Journal of Korea Multimedia Society Vol. 24, No. 1, January 2021(pp. 106-116)
https://doi.org/10.9717/kmms.2020.24.1.106

# 구문의미 분석을 활용한 복합 문단구분 시스템에 대한 연구

강 원 석†

# Research on the Hybrid Paragraph Detection System Using Syntactic-Semantic Analysis

Won Seog Kang†

## ABSTRACT

To increase the quality of the system in the subjective-type question grading and document classification, we need the paragraph detection. But it is not easy because it is accompanied by semantic analysis. Many researches on the paragraph detection solve the detection problem using the word based clustering method. However, the word based method can not use the order and dependency relation between words. This paper suggests the paragraph detection system using syntactic-semantic relation between words with the Korean syntactic-semantic analysis. This system is the hybrid system of word based, concept based, and syntactic-semantic tree based detection. The experiment result of the system shows it has the better result than the word based system. This system will be utilized in Korean subjective question grading and document classification.

**Key words:** Paragraph Detection, Syntactic-Semantic Analysis, Clustering, Similarity

## 1. 서 론

주관식 문항 채점에서 채점의 질을 향상하기 위해서는 답안에 들어있는 논지의 흐름을 파악하고 이를 비교하는 과정이 필요하다. 이를 위해서 답안에 들어있는 문단을 구분하고 문단간의 관계를 도출할 필요가 있다. 또한 문서 분류에서도 문서의 주요 특징을 추출하기 위해 주요 문단을 찾아내는 작업을 한다. 이와 같은 필요성에 따라 문단구분에 대한 연구들이 진행되었다.

문단구분에 대해 영어권에서는 많은 연구가 진행되었다. 단어를 추출하고 추출한 단어를 벡터화한 후 벡터간의 유사도를 이용하여 문단을 구분하는 연구부터 다양한 연구가 진행되었다. 단어를 기반으로 한 연구들은 단어와 단어간의 의존적 관계를 활용하지 못하고 있다[1]. 이를 개선하고자 기계학습을 이용한 연구가 진행되어 규칙기반 방법들이 갖지 못하는 강인한 시스템 결과를 얻을 수 있다. 그렇지만 단어간의 관계가 다른 요인들과 복합되어 신경망에 내장되었기 때문에 구문규칙에 근거한 의존관계 등을 활용하였다고 보기에는 어렵다. Filippova[2]의 연구는 언어정보를 기계학습에 도입하나 언어정보가 구문구조를 표현하는 것이 아니라 단어간 문장거리, 문장의 첫째 단어 등으로 분해된 정보를 속성화하여 학습한 것으로 구문의미구조가 온전히 반영된 것이라고 할 수 없다.

※ Corresponding Author : Won Seog Kang, Address: (36729) Gyungdong-ro 1375, Andong, Kyung-pook, Korea, TEL : +82-54-820-5642, FAX : +82-54-823-1766, E-mail : wskang@anu.ac.kr
Receipt date : Nov. 26, 2020, Revision date : Dec. 31, 2020
Approval date : Jan. 8, 2021
† Dept. of Computer Education, Andong National University
※ This research was supported by a Research Grant of Andong National University

한국어에서는 문장 내에서 경계를 결정짓는 연구가 있었고 형태론적 특징을 근거로 문장을 구분하는 연구가 진행되었다[3,4,5]. 임희석[6]의 연구는 kNN 알고리즘을 활용하여 문장간의 경계를 결정짓고자 하였으나 개념이나 구문적 정보를 반영하지 않고 형태론적 정보만을 사용하였다. 이일주[7]의 연구는 문장의 대표 단어들을 벡터로 구성하고 그 대표 단어를 중심으로 문장을 클러스터링하여 문서를 요약하였다. 최준호[8]의 연구는 전문 용어를 중심으로 대표 문장을 찾아 핵심문장을 선별하였다. 이와 같은 연구들은 단어 중심의 기법을 사용하여 단어간의 구문의미관계를 활용하지 못하고 있다. 박사준[9]의 연구는 의미 중심으로 문단을 구분하는 방법을 적용하였으나 대상 언어가 영어여서 한국어에 적용할 수가 없다. 이에 본 연구는 단어간의 구문의미 관계를 분석하고 이를 이용하여 문단의 경계를 결정하는 문단구분 시스템을 제안한다.

본 연구는 한국어 문장에 대해 구문의미 분석을 실시하고 그 결과인 구문의미 트리들의 유사성을 비교하는 유사성 계산기를 활용하는 복합 문단구분 시스템을 설계, 구현한다. 이 시스템은 단어간의 구문, 의미관계를 분석하는 구문의미 해석기, 구문의미 해석의 결과인 구문의미 트리의 유사성을 계산하는 유사도 계산기, 그리고 단어기반, 개념기반, 구문의미 분석기반의 복합적인 문단경계 결정기로 구성된다. 본 논문의 2장에서 본 연구와 관련되는 관련 연구를 기술하고, 3장은 단어기반, 개념기반, 구문의미 트리 기반의 복합 문단구분 시스템을 기술한다. 4장은 시스템의 실험과 결과를 기술하고 5장은 결론과 논의사항을 기술한다.

## 2. 관련 연구

문단구분과 관련된 연구로 크게 경계구분, 단어중심 문단구분, 기계학습 중심의 문단구분, 개념중심의 문단구분으로 구분할 수 있다.

### 2.1 경계구분 연구

Yimin[10]은 문서의 비주얼적인 특성을 고려하여 문단의 경계를 인식하는 연구를 하였다. 이 연구는 언어 개념적 특성이 아닌 인덴테이션 등의 비주얼적인 특성을 기반으로 구분을 시도하였다. 박의규[3]는 연관된 단어들을 이용하여 문장 내에서 구간 분할을 시도하고 이를 통해 의존관계를 도출하고자 하였다. 이현영[4]은 한국어 구문분석에서 문장 내의 단문을 구분하는 방법을 이용하여 구문분석의 애매성을 해소하였다. 박용욱[11]도 구문분석에서 의존규칙에 근거한 구간분할을 이용하여 구문분석의 애매성을 해소하였다. 박주희[12]는 한국어 복합명사의 개념경계를 구분하기 위하여 사전과 언어적 정보를 이용하였다. 이 연구들은 문장 내의 경계구분에 대해 다룬 연구로 문단의 경계구분에 대한 것은 아니다.

