들과 비교하는 방식으로 채점을 한다[7]. 이 연구는 영어 분야의 연구이고 구문 분석을 하고 있지만 구문트리를 비교하는 것이 아니라 정형화된 속성으로 변환하여 비교하는 것이므로 구문의미관계를 비교하는 것에 제한이 많다.

Akhil과 동료들(2010)은 영어 질의에 맞는 유사 SMS를 찾기 위해 두 개의 구문트리에 있는 노드들 가운데 공통되는 부분요소의 수의 합과 의미 유사도값을 합성하여 유사성을 검사한다[8]. 정창후와 동료들(2011)은 술어논항 구조의 유사도를 이용하여 객체간의 연관관계를 추출하고자 하였다[9]. 이 연구도 영어에 적용한 것이다.

2.3 구문트리 중심의 유사도 연구

김영철과 동료들(2003)은 어휘중심이 아닌 구문트리 중심의 비교 방식을 제안하였으나 트리를 역파싱한 다음 스트링매칭 알고리즘을 사용하여 유사도 검사를 실시하여 구문트리 비교가 제대로 이루어지지 않았다[10]. 또한 자연어 표현이 아닌 프로그램의 유사성 검사에 활용하여 일반 자연어 표현에 적합하지 않다.

손정우와 박성배(2009)는 온톨로지 정렬에 사용하고 파스트리에 대한 유사도를 계산하였으나 김영철과 동료들의 연구(2003)와 마찬가지로 변환 후 스트링매칭 방식을 사용하였다[11]. 스트링매칭 방식은 구문의미트리의 구조의미적 내포 관계를 충분히 반영하지 못한다.

신미해와 동료들(2010)은 문서구조 기반의 부분 또는 전체 유사도 측정방법을 제안하였다[12]. 근본적으로 텍스트는 어휘 중심의 분석을 한 후 표현하였고 단지 문서구조를 트리로 표현하고 그 유사도 비교를 시도하였다. 따라서 자연어 표현상의 구문의미표현은 추출하지 않았고 문서구조의 트리 유사도 비교도 역파싱한 후 스트링 매칭 방식을 사용하여 구문의미트리의 비교가 제대로 이루어지지 않았다.

강원석과 동료들(2014)은 구문의미분석을 시도하여 구문의미트리를 생성하였다. 유사성을 검사하기 위하여 구문의미트리를 이분법적으로 비교하였다. 즉, 정확히 트리의 구조와 노드의 의미가

일치하는 것을 같은 트리로 보고 그렇지 않은 것은 같지 않은 것으로 분리하여 유사하거나 연관이 있는 구문의미트리를 구분하지 못하였다[13].

강원석(2015)은 이전의 문제를 해결하기 위해 구문의미트리 비교기를 이용하였다[14]. 그러나 구문의미트리의 유사도에 반영될 구성요소인 루트노드와 서브트리 노드의 반영 비율 등을 직관적으로 정한 값으로 정의하여 한계가 있었다.

결론적으로 어휘 중심의 유사도 연구 방법들은 자연어 표현에 포함되어 있는 구문, 의미관계를 포착할 수 없으므로 그 한계가 있어 주관식 문항 채점에 적합하지 않다. 한편 영어 구문의미트리의 비교에 대한 연구들은 한국어에 적용하기에는 무리가 많다. 영어는 한국어와 언어적 특성이 많이 달라 구문트리의 구조적 관계와 의미가 차이나기 때문이다.

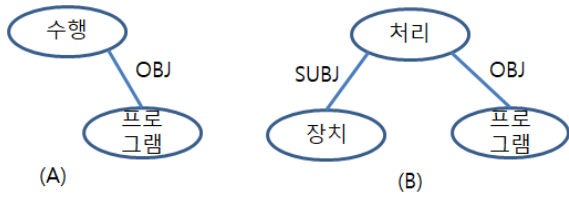
스트링 매칭 방식을 이용한 구문트리 비교에 대한 연구들은 문제점이 있다. 스트링 매칭방식은 구문의미트리에 포함되어 있는 구문적 관계를 찾아내기 어렵기 때문이다. 한국어 구문트리비교기를 이용한 연구들이 있지만 이 연구는 트리를 이분법적 비교를 하여 유사성이 어느 정도 있는지 계산할 수 없어 유사한 구문의미트리를 찾아내기 어렵다.

본 논문은 구문의미트리 이분법적 비교기의 문제를 해결하기 위해 구문의미해석의 결과인 구문의미트리기가 어느 정도 유사한지를 계산하는 비교기를 제안하였다. 이 비교기를 구축하기 위해 구문의미트리 비교에 고려해야 할 원칙을 정의한 후 실험을 통해 그 원칙을 검증하였다. 그리고 원칙에 적용할 최적의 가중치를 탐색하고 이를 적용한 구문의미트리 비교 시스템을 구축하였다.

