

CRFs를 이용한 강건한 한국어 의존구조 분석<sup>1)</sup>오진영<sup>o</sup> 차정원

창원대학교 컴퓨터공학과

shanghwoe963@changwon.ac.kr, jcha@changwon.ac.kr

## Robust Korean Dependency Analysis Based on CRFs

Jin-young Oh<sup>o</sup> Jeong-Won Cha

Dept. of Computer Engineering, Changwon National University

## 요 약

한국어 처리에서 구문분석기에 대한 요구는 많은 반면 성능의 한계와 강건함의 부족으로 인해 채택되지 못하는 것이 현실이다. 본 연구는 구문분석을 레이블링 문제로 전환하여 성능, 속도, 강건함을 모두 실현한 시스템에 대해서 설명한다. 우리는 다단계 구 단위화(Cascaded Chunking)를 통해 한국어 구문분석을 시도한다. 각 단계에서는 어절별 품사 태그와 어절 구문표지를 자질로 사용하고 Conditional Random Fields(CRFs)를 이용하여 최적의 결과를 얻는다. 98,412문장 세종 구문 코퍼스로 학습하고 1,430문장(평균 14.59어절)으로 실험한 결과 87.30%의 구문 정확도를 보였다. 이 결과는 기존에 제안되었던 구문분석기와 대등하거나 우수한 성능이며 기존 구문분석기가 처리하지 못하는 장문도 처리 가능하다.

## 1. 서 론

구문분석은 문장의 구조를 분석하는 것으로, 구문 분석의 결과를 바탕으로 다양한 자연어처리 응용분야에서 활용하고자하는 요구가 증가하고 있다. 더욱이 현재 인터넷의 발달로 방대한 인터넷 문서를 구문 분석할 수 있다면 보다 많은 응용에서 높은 성능을 얻을 수 있을 것이다.

영어권에서는 오래 전부터 많은 연구가 진행되었다. 최근의 연구는 CFG(Context Free Grammar)를 사용하고 통계 모델을 이용한 방법과 기계학습을 이용한 방법이 주류를 이루고 있다[1,2,3,4]. 또한 reranking을 통해 성능 향상시킨 방법도 제안[5]되었으며, 영어권에서 개발된 많은 방법들이 일본어와 한국어에 적용되었다.

어순이 비교적 자유로운 일본어에서도 의존 구조를 이용하는 방법이 많이 제안되었다. 의존 구조의 애매성을 해소하기 위해 통계적 방법과 기계학습을 이용한 다양한 방법이 제안되었다. 예를 들어 최대 우도 추정(Maximum Likelihood Estimation)[6], 결정 트리(Decision Tree)[7], 최대 엔트로피 모델(Maximum Entropy Model)[8,9], 지지 기반 기계(Support Vector Machine)[10] 등이다.

한국어에서도 다양한 시도가 있었다. 한국어 구문분석은 단일화 문법(Unification Grammar), 핵심어 중심 구조 문법(HPSG: Head-Driven Phrase Structure

Grammar), 어휘 기능 문법(LFG: Lexical Functional Grammar), 결합 범주 문법(CCG: Combinatorial Categorical Grammar)을 이용한 시스템들이 제안되었다[11,12,13,14,15]. 최근에는 거의 모든 연구가 의존 문법을 기반으로 하고 있다. 또한 일반어와 마찬가지로 의존 구조의 애매성을 해소하기 위해 다양한 통계 방법과 기계 학습을 이용하는 방법들이 제안되었다[16,17]. 초기의 한국어에 대한 연구는 학습 코퍼스의 부족으로 연구실 수준의 연구에 머물렀지만 최근에는 한국어정보베이스(Korean Language Information Base), 세종 구문 코퍼스 등이 제작되면서 대용량 코퍼스를 이용하는 연구가 활기를 띠고 있다[16,17].

본 연구에서는 세종 구문 코퍼스를 사용하여 학습 및 평가를 하며 기존에 방법보다 간단하지만 효율적인 한국어 구문분석 방법을 제안한다. 제안된 시스템은 다단계 구 단위화를 통해 각 단계에서 지배소를 결정하는 방법을 취한다.

본 논문의 구성은 다음과 같다. 2장은 본 논문에서 제안하는 방법 및 시스템을 설명하며, 3장은 실험 및 분석, 4장에서 결론을 맺는다.