임희석[6]은 문장의 경계를 구분하기 위하여 kNN 학습 알고리즘을 활용하였는데 개념이나 구문특징이 반영되지 않은 형태적 자질인 점, 마지막 음절, 길이 등을 사용하였다. 김주희[13]도 웹문서가 가지는 잡음에 대해서 문장의 경계를 구분할 수 있도록 음절 중심의 확률기반 기계학습을 적용하였다. 이충희[5]도 기계학습의 모델을 사용하여 문장의 경계를 인식하고자 하였다. 이 연구도 구두점, 종결어미, 경계 앞뒤 음절 등의 언어독립적인 자질을 사용하였다. 이 연구들은 형태론적 자질을 사용하여 문장의 경계를 구분한 연구로 구문 의미적 속성을 활용해야 할 문단 경계구분에는 맞지 않다.

### 2.2 단어중심 문단구분 연구

Kozima[1]는 문단을 구분하기 위해 LCP(Lexical Cohesion Profile)를 정의하고 이들의 상관관계가 낮은 부분을 문단의 경계로 하는 방법을 사용하였다. LCP는 고정된 크기의 창에 나타나는 단어들의 출현을 근거로 한 어휘 응집성을 정의한 것으로 단어들의 동시출현 정보는 이용할 수 있으나 단어간의 구조적인 관계 활용은 미흡하다. Kaufmann[14]은 일정한 범위의 context를 대표하는 대표 단어를 정의하고 그 대표단어로 벡터를 표현하여 유사도를 측정하는 방식으로 경계를 구분하였다. Tiedemann[15]은 질의응답 영역에서 원하는 답변의 구절을 찾을 때 기본적인 윈도우기반의 방법이 공기적 참조체인과 담화정보를 활용한 의미적 방법보다 나음을 보였다. 주원균[16]은 문서를 범주화하기 위하여 문단을 구분하고 구분한 문단의 타입을 이용하였다. 이 과정에서 한국어 형태소 해석의 결과인 단어들을 기반으로 타일을 정의하고 타일간의 유사도를 이용하여 병합하는 과정을 거쳐 문단을 인식하였다. 이 연구들은 단어를

기반으로 한 연구들로 단어와 단어간의 구문 의미적 관계를 활용하지 못하고 있다.

### 2.3 기계학습 중심의 문단구분 연구

Filippova[2]는 독일어를 대상으로 기계학습을 이용하여 문단을 구분하고자 하였다. 사용한 자질은 단어간 문장 거리, 관계사 위치, 마침표, 문장의 첫째 둘째 단어, 트리 깊이, 분기 수 등의 형태구문적 자질을 반영하여 학습에 활용하였다. 트리 깊이, 분기 수, 문장의 첫째 둘째 단어 등은 구문구조의 속성을 부분적으로 반영하고 있어 형태적 특징만을 사용하는 것보다는 나으나 구문구조의 속성을 온전히 반영하지 못해 부족한 점이 있다. Hashimoto[17]는 문단을 하나의 벡터로 표현하고 문단간의 유사도 계산방법으로 문단을 클러스터링하여 중심점을 찾고 중심점을 토픽으로 정하는 방법을 취한다. 이 과정에서 문서에서 문단을 구분하기 위하여 문서의 단어들을 입력으로 신경망을 통해 문단 벡터를 얻는다. 이 연구도 단어간의 관계에서 구문 의미적 구조가 반영되는 것이 아니라 단어간에 존재하는 다른 요소들과 복합되어 신경망에 반영되므로 부족한 점이 있다. Lai[18]는 대화체 문서에서 토픽간의 공기 관계, 구문적 특징 등에 기반한 어휘 정보와 순서 운율적 특징을 기반으로 학습망을 통해 문단구분을 시도하였다. 이 연구도 단어간의 구문 의미적 관계가 반영된 것이 아니라 부분적 구문속성이 다른 요소와 함께 복합되어 구문 의미적 관계 반영에는 부족하다. 또한 이 연구들은 한국어가 아닌 언어에 관한 연구로 한국어에는 적합하지 않다.

### 2.4 개념 중심의 문단구분 연구

Kaufmann[14]은 context를 대표하는 대표 단어를 정의하고 그 대표단어로 문서를 표현하여 경계를 구분하였다. 이 대표 단어는 context를 의미하는 것이므로 일종의 개념을 표현한다고 볼 수 있으나 정확한 개념을 나타낸다고 보기에는 어렵다. 이일주[7]도 지지도와 신뢰도를 이용한 공기정보 분석을 통해 대표어를 추출하고 대표어로 문장을 표현한 후 문장간의 클러스터링을 통해 문서를 요약하고자 하였다. 이때 사용한 대표어도 일종의 개념으로 볼 수 있으나 정확한 개념을 나타내기에는 부족하다. 박사준[9]은 WordNet을 이용하여 문서를 의미중심으로 문단화하고 문서의 대표 문단을 선택한 후 이를 이용하여 문서를 클러스터링하였다. 이 연구는 개념기반의 문단 분할을 시도한 것이다. 그렇지만 이 연구는 영어에 대한 연구이고 단어의 의미를 추출할 수는 있으나 단어간의 구문의미 관계를 이용하지 못하였다.

강원석[19]의 연구는 구문의미 트리의 유사성을 계산할 수 있는 구문의미 트리 비교기를 구현하여 주관식 문제 채점의 질을 높였다. 이 연구에서 더 나은 채점을 위해서는 문단을 구분하고 문단간의 구성관계를 이용해야 함을 알 수 있었다. 본 연구는 이를 개선하고자 구문의미 분석을 이용하여 문단간의 경계를 구분하는 문단구분 시스템을 설계, 구현하고자 한다.

## 3. 구문의미분석을 활용한 복합 문단구분 시스템

### 3.1 시스템 개요

문단구분 시스템의 구조는 Fig. 1과 같다. 전체 시스템은 경계 제어기, 형태소 해석기, 구문의미 분석기, 의미속성 추출기, 단어기반 유사도계산기, 개념기반 유사도계산기, 구문의미 트리 유사도 계산기, 구문의미 트리 리스트 유사도 계산기, 복합 유사도 계산기, 문단경계 결정기로 구성된다.

경계 제어기는 Fig. 2와 같이 대상 문서에서 문단의 경계가 될 지점을 기준으로 일정한 크기의 윈도우만큼을 잘라 두 세그먼트로 만들어준다. 시스템은 두 세그먼트를 대상으로 세그먼트간의 분리도를 계산하여 경계 지점인지를 결정하게 된다.

형태소 해석기와 의미속성 추출기, 구문의미 분석기는 문단의 경계후보의 두 세그먼트에 대해 분석을 실시한다.