3. 구문의미트리 비교 시스템

3.1 구문의미트리 비교 원칙

구문의미해석은 문장에 들어있는 단어간의 구문의미관계를 추출하여 구문의미트리로 표현한다. 구문의미트리는 하나의 루트 노드와 여러 개의 서브트리 노드로 구성된다.



[그림 1] 구문의미 트리 예

[그림 1]의 트리 A는 하나의 루트노드 ‘수행’과 하나의 서브트리노드 ‘프로그램’으로 구성되고 그 관계는 OBJ관계이다. 트리 B는 루트노드 ‘처리’와 두 개의 서브트리노드 ‘장치’, ‘프로그램’으로 구성되고 그 관계는 SUBJ, OBJ이다. 서브트리는 재귀적으로 구문의미트리가 될 수 있다. 두 트리의 비교는 루트노드가 유사한지 검사하고 그리고 루트노드의 자식 서브트리가 서로 유사한지 검사한다. [그림 1]의 경우 루트노드 ‘수행’과 ‘처리’의 유사성을 검사하고 서브트리는 같은 OBJ 격을 가진 서브트리 ‘프로그램’과 ‘프로그램’의 유사성을 검사할 것이다.

비교를 위해 코사인 계수 등의 비교방식을 사용하려면 구문의미트리를 벡터의 한 속성으로 나타내야 한다. 그러한 경우 조금이라도 성분이 다른 구문의미트리는 독립적인 다른 속성으로 정의하게 되어 구문의미트리간의 유사성을 비교하기 어렵다. 본 연구에서는 구문의미트리간의 유사성을 계산할 수 있는 구문의미트리 비교 시스템을 설계하고자 한다. 이를 위해 먼저 고려해야 할 원칙을 정의하고 원칙에 따른 유사성 계산법을 설계하였다.

원칙 1) 두 구문의미트리의 유사성은 두 트리의 루트의 유사성과 서브트리 유사성의 합이다.

구문의미트리는 하나의 루트노드와 여러 개의 서브트리로 이루어진다. 두 구문의미트리가 같은지는 루트노드가 서로 같은지 그리고 여러 개의 서브트리 가운데 서로 같은 것이 있는지를 비교한다. 따라서 전체의 유사도는 루트노드의 유사성과 서브트리의 유사성의 합으로 구성하는 것이 적합하다.

원칙 2) 루트의 유사성은 서브트리 유사성과 다른 가치를 가진다.

구문의미트리에서 루트는 동사로서 술어의 유형과 의미를 결정하고 서브트리는 그 유형에 따

른 역할(격)을 채우는 것이므로 역할이 서로 다르다. 따라서 서로 다른 역할의 의미가 반영되는 가중치가 필요하다. 본 연구에서는 이 원칙을 확인하고 다양한 데이터에 대해 실험을 통해 적합한 가중치를 찾으려 하였다.

원칙 3) 루트의 유사성은 노드를 구성하는 어휘의 유사성이다.

루트노드는 하나의 단어로 구성되므로 단어의 유형과 의미만 비교하면 된다. 본 연구에서는 단어의 유형과 의미를 비교하기 위해 단어의 의미 속성 정보를 이용한 어휘유사도 함수를 정의하였다.

원칙 4) 서브트리의 유사성은 격이 같은 서브트리의 유사성의 합이다.

루트노드는 여러 개의 자식 서브트리를 가진다. 각 자식 서브트리는 서로 역할(격)이 다르다. 따라서 같은 역할을 하는 서브트리끼리 비교하여 그 유사성을 반영하고 서로 다른 역할의 서브트리는 비록 그 구성성분이 같은 것이 있을지라도 비교하지 않는다. 이 원칙은 역할이 다른 것은 유사하더라도 다른 것이라는 전제를 바탕으로 정의하였다.

원칙 5) 서브트리의 유사성은 서브트리 수가 많으면 더 중요하다.

구문의미트리를 비교할 때 대상이 되는 구문의미트리의 자식 서브트리가 하나일 때와 둘일 때를 비교해보자. 둘인 경우 두 자식서브트리가 모두 일치하는 경우는 하나인 경우 하나의 자식서브트리가 일치하는 경우보다 구문트리의 유사성이 더 크다고 볼 수 있다. 이를 고려하여 서브트리의 수가 증가할수록 구문트리의 유사성이 더 증가하도록 정의한다. 이때 전체 구문트리의 유사성은 루트의 유사성과 서브트리의 유사성의 합으로 정의하였으므로 서브트리의 유사성에 이 특징을 반영한다.

원칙 6) 서브트리 수가 늘어날지라도 최종적으로 서브트리의 유사성의 최대값을 넘어서지 못한다.

원칙 5에 따르면 서브트리수가 늘어나면 유사성이 계속 증가하게 된다. 이것은 바람직하지 않다. 서브트리 개수가 증가하더라도 루트노드의 중요성은 여전히 가치를 가지므로 이를 반영하기

위해 한계값이 필요하다.