## 2. 제안 시스템

본 논문에서 제안하는 시스템은 입력 문장을 품사태깅한다. 그 결과를 입력으로 받아서 구문태그를 부착하고, 그 결과와 품사 태깅 결과를 이용하여 다단계 구 단위화를 통한 구문 분석을 한다. 구문태그와 다단계 구 단위화는 학습을 위해 다른 자질집합을 사용하여 모델을 생성한다. [그림 1]은 제안 시스템의 구조도이다.

1) 본 연구에서 제안한 시스템은 <http://thor.nlp.wo.tc/~shanghwoe963/RealParser.php>에서 실행 가능합니다.

이 장에서는 문장과 의존 구조를 정의하고 제안된 시스템에서 사용하는 자질과 구문분석 방법을 설명한다. 본 연구에서는 구문분석의 기본 단위를 어절 단위로 한다. 따라서 문장은 어절들의 집합이 된다. 즉 문장은  $S = \langle s_1, s_2, \dots, s_m \rangle$ , 의존구조는  $D = \langle dep(1), dep(2), \dots, dep(m-1) \rangle$ 와 같이 표시한다. 여기서  $dep(i) = j$ 는 어절  $s_i$ 가 어절  $s_j$ 를 지배소로 갖는 것을 의미한다. 이러한 구조에서 기존의 연구에서와 같이 다음과 같은 세 가지의 제한을  $D$ 가 만족한다고 가정한다.

1. 한국어는 지배소 후위 언어이다. 즉 지배소는 피지배소보다 항상 뒤에 위치한다.
2. 교차 의존 구조는 없다.
3. 각 어절의 머리는 유일하다.

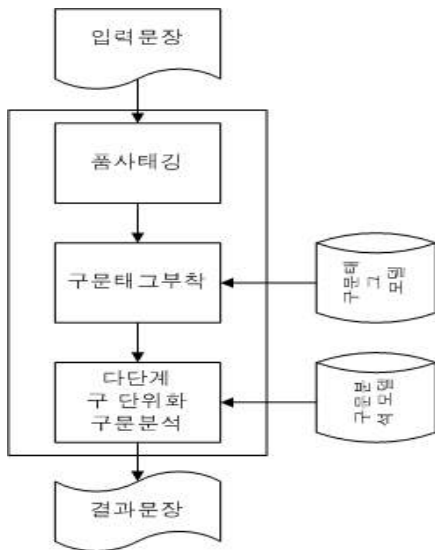


그림 53. 제안 시스템 구조도

### 2.1 다단계 구 단위화 방법

다단계 구 단위화(Cascaded Chunking) 방법은 [18]에서 영어를 위해 처음 제안되었다. 이 방법은 [19, 20]에 의해 일본어에 적용되어 좋은 성능을 보였다.

본 연구에서는 한국어의 특성에 맞게 이를 변형하여 적용한다. [그림 2]는 다단계 구 단위화의 과정을 보여준다.

어절	1단계	2단계	3단계	4단계	5단계
자기에	-	D	X	X	X
충실치	D	X	X	X	X
못하고는	-	-	-	D	X
도덕이	-	-	D	X	X
생겨날	D	X	X	X	X
수	D	D	X	X	X
없다.	-	-	-	-	-

그림 2. 다단계 구 단위화를 이용한 구문분석의 예

[그림 2]에서 'D'는 의존소에 대한 표시이다. 각 단계에 'D' 표시를 부착할 수 있는 어절은 바로 다음 어절이 지배소일 경우이다. 1단계에서 '충실치', '생겨날', '수'에 'D' 표시가 부착되었다. 그렇지 않은 어절에는 '-' 표시를 부착한다.

그리고 다음단계로 넘어갈 때 앞의 어절 표시가 '-'이고 현재 어절의 표시가 'D'인 경우 삭제된다. [그림 2]에서 1단계 '수' 어절이 삭제되지 않은 이유는 '생겨날' 어절이 의존소 표시 'D'를 가지고 있기 때문이다.

'X' 표시는 앞의 단계의 의존소를 제외한 어절로서, 학습코퍼스를 생성할 때 해당 단계에서 삭제된다. 예를 들어 두 번째 단계에서는 '충실치', '생겨날' 어절이 삭제되어 '자기에 못하고는 도덕이 수 없다.'의 문장에 대해 구문분석을 다시 수행한다. 이런 과정은 한 어절이 남을 때까지 반복한다.

