의미 애매성 해소를 이용한 WordNet 자동 매핑*

이 창 기, 이 근 배
포항공과대학교 컴퓨터공학과
경북 포항시 남구 효자동 산31번지
우: 790-784
{leeck,gblee}@postech.ac.kr

Automatic WordNet mapping using word sense disambiguation

Changki Lee, Geunbae Lee
Natural Language Processing Lab
Dept. of Computer Science and Engineering
Pohang University of Science & Technology
San 31, Hyoja-Dong, Pohang, 790-784
{leeck,gblee}@postech.ac.kr

요약

본 논문에서는 어휘 의미 애매성 해소와 영어 대역어 사전 그리고 외국언어에 존재하는 개념체계를 이용하여 한국어 개념체계를 자동으로 구축하는 방법을 기술한다. 본 논문에서 사용하는 방법은 기존의 개념체계 구축 방법들에 비해 적은 노력과 시간을 필요로 한다. 또한 상기한 자동 구축 방법에서 사용하는 어휘 의미 애매성 해소를 위한 6가지 feature도 함께 설명한다.

1 서론

일반적인 사람들이 가진 지식 베이스를 컴퓨터에 도입하여 컴퓨터가 이를 참조하여 자연어 문장을 이해할 수 있도록 하려는 노력이 많이 있었는데, 이러한 지식 베이스를 개념체계(ontology) 혹은 시소러스(thesaurus) 라고 한다. 영어권의 경우 이에 대한 연구가 오래 전부터 있어 왔고, 그 결과로 현재 주로 사용되고 있는 개념체계에는 Roget's Thesaurus와 WordNet 등이 있다.

이러한 개념체계를 구축하는 데에는 여러 가지 방법이 있을 수 있다. 그 중에서 가장 정확하고 신뢰할 수 있는 방법은 사람이 단어간의 상하위 관계를 구하여 수동으로 구축하는 것이다. 그러나 이러한 방법은 언어학자나 심리학자의 도움이 필요한 매우 어려운 작업으로 많은 시간과 인력을 필요로 한다. 이러한 이유로 기존에 존재하는 어휘 지식 정보를 가지고 자동이나 반자동으로 개념체계를 구축하려는 많은 시도가 있었다. 이러한 방법 중에 하나가 기존의 사전을 이용하여, 단어의 상위어를 뽑아내어, 이 상위어들을 연결함으로써 개념체계를 구축하는 것이다. 그러나 이러한 방법을 사용하기 위해서는 사전으로부터 정확한 상위어를 추출해야 하며, 또한 추출한 상위어가 다의어인 경우 올바른 의미를 선택하는 작업이 필요한데, 이 역시 매우 어려운 작업이다. 또한 이러한 방법은 사용하는 사전에 의존적이라는 문제가 있고, 사전의 정의문이 부정형으로 나오거나 상위어의 링크에 loop가 존재하는 등의 문제가 있다.

본 논문에서는 한국어 개념체계를 구축하기 위해서 다른 언어에 이미 존재하는 개념체계와 대역어 사전을 이용하여, 한국어 단어의 의미를 다른 언어의 개념체계에 매핑시키는 방법을 사용하며, 이때 다른 언어의 개념체계로의 매핑시 발생하는 어휘 의미 애매성 해소(WSD; word sense disambiguation)를 위하여 대역어 사전과 다른 언어의 개념체계로부터 구할 수 있는 여섯 가지의 feature를 제안한다. 이러한 개념체계 구축 방법은 개념체계 구축 문제를 대역어의 여러 의미 중에서 올바른 의미를 선택하는 문제인 Word Ssense Disambiguation 문제로 치환시키게 된다.