위와 같은 6개의 원칙을 고려하여 구문의미트리 비교 시스템에 사용할 유사도 계산식을 다음과 같이 정의하였다.

$$ST(T_1, T_2) = \frac{SL(r(T_1), r(T_2)) \times \alpha + \frac{\sum_{(T_{1i}, T_{2j}) \in T_1 \cap T_2} ST(T_{1i}, T_{2j}) \times 2}{n} \times \beta \times \gamma}{\alpha + \beta \times \gamma} \quad (식1)$$

$ST(T_1, T_2)$ 는 구문의미트리 T_1 과 T_2 의 유사도를 계산하는 함수이다. $ST(T_1, T_2)$ 는 원칙 1에서 제시한 것과 같이 루트의 유사성과 서브트리의 유사성의 합으로 구성된다. 그리고 원칙 2에서 제시한 유사성의 가중치는 α 와 β 로 표현한다. α 는 루트의 가중치이고 β 는 서브트리의 가중치이다.

$SL(r(T_1), r(T_2))$ 는 T_1 의 루트단어와 T_2 의 루트단어의 어휘 유사성을 계산하는 원칙 3의 함수이다.

$$SL(a, b) = \frac{2 * |sem(a) \cap sem(b)|}{|sem(a)| + |sem(b)|} \quad (식2)$$

이 함수는 어휘를 비교하기 위해 어휘가 가지는 의미속성의 집합을 이용하므로 집합의 유사성을 표현하기 적합한 다이스 계수를 이용하여 정의하였다. $sem(a)$ 는 단어 a 의 의미속성집합을 나타내고 $|sem(a)|$ 는 의미속성집합의 원소수를 나타낸다.

원칙 4에 의하면 서브트리의 유사성은 격이 같은 서브트리의 유사성의 합으로 정의된다. 식(1)의 Σ 가 합을 표현하고 교집합은 격이 같은 서브트리를 의미한다. $(T_{1i}, T_{2j}) \in T_1 \cap T_2$ 는 T_1 트리의 서브트리의 격이 T_2 트리의 서브트리의 격과 같은 경우를 의미한다. 그 때 각 경우의 T_1 의 서브트리를 T_{1i} 로 표현하고 T_2 의 서브트리는 T_{2j} 로 표현한다. 이때 격이 같은 트리는 양 구문트리에 모두 있으므로 2로 곱하여 가중한 후 정규화를 위해 전체 서브트리수 n 으로 나눈다. 최종적으로 계산된 값에 서브트리 유사성 가중치 β 를 곱한다. 그리고 격이 같은 서브트리의 유사도는 재귀적인 ST 함수를 사용한다.

$$n = |T_1 \cap T_2| \times 2 + |T_1 - T_2| + |T_2 - T_1| \quad (식3)$$

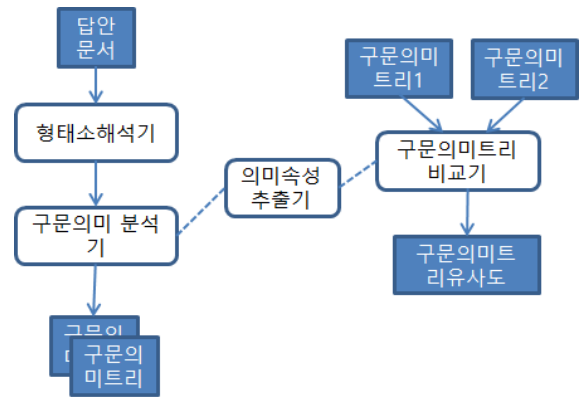
n 의 계산에 사용한 $|T|$ 기호는 트리 T 의 서브트리수를 의미한다. n 은 비교할 두 구문의미트리에서 전체 서브트리수를 나타내는 것으로 격이 같은 공통 서브트리수의 2배에 T_1 에만 있는 서브트리수와 T_2 에만 있는 서브트리수를 더한 값이다.

$$\gamma = (1 - \frac{1}{e^{nk}}), k = \frac{1}{4}, \frac{2}{4}, \frac{3}{4}, \frac{4}{4}, \frac{100}{4} \quad (식4)$$

가중치 γ 는 원칙 5와 6을 반영한 것으로 서브트리의 수가 증가함에 따라 가중치가 증가하도록 설계하였다. 또한 계속 증가하는 것이 아니라 한계값까지만 증가하도록 서브트리수의 지수승의 역수를 1에서 감산하는 식으로 설계하였다. 서브트리수가 아주 큰 경우 γ 는 1에 수렴하게 된다. 본 연구에서는 5 종류의 상수 k 를 설정하여 어떤 k 의 γ 가 좋은지 실험하였다.