다단계 일반적인 구문분석 방법의 시간 복잡도가  $O(n^3)$ 임에 비하여 구 단위화 기법은  $O(n^2)$ 이므로 매우 빠르다. 실제 본 연구에서 제안한 시스템은 평균 15어절의 문장을 초당 50문장 이상 분석할 수 있다. 또한 구문 요소의 결합이 아니라 레이블링 문제이므로 입력 문장에 대해서 매우 강건하다.

### 2.2 CRFs 학습 및 평가

CRFs는 조건부 확률을 최대로 하는 방향성이 없는 그래프 모델이다[21]. 입력열  $X = x_1 x_2 \dots x_n$ , 상태열  $T = t_1 t_2 \dots t_n$ 가 주어지고 가중치  $\Lambda = \{\lambda, \dots\}$ 가 주어졌을 때, CRFs에서는 조건 확률로 식 (1)과 같이 정의된다.

$$P(T|X) = \frac{1}{Z(X)} \exp\left(\sum_{i=1}^n \sum_k \lambda_k f_k(t_{i-1}, t_i, x, i)\right) \quad (1)$$

여기서  $Z(X)$ 는 확률 값으로 만들어 주는 정규화 값이고  $f_k(t_{i-1}, t_i, x, i)$ 는 자질 함수이다. 또한  $\lambda_k$ 는 각 자질에 대한 가중치를 나타낸다.  $k$ 는  $k$ 번째 자질이며, 자질 함수는 현재 시간  $i$ 에 대해 관측열  $x$ , 상태변이  $t_{i-1} \rightarrow t_i$

에 대해서 전이의 양상을 측정할 수 있다.

매개변수들은 주어진 입력 열과 이에 대응하는 상태 열에 대한 조건부 확률이 최대화하는 최대 우도(maximum likelihood)에 의해서 추정된다. 훈련 집합  $\{(t_i, x_i)\}_{i=1}^N$ 에 대해서 다음과 같은 로그 유사도(log-likelihood)를 계산한다.

$$L(A) = \sum_i \log P_A(t_i|x_i) = \sum_i \left( \sum_{j=1}^n \sum_k \lambda_k f_k(t_{i-1}, t_i, x, i) - \log Z_{x_i} \right) \quad (2)$$

식 (2)를 최대로 하도록 학습한다. 일반적으로 CRFs는 IIS(Improved Iterative Scaling)나 GIS(Generalized Iterative Scaling)[22]를 사용하여 학습한다.

또한 학습 데이터의 과적합(overfitting) 문제를 해결하기 위해서 가우스 사전 평활(Gaussian prior smoothing)[23]을 적용한다. 본 연구에서 [24]의 CRF++을 사용하였다.

### 2.3 자질 집합

본 연구에서는 세종 구문 코퍼스[25]를 사용하였다. 구문 분석은 형태소 단위가 아닌 띄어쓰기로 구분된 어절 단위로서 분석하게 된다. 따라서 구문태그와 의존관계에 대해서 기계 학습한 두 모델 결과에 따라 분석을 하게 되므로 각각에 맞는 자질집합 생성이 중요하다.

구문태그와 의존관계를 위해 사용한 자질은 서로 비슷하게 구성되어 있다. 자질의 기본은 형태소 분석기의 결과로서, 문장을 이루는 어절들의 모든 형태소 중에서 각 단계의 모델 생성에 있어 많은 영향을 주는 것을 선택하였다. 다음 [그림 3]는 구문태그에 대한 자질 예이다.

형태소 분석	1	2	3	4	5	구문태그
물론/MAG	MAG	-	-	MAG	-	AP
스포츠/NNG	NNG	-	-	JKB	있	NP_AJT
+에/JKB					/VV	
있/VV+어	VV	-	EC	JX	-	VP
/EC+도/JX						
이것/NP	NP	-	-	JX	-	NP_SBJ
+은/JX						
예외/NNG	NNG	-	-	JKC	아니	NP_CMP
+가/JKC					/VCN	
아니/VCN	VCN	-	-	-	-	VP
+다/EF+./SF						

그림 3. 구문태그 코퍼스 예

구문태그는 5개의 자질을 사용하였으며, 자질들을 생성함에 있어서 기호에 대한 품사는 추가하지 않았다.