본 논문의 방법은 기존의 개념체계 구축 방법보다 훨씬 적은 시간과 비용을 필요로 하고, 상기한 사전으로부터 상위어를 추출하여 개념체계를 구축할 때 발생하는 여러 문제점 등이 발생하지 않게 된다. 그리고 다른 언어의 개념체계에 상하위 관계가 아닌 다른 관계 등이 포함되어 있을 경우, 이러한 관계를

* 본 연구는 과학재단 특정기초(1997.9 – 2000.8 #970-1020-301-3) 연구비 지원으로 이루어진 것임

그대로 사용할 수 있다는 장점이 있다. 수동 구축이나 사전의 상위어를 이용하는 방법에서는 이러한 관계를 구축하려면 상하위 관계를 구축하는 것보다 더욱 많은 시간과 비용이 들게 된다는 단점이 있다

2 어휘 의미 애매성 해소를 위한 다양한 휴리스틱

본 논문에서는 한국어 단어를 WordNet synset에 매핑시키기 위해서 6가지의 휴리스틱을 사용하여, 이들을 결합한 결과를 이용한다. 따라서 각 휴리스틱들은 언어적 지식이나 확률 정보의 일부를 사용하며 한영사전에 있는 모든 표제어에 적용되지 않을 수도 있다.

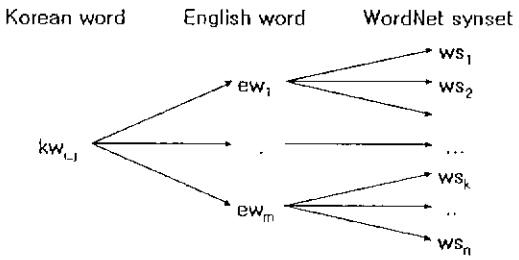


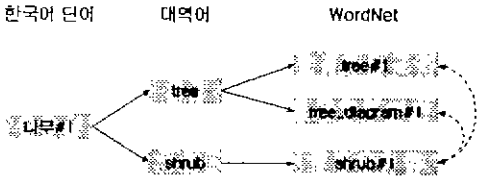
그림 1: Word-to-Concept Association

그림 1은 한국어 단어와 WordNet synset간의 관계를 보여준다. 한국어 단어 kw_i의 j번째 의미에 대해서 m개의 영어 대역어가 존재하며, 이 m개의 영어 대역어들마다 여러 개의 WordNet synset이 존재할 수 있으므로, kw_{ij}는 총 n개의 WordNet synset 후보를 갖게 된다. 본 논문에서 사용하는 휴리스틱들은 이 n개의 WordNet synset 후보에 적용이 되고 그 점수를 계산하게 된다.

2.1 Heuristic 1: Maximum Similarity

영어 대역어의 어휘 의미 애매성 해소를 위한 휴리스틱 1은 이전의 연구인 (이창기, 이근배 1999)에서 도용한 것이다. 이 휴리스틱은 하나의 한국어 단어 의미에 관한 영어 대역어들은 의미적으로 유사하다는 가정을 하여, 영어 대역어들의 WordNet synset 후보들간의 의미 유사도가 높은 synset 후보에 높은 점수를 준다. 이 휴리스틱은 하나의 한국어 단어 의미에 2개 이상의 영어 대역어가 존재할 경우에만 적용될 수 있다.

그림 2는 이 휴리스틱을 예를 들어 설명한 것이다. "나무"의 첫 번째 의미의 영어 대역어인 "tree"와 "shrub"의 의미 후보간의 유사도를 WordNet을 이용해 산술적으로 구해보면, "tree#1"과 "shrub#1"의 의미 유사도가 "tree_diagram#1"과 "shrub#1"의 의미 유사도보다 높으므로 "tree"의 의미 후보인 "tree#1"이 "tree_diagram#1"보다 높은 점수를 얻게 된다 (그림에서 Sim은 산술적으로 의미 유사도를 구하는 식이다).



