Journal of the Korea Academia-Industrial cooperation Society
Vol. 16, No. 11 pp. 7718-7728, 2015

http://dx.doi.org/10.5762/KAIS.2015.16.11.7718
ISSN 1975-4701 / eISSN 2288-4688

워드넷을 이용한 문서내에서 단어 사이의 의미적 유사도 측정

강석훈[1*], 박종민[1]
[1]인천대학교 임베디드시스템공학과

Semantic Similarity Measures Between Words within a Document using WordNet

SeokHoon Kang[1*], JongMin Park[1]
[1]**Dept. of Embedded Systems Engineering, Incheon National Univ.**

요 약 단어 사이의 의미적 유사성은 많은 분야에 적용 될 수 있다. 예를 들면 컴퓨터 언어학, 인공지능, 정보처리 분야이다. 본 논문에서 우리는 단어 사이의 의미적 유사성을 측정하는 문서 내의 단어 가중치 적용 방법을 제시한다. 이 방법은 워드넷의 간선의 거리와 깊이를 고려한다. 그리고 문서 내의 정보를 기반으로 단어 사이의 의미적 유사성을 구한다. 문서 내의 정보는 단어의 빈도수와 단어의 의미 빈도수를 사용한다. 문서 내에서 단어 마다 단어 빈도수와 의미 빈도수를 통해 각 단어의 가중치를 구한다. 본 방법은 단어 사이의 거리, 깊이, 그리고 문서 내의 단어 가중치 3가지를 혼합한 유사도 측정 방법이다. 실험을 통하여 기존의 다른 방법과 성능을 비교하였다. 그 결과 기존 방법에 대비하여 성능의 향상을 가져왔다. 이를 통해 문서 내에서 단어의 가중치를 문서 마다 구할 수 있다. 단순한 최단거리 기반의 방법들과 깊이를 고려한 기존의 방법들은, 정보에 대한 특성을 제대로 표현하지 못했거나 다른 정보를 제대로 융합하지 못했다. 본 논문에서는 최단거리와 깊이 그리고 문서 내에서 단어의 정보량까지 고려하였고, 성능의 개선을 보였다.

Abstract Semantic similarity between words can be applied in many fields including computational linguistics, artificial intelligence, and information retrieval. In this paper, we present weighted method for measuring a semantic similarity between words in a document. This method uses edge distance and depth of WordNet. The method calculates a semantic similarity between words on the basis of document information. Document information uses word term frequencies(TF) and word concept frequencies(CF). Each word weight value is calculated by TF and CF in the document. The method includes the edge distance between words , the depth of subsumer, and the word weight in the document. We compared out scheme with the other method by experiments. As the result, the proposed method outperforms other similarity measures. In the document, the word weight value is calculated by the proposed method. Other methods which based simple shortest distance or depth had difficult to represent the information or merge informations. This paper considered shortest distance, depth and information of words in the document, and also improved the performance.

Keywords : Corpus statistics, Information content, Lexical database, Semantic similarity, WordNet

1. 서론

단어 사이의 의미적 유사성에 대한 연구는 자연언어처리와 정보 검색 분야에서 중요한 역할을 한다. 의미적 유사성은 컴퓨터 언어학과 인공지능에 관련된 다양한 응용프로그램에서 자주 사용되는 일반적인 방법이다. 예를 들어 단어의 의미 명확화, 단어의 철자 오류 검출, 단어의 철자 오류 정정, 이미지 검색, 문서 검색, 자동 하이퍼텍스트 연결, 문서의 분류이다[1].

두 단어의 유사성은 두 단어와 연관된 개념들 사이에

본 논문은 인천대학교 2013년도 자체연구비 지원에 의하여 연구되었음.
*Corresponding Author : SeokHoon Kang (Incheon National University)
Tel: +82-10-8735-1571 email: hana@incheon.ac.kr
Received October 19, 2015 Revised November 6, 2015
Accepted November 6, 2015 Published November 30, 2015

서 관련성을 계산하여 나타낸다. 많은 의미적 유사성 측정 방법은 다양한 연구자들에 의해서 개발되어 왔다. 다른 유사도 측정 방법들은 컴퓨터 인공지능 분야의 응용 프로그램과 정보 검색 분야에서 그 가치를 증명했다[1].

의미적 관계와 단어 또는 개념 사이의 유사성을 얻기 위해서는 둘 사이에서 눈에 띄지 않는 관계를 파악해야 한다. 그 관계로는 계층 관계, 연상 관계, 등가 관계 등이 있다. 특히 분류체계에서 자주 쓰이는 계층 관계는 IS-A 관계와 상위어-하위어 관계가 있다. 어휘 데이터베이스 중에서 워드넷은 이러한 관계를 갖고 있다. 이 워드넷을 이용하여 단어의 의미적 관계의 유사성을 계산 할 수 있다. 의미적 유사도 측정에 대한 연구가 다방면에서 진행됐음에도 두 개념 사이의 간단하면서 효과적인 의미적 유사도 측정방법을 만드는 것은 큰 일로 남아있다[26]. 일반 적으로 측정 방법들은 2가지 그룹으로 나눌 수 있다. 하나는 간선 기반의 방법과 정보량 이론 기반의 방법이다. 가장 직관적인 방법은 워드넷에서 분리된 두 개념 사이의 간선을 계산하는 것이다[26]. 하지만 이 방법은 인접한 개념에 대해서 모두 동일한 거리로 생각하여 계산하는 문제가 있다. 이 문제를 해결하기 위해서 최단거리와 깊이를 고려한 여러 방법들이 제안되었다[2,14].

정보량을 기반으로 두 개념에 대해 의미적 유사성을 측정하는 방법은 단어 자체의 정보량을 어휘 데이터베이스에서 계산하는 방법이다. 정보량 기반 접근 방법은 간선 거리가 변하는 문제에 대해서 의식하지 않는다. 그래서 표준기준에 대해서 향상된 상관계수 값을 얻는다. 이후에 정보량에 기초하여 많은 방법들이 제안되었다[5,11].

하지만 이러한 방법들은 특정 상황을 고려하지 않는다. 항상 보편적인 단어 사이의 관계만 의식한다. 그렇기 때문에 다음과 같은 문제가 발생 할 수 있다. 예를 들어, ‘automobile’과 ‘cushion’ 이라는 단어를 보면 두 단어가 유사한 의미를 갖지 않아 보인다. 만약 어떤 문서의 내용이 자동차와 관련이 깊고 ‘cushion’ 이 자동차의 좌석을 의미 한다면 두 단어 ‘automobile’과 ‘cushion’의 의미적 유사성은 높아질 것이다.

기존의 단어 사이의 의미적 유사성을 측정하는 방법들은 이런 상황을 생각하지 않았다. 사람이 일반적으로 생각하는 단어의 사이의 의미적 유사성을 구하였다. 하지만 단어의 의미는 항상 같지 않다. 상황과 문맥에 따라서 단어의 의미가 달라지고, 두 단어의 의미적 유사성도 달라진다.