형태소 해석기는 [20]에서 활용한 시스템을 이용하였으며 구문의미 분석기는 [19]의 연구 시스템을 이용하였다. 형태소 해석과 구문의미 분석기의 수행예는 Table 1과 같다.

형태소 해석기의 결과의 nq, ncn, jcs, xsn, jca, pvg, ep, ef, sf 등은 형태소의 태그이고 구문의미 분석의 결과에 있는 FINEN, SUBJ, TARG_FOR은 구문구조의 태그이다. 구문의미 분석기는 구문의미 분석에 필요한 제한 조건 등의 검사를 위해 단어에 대한 의미속성을 추출하는 의미속성 추출기를 사용한다. 또한 의미속성 추출기는 개념기반 유사도를 계산

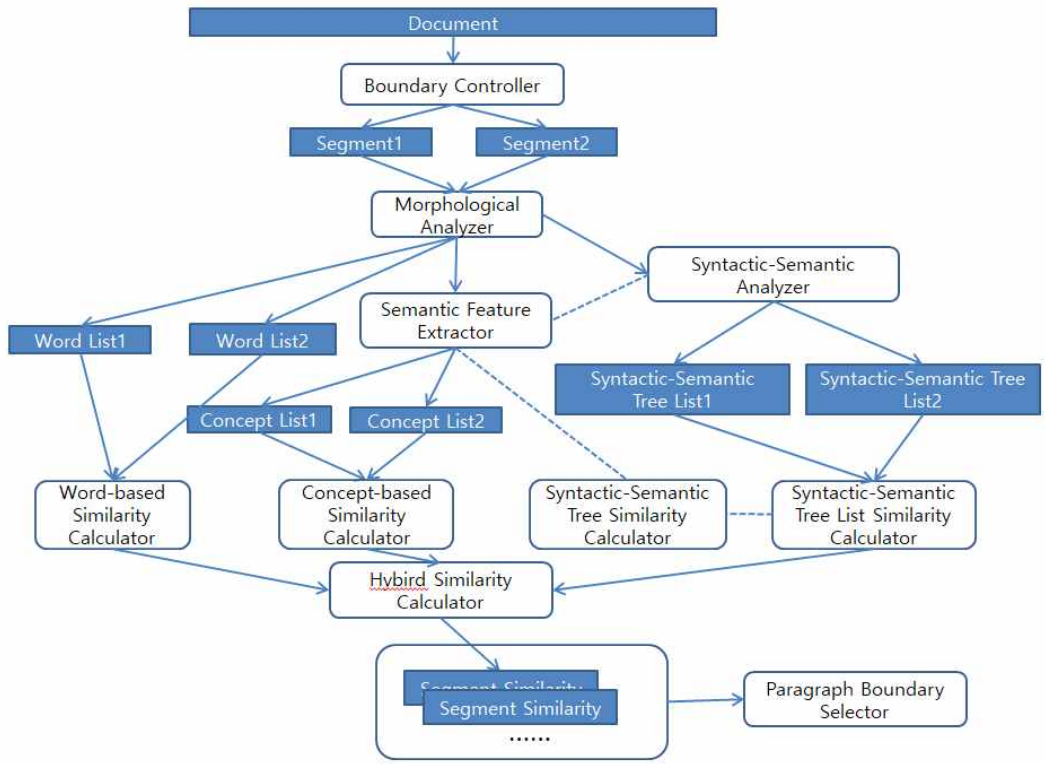


Fig. 1. Overview of the Boundary Decision System.


Fig. 2. Boundary Candidate Assignment.

하기 위해 필요한 문단의 개념속성들을 추출한다. 이 추출기는 단어에 대해 그 의미와 상위의미를 추출하는 것으로 [21]의 시소러스 사전을 이용하였다. 의미속성 추출기의 수행 예는 Table 2와 같다.

단어기반 유사도 계산기는 형태소 해석의 결과에서 추출된 단어를 벡터로 변환하고 벡터간의 유사도를 계산한다. 유사성의 계산식은 코사인 계수를 이용한 것으로 식 (1)과 같다.

$$wsim(A,B) = \frac{\sum_{i=1}^{n} w(A_i) \times w(B_i)}{\sqrt{\sum_{i=1}^{n} w(A_i)^2} \times \sqrt{\sum_{i=1}^{n} w(B_i)^2}} \quad (1)$$

A, B는 세그먼트1의 단어 리스트, 세그먼트2의 단어 리스트의 벡터이다. Ai는 세그먼트1의 i번째 단어이고 w(Ai)는 그 단어에 대한 가중치이다. 추출 단어를 벡터로 변환할 때 가중치 값은 다음 식과 같이 정의한다. 빈도수 tf(Ai)는 대상 문서에 단어 Ai가 몇 번 출현되었는지의 빈도수이고 역문헌 빈도수 idf(Ai)는 코퍼스 문서에서 단어 Ai가 나타난 문서수의 역수이다.

$$w(A_i) = tf(A_i) \times idf(A_i) \quad (2)$$

Table 1. Example of morphological and syntactic-semantic analysis.

| item | Content |
|---|---|
| sentence | 박교사가 학생들에게 말하였다. |
| morphological analysis | 박/nq+교사/ncn+가/jcs, 학생/ncn+들/xsn+에게/jca, 말하/pvg+었/ep+다/ef+./sf |
| syntactic-semantic analysis | ( FINEN 말하 ( SUBJ 교사 ) ( TARG_FOR 학생 ) ) |

Table 2. Example of semantic feature extraction.

| item | content |
|---|---|
| 교사 | animal animate-feature animate-thing domain education event feature human ... |
| 학생 | animal animate-thing domain education human physical-thing society thing 학생 |
| 말하 | event intellectual-behavior utterance 표현 |

개념기반 유사도 계산기는 두 세그먼트의 단어들에 대해 개념(의미 속성)들을 구한 후 개념리스트간의 유사도를 계산하는 것으로 단어기반 유사도 계산기와 같이 코사인 계수를 이용한다. CA, CB는 두 문단에 들어있는 단어들의 의미 속성들을 추출한 개념리스트의 벡터이다.

$$csim(CA, CB) = \frac{\sum_{i=1}^{n} w(CA_i) \times w(CB_i)}{\sqrt{\sum_{i=1}^{n} w(CA_i)^2} \times \sqrt{\sum_{i=1}^{n} w(CB_i)^2}} \quad (3)$$

### 3.2 구문의미 분석을 활용한 복합 문단구분

구문의미 트리 유사도 계산기는 단어간의 의존관계를 표현한 구문의미 트리간의 유사성을 비교하는 것으로 [19]의 도구를 사용한다. 이 도구는 구문의미 트리 간의 유사성을 비교하기 위해 식 (4)로 정의된다.

$$ST(T_1, T_2) = \frac{SL(r(T_1), r(T_2)) \times \alpha + \frac{\sum_{(T_{1i}, T_{2j}) \in T_1 \cap T_2} ST(T_{1i}, T_{2j}) \times 2}{n} \times \beta \times \gamma}{\alpha + \beta \times \gamma} \quad (4)$$

$ST(T_1, T_2)$는 구문의미 트리 $T_1$과 $T_2$의 유사도를 계산하는 함수이다. $ST(T_1, T_2)$는 루트의 유사성과 서브트리의 유사성의 합으로 구성된다. 두 요소에 대한 가중치를 부여하여 각 시스템의 성능을 점검할 수 있도록 하였다. 루트노드의 가중치는 $\alpha$로 표현하고 서브트리의 가중치는 $\beta$로 표현하였다.