Sim(tree#1, shrub#1) > *Sim*(tree_diagram#1, shrub#1)

Score(tree#1) > *Score*(tree_diagram#1)

그림 2: Heuristic 1

다음은 실제 휴리스틱 1의 점수를 계산하는 식이다.

$$H_1(s_i) = \max_{ew \in EW_i} \frac{1}{(n-1)+\alpha} \cdot \left(\sum_{j=1}^{n} support(s_i, ew_j) - 1 \right)$$

where $EW_i = \{ew \mid s_i \in synset(ew)\}$

위 식에서 $H_1(s_i)$는 synset s_i의 점수이며, s_i는 WordNet synset 후보이며, ew는 영어 대역어이고, n은 영어 대역어들의 수를 나타내고, $synset(ew)$는 영어 대역어 ew의 WordNet synset의 집합이다. 따라서 EW_i는 synset s_i를 가지고 있는 영어 대역어들의 집합이 된다. α는 영어 대역어의 수에 따라서 후보 synset의 기여도를 조정하게 된다. 즉 α 값이 증가하게 되면 영어 대역어의 수가 적은 쪽의 후보 synset들에 상대적으로 적은 weight가 주어지게 된다. α의 값은 실험을 통하여 0.5의 값을 주었다. $support(s_i, ew)$는 synset s_i와 대역어 ew간의 최대 의미 유사도를 구하며 다음과 같이 정의된다.

$$support(s_1, ew) = \max_{s \in synset(ew)} S(s_i, s)$$

$$S(s_1, s_2) = \begin{cases} sim(s_1, s_2) & \text{if } sim(s_1, s_2) \geq \theta \\ 0 & \text{otherwise} \end{cases}$$

의미 유사도가 임계값 θ 보다 낮은 경우에는 잡음으로 간주하여 무시하게 되며, 본 실험에서는 θ=0.3 을 사용하였다. $sim(s_1, s_2)$ 은 synset s_1와 synset s_2의 의미 유사도를 구한는 식으로 다음과 같은 식을 사용한다.

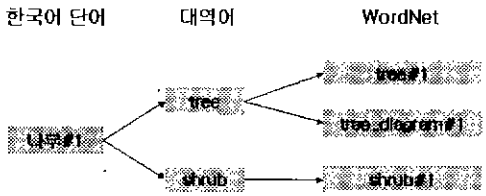
$$sim(s_1, s_2) = \frac{2 \times level(MSCA(s_1, s_2))}{level(s_1) + level(s_2)}$$

$MSCA(s_1,s_2)$는 synset s_1와 synset s_2의 가장 구체적인 공통 조상 노드를 의미하고, $level(s)$는 WordNet[1]의 root 노드로부터 synset s까지의 깊이를 나타낸다.

2.2 Heuristic 2: Prior Probability

영어 대역어의 어휘 의미 애매성 해소를 위한 휴리스틱 2는 하나의 대역어의 각 synset에 prior probability를 점수로 주게 된다. 즉, 영어 대역어의 synset 후보 중에서 의미 애매성이 적은 synset에 높은 점수를 준다.

그림 3에서 "나무"의 첫 번째 의미에 해당하는 대역어가 "tree"와 "shrub"가 있고, 대역어의 의미 후보의 수는 "tree"가 2개, "shrub"가 1개이므로, "tree"의 의미 애매성이 "shrub"보다 많다. 따라서 "shrub"의 의미 후보가 "tree"의 의미 후보보다 높은 점수를 얻게 된다.



Score(tree#1) = *Score*(tree_diagram#1)
Score(tree#1) < *Score*(shrub#1)

그림 3: Heuristic 2

다음은 실제 Heuristic 2의 점수를 계산하는 식이다.

$$H_2(s_i) = \max_{ew \in EW_i} P(s_i \mid ew)$$

$$\text{where } EW_i = \{ew \mid s_i \in synset(ew)\}$$

$$P(s_i \mid ew_j) \approx \frac{1}{n_j}$$

$$\text{where } s_i \in synset(ew_j),\ n_j = |synset(ew_j)|$$

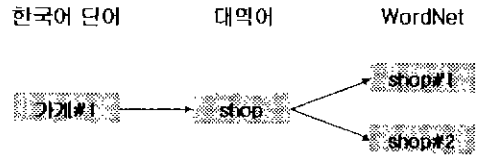
위 식에서 n_j는 영어 대역어 ew_j의 WordNet synset의 수를 나타낸다.