그래서 이 논문에서 우리는 문서 내에서 두 단어 사이의 의미적 유사성을 측정하는 방법을 제안하였다. 일반적인 단어의 의미적 유사성을 측정하는 방법에 문서 내에서 단어의 의미적 유사성을 다시 평가 할 수 있도록 가중치 *Ip*값을 구하였다. 이 가중치 *Ip*값을 통해 단어가 포함된 문서의 상황에 맞도록 단어 사이의 의미적 유사성을 계산 할 수 있다.

2. 관련연구

지금까지 단어 사이의 유사성을 측정하는 방법들은 많이 제안되었다[2]. 1995년 어휘 데이타베이스인 워드넷이 발표 되었다. 워드넷은 상위 개념으로 올라가면, 포괄적인 의미를 갖고 하위 개념으로 내려가면 구체적인 단어를 갖는 계층적인 형태를 이루고 있다. 개념이란 워드넷에서 단어를 포함하는 노드를 의미 한다. 이 형태를 사용 할 수 있도록 워드넷을 이용한 의미적 유사성 평가 방법은 다양하게 나와 있다[2,4,5,11,14]. 대부분의 방법들은 단어 사이의 최단거리를 사용하거나 깊이를 사용한다. 이 방법을 통해, 단어 사이의 의미적 관계에 대한 유사성을 평가 할 수 있다.

의미 분류체계에서 2개의 개념 사이의 최소 거리를 구하여 단어 유사성을 구한다. 이 방법은 오로지 간선의 거리만 사용한다[14].

$$sim(A,B) = 2*Distance - dist(A,B)$$
$$dist(A,B) = \min imum\ number\ of\ edges\ separating\ A\ and\ B \quad (1)$$

위에서 Distance는 거리의 최댓값이다.

최소거리로만 의미적 유사성을 구하는 것에 한계가 있다. 그래서 간선거리와 깊이를 고려한 방법이 제안되었다[2].

$$(A,B) = \frac{2*d}{l_1 + l_2 + 2*d} \quad (2)$$

l_1, l_2는 A와B의 LCS까지의 최단거리 일 때 간선의 수이다. LCS는 Lowest Common Subsumer이다. 이는 워드넷에서 두 단어 A와B를 모두 포함하는 개념을 의미한

다. d는 LCS까지의 depth를 뜻한다.

Leacock and Chodorow는 최단거리를 기반으로 하는 방법을 연구하였고, 거기에 깊이의 최댓값을 추가 하였다[4].

$$sim(A,B) = -\log\frac{L}{2D} \quad (3)$$

L은 A와B사이의 최단 거리이다. D는 워드넷에서의 A와B의 LCS의 최대 깊이이다. 이 방법 또한 간선기반의 방식이기 때문에 성공적인 의미적 유사성 측정은 하지 못하였다.

간선 기반의 유사성 측정을 벗어난 방법으로 어휘 데이터베이스를 통해 단어의 정보량을 측정 하고자 하였다[5]. 이 방법은 다양한 논문에 영향을 미쳤다. 이를 통해 정보량 기반의 방법에 간선의 개념을 추가하여 계산하도록 하였다[10].

$$sim(A,B) = IC(A) + IC(B) - 2IC(LCS(A,B)) \quad (4)$$

IC는 단어 개념에 대한 정보량을 의미한다.

정보량과 간선개념의 수식을 수정하여 확률의 개념을 갖도록 하였다[11].

$$sim(A,B) = \frac{2IC(LCS(A,B))}{IC(A) + IC(B)} \quad (5)$$

지금까지 두 단어를 비교하는 많은 연구들이 있었다. 이 방법들은 5장에서 실험을 통해 성능비교를 할 때 다시 언급할 것이다.

3. 비선형적인 의미적 유사도

사람이 생각하고 판단하는 의미적 정보와 의미적 유사성은 선형적인 방법으로는 표현 할 수 없다. 그렇기 때문에, 비선형적인 방법으로 정보 들을 다루고 조합하여야 한다. 최단거리와 LCS의 깊이를 융합한 비선형적인 유사도 측정은 아래와 같다[1].

$$f_1(l) = e^{-\alpha l} \quad (6)$$

α는 상수이다, l은 단어 w_1과 w_2사이의 최단거리이다. 비선형함수에 최단거리를 사용하여 두 단어 사이의 유사성을 구하는 식이다. 그 값은 0 과 1사이이다.

다음은 깊이를 사용하여 두 단어 사이의 의미적 유사성을 구하는 수식이다.

$$f_2(h) = \frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}} \quad (7)$$

β는 상수이다. h 는 워드넷에서 LCS의 깊이이다. 이 수식 또한 비선형함수의 조합으로 나타내었다.

두 수식을 곱하여 최단거리와 깊이를 같이 고려하는 방법을 제안하였다.

$$sim(w_1,w_2) = f_1(l) \cdot f_2(h) = e^{-\alpha l} \cdot \frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}} \quad (8)$$

연구에서 진행한 실험들을 통해 가장 성능이 좋은 유사도 사용한 방법 이었다. 유사성 값이 커지기 위해서는 l은 줄어들어야 한다. 그 의미는 두 단어 사이의 거리가 가깝다는 것을 뜻한다. h는 커져서 LCS가 구체적인 의미가 되어야 한다. 하지만 경로 거리와 깊이를 이용한 방법은 계층적인 구조에서 개념들 사이의 위치를 기반으로 유사도를 계산하기 때문에, 큰 의미는 없다. 그렇기 때문에 단어 사이의 유사도를 평가 할 수 있는 정보를 추가해야 한다.

정보량 기반의 유사도 측정은 노드의 확률을 기반으로 하는 방법이다[5]. 다음은 단어 집합 w를 모두 포함하는 개념 c에 대한 수식이다.

$$IC = \log p(c) \quad (9)$$

$$p(c) = \frac{freq(c)}{N} \quad (10)$$

$$freq(c) = \sum_{w \in W(c)} count(w) \quad (11)$$

*p(c)*는 개념 c와 마주칠 확률이고, N은 어휘 데이터베이스에서 볼 수 있는 모든 명사의 종류이다. *freq(c)*는 분류에서 발생하는 개념 c의 모든 하위 개념의 빈도수이다. *W(c)*는 개념 c에 포함된 모든 단어 집합을 뜻한다. IC 값을 통해 워드넷의 두 개념 c_1의 c_2의 유사도를 측정할 수 있다. 그 값은 *subsumer concept*의 정보량의 최댓

값으로 계산된다.