$SL(r(T_1), r(T_2))$는 $T_1$의 루트와 $T_2$의 루트의 어휘의미 유사성을 계산하는 함수로 식 (5)와 같이 정의된다[19].

$$SL(a, b) = \frac{2 * |sem(a) \cap sem(b)|}{|sem(a)| + |sem(b)|} \quad (5)$$

이 함수는 다이스 계수를 이용하여 정의하였다. sem(a)는 단어 a의 의미 속성을 추출하여 얻은 의미 속성 집합을 나타내고 |sem(a)|는 의미 속성 집합의 원소수를 나타낸다.

서브트리의 유사성은 격이 같은 서브트리의 유사성의 합으로 정의된다. 식(4)의 Σ가 그 합을 표현하고 Σ 표기의 조건에 있는 교집합은 격이 같은 서브트리를 의미한다. 즉, $(T_{1i}, T_{2j}) \in T_1 \cap T_2$는 $T_1$ 트리의 i번째 서브트리가 $T_2$ 트리의 j번째 서브트리와 격이 같음을 나타낸다. 이때 격이 같은 트리는 양 구문 트리에 모두 있으므로 2로 곱하여 가중한 후 정규화를 위해 전체 서브트리 수 n으로 나눈다. 최종적으로 계산된 값에 서브트리 유사성 가중치 $\beta$를 곱한다.

$$n = |T_1 \cap T_2| \times 2 + |T_1 - T_2| + |T_2 - T_1| \quad (6)$$

n의 계산에 사용한 |T| 기호는 트리 T의 서브트리 수를 의미한다. n은 비교할 두 구문의미 트리에서 전체 서브트리 수를 나타내는 것으로 격이 같은 공통 서브트리 수의 2배에 $T_1$에만 있는 서브트리 수와 $T_2$에만 있는 서브트리 수를 더한 값이다.

$$\gamma = (1 - \frac{1}{e^{nk}}) \quad (7)$$

가중치 $\gamma$는 서브트리의 수가 증가함에 따라 가중치가 증가하나 한계 값까지만 증가하도록 설계하였다. k는 감마값을 결정짓는 감마계수 상수이다[19].

구문의미 분석기는 문장에 대한 구문의미 트리들을 생성해낸다. 문단에는 여러 개의 문장으로 구성되므로 여러 개의 구문의미 트리들이 들어있다. 두 문단 TA, TB에 대한 구문의미 트리들을 정의하면 다음과 같다.

$$TA = [TA_1, TA_2, TA_3, \ldots, TA_m]$$
$$TB = [TB_1, TB_2, TB_3, \ldots, TB_n]$$

구문의미 트리 유사도 계산기는 한 쌍의 구문의미 트리를 비교한다. 그러나 문단은 하나의 구문의미 트리로 구성되는 것이 아니라 여러 개의 구문의미 트리로 구성되므로 문단의 유사도를 비교하기 위해서는 구문의미 트리 리스트 유사도 계산기를 사용해야 한다. 구문의미트리 리스트 유사도 계산식 tsim은 식 (8)과 같이 정의된다[19].

$$tsim(TA, TB) = \frac{\sum_{i=1}^{m} cal(TA_i, TB) + \sum_{j=1}^{n} cal(TB_j, TA)}{m + n} \quad (8)$$

cal(TAi, TB)은 TB의 구문의미 트리들 가운데 구문의미 트리 TAi와 가장 유사한 트리를 찾아 그 유사도 값을 계산한다. 가장 유사한 것은 최대값이 될 것이다. 반대로 cal(TBj, TA)는 TA의 구문의미 트리들 가운데 TBj와 가장 유사한 트리를 찾아 그 유사도 값을 돌려준다. 각 값들을 합한 후 전체 구문의미 트리의 수로 나누어 평균값을 취한다. 이 방법으로 문단간의 구문의미 트리들의 유사성을 계산할 수 있다.

문단의 경계를 구분하기 위하여 문서를 세그먼트 단위로 구분한다. 이때 세그먼트 간의 유사도 계산을 위해 단어기반 유사도 값과 개념기반 유사도 값, 구문의미 트리기반 유사도 값의 복합 계산으로 유사도 계산을 한다. 그 계산식은 식 (9)와 같다.

$$sim(A,B)=\frac{\alpha\times wsim(WA,WB)+\beta\times csim(CA,CB)+\gamma\times tsim(TA,TB)}{\alpha+\beta+\gamma} \quad (9)$$

복합 유사도는 각 요소에 가중치를 곱하여 평균을 취한 값으로 정하였다. 각 가중치를 달리하여 여러 가지의 시스템으로 구성한 후 각 시스템을 실험하였다.

문단경계 결정기는 대상 문서의 문단 경계를 결정해주는 것으로 경계제어기와 맞물려 돌아간다. 경계제어기는 대상 문서에 대해 경계후보 지점을 기준으로 앞 세그먼트와 뒤 세그먼트로 찾아낸다. 세그먼트는 일정한 크기의 윈도우에 해당하는 것이다. 그 세그먼트는 형태소 해석과 구문의미 분석을 통해 단어 리스트와 개념리스트, 구문의미 트리 리스트로 변환된 후 복합유사도 계산기를 통해 두 세그먼트간의 유사도를 내어놓는다. 다른 경계 후보에 대해서도 두 세그먼트를 찾고 찾은 두 세그먼트간의 유사도를 계산하여 각 경계후보에 대한 앞 세그먼트와 뒤 세그먼트의 분리도들이 계산된다.