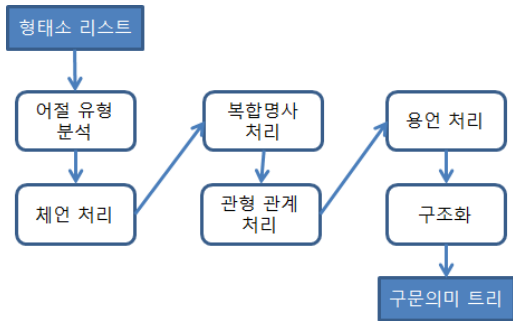
3.2 구문의미트리 비교 시스템 구성

구문의미트리를 비교하기 위해서 시스템을 다음과 같이 구성하였다.



[그림 2] 구문의미트리 비교 시스템

먼저 구문의미트리를 생성하기 위해 형태소해석기와 구문의미 분석기를 사용한다. 형태소 해석기는 김재훈과 동료들의 시스템[15]을 이용하였다. 해석결과를 입력으로 구문의미 분석기를 실행한다. 구문의미 분석기의 처리과정은 [그림 3]과 같다.



[그림 3] 구문의미 분석기 처리과정

구문의미 분석기는 여섯 단계의 처리로 구성된다. 첫 단계는 어절 유형 분석이다. 즉, 어절이 체언 중심의 유형인지 용언 중심의 유형인지 구분한다. 다음 단계는 체언처리 부분이다. 체언 유형으로 파악된 어절에 대해 부가된 격조사나 어미 등에 근거하여 구문/의미관계를 첨부한다. 이때 의미적 조건을 검사하기 위해 의미속성 추출기를 활용한다. 구문/의미관계는 24개로 정의하였고 <표 1>과 같다.

<표 1> 시스템의 구문/의미 관계

명칭	구문/의미관계	격조사, 어미
SUBJ	주격	가, 이
OBJ	목적격	을, 를
BLA	복합명사	blank
TARG	방향격	로, 으, 예
ADNM	관형격	의, 은, 는
...

다음 단계는 복합명사 처리 부분이다. 연이어 두 명사가 복합명사인 경우 두 명사를 연결시켜 구조화한다. 다음은 관형 관계 처리이다. 관형형은 두 가지 종류로 구분할 수 있다. 격조사가 붙은 체언 유형과 관형형 어미가 붙은 용언 유형이다. 관형형의 대상이 되는 체언과 연결하여 구조화한다.

다음 단계는 용언 처리이다. 용언은 하다가 붙는 유형과 그렇지 않은 유형으로 구분하여 태그를 붙인다. 마지막 단계는 각 어절의 성분이 어느 용언과 관계를 맺는지 찾아내어 구조화하는 단계이다. 구조화 과정에도 의미조건을 검사하기 위해 의미속성 추출기를 활용한다.

구문의미트리 비교기는 생성된 구문의미트리

간의 유사성을 계산한다. 다음 예 문장에 대해 계산과정을 살펴보자.

- (1) 프로그램을 수행한다.
- (2) 장치가 프로그램을 처리한다.
- (3) 장치가 수행하였다.

<표 2>는 앞의 세 문장에 대한 구문의미 분석의 결과인 구문의미 트리와 구문의미 트리간의 유사도 계산과정을 보여준다. T1, T2, T3는 앞의 세 문장에 대한 구문의미트리를 의미한다. 시스템에서는 괄호로 싸인 형태로 트리를 표현하였다. ‘(SUCO 수행 (OBJ 프로그램))’의 표현은 하나의 구문의미트리를 나타낸다. 그 표현을 보면 괄호 다음에 나오는 첫 요소는 루트노드의 유형을 나타내고 두 번째 요소는 루트노드에 해당하는 단어를 나타낸다. 그리고 세 번째 요소부터는 서브트리를 표현하는 것으로 괄호로 싸인 형태로 표현된다. 여러 개의 서브트리가 있는 경우 연이어 괄호로 싸인 형태가 계속 나타날 것이다.

<표 2> 구문의미트리 유사도 계산 예

항목	값
T1	(SUCO 수행 (OBJ 프로그램))
T2	(SUCO 처리 (SUBJ 장치) (OBJ 프로그램))
T3	(SUCO 수행 (SUBJ 장치))
ST(T1, T2)	$(SL(수행,처리) \times 6 + ST(T11,T22) \times 2) / (1 \times 2 + 0 + 1) \times (4 \times (1 - 1/e^{(3/2)})) / (6 + 4 \times (1 - 1/e^{(3/2)})) = 0.447$
ST(T1, T3)	$(SL(수행,수행) \times 6 / (6 + 4 \times (1-1/e^{(2/2)}))) = 0.704$
ST(T2, T3)	$(SL(처리,수행) \times 6 + ST(T21,T31) \times 2) / (1 \times 2 + 1 + 0) \times (4 \times (1-1/e^{(3/2)}))/(6+ 4 \times (1-1/e^{(3/2)})) = 0.447$

T1과 T2의 유사도 계산 ST(T1, T2)는 루트노드의 유사성 계산의 값과 서브트리노드의 유사성 계산의 값의 가중치를 곱한 값의 합으로 계산된다. <표 2>에서는 $\alpha = 6, \beta = 4, k = 1/2$ 로 하여 계산하였다. SL(수행, 처리)의 값은 (식 2)에 따라 의미속성 추출기를 통해 얻어진 의미속성 가운데 공통의 의미속성의 수와 각 의미속성의 수의 식으로 계산한다. 그 계산은 <표 3>과 같다.