$$sim(c_1,c_2)=\max[-\log p(c)] \quad (12)$$

$$wsim(w_1,w_2)=\max[sim(c_1,c_2)] \quad (13)$$

c_1과 c_2는 w_1과 w_2각각의 뜻을 포함 가능하다. 이 정보량 개념을 착안하여 새로운 방법을 제시하였다. 이 방법은 앞선 거리와 깊이를 이용한 방법과 같이 비선형적인 방법을 적용하였다.

$$f_3(wsim)=\frac{e^{\lambda wsim}-e^{-\lambda wsim}}{e^{\lambda wsim}+e^{-\lambda wsim}} \quad (14)$$

$\lambda>0$, λ는 상수이다. $f_3(wsim)$과 $f_1(l),f_2(h)$를 혼합하여 유사도 측정법을 제안 하였다.

$$\begin{aligned} sim(w_1,w_2) &= f(l,h,d) \\ &= f(f_1(l),f_2(h),f_3(d)) \\ &= f_1(l)\bullet f_2(h)\bullet f_3(d) \end{aligned} \quad (15)$$

이 수식을 통해 단순히 최단거리 와 깊이 기반의 두 단어 사이의 유사도 측정 방법에서 단어의 정보량을 적용한 방법을 얻었다. 최근에 단어 사이의 의미적 유사도 측정법은 단순히 단어를 비교하는 것보다 실제 상황에 적용하는데 목적을 두고 있다. 문서에서 단어 유사도를 평가하는 것처럼 그 활용 분야가 확장되고 있다. 하지만 위 방법들은 문서에서의 단어 유사도를 평가하기에는 적합한 방법이 아니기 때문에 그 상황에 맞는 다른 방법을 필요로 한다.

4. 의미적 유사도 측정을 위한 문서의 정보량

본 연구의 주 목적은 문서의 성향을 반영하여 문서 내 단어들의 유사도를 워드넷을 사용하여 구하는 것이다. 이전 연구들의 두 단어의 의미적 유사도 측정은 단어 쌍 사이에서 최단거리 와 깊이를 사용하여 계산한다. 이 방식을 사용하면 오로지 단어 쌍 사이만 생각하고 유사도 측정을 하게 된다. 그러면 이 유사도를 사용하기 위해서는 고려해야 할 상황들이 많아지게 되는 문제가 발생한다. 그렇기 때문에 본 논문에서는 여러 상황들 중에서 문서 내에서의 상황을 고려 할 수 있는 방법을 제안한다. 그 방법은 문서 내에서 단어의 정보량을 구하는 것이다.

4.1 문서의 정보

문서에서 얻을 수 있는 정보는 단어, 단어 빈도 수, 단어 개념 등이 있다. 이 정보들은 문서의 주제나 성격을 나타내는 중요한 역할을 한다. 예를 들어, 문서 A에서 'car'라는 단어의 빈도수가 20번이고, 나머지 다른 단어들의 빈도수가 미미 하다. 이 문서는 'car'와 관련된 문서일 확률이 높다. 실제로 이러한 아이디어를 통해 만든 방법이 문서의 주제를 분류하는 네이브 베이지안 분류기이다[6].

하지만 단순히 단어의 출현 확률만을 가지고 문서의 성격을 판단하기에는 무리가 있다. 예를 들어 문서 d에서 'car'라는 단어가 20번 출현 하고, 다른 모든 문서에서도 15번 이상 발생하였다. 그렇다면 실제로 자동차 관련 문서에서는 단어 'car'가 25번이상 발생한다고 하면 문서 d는 자동차와 관련성이 높다가 판단하기에 무리가 있다.

문서집합 $T=d_1,d_2,d_3...d_n$에서 d_1의 단어 w_1와 w_2가 있다고 하자 w_1의 빈도수는 15, w_2의 빈도수도 15이다. 문서집합 *T*에서 w_1을 갖는 문서는 10000개 w_2을 갖는 문서는 100개이다. 단어의 문서 d_1에서 중요도는 w_2가 더 높을 것이다. 그래서 *TF-IDF*라는 방법이 나오게 되었다[7].

이 단어가 문서 집합에서 중요한 단어인지 평범한 단어인지 판단하는 수식이다. 이러한 개념을 가지고 본 논문에서는 하나의 문서 안에서 단어들의 중요도를 평가하려고 한다. 위에서 말한 문서가 가지고 있는 정보들 중에서 단어 빈도수와 단어의 개념을 사용 할 것이다. 단어 빈도수란 그 단어의 출현도를 나타내는 용어로 *TF*라고 나타낸다. 그리고 아래에 한 가지 개념을 정의 한다. *CF*는 개념 빈도수(Concept Frequency)로 워드넷의 개념 *c*가 얼마나 출현했는가를 나타내는 것이다. 문서 안에서 개념 c인 단어들의 종류로 구할 수 있다.

문서에서 단어가 중요하다는 의미는 단어가 많이 언급되어 단어의 *TF*가 높거나, 하나의 개념을 여러 개의 단어로 언급하여 *CF*값도 높은 경우이다. 특정 단어가 많이 언급되거나 특정 개념을 여러 단어로 표현 했다는 의미이다. 이것은 단어에 해당하는 개념이 중요하다는 뜻으로 해석 할 수 있다. 문서 D에서의 CF 값을 수식으

로 표현하면 아래와 같다.

$$CF(C) = the\ number\ of\ C \quad (16)$$
$$C = \{w \in D | w는\ concept\ c에 \ 해당되는\ 단어의\ 종류\}$$

이제 단어의 중요도를 측정하기 위해서 단어에 맞게 표현해야 한다. 그 과정으로 확률적 표현을 사용 하였다. 문서 내에서 단어가 개념 *c*일 확률은 아래와 같다.

$$\Pr(c) = \frac{CF(c)}{the\ number\ of\ W} \quad (17)$$
$$W = w \in D | W는\ 단어의\ 종류$$

이 확률 값을 통해 단어의 중요도를 표현해야 한다. 그 수식은 *CF*값과 *TF*값을 고려하여 단어의 중요성을 상황마다 판단 할 수 있도록 하였다.

$$Ip(c,w) = \log(1 + \Pr(c)) \bullet TF(w) \quad (18)$$

Pr(c)의 값이 클수록 *Ip*의 값도 커져야 한다. 그 이유는 개념 *c*가 될 확률이 크다는 것은 그 문서가 c와 관련된 문서라는 뜻이기도 하다. 원래 이 *Ip(c,w)*은 정보량 이론에서 아이디어를 얻었다. 하지만 *IC*는 확률 값이 크면 클수록 그 값은 작아진다. 반대로 *Ip(c,w)*은 확률 값이 커지면 그 값도 커져야 한다. 그렇기 때문에 확률의 역을 취하지 않고 순수 확률 값으로 계산하였다. 그리고 *TF*값도 고려하여 *TF*값이 커질수록 큰 결과 값을 얻을 수 있도록 했다. 두 값의 곱을 통해, *TF*와 *CF* 결과를 나누었는데, 이 형태는 앞에서 언급한 *TF-IDF*와 유사한 형태를 취하고 있다.