문단경계 결정기는 계산된 분리도들에 근거하여 편차가 크게 차이가 나는 부분을 경계로 결정한다. 그 계산식은 식 (10)과 같다. 이 식은 한 경계를 기준으로 현재의 분리도 값이 분리도들의 평균값보다 크게 차이가 나는 부분을 찾는 것으로 표준편차로 나누어 정규화하였다. 본 연구에서는 식의 한계값 Θ를 정하여 실험하였다.

$$\frac{sep(BD_i)-\dfrac{\sum_{BD_k\in BW_b} sep(BD_k)+\sum_{BD_j\in BW_f} sep(BD_j)}{|BW_b|+|BW_f|}}{std(BW_b,BW_f)}>\theta \quad (10)$$

BDi는 경계 i를 의미하는 것이고 sep(BDi)는 경계 i를 기준으로 나누어진 정해진 크기의 앞 세그먼트와 뒤 세그먼트간의 분리도를 의미한다. 분리도는 복합 유사도 계산값의 역수(1-유사도)로 정의된다. 편차가 크게 나는 부분을 찾기 위해서는 분리도들의 평균을 구해야 한다. BWb와 BWf가 분리도 값들의 평균을 계산하기 위한 범위를 의미한다. BWb는 경계 앞 부분의 범위이고 BWf는 뒤 부분의 범위이다. |BWb|와 |BWf|는 범위에 포함되는 분리도들의 개수를 말한다. std(BWb, BWf)는 편차를 정규화하기 위한 표준편차이다.

## 4. 실험 및 결과 분석

본 연구에서 제안한 문단구분 시스템의 실험을 위해 35개의 영역(건축, 경영, 경제, 교육 등)에서 1400개의 문장을 뉴스기사와 백과사전에서 발췌하였다. 영역을 선정할 때 전문용어가 출현하는 자연과학 영역은 배제하였다. 이는 일반의 영역에서 본 시스템의 효과를 보기 위함이다. 문단은 총 151개의 문단이고 한 문단은 평균적으로 9개의 문장으로 구성되었다. 그리고 한 문장은 평균적으로 27개의 어절로 구성되었다.

시스템의 성능은 정확률과 재현율에 가중치를 균등하게 배분한 F1-measure를 사용하여 검사하였다. 그 식은 다음과 같다.

$$\text{정확률}(P)=\frac{\text{옳은 문단경계의 수}}{\text{시스템결과 문단경계의 수}} \quad (11)$$

$$\text{재현율}(R)=\frac{\text{시스템결과 문단경계의 수}}{\text{정답 문단경계의 수}} \quad (12)$$

$$F1-measure=\frac{2\times P\times R}{P+R} \quad (13)$$

정확률과 재현율을 계산하기 위해 시스템은 식 (10)을 기준으로 문단 경계를 판단한다. 현재의 경계 후보에 대해 분리도 평균과의 편차가 한계값 이상이 되면 문단 경계이다. 정확률의 경우 본 시스템이 문단 경계로 인식한 결과 가운데 옳게 문단경계로 판단한 것이 얼마인지를 나타낸 식이고 재현율은 옳은 문단경계 가운데 몇 개가 시스템의 문단경계 결과에 들어있는지를 나타낸 식이다. 예를 들면 실험 데이터에서 옳은 문단 경계가 151개이고 시스템이 문단 경계로 제시한 것이 140이고 그 중 100개가 옳은 문단경계이면 정확률은 100/140이고 재현율은 100/151

이 된다.

본 연구에서 제안한 복합 문단구분 시스템의 성능을 검사하기 위해 먼저 단어 기반의 기존 시스템과 비교하는 실험을 하였다. 그리고 제안한 복합 문단구분 시스템이 어떠한 인자를 가질 때 최적이 되는지를 알아보기 위한 실험을 하였다. 시스템에 고려할 인자는 세그먼트 크기, 분리도 평균 대상 범위, 분리도 편차 한계값이다. 세그먼트의 크기는 식 (9)의 A, B의 크기를 의미하고, 분리도 평균 대상 범위는 식 (10)의 BWb, BWf를 의미하고, 분리도 편차 한계값은 식 (10)의 $\theta$를 의미한다.

- 실험1 : 제안한 복합 문단구분 시스템의 실험
- 실험2 : 세그먼트의 크기 별 실험
- 실험3 : 분리도 평균의 대상범위 별 실험
- 실험4 : 분리도 평균과의 편차 한계값 별 실험

### 4.1 제안한 복합 문단구분 시스템의 실험

본 시스템은 단어기반의 문단구분, 개념기반의 문단구분, 구문의미분석 기반의 문단구분 시스템을 구성하고 이를 복합한 복합문단구분 시스템을 제안하였다. 이 시스템의 성능을 평가하기 위하여 식 (9)의 가중치 $\alpha$, $\beta$, $\gamma$를 달리하여 실험하였다. Table 3의 시스템열의 Word는 단어기반 문단구분 시스템을 의미하고 Concept는 개념기반, Tree는 구문의미 분석 기반 문단구분 시스템을 의미한다. 복합(Hybrid) 시스템은 그 가중치 값별로 다르게 구축되었다. 그리고 시스템에서 사용한 세그먼트의 크기는 3 문장으로 지정하였고 분리도의 평균의 대상범위는 앞뒤 2개씩 모두 4개의 분리도들의 대상으로 하였으며 분리도 편차 한계값은 0.7로 하였다.

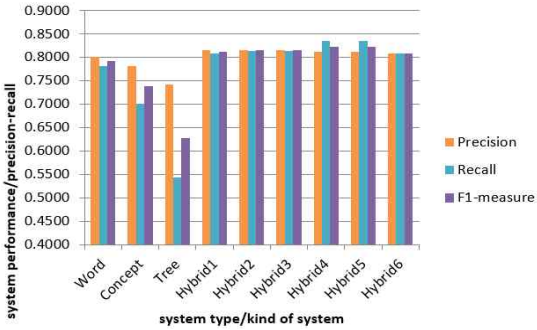


Fig. 3. Experiment result of hybrid systems.

실험 결과를 그래프로 표현한 것이 Fig. 3이다. 실험 결과 구문의미 분석 결과만을 이용한 실험은 단어기반 시스템이나 개념기반 시스템보다 못한 결과를 가져왔다. 그렇지만 복합한 시스템, 복합5(0.8234)의 경우 단어기반(0.7915)이나 개념기반 시스템(0.7394)보다 더 나은 결과를 보였다. 복합 시스템이 단어기반 시스템보다 나은 결과를 확인하기 위해 통계 검증을 해 본 결과 p-value가 0.03406으로 0.05보다 작으므로 95% 신뢰수준으로 복합 시스템이 단어기반 시스템보다 더 나은 결과를 보인다고 할 수 있다.