<표 3> 어휘의미유사도 계산 예

항목	값
sem(수행)	{accomplishment, event, 수행}
sem(처리)	{control-behavior, event, 처리}
SL(수행, 처리)	$\frac{2*1}{3+3} = 0.33$
SL(수행, 수행)	$\frac{2*3}{3+3} = 1$

서브트리의 유사성은 격이 같은 서브트리가 하나밖에 없으므로 하나의 유사도 계산값만 고려하면 된다. 즉, T1의 첫 번째 서브트리와 T2의 2번째 서브트리의 유사성을 검사하면 되는데 이 경우 같은 내용의 서브트리이므로 1이 된다. 이 값에 가중치 α 와 β , γ 를 곱하여 계산한다. γ 는 서브트리의 수 n 과 상수 k 에 대한 식(식 4)으로 계산된다.

4. 실험 및 결과 분석

4.1 여러 조건의 구문의미트리 비교기 실험

본 연구에서는 최적의 구문의미트리 비교기를 찾기 위해 여러 조건의 구문의미트리 비교기를 구성하여 실험하였다. 각 구문의미트리 비교기는 두 가지 요소에 근거하여 정의된다. 하나는 가중치 α 이고 다른 하나는 상수 k 이다. 가중치 α 는 루트의 유사성에 적용되는 가중치이다. β 는 서브트리의 유사성에 적용되는 가중치로써 두 가중치의 합을 10으로 정하였기 때문에 α 가 정해지면 β 도 정해진다. 따라서 β 는 고려하지 않았다.

다른 하나는 상수 k 이다. 이 상수는 서브트리의 개수에 따라 변하는 영향력을 어느정도 반영할지에 대한 것이다. 상수 k 가 정해지면 가중치 γ 도 정해지므로 두 요소(α , k)를 기준으로 여러 조건의 구문의미트리 비교기를 구성하였다. <표 4>는 구성한 구문의미트리 비교기를 기술한 것으로 행으로 α 가중치 값을 나타내고 열로 상수 k 값을 나타낸다. 본 연구에서 구성한 비교기는 $8 \times 5 = 40$ 종류이다.

<표 4> 본 연구의 구문의미트리 비교기 종류

$\alpha \backslash k$	1/4	2/4	3/4	4/4	100/4
1	C11	C12	C13	C14	C15
2	C21	C22	C23	C24	C25
...
8	C81	C82	C83	C84	C85

4.2 여러 종류의 검사 데이터 실험

실험을 위해 본 연구에서는 세 종류의 검사 데이터를 정의하였다.

- (가) 24쌍의 유사 데이터
- (나) 300쌍의 정보처리 분야의 데이터
- (다) 400쌍의 일반 영역의 데이터

(가)는 정보처리 분야에서 선별된 유사 데이터이고 (나)는 정보처리 분야에서 임의로 수집된 데이터이고 (다)는 일반 분야에서 임의로 수집된 데이터이다. 각 데이터에 따라 각 시스템이 어떠한 차이가 있는지 살펴보고자 하였다. 검사 데이터의 일부 예는 <표5>와 같다.

<표 5> 검사쌍 데이터

종류	검사쌍	
(가) 24쌍	시스템이 입력을 수행한다 채널은 시스템에 신호를 전달한다 ...	시스템이 프로그램을 수행한다 장치는 시스템에 신호를 전달한다 ...
(나) 300쌍	컴퓨터가 입력 프로그램을 만난다 채널은 수행완료를 알린다 ...	데이터가 입력장치로 처리된다 채널은 입력완료를 보고한다 ...
(다) 400쌍	마우스 작업은 무리가 된다 전자기파로 질병이 생긴다 ...	컴퓨터를 잘못 사용한다 통증이나 두통을 겪는다 ...

4.3 구문의미트리 비교기 검증 방법

구문의미트리 비교기의 성능을 검사하기 위해 먼저 사람이 계산한 유사도 값을 정의한 후 각 비교기의 유사도 계산값과 비교를 하였다. 사람이 계산한 유사도 값은 3명의 유사도 계산값의 평균을 취하였다. 두 유사도 값들의 비교는 다음과 같은 상관계수를 사용하였다.

$$r = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \sigma_y} \quad (\text{식5})$$

상관계수는 시스템이 계산한 유사도가 사람이 계산한 유사도와 얼마나 일치하는지를 알려주므로 이 값으로 시스템의 성능을 평가한다.