하지만 *TF*와 *CF* 두 요소의 관계는 *TF-IDF*와는 다르다. 예를 들어 *TF* 값이 크고 *CF*값도 크게 되면 그 단어는 중요한 단어 이다. *TF* 작은데, *CF* 값이 크면 두 값이 클 때보다는 값이 작지만 중요한 단어를 뜻한다. 반대로 *TF* 크고 *CF* 작을 때도 마찬가지이다. 하지만 *TF* 와 *CF* 두 값 모두 작을 경우에는 중요하지 않은 단어 이므로 결과 값이 많이 작아야 할 필요가 있다.

단어 쌍 'gem-jewel'에 대해서 *wip*값을 구하였다. d_0는 d_1보다 상대적으로 *TF*와 *CF*가 높은 문서 집합이고, d_0은 d_1보다 낮은 문서 집합이다. 이 때의 wip값을 계산하면, $wip(d_0)$는 0.1561이고, $wip(d_1)$ 0.0223으로 $wip(d_0)$가 더 높은 것을 확인 할 수 있다. 이것은 *TF*와 *CF*값이 높음에 따라 값도 높아진다는 것을 의미한다.

4.2 문서에서 의미적 유사도

4.1장에서 구한 *Ip*값을 통해 실제 단어 사이 유사도를 평가한다. 두 단어에서 *Ip*값을 구하기 때문에, *Ip*값도 2개가 나오게 된다. 하지만 단어 유사도를 구하기 위해서는 1개의 값만 필요하다. 1개의 값을 고르기 위해 문서와 더 많은 개념과 단어를 공유하는 값을 선택해야 한다. 그래서 두 값 중 더 큰 값을 선택한다. 그 과정은 아래와 같다.

$$wIp(c,w) = \max_{(c,w) \in (c_1,w_1),(c_2,w_2)} Ip(c,w) \quad (19)$$

wIp는 문서에서 단어 중요도이다. 이 중요도만 가지고는 단어 의미 유사도를 구할 수 없다. 유사도를 구하기 위해서 가중치 함수를 적용해야 한다. 그 함수는 아래와 같다.

$$\widehat{W}(wIp) = e^{\lambda\, wIp} \quad (20)$$

λ는 상수이다. λ는 0과1 사이이다. 가중치 함수는 앞에 언급한 것 내용처럼 비선형함수로 하였다. 이 가중치를 이용하여 단어 의미 유사도를 구해야 한다. 최단거리와 깊이를 사용한 $f_1(l), f_2(h)$에 가중치를 적용하였다.

$$sim = \widehat{W} \bullet f_1(l) \bullet f_2(h) \quad (21)$$

*wIp*값을 통해 문서의 성격을 파악하거나 문서의 분류를 하게 될 때, 단순히 단어가 일치 하는 것을 계산하는 것이 아니라 문서의 성격을 그 문서의 정보를 통해 파악한다. 그리고 단어의 유사도를 계산 할 수 있는 새로운 방법을 얻었다.

*TF*값과 *CF*값이 높은 단어 쌍의 경우 높은 *wIp*값을 갖게 된다. 가중치 값 $\widehat{W}$ 은 *wIp*값이 커질수록 증가하는 비례 관계를 갖는다. 그렇기 때문에 *wIp*이 커지면, 높은 유사도 값을 얻을 수 있다. $\widehat{W}$ 의 함수로 $e^{\lambda wIp}$를 사용한 이유는 단어 중요도 *wIp*와 비례 관계를 갖는 가중치 함수여야 한다. 그래서 비선형적인 형태를 갖는 함수들 중에서 $f_1(l), f_2(h)$같은 밑을 갖는 $e^{\lambda wIp}$로 하였다. $f_1(l)$

의 경우는 *l*과 결과 값이 반비례 관계이다. 거리가 멀수록 유사도 값이 낮아진다. $f_1(l)$과 $\widehat{W}$ 은 이러한 차이를 갖는다.

예를 들어, 실제로 'automobile-car'를 포함하고 있는 문서 2개가 있다. 이때 한 문서의 주제는 'Car'였다. 다른 문서의 주제는 'Exports'였다. 이때 각 문서의 *wIp*값을 계산하였더니 'car'를 주제로 하는 문서의 경우 0.2769가 나왔다. 다른 문서의 경우에는 0.0313이 나왔다. *TF*와 *CF*값을 비교해 보면, *TF*값은 'Car'주제의 문서일 때, 'automobile'은 7번 'car는 9번이 나왔다. 'Exports'의 문서일 때는 'automobile' 2번 'car' 2번 나왔다. *CF*의 경우에는 'Car'주제의 문서일 때, 'automobile'은 3번 'car는 6번이 나왔다. 'Exports'의 문서일 때는 'automobile' 2번 'car' 2번 나왔다. 이렇게 *TF,CF*값이 차이 나기 때문에 *wIp*의 값도 큰 차이를 갖게 되었다. 이 내용을 통해서 *wIp*값이 큰 경우에 단어 쌍과 문서의 주제가 연관이 있다는 것을 확인 할 수 있다.

5. 실험

이 장에서는 4장에서 언급한 수식에 대해서 성능을 실험하고 그 결과를 분석한다. 진행할 내용은 이미 발표된 다른 유사도 측정법과 성능을 비교한다. 그리고 가장 좋은 성능을 낼 수 있는 상수를 얻기 위한 실험을 한다.

이 실험들을 하기 위해서 필요한 데이터는 단어 의미 유사성 측정분야에서 표준 기준으로 사용되는 MC set[9]과 RG set[8] 이다. 이 데이터들은 많은 연구자들이 단어의 의미적 유사도를 측정하기 위해 사용했다 [10,11, 12,13]. RG set은 단어 쌍에 대하여 사람이 직접 상관성을 채점한 표본이다. 컴퓨터의 계산 방법과 사람의 생각이 얼마나 일치하는지 평가하는 데이터로 사용된다.

RG set의 37개의 단어 쌍을 표본.1 이라고 하였으며, 표본.1은 본 논문에서 제안한 방법의 인자 값을 구하는 실험 데이터로 사용하였다. 표본.2는 MC set의 28개 단어 쌍이다. 본 논문의 성능을 측정하는 용도로 사용 하였다. 표본.1과 표본.2는. Table 1과 Table 2에는 각 단어 쌍과 단어 쌍 사이의 최단거리, 깊이 그리고 *IC* 값이 나와 있다. 두 단어 쌍의 유사성은 표본 1에 경우에는 RG rating으로 확인 할 수 있고, 표본 2는 MC rating으로 확인 할 수 있다.