한편 시스템의 실험 결과가 정확률에는 각 기반 시스템별로 큰 차이가 없으나 재현율에는 구문의미 분석기반이 부족함을 보여주고 있다. 재현율이 부족하다는 것은 문단을 문단으로 구분하지 못한다는 것이다. 실패의 경우를 분석해 보니, 다른 문단이지만 같은 내용을 다루고 있어 이를 구분하지 못하는 경우가 있었다. 그리고 문단의 길이가 짧아 세그먼트내에 들어가지 못하는 경우가 발생하였다. 세그먼트의 길

Table 3. Experiment result of the hybrid paragraph boundary detection systems.

| System | α | β | γ | Precision | Recall | F1-measure |
|---|---|---|---|---|---|---|
| Word | 10 | 0 | 0 | 0.8019 | 0.7815 | 0.7915 |
| Concept | 0 | 10 | 0 | 0.7811 | 0.7020 | 0.7394 |
| Tree | 0 | 0 | 10 | 0.7425 | 0.5430 | 0.6273 |
| Hybrid 1 | 6 | 1 | 3 | 0.8147 | 0.8079 | 0.8113 |
| Hybrid 2 | 6 | 2 | 2 | 0.8155 | 0.8146 | 0.8150 |
| Hybrid 3 | 6 | 3 | 1 | 0.8162 | 0.8146 | 0.8154 |
| Hybrid 4 | 7 | 2 | 1 | 0.8112 | 0.8344 | 0.8226 |
| Hybrid 5 | 8 | 1 | 1 | 0.8126 | 0.8344 | 0.8234 |
| Hybrid 6 | 9 | 0.5 | 0.5 | 0.8090 | 0.8079 | 0.8085 |

이를 좀 더 작게 하면 되나 전체적 문단의 길이를 고려해야 하므로 다른 방법이 필요하다. 그리고 같은 내용을 다루는 다른 문단의 경우도 구문의미 분석 이외의 방법이 필요하다. 기존 연구에서 다룬 것과 같이 토픽 등의 언어적 정보를 구문의미 분석과 함께 활용하면 더 나은 결과를 가져올 것으로 예상된다.

### 4.2 세그먼트의 크기 별 실험

시스템에서 경계를 기준으로 앞 부분의 세그먼트와 뒤 부분의 세그먼트로 구분하고 이들의 유사도를 이용하여 분리도를 계산하게 된다. 이때 세그먼트의 크기를 어느 정도로 하는 것이 최적인지 알아보기 위해 실험을 하였다. 그 실험 결과는 Table 4와 같다.

Table 4의 시스템 이름에 붙은 2, 3, 4는 세그먼트 크기를 의미한다. 실험에 사용하는 시스템의 다른 선택요소는 동일한 값으로 실험하였다. 복합 시스템의 선택요소인 $\alpha$, $\beta$, $\gamma$ 값은 8, 1, 1로 정하였고 분리도의 평균의 대상 범위는 세그먼트 크기-1로 정하였으며 분리도 편차 한계값은 0.7로 하였다. 실험 결과, 단어기반과 개념기반과 달리 구문의미 분석기반 시스템의 경우 세그먼트 크기를 늘려갈수록 좋은 결과를 얻었다. 그렇지만 단어기반, 개념기반의 복합시스템이므로 전체를 고려한다면 세그먼트 크기가 3일 경우 가장 좋은 결과를 얻을 수 있었다. 또한 실험 시스템 가운데 성능이 가장 좋은 것도 복합시스템3임을 알 수 있었다. 이 결과는 실험 데이타가 더 다양할수록 변할 소지는 있는 것으로 보인다. 본 연구의 문장의 평균 길이는 27어절로 비교적 길다. 만약 좀 더 짧은 문장으로 실험할 경우 세그먼트의 크기는 좀 더 커질 것으로 예상된다.

Table 4. Experiment result of the systems per segment size.

| System | Precision | Recall | F1-measure |
|---|---|---|---|
| Word2 | 0.7697 | 0.7748 | 0.7722 |
| Word3 | 0.8019 | 0.7815 | 0.7915 |
| Word4 | 0.8016 | 0.7947 | 0.7981 |
| Concept2 | 0.7103 | 0.6755 | 0.6925 |
| Concept3 | 0.7811 | 0.7020 | 0.7394 |
| Concept4 | 0.7822 | 0.6623 | 0.7173 |
| Tree2 | 0.6881 | 0.4901 | 0.5724 |
| Tree3 | 0.7425 | 0.5430 | 0.6273 |
| Tree4 | 0.7622 | 0.5430 | 0.6342 |
| Hybrid2 | 0.7446 | 0.7881 | 0.7657 |
| Hybrid3 | 0.8126 | 0.8344 | 0.8234 |
| Hybrid4 | 0.8123 | 0.7881 | 0.8000 |

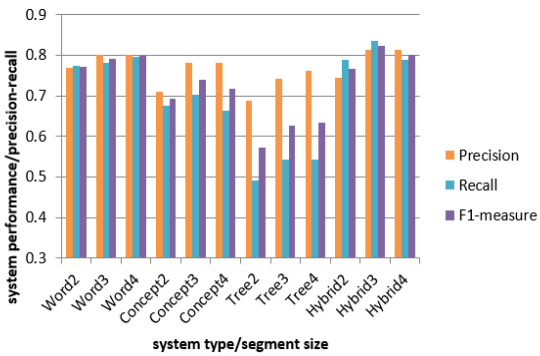


Fig. 4. Experiment result of the systems per segment size.

### 4.3 분리도 평균의 대상범위 별 실험

본 시스템의 문단경계 결정기는 먼저 문단 경계가 되는 지점에 대해 앞 뒤 세그먼트의 분리도를 계산한다. 그리고 각 경계 지점의 분리도 값이 급격히 변하는 부분을 찾아야 한다. 이를 위해 식 (10)에서 분리도의 평균과 편차를 계산한다. 이때 분리도들의 평균을 구하기 위한 대상 범위를 어느 정도로 정하는 것이 좋은지를 실험하였다. 그 결과는 Table 5와 같다.