4.4 실험 결과 및 분석

세 종류의 데이터에 대해 40개 종류의 구문의 미트리 비교기의 실험에 대한 상관계수값은 다음 표와 같다.

<표 6> (가) 데이터의 실험 결과

$\alpha \setminus k$	1/4	2/4	3/4	4/4	100/4
1	0.869308	0.84535	0.838211	0.835673	0.833991
2	0.899352	0.895237	0.89054	0.888409	0.886786
3	0.851685	0.879132	0.882401	0.882641	0.88225
4	0.777733	0.820703	0.829808	0.832003	0.832656
5	0.706007	0.748925	0.7595	0.762315	0.763299
6	0.644736	0.680112	0.689359	0.691866	0.692706
7	0.59424	0.61989	0.626729	0.628561	0.629104
8	0.552803	0.568886	0.573173	0.57429	0.57456

<표 7> (나) 데이터의 실험 결과

$\alpha \setminus k$	1/4	2/4	3/4	4/4	100/4
1	0.798711	0.777827	0.771059	0.768204	0.764456
2	0.852883	0.832022	0.824434	0.821149	0.816804
3	0.879093	0.866729	0.861389	0.859011	0.855842
4	0.885136	0.883572	0.881573	0.880551	0.878999
5	0.877112	0.884885	0.886053	0.886266	0.885958
6	0.860013	0.873604	0.877021	0.87808	0.878432
7	0.837676	0.852903	0.857305	0.858742	0.859322
8	0.812829	0.8258	0.829862	0.83122	0.831794

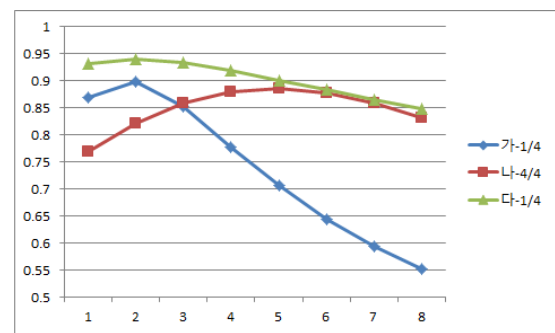
<표 8> (다) 데이터의 실험 결과

$\alpha \setminus k$	1/4	2/4	3/4	4/4	100/4
1	0.931849	0.922676	0.918868	0.917023	0.91422
2	0.940757	0.938195	0.935701	0.934208	0.931526
3	0.933386	0.93903	0.939266	0.938902	0.937558
4	0.918855	0.930521	0.93324	0.934	0.934092
5	0.901398	0.916212	0.920579	0.922136	0.923251
6	0.883102	0.898423	0.90349	0.905437	0.907051
7	0.865015	0.878669	0.883545	0.885494	0.887162
8	0.847652	0.857935	0.861835	0.863441	0.864826

1) 분석 1 : 최적의 시스템 검증

결과 값을 분석하여 본 결과 (가) 데이터의 경우는 C21(0.899352), (나) 데이터의 경우는 C54(0.886266), (다) 데이터의 경우는 C21(0.940757)이 가장 좋은 결과를 가져옴을 알 수 있었다. 그중 일반 분야의 데이터인 (다) 데이터에서 가장 좋은 결과가 나옴을 제안한 구문트리 비교기가 특정한 자료에만 좋은 결과를 가져오는 시스템이 아님을 보여준다.

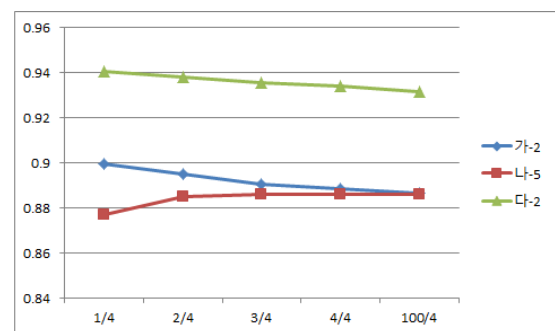
2) 분석 2 : 제안한 가중치 α 의 검증



[그림 5] 구문의미트리 비교기 분석2

[그림 5]는 α 값의 변화에 따른 실험결과를 보기 위해, 관련된 결과를 발췌하여 그래프로 도시한 것이다. 그래프의 가로축은 α 값을 나타내고 세로축은 상관계수 값이다. ‘가-1/4’ 그래프는 k값이 1/4인 경우의 α 값 변화에 따른 시스템의 성능을 보여준다. 이 그래프에서 α 가 2일 때 가장 좋은 결과가 나옴을 보여주고 있다. ‘나-4/4’는 α 가 5일 때 가장 좋은 결과가 나오고 ‘다-1/4’는 α 가 2일 때 가장 좋은 결과가 나온다. 이것은 구문트리의 유사성이 루트의 유사성과 서브트리의 유사성이 가중치를 반영한 합으로 정의한 원칙 1, 2가 타당함을 보여주고 있다.