Table 1. Sample 1 - RG rating, length, depth, IC

Word1	Word2	RG	length	depth	IC
Fruit	furnace	0.05	8	4	2.4934
autograph	shore	0.06	9	0	0
automobile	wizard	0.11	12	3	1.3696
mound	stove	0.14	6	4	2.4934
grin	implement	0.18	12	0	0
asylum	fruit	0.19	6	4	2.4934
asylum	monk	0.39	10	3	1.3696
graveyard	madhouse	0.42	14	2	1.1692
boy	rooster	0.44	11	5	1.8231
cushion	jewel	0.45	6	4	2.4934
Asylum	cemetery	0.79	11	2	1.1692
Grin	lad	0.88	11	0	0
shore	woodland	0.9	4	2	1.1692
boy	sage	0.96	5	3	1.9033
automobile	cushion	0.97	8	5	3.4451
mound	shore	0.97	4	3	6.1381
cemetery	woodland	1.18	8	2	1.1692
shore	voyage	1.22	14	0	0
bird	woodland	1.24	8	1	0.6144
Furnace	implement	1.37	7	4	2.4934
crane	rooster	1.41	7	9	6.9374
hill	woodland	1.48	5	2	1.1692
cemetery	mound	1.69	10	2	1.1692
glass	jewel	1.78	6	1	0.144
magician	oracle	1.82	6	3	1.9033
sage	wizard	2.46	5	3	1.9033
oracle	sage	2.61	5	4	6.601
hill	mound	3.29	0	6	10.1563
cord	string	3.41	1	6	8.6303
glass	tumbler	3.45	1	7	9.1267
grin	smile	3.46	0	6	8.1282
serf	slave	3.46	3	4	8.77
autograph	signature	3.59	1	6	11.7658
forest	woodland	3.65	0	4	11.0726
cock	rooster	3.68	0	13	10.6671
cushion	pillow	3.84	1	6	9.5683
cemetery	graveyard	3.88	0	8	9.8198

Table 2를 보면 28개의 단어 쌍과 MC rating이 나와 있다. 이 값을 보면 각 단어 쌍이 사람의 생각과 얼마나 비슷한지를 확인 할 수 있다. 단어 쌍의 거리와 깊이를 워드넷에서 구한 값이 나와 있다. 이전 연구들에서 이용한 IC 값도 확인 할 수 있다. 단순히 IC 값의 높고 낮음으로 단어의 유사성을 평가하기에는 무리가 있다.

실험에서 필요한 문서는 65개의 단어 쌍을 각각 웹에서 검색하였다. 검색한 문서는 d_0와 d_1으로 나누었다. d_0는 단어 쌍을 많이 포함하고 있는 집합으로 단어 쌍과 관련성이 높은 주제의 문서로 구성되어 있다. 반면에, d_0와 d_1보다 단어 쌍을 적게 포함한다. 단어 쌍과 관련성이 낮은 주제를 갖는 문서 집합이다.

Table 2. Sample 2 - RG rating, length, depth, IC

Word1	Word2	MC	length	depth	IC
cord	smile	0.13	10	1	0.7794
Rooster	voyage	0.08	23	0	0
noon	string	0.08	11	1	0.7794
glass	magician	0.11	7	1	0.6144
monk	slave	0.55	4	3	1.9033
coast	forest	0.42	5	2	1.1692
monk	oracle	1.1	7	3	1.9033
lad	wizard	0.42	4	3	1.9033
forest	graveyard	0.84	8	2	1.1692
food	rooster	0.89	15	1	0.6144
coast	hill	0.87	4	3	6.1381
car	journey	1.16	17	0	0
crane	implement	1.68	4	5	3.4451
brother	lad	1.66	4	3	1.9033
bird	crane	2.97	3	9	6.9374
bird	cock	3.05	1	9	6.9374
food	fruit	3.08	9	2	1.7798
brother	monk	2.82	1	9	10.1563
asylum	madhouse	3.61	1	9	10.6671
furnace	stove	3.11	2	4	2.4934
magician	wizard	3.5	0	8	11.0726
journey	voyage	3.84	1	9	7.1408
coast	shore	3.7	1	4	8.1022
implement	tool	2.95	1	6	6.3104
boy	lad	3.76	1	8	6.7419
automobile	car	3.92	0	11	7.0036
midday	noon	3.42	0	9	9.5685
gem	jewel	3.84	0	8	9.8198

단어 유사성을 비교하는 수단으로는 워드넷 2.1을 사용하였다. 특히 직접 실험을 하기 위하여 워드넷2.1의 JAWS[15] API를 사용하였다. 이것을 사용하여 두 개념 사이의 최단 거리 값과 깊이 값을 구하고 *wIp*값과 두 단어 사이의 유사도를 구하였다.

유사도 성능을 평가하는 상관관계 값은 Pearson's correlation을 사용 하였다. 이 방법은 많은 연구자들이 유사도 검증을 위하여 선택한 방법이다[17,18].

$$\rho(S_1,S_2)=\frac{cov(S_1,S_2)}{\sigma_{S_1}\sigma_{S_2}} \tag{22}$$

σ_{S_1}과 σ_{S_2}는 비교 할 두 가지 유사도에 대한 표준 편차이다. $cov(S_1,S_2)$는 S_1,S_2의 공분산이다. 상관관계 값이 1에 가까울수록 두 유사도 결과가 일치한다는 것이다. 실험에서 S_1은 표본.1의 human judgment rating이고 S_2는 실험할 각 유사도 값이다. 상관계수의 결과 값이 만약 1에 가깝다면, 그 것은 표본.1의 human judgment rating과 유사도가 일치하여 값을 잘 반영한다는 뜻이고, 반대로 0에 가깝다면, human judgment rating을 제대로 반영하지 못하여 전혀 상관없는 값이라는 뜻이다. 이 상관관계 값을 통해 유사도 성능을 평가 한다.

Table 3. Sample 2 - $wIp(d_0)$와 $wIp(d_1)$

Word1	Word2	$wIp(d_0)$	$wIp(d_1)$
cord	smile	0.2748	0.0741
Rooster	voyage	0.1521	0.0209
noon	string	0.4082	0.0513
glass	magician	0.2598	0.0732
monk	slave	0.1297	0.0325
coast	forest	0.2894	0.0179
monk	oracle	0.1297	0.0325
lad	wizard	0.2847	0.0876
forest	graveyard	0.2748	0.0732
food	rooster	0.2791	0.1078
coast	hill	0.1873	0.0442
car	journey	0.2388	0.0591
crane	implement	0.2432	0.0221
brother	lad	0.1928	0.0683
bird	crane	0.4061	0.0995
bird	cock	0.2041	0.0760
food	fruit	0.2703	0.0678
brother	monk	0.3345	0.0468
asylum	madhouse	0.1774	0.0580
furnace	stove	0.2682	0.0504
magician	wizard	0.1197	0.0424
journey	voyage	0.2208	0.0428
coast	shore	0.3371	0.0236
implement	tool	0.3690	0.0545
boy	lad	0.1650	0.0460
automobile	car	0.2769	0.0313
midday	noon	0.1918	0.0442
gem	jewel	0.1561	0.0223

5.1 인자 값 설정

인자 값을 구하기 위한 실험이다. $f_1(l),f_2(h)$ 유사도 값을 최고로 잘 나타내는 상수 값 α와β를 먼저 구한다. 그 후에, $\hat{W}(wsim)$에서 상수 값 λ를 구한다. 표본.1의 단어 쌍에 대해 $f_1(l),f_2(h)$의 유사도 값을 구하여 표본.1과 상관관계 값을 Fig.1a로 표현 하였다. X축은 $f_1(l)$의 상수 α를 나타낸다. Y축은 $f_2(h)$의 상수 β를 나타낸다. Z축은 상관관계 값을 나타낸다. 상관관계 값을 구하여 성능을 평가 하였다.