Table 5의 시스템 이름에 붙은 1, 2, 3은 분리도의 평균의 대상범위의 크기를 뜻한다. 3의 경우 현재의 분리도 값 이외에 앞뒤 3개씩 모두 6개의 분리도들의 평균을 계산함을 의미한다. 실험에 사용하는 시스템의 다른 선택요소는 동일한 값으로 실험하였다. 복합 시스템의 $\alpha$, $\beta$, $\gamma$ 값은 8, 1, 1로 정하였고 세그먼트 크기는 3으로 실험하였으며 분리도 편차 한계값은 0.7로 하였다. 실험 결과, 단어시스템, 개념시스템, 트리시스템, 복합시스템 모두 분리도 평균의 대상을 앞뒤 2개씩 모두 4개로 하는 것이 가장 좋았다. 세그먼트 크기를 4로 하였을 경우도 검사한 결과, 분리도 평균의 범위를 세그먼트 크기-1로 하는 것이 좋음을 알 수 있었다.

### 4.4 분리도 평균과의 편차 한계값 별 실험

본 시스템의 문단경계 결정기는 식 (10)에서와 같이 현 경계 후보의 분리도가 평균 분리도보다 한계값

이상 클 때 문단 경계로 판단한다. 이 때 한계값($\theta$)을 어느 정도로 하는 것이 좋은지 알아보기 위해 실험을 하였다. 그 결과는 Table 6과 같다.

Table 6의 시스템 이름에 붙은 3, 5, 7, 9는 한계값을 의미하는 것으로 0.3, 0.5, 0.7, 0.9를 나타낸다. 실험에 사용하는 시스템의 다른 선택요소는 동일한 값으로 실험하였다. 복합 시스템의 $\alpha$, $\beta$, $\gamma$ 값은 8, 1, 1로 정하였고 세그먼트 크기는 3이고 분리도 평균의 대상 범위는 2로 하였다. Fig. 6은 Table 6을 그래프로 그린 것이다. 그래프에서 알 수 있듯이 한계값이 0.7일 때 가장 좋은 F1-measure값(0.8234)을 가져온다. 결론적으로 본 연구의 실험에서 복합 시스템의 선택요소인 $\alpha$, $\beta$, $\gamma$ 값은 8, 1, 1로, 세그먼트의 크기는 3으로, 분리도 평균을 계산하기 위한 범위는 세그먼트-1로, 분리도 평균과의 편차 한계값은 0.7로 하는 것이 가장 좋은 결과를 가져옴을 알 수 있었다.

Table 5. Experiment result of the systems per range of separation average.

| System | Precision | Recall | F1-measure |
|---|---|---|---|
| Word1 | 0.7926 | 0.7219 | 0.7556 |
| Word2 | 0.8019 | 0.7815 | 0.7915 |
| Word3 | 0.6924 | 0.2848 | 0.4036 |
| Concept1 | 0.7489 | 0.6689 | 0.7066 |
| Concept2 | 0.7811 | 0.7020 | 0.7394 |
| Concept3 | 0.6874 | 0.2053 | 0.3162 |
| Tree1 | 0.7182 | 0.4768 | 0.5731 |
| Tres2 | 0.7425 | 0.5430 | 0.6273 |
| Tres3 | 0.6946 | 0.2517 | 0.3695 |
| Hybrid1 | 0.7618 | 0.7682 | 0.7650 |
| Hybrid2 | 0.8126 | 0.8344 | 0.8234 |
| Hybrid3 | 0.7074 | 0.2384 | 0.3566 |

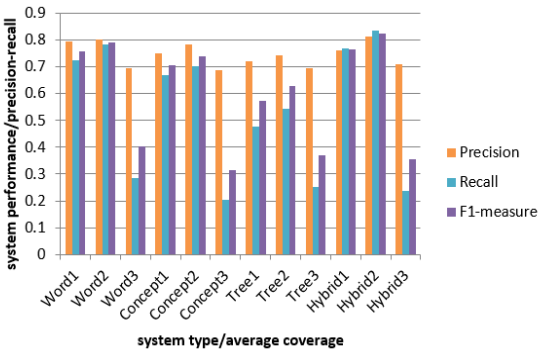


Fig. 5. Experiment result of the systems per range of separation average.

## 5. 결 론

본 연구에서 기존의 연구에서 반영하지 못하였던 구문의미분석을 활용하는 문단구분 시스템을 제안하였다. 이를 위해 한국어 구문의미 분석기와 결과인 구문의미 트리의 유사성을 계산하는 구문의미 트리 유사도 계산기를 구현하였고 이를 바탕으로 복합 문

Table 6. Experiment result of the systems per deviation threshold about separation average.

| System | Precision | Recall | F1-measure |
|---|---|---|---|
| Word3 | 0.6967 | 0.9073 | 0.7882 |
| Word5 | 0.7575 | 0.8275 | 0.7910 |
| Word7 | 0.8019 | 0.7815 | 0.7915 |
| Word9 | 0.8441 | 0.7351 | 0.7858 |
| Concept3 | 0.7003 | 0.8146 | 0.7531 |
| Concept5 | 0.7425 | 0.7483 | 0.7454 |
| Concept7 | 0.7811 | 0.7020 | 0.7394 |
| Concept9 | 0.8126 | 0.6755 | 0.7377 |
| Tree3 | 0.6609 | 0.6954 | 0.6777 |
| Tree5 | 0.7074 | 0.6291 | 0.6660 |
| Tree7 | 0.7425 | 0.5430 | 0.6273 |
| Tree9 | 0.7861 | 0.4503 | 0.5726 |
| Hybrid3 | 0.6924 | 0.9139 | 0.7879 |
| Hybrid5 | 0.7611 | 0.8808 | 0.8166 |
| Hybrid7 | 0.8126 | 0.8344 | 0.8234 |
| Hybrid9 | 0.8519 | 0.7682 | 0.8079 |

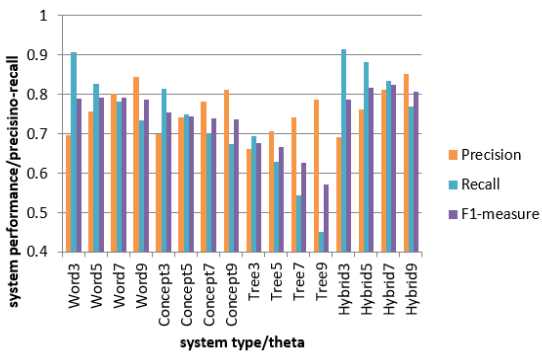


Fig. 6. Experiment result of the systems per deviation threshold about separation average.