2.3 Heuristic 3: Sense Ordering

(Gale et al., 1992) 에서는 만약 단어의 의미 중에서 가장 자주 쓰이는 의미만으로 의미를 결정하는 WSD 시스템을 만들 경우 적어도 75%의 정확도를 얻을 수 있음을 보고했으며, (Miller et al., 1994) 에서는 Brown Corpus에 있는 의미적으로 애매성이 있는 단어들에 대해서 가장 자주 쓰이는 의미를 그 단어의 의미로 결정하는 경우 58%의 정확도를 보임을 보고했다. Heuristic 3에서는 이러한 결과를 이용하였다.

Heuristic 3은 영어 대역어의 synset 후보 중에서 실제로 자주 쓰이는 synset에 높은 점수를 주게 된다. 이러한 정보는 WordNet에서 제공하고 있으므로 바로 사용할 수 있다.

그림 4에서 한국어 단어 "가게"의 첫 번째 의미의 영어 대역어인 "shop"의 의미 후보는 2개가 있고, 이중에서 "shop"의 첫 번째 의미인 "shop#1"이 "shop#2"보다 자주 쓰이므로 "shop#1"의 점수가 "shop#2"보다 높게 된다.



Freq(shop#1) > *Freq*(shop#2)
Score(shop#1) > *Score*(shop#2)

그림 4: Heuristic 3

다음은 실제 Heuristic 3의 점수를 계산하는 식이다.

$$H_3(s_i) = \max_{ew \in EW_i} SO(s_i, ew)$$

$$\text{where } EW_i = \{ew \mid s_i \in synset(ew)\}$$

$$SO(s_i, ew) = \frac{\alpha}{x^{\beta}}$$

$$\text{where } s_i \in synset(ew)$$

$$\wedge\ synset(ew) \text{ is sorted by frequency}$$

$$\wedge\ s_i \text{ is the } x\text{-th synset in } synset(ew)$$

위 식에서 x는 $synset(ew)$ 안에 있는 synset s_i의 순서를 나타낸다. 즉 s_i가 대역어 ew의 가장 자주 쓰이는 의미인 경우 x는 1이 된다. α와 β는 상수로 이들의 값은 그림 5에 있는 SemCor corpus[2]의 data distribution으로부터 regression을 이용하여 구하였고, 그 값은 α=0.705, β=2.2 이다.

[1] 본 논문에서는 영어 WordNet version 1.6을 사용하였다.

[2] SemCor는 Brown corpus의 일부분으로써 WordNet의 synset으로 sense tagging된 corpus이다.

그림 5에서 X축은 SemCor corpus에 있는 단어들의 의미 순서를 나타내며, Y축은 의미 순서에 따른 확률 분포를 나타낸다. 예를 들어 Semcor corpus에 있는 단어들 중에서 첫번째 의미, 즉 가장 자주 쓰이는 의미로 사용된 것이 전체 corpus에서 70.5%에 해당하게 된다.

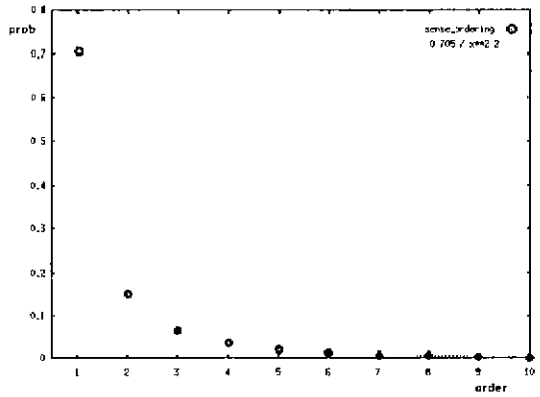
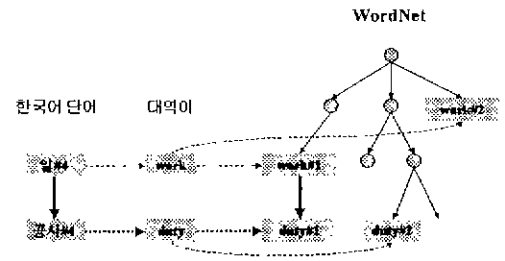


그림 5: Sense distribution in SemCor

2.4 Heuristic 4: IS-A relation

영어 대역어의 어휘 의미 애매성 해소를 위한 휴리스틱 4는 "두 개의 한국어 단어가 상하위 관계를 갖는다면, 그들의 영어 대역어들 중에도 상하위 관계를 갖는 것이 존재한다"라는 가정을 사용하여, 실제로 상하위 관계를 갖는 synset 후보에 높은 점수를 준다.