1은 현재 어절의 첫 번째 형태소의 품사 자질이다. 3은 마지막 형태소 앞의 품사이며, 2는 1과 3의 자질 사이에 'XSA, XSV, VCP' 중 하나의 품사가 있을 경우에 추가된다. 이것은 문장 분석에 있어 용언에 대한 영향을 고려하였을 때, 어절을 이루는 품사가 많을 경우 위의 자질이 추가되지 않는 것을 막기 위해 2번째 자질로 추가하였다. 4는 마지막 품사에 대한 자질이다. 여기에서 4번 자질은 어절을 이루는 형태소가 단일일 경우 1번 자질이 4번에도 추가된다. 어절태그를 예측하는 모델 결과를 분석해보았을 때, 어미 또는 조사가 없는 어절에 대한 오류가 가장 많았다. 이를 위해 4번 자질에 반복 추가함으로써, 오류를 줄이고자 하였다.

그리고 5번째 자질은 다음어절 첫 번째 형태소에 대한 형태소와 품사로서 이루어진 자질이다. 다음 어절의 첫 번째 형태소가 'V'로 시작하는 경우 5번 자질에 추가하는 것으로서, 조용사(XSA, XSV)가 붙어서 형용사, 또는 동사가 되는 경우에도 5번 자질에 추가하였다. 예를 들어 다음 어절이 '입장/NNG+하/XSV+다/EF+./SF'일 경우 이전 어절의 5번 자질에는 '입장하/VV'가 추가된다.

의존관계를 위한 코퍼스의 자질은 4개의 자질을 사용하는데, [그림 3]의 3~5번 자질과 구문태그 결과를 자질로 사용한다. 구문태그 결과만을 자질로 사용하였을 경우 구문태그 자질의 오류에 대해서 구문분석 예측 확률이 낮아서 품사에 대한 자질을 추가하였다.

## 3. 실험 및 분석

### 3.1 실험 환경

실험에서 사용된 코퍼스는 어절태그를 학습하기 위해서는 9,889문장(90,112어절)을 사용하였고, 의존관계를 위해 1,120,777어절(96,412문장)을 사용하였다. 평가 코퍼스는 20,685어절(1,430문장)을 사용하였고, 한 문장당 평균 14.59어절로서 구성되어 있다.

### 3.2 실험 및 토의

본 논문에서는 세 가지의 큰 분류로 실험을 나누었다.

먼저 형태소분석 및 구문태그의 결과가 정답인 문서에 대해서 구문분석의 성능측정(P)과, 형태소분석만 정답인 문서에 대한 성능측정(E+P), 형태소분석부터 구문태그 모두 시스템의 결과로서 측정된 성능평가(T+E+P) 실험으로 나누어진다. 본 논문에서는 [26]의 품사 태거를 사용하였다.

또한 각 실험에 대해서 20어절 이상이 포함된 장문 문장과 20어절 미만인 문장으로 분리하여 따로 성능을 측정하였다.

제안한 시스템의 성능 평가를 위해 아크-정확도와 아크-재현율을 결합한  $F_1$ -measure와 문장 정확도(Exact-Matching)를 사용하였다. 평가 척도는 식 (3)과 같다. 본 연구에서는 구문 분석을 레이블링 문제로 해결하기 때문에 아크-정확도와 아크-재현율이 같다.

$$\text{아크-정확도 (Arc Precision, AP)} = \frac{\text{구문 분석 파스트리에서 올바른 아크의 수}}{\text{구문 분석 파스트리에서 모든 아크의 수}}$$

$$\text{아크-재현율 (Arc Recall, AR)} = \frac{\text{구문 분석 파스트리에서 올바른 아크의 수}}{\text{정답 파스트리에서 모든 아크의 수}} \quad (3)$$

$$F_1\text{-measure} = \frac{2 \cdot AP \cdot AR}{AP + AR}$$

$$\text{Exact-Matching} = \frac{\text{정확히 분석된 문장의 수}}{\text{문장의 수}}$$

표 1. 구문 분석기의 성능(%). '<20'은 20어절 미만의 성능을 나타내고 '>20'은 20어절 이상의 성능을 나타낸다. ()는 문장의 성능을 나타낸다.

	<20	>20	전체
<b>T+E+P</b>	75.06 (18.67)	68.77 (0)	72.74 (14.90)
<b>E+P</b>	85.82 (32.91)	80.06 (2.80)	83.52 (26.15)
<b>P</b>	89.08 (40.04)	84.63 (4.05)	<b>87.30</b> <b>(31.95)</b>

[표 1] T+E+P에 관한 어절 오류를 분석을 해보았을 때, 품사 태깅 오류에 의해 구문분석 오류가 발생한다는 것을 알 수 있었다. 이것은 품사태깅 결과를 자질로 사용하기 때문에 당연한 결과이다. 따라서 품사 태깅 성능을 향상시키는 것이 당면 과제이다.