Fig.1을 보면 2개의 그래프가 있다. Fig.1a는

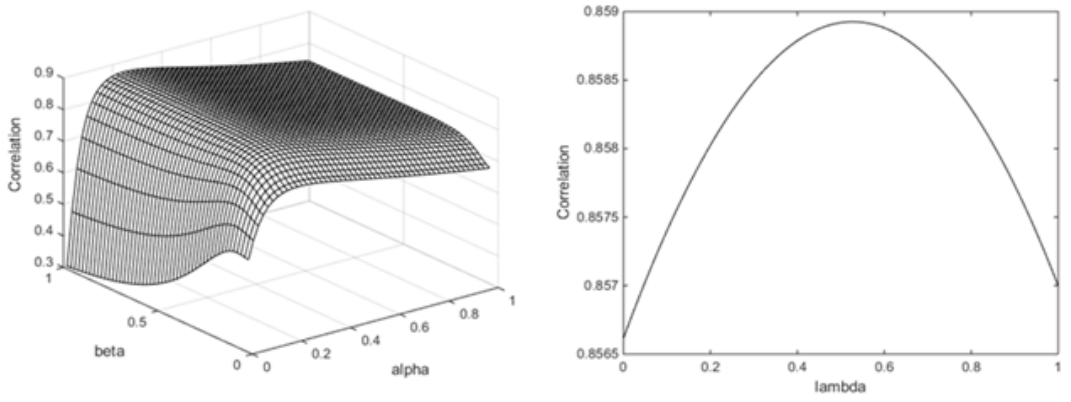

Fig. 1. Correlation coefficient versus $Sim(\widehat{W})$ parameters on Sample 1
(a) Correlation between $f_1(l) \bullet f_2(h)$ and human judgments rating versus α and β. (b) Correlation between $Sim(\widehat{W})$ and human judgments rating versus λ.

$f_1(l) \bullet f_2(h)$의 α와β를 구한 실험을 표현한 것이다. X축은 α에 대한 것이다. Y축은 β에 대한 것이다. Z축은 $f_1(l) \bullet f_2(h)$과 표본.1과의 상관계수 값으로 Z가 가장 큰 값을 갖는 X와Y의 값을 구하기 위한 그래프이다.

Fig.1을 통해, 최적의 상관계수 값을 갖는 α와β를 먼저 구해야 한다. 그 값을 상수로 정함으로써 $f_1(l) \bullet f_2(h)$와 표본.1의 상관관계를 구한다. 그 후에 인자 λ를 정해야 한다. Fig.1a를 보면 3차원 그래프에서 가장 높은 상관계수 값을 갖는 때는 α = 0.23 이고, β = 0.99 이다. $f_1(l) \bullet f_2(h)$는 이 때에 가장 큰 상관관계 값을 갖는다.

α와β을 통해 λ를 구한다. 최적의 λ를 구하여 제일 성능이 좋은 유사도 식을 완성하는 것이 목표이다. 성능을 가장 좋게 하는 것이 목표이기 때문에 단어 쌍을 많이 포함하고 있는 문서집합 d_0를 사용하였다. Table 3을 보면 d_0와 d_1에 대한 *wIp*값이 있는 것을 확인 할 수 있다. 이 *wIp*값을 통해 $\widehat{W}$ 값을 구하고 유사도 값을 계산하였다. 계산한 유사도 값과 표본.1의 상관계수 값을 구하였다. Fig.1b를 보면, X축은 $Sim(\widehat{W})$의 상수 lambda 이다. Y축은 상관계수 값이다. 가장 높은 상관계수 값을 갖는 경우는 Fig.1b를 보면 λ = 0.52 일 때이다. 그 상수 값을 통해서 표본.2와 상관계수 값을 구하면 그 값은 0.8571이다. 실험 집합 d_0에 대한 실험을 통해서 얻은 인자 $\alpha,\beta,\lambda\lambda$는 가장 좋은 성능을 얻는 설정이다. 물론 이 설정은 실험 세트에 따라서 바뀔 수 있다. 하지만 다른 유사도 측정 방법보다 본 논문이 제안한 방법의 성능이 더 좋다는 것은 확인 할 수 있다.

5.2 정보량 기반의 유사도 측정 방법의 성능 비교

이제 다음 실험으로 다른 방법들과 실험 결과를 비교한다. 그 대상으로는 Li et al.에서 실험한 수식들 중에서 정보량을 사용한 수식들과 비교하였다. 그 수식은 다음과 같다.

비선형적인 정보량 측정 방법과 선형적인 방법의 조합이다.

$$sim_1 = (2M - l) \bullet f_3(wsim) \tag{23}$$

M은 최대 깊이 값을 뜻하며, 이 때의 M 은 16이다. l은 두 단어 사이의 최단거리를 나타낸다. 표본.1과 실험을 통해서 얻은 λ=0.01이다. 표본.2와 상관계수 값 0.8241를 갖는다.

비선형적인 정보량 측정 방법과 거리와 깊이를 사용하는 선형적인 방법의 조합이다.

$$sim_2 = (\alpha(2M - l) + \beta d) \bullet f_3(wsim) \tag{24}$$

α와β는 [0 1] 이고, 표본.1에서 얻은 α = 0.52 , β = 0.12, λ가 0.01일 때 최고 값을 갖는다. 표본.2와

human similarity judgment의 상관계수 값은 0.8251이다.

비선형적인 정보량 측정 방법과 최단거리를 사용하는 비선형적인 방법의 조합이다.

$$sim_3 = e^{-\alpha l} \cdot f_3(wsim) \quad (25)$$

Sample.1에서 구한 인자 값은 α = 0.14, λ = 0.10이다. 그 때의 최고 상관관계 값은 0.8482 이다.