그림 6에서 한국어 단어 "공사"의 네 번째 의미의 영어 대역어는 "duty"이고, "공사#4"의 상위어는 "일"의 네 번째 의미이고 대역어는 "work"이다. 이때 "work"와 "duty"가 다의어이므로 상하위 관계를 가질 수 있는 후보는 "work#1" --- "duty#1", "work#1" --- "duty#2", ... , "work#2" --- "duty#1", ... 등이 있다. 이 중에서 실제로 WordNet상에서 상하위 관계를 갖는 것은 "work#1" --- "duty#1" 이므로, "duty#1"이 "duty#2"보다 높은 점수를 얻게 된다.



Score(duty#1) > *Score*(duty#2)

그림 6: Heuristic 4

다음은 실제 휴리스틱 4에서 점수를 계산하는 식이다.

$$H_4(s_i) = \max_{ew \in EW_i} IR(s_i, ew)$$

$$\text{where } EW_i = \{ew \mid s_i \in synset(ew)\}$$

$$IR(s_i, ew) = \begin{cases} 1 & \text{if } IsA(s_i, s_j) \\ 0 & \text{otherwise} \end{cases}$$

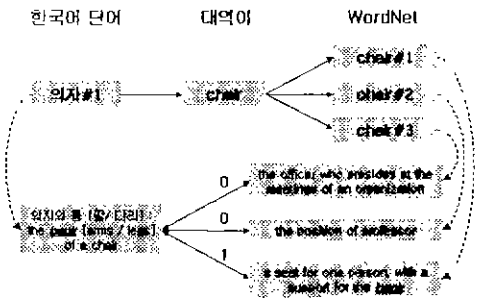
$$\text{where } s_i \in synset(ew),\ s_j \in synset(hew)$$

위 식에서 $IsA(s_i,s_j)$는 s_i가 s_j의 하위어인 경우 true 값을 반환하고 그렇지 않은 경우 false 값을 반환한다. *hew*는 한국어 단어의 상위어에 대한 영어 대역어이다. 이 휴리스틱을 사용하기 위해서는 한국어 단어에 대한 상위어들을 구해야 하는데, 이것은 국어사전의 정의문으로부터 간단한 rule을 적용하여 추출한 것을 사용하였다. 또한 한국어 단어가 다의어인 경우, 나머지 의미들이 잡음으로 작용을 하기 때문에 이 휴리스틱은 한국어 단어가 다의어가 아닌 경우에만 적용되었다.

2.5 Heuristic 5: Word Match

영어 대역어의 어휘 의미 애매성 해소를 위한 휴리스틱 5에서는 "유사한 의미들은 비슷한 어휘들을 사용하여 표현된다" 라는 가정을 사용하여, 같은 어휘들이 많이 사용된 의미에 높은 점수를 준다

그림 7에서 한국어 단어 "의자"의 첫 번째 의미의 영어 대역어는 "chair"이고, "chair"는 WordNet 상에서 3개의 의미 후보를 갖는다. 이 3개의 후보들의 WordNet의 정의문 및 예제와 "의자#1"의 영어 대역어 사전의 영어 예제간의 공통된 단어를 구해보면 "chair#1"이 가장 많으므로 "chair#1"이 높은 점수를 얻게 된다.