3) 분석 3 : 제안한 가중치 γ 의 검증

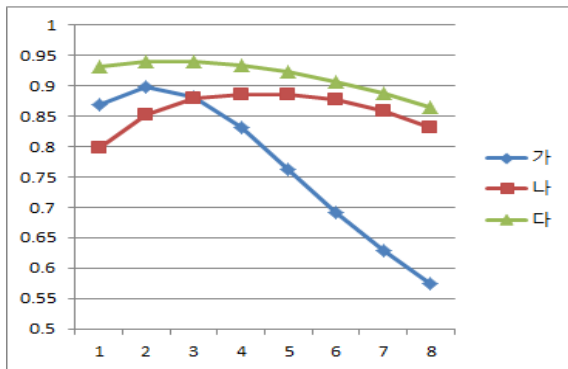


[그림 6] 구문의미트리 비교기 분석3

[그림 6]은 k 값에 따른 γ 의 변화에 따라 시스템이 어떤 성능을 보이는지를 보여주기 위한 것이다. 그래프의 가로축은 k 값을 나타내고 세로축은 상관계수 값이다. ‘가-2’ 그래프는 α 가 2인 경우 k 변화에 따른 시스템의 성능을 보여준다.

그래프에서 k 가 100/4인 경우 계산식에 따라 γ 값이 1에 이른다. 즉, 서브트리 수에 관계없이 k 값이 충분히 커서 γ 값은 항상 1에 근접한다. 이것은 서브트리수의 변화가 유사성에 미치는 원칙 5, 6을 배제한 것이 된다. 만약 이 경우의 시스템의 성능이 다른 경우보다 더 좋다면 제안한 원칙 5, 6이 무의미함을 뜻할 것이다. 그러나 실험결과에서 알 수 있듯이 k 가 1/4(가-2), 4/4(나-5), 1/4(다-2)일 때 시스템의 성능이 더 좋으므로 본 연구에서 제안한 원칙 5, 6이 타당함을 알 수 있다. 그래프에서 ‘가-2’와 ‘다-2’의 경우는 k 가 0으로 갈수록 시스템의 성능이 좋아지고 있다. 그렇지만 k 가 0인 경우는 γ 가 0이 되고 β 를 0으로 만들어 α 값을 큰 값으로 정하는 효과가 온다. 분석2에서 검증하였듯이 α 가 아주 커지면 오히려 시스템의 성능이 떨어짐을 알 수 있었다. 따라서 k 가 0에 근접하게 되면 최종적으로 시스템의 성능이 떨어짐을 유추할 수 있으므로 최적의 시스템은 k 가 0과 1/4를 포함한 사이에 있음을 알 수 있다. 따라서 (가)와 (다) 데이터에 대해서 최고 성능의 시스템을 찾기 위해서는 k 값을 0과 1/4사이의 값으로 실험을 더 해야 판명이 날 것이다.

4) 분석 4 : α 와 γ 의 관계



[그림 7] 구문의미트리 비교기 분석4

추가적으로 세 종류의 데이터에 대한 시스템의 성능을 분석해 보면 α 값과 γ 값과의 관계를 파악할 수 있다. [그림 7]의 가로축은 알파값, 세로축

은 상관계수값을 나타낸다. [그림 7]의 그래프는 [그림 5]와 달리 <표 6>, <표 7>, <표 8>에서 각 α 값에 대해 가장 좋은 성능을 가지는 γ 값의 시스템을 골라 상관계수값을 그린 그래프이다. ‘가’의 경우는 α 값이 1, 2일 때는 k 를 1/4로 한 γ 값의 시스템의 성능이 제일 좋아 그 상관계수를 보이고 α 값이 3일 때는 k 가 4/4로 한 γ 값의 시스템의 성능이 제일 좋아 그 상관계수를 보인다. 그리고 알파가 4이상일 때는 k 가 100/4로 한 γ 값의 시스템의 성능이 제일 좋아 그 상관계수를 도시하였다. 그래프가 이루는 곡선에서 알 수 있듯이 시스템이 가장 좋은 성능을 가지기 위해서는 적절한 α 값과 γ 값을 설정하여야 함을 알 수 있다. 또한 서브트리 수가 늘어남에 따라 유사성에 끼치는 영향력인 γ 값이 α 를 고려하여 결정되어야 함을 알 수 있다.