비선형적인 정보량 측정 방법과 최단거리와 깊이를 사용하는 비선형적인 방법의 조합이다.

$$sim_4 = \frac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} + e^{-\beta h}} \cdot f_3(wsim) \quad (26)$$

β는 (0 1] 이고 표본.1에서 얻은 β = 0.34 λ가 0.01이며, 그 때의 상관계수 값은 0.8298이다.

정보량을 사용한 위 4개의 방법들과 본 논문이 제안한 무서 정보량 방법을 비교하면, 본 논문의 방법이 표본.2와의 상관계수 값 0.8571로 더 높은 것을 확인 할 수 있다.

5.3 유사도 성능 비교

우리는 본 논문의 방법을 검증하기 위해서 여러 가지 실험을 진행하였다. 이제 그 실험 결과에 대해서 확인하려고 한다. 확인하고자 하는 내용은 크게 2가지 이다. 먼저, 본 논문의 유사도 성능을 다른 유사도 측정 방법들과 비교하여 향상되었는지 확인한다. 그리고, 문서의 성격을 잘 반영하여 단어 쌍의 유사도를 나타냈는지도 확인한다.

Table 4를 보면 왼쪽에는 여러 가지 단어 유사도 측정 방법들이 나와 있고, 오른쪽에는 그 값과 MC set의 28개의 단어 쌍과 상관계수 값이 나와 있다. 각기 사용한 방법이 다르다. 크게 간선 계산 방식과 정보량 기반의 방법으로 나눌 수 있다. 이렇게 여러 가지 방법과 비교하였다. Table 4를 보면 대부분의 간선 계산 방식은 상관계수 값이 상대적으로 낮은 것을 확인 할 수 있다. 이 뜻은 의미망에서 두 단어를 비교할 때 간선 거리만으로 판단하는 것은 좋지 않다는 것이다. 또, 정보량 기반 방식은 간선 계산 방식 보다는 상대적으로 상관계수 값이 높게 나왔지만, 이 것 또한 충분치 못하다.

이러한 문제를 해결하기 위해서는 워드넷에서 두 단어 사이의 간선 관계를 고려해야 하고, 정보량도 함께 생각해야 한다. 본 논문에서 제안한 최단거리, 깊이 그리고 문서에서 단어 가중치를 고려한 방법이 다른 방법들 보다 유사도 성능이 높게 나온 것을 확인 할 수 있다.

Table 4. Correlation Result and type

Similarity method	Type		Correlation
Rada et al	Edge	[1]	0.6640
Wu and Palmer	Edge	[11]	0.8030
Li et al.	Edge	[1]	0.8550
Leacock & Chodorow	Edge	[17]	0.7400
Resnik	IC	[11]	0.7950
Lin	IC	[11]	0.8340
Jiang and Conrath	IC	[11]	0.8280
sim_1	IC	[1]	0.8241
sim_2	IC	[1]	0.8251
sim_3	IC	[1]	0.8482
sim_4	IC	[1]	0.8298
Our method			0.8571

다음으로 확인 할 내용은 문서의 주제에 따른 단어 쌍의 가중치 값이 제대로 작동하는지 성능을 비교 하는 것이다. 예를 들어, 단어 쌍이 'car-automobile'인 경우에 문서의 주제가 자동차와 관련 있는 경우의 두 단어의 가중치 값과 관련성이 낮은 주제의 문서에서 두 단어의 가중치 값이 어떻게 차이가 있는지 확인하는 것이다.

Table 5을 보면, 각 단어 쌍을 포함하는 문서 세트 d_0와 d_1이 있다. d_0는 주제와 단어 쌍의 상관관계가 높은 집합이다. d_1는 단순히 주제와 상관없이 해당 단어 쌍만 문서에 포함된 집합이다. Table 5의 각 단어 쌍과 주제에 대한 *sim*값을 보면, 분명한 차이가 있는 것을 확인 할 수 있다. 'automobile'-'car'에서 d_0는 미래 자동차와 관련된 문서였다. 'car'와 관련된 문서였다. 반면에 d_1의 문서는 자동차와 관련이 있기 보다는 수출과 관련이 있는 문서였다. 이러한 결과로 두 *sim*값을 보면 0.9925와 0.8141으로 차이가 나는 것을 확인 할 수 있다.

기존까지 단순하게 최단거리 기반의 방법들과 깊이를 고려한 방법들은 한계가 있었다. 그렇기 때문에 정보량을 사용한 방법들이 나왔지만, 이 역시 정보에 대한 특성을 제대로 표현하지 못했거나 최단거리와 깊이가 같은 다른 정보를 제대로 융합하지 못했다. 하지만 본 논문의 방법은 최단거리 와 깊이 그리고 문서 내에서 단어의 정보량까지 고려하였고, 좋은 성능을 낸다는 것을 확인하였다.

Table 5. Word Semantic Similarity by topic of document

Word	pair	D	Topic	
Cord	Smile	d_0	TV show	0.0876
		d_1	Orthodontics	0.0789
rooster	Voyage	d_0	foreign rooster campaign	0.0000
		d_1	Christmas	0.0000
noon	string	d_0	Diary-music	0.0746
		d_1	basketball,	0.0620
Glass	magician	d_0	Magic	0.1733
		d_1	ex-boyfriend	0.1573
Monk	Oracle	d_0	Dalai Lama	0.4241
		d_1	Movie-matrix	0.4032
Lad	Wizard	d_0	Football-zidane	0.3543
		d_1	Hurdle	0.3076
Forest	Graveyard	d_0	tour	0.2127
		d_1	Wildlife	0.2022
Food	Rooster	d_0	Recipe	0.4597
		d_1	Shop information	0.4149
Coast	Hill	d_0	travel	0.1764
		d_1	Funding	0.1588
Car	Journey	d_0	transport	0.0278
		d_1	cook	0.0254
Crane	Implement	d_0	construct	0.4370
		d_1	traffic	0.4056
Brother	Lad	d_0	teenager	0.0000
			Cinematheque	0.0000
Bird	Crane	d_0	Bird migration	0.4522
		d_1	TV	0.4031
Bird	Cock	d_0	bird	0.4382
		d_1	Golf club	0.4108
Food	Fruit	d_0	nutrition,	0.6195
		d_1	Allergy	0.5282
Brother	Monk	d_0	Religious life	0.8835
		d_1	business	0.8266
Asylum	Madhouse	d_0	mental health	0.1398
		d_1	civil war	0.1258
Furnace	Stove	d_0	Heating with wood	0.9455
		d_1	maple syrup	0.8141
Magician	Wizard	d_0	Magic show	0.8713
		d_1	Comic Relief	0.8189
Journey	Voyage	d_0	life on seas	0.7252
		d_1	swimming pool	0.6476
Coast	Shore	d_0	seafarers	1.0642
		d_1	cyclones	1.0223
Implement	Tool	d_0	Education	0.8912
		d_1	government	0.8124
Boy	Lad	d_0	juvenile delinquency	0.9461
		d_1	Love story	0.8038
Automobile	Car	d_0	Car	0.9626
		d_1	Exports	0.8174
Midday	Noon	d_0	forecast	0.8657
		d_1	criminal	0.8138
gem	Jewel	d_0	Jewel	1.1549
		d_1	Beer	1.0164

6. 결 론

이 논문은 단어의 의미적 유사도를 측정할 수 있는 정보를 제안하였다. 그 정보는 단어 빈도수와 단어의 개념 빈도수이다. 이 정보를 토대로 문서 안에서 단어의 중요도를 측정하는 수식을 제안했다. 문서마다 같은 단어라도 중요도가 다르기 때문에 필요한 정보이다. 또, 중요도를 토대로 단어의 의미적 유사도를 측정하는 방법까지 제안하였다.