Score(chair#1) > *Score*(chair#2)
Score(chair#1) > *Score*(chair#3)

그림 7: Heuristic 5

다음은 실제 휴리스틱 5에서 점수를 계산하는 식이다.

$$H_5(s_i) = \max_{ew \in EW_i} WM(s_i, ew)$$

$$\text{where } EW_i = \{ew \mid s_i \in synset(ew)\}$$

$$WM(s_i, ew) = sim(X, Y_i)$$

$$sim(X,Y) = \frac{|X \cap Y|}{|X \cup Y|}$$

위 식에서 X는 대역어 사전에 있는 영어 예제의 content word들의 집합이고, Y_i는 WordNet에 있는 synset s_i의 정의문과 예제에 있는 content word들의 집합이다.

2.6 Heuristic 6: Cooccurrence

영어 대역어의 어휘 의미 애매성 해소를 위한 휴리스틱 6은 공기 정보(co-occurrence information)를 사용하여, 높은 공기도를 갖는 의미에 높은 점수를 준다. 그림 8에서 "마음"이 한영사전의 정의문에 쓰였을 경우, "마음"을 정의문에 포함하는 단어인 "명예심#1"과 "공포심#1" 등의 의미가 WordNet의 "feeling#1"의 하위어로 매핑된다면, "마음"을 정의문에 포함하는 새로운 단어인 "근심#1"의 의미도 "feeling#1"의 하위어로 쓰일 가능성이 많으므로 "feeling#1"의 하위어인 "anxiety#2"의 점수가 "anxiety#1"보다 높게 된다.

이 휴리스틱을 적용하기 위해서는 대역어 사전의 한국어 단어가 WordNet synset에 매핑된 corpus가 필요한데, 이것을 구축하기 위해서 한국어 단어에 대한 영어 대역어가 의미 애매성이 없는 것들을 이용하였다. 그리고 WordNet의 명사부분에 쓰인 25개의 semantic tag들을 sense tag로 사용하였다.

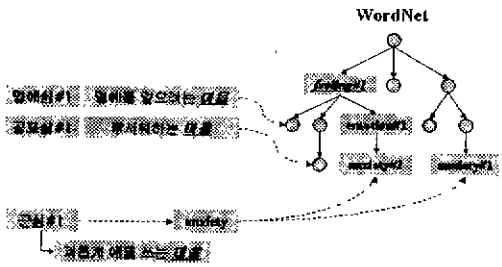


그림 8: Heuristic 6

다음은 실제 휴리스틱 6에서 점수를 계산하는 식이다.

$$H_6(s_i) = \max_{x \in Def} p(t_i \mid x)$$

$$\text{with } p = \hat{p} - Z_{(1-\alpha)/2} \times \sqrt{\frac{\hat{p}(1-\hat{p})}{n}}$$

$$\hat{p}(t_i \mid x) = \frac{Freq(t_i, x)}{Freq(x)}$$

위 식에서 Def 는 대역어 사전의 한국어 정의문에 있는 content word들의 집합이고, t_i는 synset s_i에 해당하는 WordNet의 semantic tag이며, n은 $Freq(x)$를 나타낸다.

3 Decision tree를 이용한 다양한 휴리스틱들의 결합

위에서 설명한 6가지 휴리스틱들을 각각 영어 대역어의 어휘 의미 애매성 해소에 적용할 수도 있지만, 본 논문에서는 이러한 6가지 휴리스틱들을 결합시켜서 사용하도록 하였다. 본 논문에서는 한국어 단어의 WordNet mapping 문제를 대역어들의 synset 후보들을 연결할 것인지 (linking) 아니면 버릴 것인지(discarding)를 결정하는 binary classification 문제로 보았다.

6가지의 휴리스틱을 결합시키기 위해서 본 논문에서는 decision tree를 이용하였는데, 그 이유는 decision tree를 이용할 경우 feature들의 non-linear 한 관계 등을 위해서 이다. 본 논문에서는 가장 널리 쓰이고 있는 C4.5 (Quinlan, 1993)를 사용하였다.