단구분 시스템을 구축하였다. 실험 결과 본 연구의 구문의미 분석을 활용한 복합 문단구분 시스템이 단어기반, 개념기반의 시스템보다 더 나은 결과를 가져옴을 알 수 있었다. 또한 문단의 경계후보를 기준으로 분리도를 계산할 때 세그먼트의 크기를 3 문장으로 할 때 가장 성능이 좋아짐을 알 수 있었다. 그리고 분리도가 급격히 변하는 부분이 경계임을 판단하기 위해 분리도의 평균을 구해야 하는데 분리도의 평균을 구하는 범위의 크기를 세그먼트 크기-1로 하는 것이 가장 좋음을 알 수 있었고 분리도 평균과의 편차 한계값은 0.7로 할 때 좋은 결과를 가져옴을 알 수 있었다. 또한 더 좋은 문단구분을 하기 위해서는 구문의미 분석과 함께 토픽 등의 언어적 문단구분 실마리를 활용해야 함을 알 수 있었다.

본 연구를 통해 구축된 구문의미 분석기, 의미속성 추출기, 구문의미 트리 리스트 유사도 계산기는 주관식 채점 영역과 자연어 처리 영역에 활용도가 높을 것으로 예상된다. 또한 문서 분류의 영역과 문서표절의 영역에도 활용할 수 있을 것이다. 앞으로 문단구분 시스템의 성능을 향상하기 위해 토픽 등의 문단구분을 위한 언어적 실마리 정보의 연구가 필요하고 이를 구문의미 분석과 함께 반영하는 시스템의 연구를 해야 할 것이다.

## REFERENCE


[1] H. Kozima, "Text Segmentation based on Similarity between Words," *Proceeding of 31st Annual Meeting on ACL*, pp. 286-288, 1993.

[2] K. Filippova and M. Strube, "Using Linguistically Motivated Features for Paragraph Boundary Identification," *Proceeding of 2006 Conference on Empirical Methods in NLP*, pp. 267-274, 2006.

[3] E.G. Park, M.H. Cho, S.W. Kim, and D.R. Na, "A Method for Extracting Dependency Relations Using Chunking and Segmentation," *Proceeding of KISS Conference in Language Engineering Research*, Vol. 16, No. 1, pp. 131-137, 2004.

[4] H.Y. Lee and Y.S. Lee, "Korean Syntactic Analysis by Using Clausal Segmentation of Embedded Clause," *Journal of KISS: Software and Application*, Vol. 35, No. 1, pp. 50-58, 2008.

[5] C.H. Lee, M.G. Jang, and Y.H. Seo, "Improved Sentence Boundary Detection Method for Web Documents," *Journal of KISS: Software and Application*, Vol. 37, No. 6, pp. 455-463, 2010.

[6] H.S. Lim and G.H. Han, "Korean Sentence Boundary Detection Using Memory-based Machine Learning," *Journal of KCA*, Vol. 4, No. 4, pp. 133-139, 2004.

[7] I.J. Lee and M.G. Kim, "Document Summarization Based on Sentence Clustering Using Graph Division," *Journal of KISS B*, Vol. 13-B, No. 2, pp. 149-154, 2006.

[8] J.H. Choi and P.K. Kim, "Core Sentence Extraction for Expanding Knowledge Base," *Proceeding of Conference on Korea Multimedia Society*, pp. 446-449, 2010.

[9] S.J. Park and J.H. Kim, "Paragraph-based K-Means Clustering by using Meaning-based Paragraph Division," *Journal of Knowledge Information Technology and System(JKITS)*, Vol. 12, No. 1, pp. 157-164, 2017.

[10] C. Yimin, J. Adachi, and A. Takasu, "Detection of Paragraph Boundaries in Complex Page Layouts for Electronic Documents," *Proceeding of 74th Conference on JIPS*, pp. 539-540, 2012.

[11] Y.U. Park and H.C. Kwon, "A Study of Parsing System Implementation Using Segmentation and Argument Information," *Journal of Korea Multimedia Society*, Vol. 16, No. 3, pp. 366-374, 2013.

[12] J.H. Park and S.H. Maeng, "A Method for Establishing Korean Multi-word Concept Boundary Harnessing Dictionaries and Sentence Segmentation for Constructing Concept Graph," *Proceeding of Conference on KISS*, pp. 651- 653, 2017.

[13] J.H. Kim and J.Y. Seo, "Robust Method for Sentence Boundary Identification in Informal Documents," *Proceeding of Conference on*

*KISS*, Vol. 37, No. 1C, pp. 266-270, 2010.

[14] S. Kaufmann, "Cohesion and Collocation: Using Context Vectors in Text Segmentation," *Proceeding of 37th Annual Meeting of ACL on Computational Lingistics*, pp. 591-595, 1999.

[15] J. Tiedemann and J. Mur, "Simple is Best: Experiments with Different Document Segmentation Strategies for Passage Retrieval," *Coling 2008: Proceeding of 2nd Workshop on Information Retrieval for Question Answering (IR4QA)*, pp. 17-25, 2008.

[16] W.G. Joo, J.S. Kim, and K.S. Choi, "Automatic Text Categorization Using Passage-based Weight Function and Passage Type," *Journal of KIPS B*, Vol. 12-B, No. 6, pp. 703-714, 2005.

[17] K. Hashimoto, G. Kontonatsios, M. Miwa, and S. Ananiadou, "Topic detection using paragraph vectors to support active learning in systematic reviews," *Journal of Biomedical Informatics*, Vol. 62, pp. 59-65, 2016.

[18] C. Lai, M. Farrus, and J.D. Moore, "Automatic Paragraph Segmentation with Lexical and Prosodic Features," *Interspeech 2016*, pp. 1034-1038, 2016.

[19] W.S. Kang, "Research on Subjective-type Grading System Using Syntactic-Semantic Tree Comparator," *Journal of Korean Association of Computer Education*, Vol. 21, No. 6, pp. 83-92, 2018.

[20] J.H. Kim, C.N. Sun, S.W. Hong, S.W. Lee, J.Y. Seo, J.M. Cho, "KTAG99: Highly-Adaptable Korean POS Tagging System to New Environments," *Proceeding of 11th Conference on Hangeul and Korean Information Processing*, pp. 99-105, 1999.

[21] W.S. Kang, J.H. No, H.J. Je, H. Cho, S.Y. Hwang, and B.C. Jung, "Design and Implementation of a Keyword Relevant Word Extractor for Information Search Engine," *Proceeding of 2007 Fall Conference on KACE*, pp. 241-246, 2007.

강 원 석

1985년 2월 경북대학교 전자공학과 공학사
1988년 2월 KAIST 전산학과 공학석사
1995년 2월 KAIST 전산학과 공학박사

1994년~1995년 KAIST CAIR 위촉연구원
1995년~현재 안동대학교 교수
관심분야 : 자연어처리, 정보검색