단어의 의미적 유사도 측정은 많은 관심을 받고 있지만, 정확도가 좋지 않고, 활용 범위가 특화되지 않은 문제가 있었다. 하지만 본 논문은 문서 안에서 단어들의 유사도를 비교 할 수 있는 방법을 제안하였기 때문에 이 문제를 해결 할 수 있다.

실험을 통해 수식에 맞는 최적의 설정을 찾고, 기존에 제시된 여러 유사도 측정 방법들과 상관계수 값을 비교하여 유사도 측정의 성능을 검증 하였다. 그 결과 기존의 방법보다 성능이 향상된 것을 확인 할 수 있었다. 그리고 실제 문서에서 문서의 성향을 반영하여 단어 사이의 유사성을 재평가 할 수 있는지 또한 실험을 통해 확인 하였다. 그 결과 문서의 주제와 단어 쌍이 관련이 있는 문서의 경우 단어 쌍과의 유사도가 높은 것을 확인 할 수 있었다.

기존의 방법은 문맥 또는 문서 같은 단어가 속해있는 상황을 고려하지 않고 단어 사이의 의미적 유사도를 측정하는 문제가 있었다. 이러한 문제를 본 논문의 방법을 통해 개선하였다.

본 논문의 방법은 문서 내에서 의미적 유사성을 측정하여 문서라는 특정 상황을 고려하도록 하였다. 그 방법으로 문서의 성향을 반영 하여 단어의 의미적 중요도를 다시 평가 할 수 있는 가중치 Ip값을 구하였다. 이 Ip값을 가중치로 사용하여 두 단어 사이의 유사도 값을 계산한다.

이 유사도 값을 통해 문서 내에서 단어의 의미적 관계를 다시 파악 할 수 있다. 다음 연구에서는 성능을 더 향상시키고, 활용 범위 또한 넓힐 것이다.

References

[1] Yuhua Li, “an approach for measuring semantic similarity between words using multiple information

sources," IEEE Trans. Knowl. DataEng. 15(4) ,871-882 ,2003
DOI: http://dx.doi.org/10.1109/TKDE.2003.1209005

[2] Wu and M. Palmer, "Verb semantics and lexical selection," In: Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics, LasCruces, New Mexico , 133-138, 1994

[3] G.A. Miller, "wordnet: a lexical database for English," Comm. ACM, Vol. 38, no. 11, 39-41, 1995.
DOI: http://dx.doi.org/10.1145/219717.219748

[4] C.Leacock and M. Chodorow, "Combining local context and wordnet similarity for word sense identification," WordNet: An Electronic Lexical Database. The MIT Press, Cambridge, MA, 265‐283 , 1998

[5] P. Resnik, "Using information content to evaluate semantic similarity," Proc. 14th Int'l Joint Conf. Artificial Intelligence, 1995

[6] A. McCallum and K. Nigam "A comparison of event model for naive Bayes text classification", AAAI Workshop on Learning for Text Categorization, 1998

[7] R.Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval. Addison-Wesley, 1999.

[8] H. Rubenstein and J.B. Goodenough, "Contextual Correlates of Synonymy," Comm. ACM, vol. 8, 627-633, 1965
DOI: http://dx.doi.org/10.1145/365628.365657

[9] G.A. Miller and W.G. Charles, "Contextual Correlates of Semantic Similarity," Language and Cognitive Processes, vol. 6, no. 1, 1-28, 1991.
DOI: http://dx.doi.org/10.1080/01690969108406936

[10] J.J. Jiang and D.W. Conrath, "Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy," Proc. ROCLING X, 1997.

[11] D. Lin, "An Information-Theoretic Definition of Similarity," Proc. Int'l Conf. Machine Learning, July 1998

[12] M. McHale, "A Comparison of WordNet and Roget's Taxonomy for Measuring Semantic Similarity," Proc. COLING/ACL Workshop Usage of WordNet in Natural Language Processing Systems, 1998.

[13] P. Resnik, "Semantic Similarity in a Taxonomy: An Information-Based Measure and Its Application to Problems of Ambiguity in Natural Language," J. Artificial Intelligence Research, vol. 11, pp. 95- 130, 1999.

[14] R. Rada, H. Mili, E. Bichnell, and M. Blettner, "Development and Application of a Metric on Semantic Nets," IEEE Trans. Systems, Man, and Cybernetics, vol. 9, no. 1, pp. 17-30, Jan. 1989.
DOI: http://dx.doi.org/10.1109/21.24528

[15] B. Spell. Java API for WordNet Searching (JAWS). http://lyle.smu.edu/ tspell/jaws/index.html, 2009.

[16] JB Gao, BW Zhang and XH Chen, "A WordNet-based semantic similarity measurement combining edge-counting and information content theory," 2015

[17] Pirro,G. A Semantic similarity metric comibing features and intrinsic information content. Data & Knowledge Engineering, 1289-1308, 2009.
DOI: http://dx.doi.org/10.1016/j.datak.2009.06.008

[18] David D Sánchez, Montserrat Batet, David Isern, Aida Valls, Ontology-based semantic similarity: A new feature-based approach, Expert Systems with Applications 39 ,7718-7728, 2012.
DOI: http://dx.doi.org/10.1016/j.eswa.2012.01.082

강 석 훈(SeokHoon Kang) **[정회원]**

- 1989년 2월 : 한양대학교 전자통신공학과 (공학사)
- 1995년 8월 : 한양대학교 전자통신공학과 (공학박사, 인공지능)
- 2004년 3월 ~ 현재 : 인천대학교 정보기술대학 임베디드시스템공학과 교수
- 2000년 4월 ~ 현재 : (주) 테크에이스솔루션 대표이사

<관심분야>
임베디드시스템, 인공지능, 언어처리, 영상처리, 모바일소프트웨어, 웨어러블

박 종 민(JongMin Park) **[정회원]**

- 2013년 2월 : 인천대학교 임베디드시스템공학과 (공학사)
- 2015년 8월 : 인천대학교 대학원 임베디드시스템공학과(공학석사)

<관심분야>
임베디드시스템, 인공지능, 언어처리, 모바일소프트웨어