그림 9는 decision tree를 이용한 6가지 휴리스틱의 결합의 학습 단계를 나타낸다. 학습 단계에서는 영어 대역어의 후보 synset들에 대해서 위에서 설명한 6가지의 휴리스틱들의 점수와 수작업으로 분류된 결과 (linking or discarding)가 사용된다. 이러한 training data를 이용하여 C4.5로 decision tree를 만들게 된다.

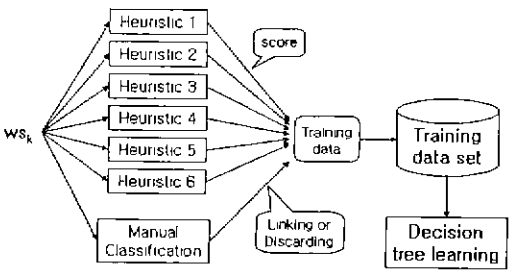


그림 9: 학습 단계

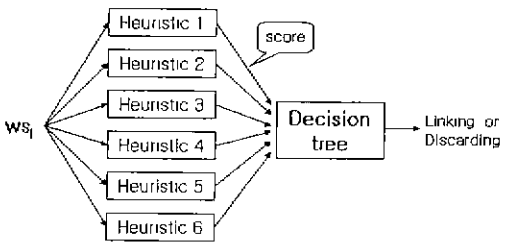


그림 10: 매핑 단계

그림 10은 매핑 단계를 나타낸다. 매핑 단계에서는 학습 단계에서 생성된 decision tree를 이용하여 새로운 영어 대역어의 synset 후보 ws_i가 입력 되었을 때 위에서 설명한 6가지 휴리스틱을 이용하여 각각의 점수를 계산하여 decision tree를 이용하여 이 synset 후보를 연결할 것인지 버릴 것인지를 판단하게 된다.

4 실험 및 결과

본 논문에서는 6가지의 휴리스틱 각각에 대한 성능과 이 휴리스틱들을 결합시켰을 때의 성능을 평가하였다. 이러한 평가를 위해서 영어 대역어 사전의 한국어 표제어들의 3260 의미에 대한 그들의 영어 대역어의 synset 후보들에 대해서 수작업으로 분류하였다(discarding or discarding).

본 논문에서는 정확도(precision)를 test set에 있는 한국어 단어들의 의미들에 대해서 옳바르게 연결된 의미의 비율로 정의하고, 적용율(coverage)은 test set에 있는 한국어 단어들의 의미들 중에서 실제로 연결이 된 의미들의 비율로 정의한다.

	Precision(%)	Coverage(%)
Random mapping	49.85	100.0
Heuristic 1	75.21	59.51
Heuristic 2	74.66	100.0
Heuristic 3	71.87	100.0
Heuristic 4	55.49	29.36
Heuristic 5	56.48	63.01
Heuristic 6	67.24	64.14

Table 1: Individual heuristics performance

Table 1은 각각의 휴리스틱에 대한 성능을 나타낸다. 본 논문에서 사용한 휴리스틱들은 모두 비교사 학습 방법(Unsupervised method)이므로 수작업으로 분류된 3260개의 한국어 의미 모두를 test set으로 사용하였다. Table 1에서 각각의 휴리스틱들의 성능은 좋게 나오지는 않지만 모든 휴리스틱들이 random으로 선택하는 경우보다 높은 정확도를 보인다. 가장 높은 정확도를 보인 것은 maximum similarity heuristic으로 75.21%의 정확도를 보이지만, 적용율은 59.51%로 낮게 나왔다. 각각의 휴리스틱들이 random mapping보다 높은 정확도를 보이지만 이러한 수치가 통계적으로 중요한 가를 판단하기 위해서 statistically significance test를 수행한 결과 6가지 휴리스틱 모두 99%의 level에서 random mapping과의 차이가 중요하다는 결과가 나왔다.