5. 결론 및 논의

본 연구는 주관식 문제 채점을 위해 구문의미 분석이 필요함을 제시하였고 구문의미분석의 결과인 구문의미트리를 비교하는 비교기를 제안하였다. 이를 위해 비교기에 적용할 원칙을 설계하고 그 원칙이 타당함을 실험을 통해 검증하였다. 본 연구의 실험 결과 사람이 계산한 유사도와 비교하여 상관계수 0.94의 값의 시스템을 구축할 수 있었다. 또한 구문의미트리 유사성이 루트의 유사성, 서브트리의 유사성, 서브트리 수에 따른 영향력에 따라 결정됨을 알 수 있었다.

본 연구의 구문의미트리 비교기는 주관식 문제 채점에 바로 적용하여 채점의 질을 향상할 수 있을 것으로 기대된다. 또한 문서의 유사도를 계산하는 분야에도 활용할 수 있을 것이다. 앞으로 복문과 다양한 유형의 구문의미트리에 대한 비교기로 개선할 필요가 있다. 또한 주관식 문제 유형에 따른 유형별 처리 방법과 구문의미트리 비교기를 복합한 채점 시스템에 대한 연구도 필요하다.

참 고 문 헌

- [1] 장성호 · 강승식 (2003). 용어 선별기법에 의한 유사문서 판별시스템. **2003년도 정보과학회 봄학술발표논문집**. 30(1), 534-536.
- [2] 김혜숙 · 박상철 · 김수형 (2003). 단어가중치기 반 문서간 유사도 측정에 관한 연구. **2003년 한국멀티미디어학회 춘계학술발표논문집**. 198-201.
- [3] 조우진 · 오정석 · 이재영 · 김유섭 (2005). 의미 커널과 한글 워드넷에 기반한 지능형 채점 시스템. **정보처리학회논문지A**. 12-A(6), 539-546.
- [4] 박일남 · 강승식 · 노은희 · 김명화 · 성태제 (2013). 정답 템플릿 작성 방식에 의한 한국어 서답형 문항 자동채점 시스템. **정보과학회논문지 : 컴퓨팅의 실제 및 리더**. 19(12), 630-636.
- [5] 강승식 · 장은서 (2016). 학생 답안 분석과 정답 템플릿 생성에 의한 한국어 서답형 문항의 자동채점 시스템. **정보과학회 컴퓨팅의 실제 논문지**. 22(5), 218-224.
- [6] 김영철 · 최재영 (2005). 구문트리에서 키워드 추출을 이용한 프로그램 유사도 평가. **정보처리학회논문지A**. 12-A(2), 109-116.
- [7] Attali, Y. & Burstein, J. (2006). Automated Essay Scoring With e-rater® V.2. *Journal of Technology, Learning, and Assessment*. 4(3).
- [8] Akhil Langer, Rohit Banga, Ankush Mittal, L.V.Subramaniam (2010). Variant Search and Syntactic Tree Similarity Based Approach to Retrieve Matching Questions for SMS queries. *AND '10 Proceedings of the fourth workshop on Analytics for noisy unstructured text data*. 67-72.
- [9] 정창후 · 최성필 · 최윤수 · 송사광 · 전홍우 (2011). 술어-논항 구조의 패턴 유사도를 결합한 혼합 커널 기반 관계 추출. **한국인터넷정보학회**. 12(5), 73-85.
- [10] 김영철 · 김성근 · 염세훈 · 최중명 · 유재우 (2003). 구문트리 비교를 통한 프로그램 유형 복제 검사. **정보 과학회논문지 : 소프트웨어 및 응용**. 30(7-8), 792-802.
- [11] 손정우 · 박성배 (2009). 구조 및 의미 정보를 활용한 파스 트리 커널 기반의 온톨로지 정렬 방법. **정보 과학회논문지 : 소프트웨어 및 응용**. 36(4), 329-334.
- [12] 신미해 · 고방원 · 김영철 · 정진영 (2010). 문서구조 정보 기반의 유사도 측정. **한국컴퓨터정보학회 학술발표논문집**. 18(2), 499-502.
- [13] 강원석 · 황도삼 · Jung H Kim (2014). 구문 의미분석을 이용한 유사문서 판별기. **한국콘텐츠학회논문지**. 14(3), 40-51.
- [14] 강원석 (2015). 구문의미트리 비교기를 이용한 유사문서 판별기. **한국콘텐츠학회논문지**. 15(10), 636-646.
- [15] 김재훈 · 선충녕 · 홍상욱 · 이성욱 · 서정연 · 조정미 (1999). KTAG99: 새로운 환경에 쉽게 적용하는 한국어 품사 태킹 시스템. **제11회 한글 및 한국어정보처리 학술대회논문집**. 99-105.



강 원 석

1985 경북대학교
전자공학과(공학학사)
1988 한국과학기술원
전산학과(공학석사)

1995 한국과학기술원 전산학과(공학박사)
1995~현재 안동대학교 컴퓨터교육과 교수
관심분야: 컴퓨터교육, 자연어처리, 정보검색
E-Mail: wskang@anu.ac.kr