	Precision(%)	Coverage(%)
Summing	84.61	100.0
Logistic regression	86.41	100.0
Decisioin tree	93.59	77.12

Table 2: Performance and comparison of the decision tree based combination

본 논문에서는 decision tree를 이용한 6가지 휴리스틱의 결합의 성능을 평가하기 위해서 10-fold cross validation을 수행하였다. 즉, 3260개의 수작업으로 분류된 data를 열등분으로 쪼개어 하나는 test set으로 사용하고 나머지를 decision tree의 training data로 사용하였다. 그리고 나머지 9개의 부분이 한 번씩 test set이 되도록하여 위와 같은 작업을 9번 더 수행하였다.

Table 2는 decision tree를 이용한 휴리스틱의 결합 성능과 휴리스틱을 결합하기 위해서 다른 방법을 수행한 결과를 나타낸다. 'Summing'은 6가지 휴리스틱의 점수를 단순히 더하는 방법으로, 더한 값이 가장 높은 synset 후보를 선택하게 된다. Logistic regression은 binary classification에 널리 사용되는 방법이다(Hosmer and Lemeshow, 1989).

'Summing'을 이용하여 휴리스틱을 결합시켰을 경우 가장 높은 정확도를 보인 maximum similarity heuristic (heuristic 1)보다 9%의 향상을 보였으며 적용율이 100%가 나왔다. Decision tree를 이용한 경우 적용율이 77.12%로 나왔지만 정확도는 가장 높은 93.59%를 보였다.

Decision tree를 이용한 휴리스틱 결합 모델을 이용하여 전체 영어 대역어 사전에 적용한 결과 17696개의 한국어 명사의 21654개의 의미로 이루어진 한국어 WordNet을 얻게 되었으며 이때의 정확도는 93.59%이다((±0.84% 신뢰도 99%)

6 결론

본 논문에서는 어휘 의미 에매싱 해소와 영어 대역어 사전 그리고 외국언어에 존재하는 개념체계를 이용하여 한국어 개념체계를 자동으로 구축하는 방법을 기술했다. 본 논문에서 사용하는 방법은 기존의 개념체계 구축 방법들에 비해 적은 노력과 시간을 필요로 한다. 또한 본 논문에서 사용하는 방법들은 특정 언어에 국한되지 않는 방법이기 때문에 이 방법을 다른 언어에도 쉽게 적용할 수 있을 것이다.

참고문헌

이창기, 이근배 (1999) WordNet을 이용한 한국어 시소러스 자동 구축. 한글 및 한국어 학회.

Atserias J., Climent S., Farreras J., Rigau G. and Rodriguez H. (1997) Combining Multiple Methods for the Automatic Construction of Multilingual WordNets. In *proceeding of the Conference on Recent Advances on NLP*.

Changki Lee, Geunbae Lee and Seo JungYun. (2000) Automatic WordNet mapping using word sense disambiguation. In *Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-2000)*.

Farreres X., Rigau G., and Rodriguez H.. (1998) Using WordNet for building WordNets. In *Proceedings of COLING-ACL Workshop on Usage of WordNet in Natural Language Processing Systems*.

Gale W., Church K., and Yarowsky D. (1992) Estimating upper and lower bounds on the performance of word-sense disambiguation programs. In *Proceeding of 30th Annual Meeting of the Association for Computational Linguistics*.

Hosmer Jr. and Lemeshow S. (1989) *Applied Logistic Regression*. Wiley, New York.

Knight K. and Luk S. (1994) Building a large-scale knowledge base for machine translation. In *Proceeding of the American Association for Artificial Intelligence*.

Miller G. (1990) Five papers on WordNet. *Special Issue of International Journal of Lexicography*.

Miller G., Chodorow M., Landes S., Leacock C. and Thomas R.. (1994) Using a semantic concordance for sense identification. In *Proceedings of the Human Language Technology Workshop*.

Okumura A. and Hovy E. (1994) Building Japanese-English Dictionary based on Ontology for Machine Translation. In *Proceedings of ARPA Workshop on Human Language Technology*.

Quinlan R.. (1993) *C4.5: Programs For Machine Learning*. Morgan Kaufmann Publishers.

Seungwoo Lee and Geunbae Lee. (2000) Unsupervised Noun Sense Disambiguation Using Local Context and Co-occurrence. In *Journal of Korean Information Science Society*. (in press)