어절태그에 오류는 어절에서 조사와 어미가 없는 형태에 대한 오류가 대부분이지만, [그림 3]에서의 5번 자질이 추가되어 있는 경우에 대해서는 구문분석이 정확한 것으로 보아, 의존관계 분석은 5번 자질에 대해 영향을 많이 받는다는 것을 알 수 있다. 그리고 가장 많은 오류를 보이는 의존관계 분석은 형태소 분석의 오류에 따른 오류도 존재하였으며, 문장의 중간에 나타나는 용언구에 대한 의존관계 분석 오류가 절반을 차지하였다. 또한 어절태그 오류와 같은 조사와 어미가 없는 체언구 오류가 많다.

또한 문장의 어절과 비교하여 50%이상이 오류인 문장을 분석해보면, 오류 분포가 큰 문장일수록 문장에서 명

사인 어절이 많으며, 명사가 나열된 구조를 나타내고 있다. 이러한 오류에 대해서는 차후에 개선해야할 문제로서 자질의 수정 또는 추가로 보완해야한다.

본 논문의 시스템이 문장길이에 대한 구문분석의 성능을 확인하기 각 길이별 평균을 측정하였다.

평가 문서는 평균 한 문장에 14.59어절로 구성되어 있으며, 다양한 길이의 문장들로 구성되어 있다.

표 2. 문장별 구문 분석성능

어절 수	20미만	20~30	30~40	40이상
<b>문장 수</b>	1109	259	48	14
<b>구문 정확도</b>	89.08	85.58	83.06	79.24
<b>문장 정확도</b>	40.04	5.02	0	0

단문일수록 장문에 비해 좋은 성능을 보이고 있다. 또한 형태소 분석과 어절태그의 성능이 보장 된다면 이와 같은 성능을 확보 할 수 있음을 보여준다.

표 3. 구문분석 성능 비교

구분	의존관계 성능(%)	비고
<b>제안방법</b>	<b>87.30</b>	<b>한국어</b>
Lee [17]	87.46	한국어
Lex.(Chung) [16]	86.74	한국어
Unlex.(Chung) [16]	85.62	한국어
Kudo [10]	89.09	일본어
Kudo [19]	89.29	일본어
Yamada [27]	90.30	영어
Unlex.(Klein & Manning 2003) [2]	85.7	영어
Charniak 2000 [3]	89.5	영어
Petrov & Klein 2007 [4]	90.1	영어
Charniak 2005 [5]	91.4	영어

[표 3]에서는 각 언어의 대표적인 시스템과 성능을 비교하였다. 코퍼스가 서로 다르기 때문에 직접적인 비교를 하는 것은 무리가 있지만 상대적인 비교는 가능하다. 영어권의 성능과는 아직 거리가 있다. 일본어 시스템과도 2% 정도의 차이가 난다. 하지만 이것은 언어 복잡성과 같은 요소가 작용할 수 있다. 기타 한국어 시스템과는 거의 대등하거나 우수한 성능을 보이고 있다.

#### 4. 결론

본 논문에서는 한국어 의존구조 분석을 위해 기계학습을 이용하였다. 어절태그를 사용하여 문장의 요소를 분리 하였으며, 다단계 구 단위화(Cascaded Chunking) 방법을 사용하여 강건함을 획득할 수 있었다. 실험결과로

보았을 때, 어휘를 사용하지 않은 자질 학습으로 모델의 사이즈가 작으며, 고속으로 처리할 수 있었다.

또한 이전 논문에서 처리하기 힘든 장문에 대한 분석이 가능하며, 이 또한 분석 처리속도가 빠르게 이루어짐을 확인하였다.

향후 연구로서는 성능을 향상시킬 수 있는 자질에 대한 추가 연구가 필요하며, 조사와 어미가 없는 어절에 대한 오류분석과 보완을 할 것이다. 또한 대등문 처리에 대한 방법을 연구해야 할 것이다. 이를 보완하면, 여러 응용분야에 기반이 되며, 응용시스템에서 사용가능한 구문분석 시스템이 될 것이라 기대한다.

### 참고문헌

- [1] Charniak, E. "Statistical parsing with a context-free grammar and word statistics." in Proceedings of the Fourteenth National Conference on Artificial Intelligence. Menlo Park, AAAI Press/MIT pp. 598-603, 1997.
- [2] Dan Klein and Christopher D. Manning. 2003. "Accurate Unlexicalized Parsing." ACL 2003, pp. 423-430.
- [3] Charniak, E. "A Maximum-Entropy-Inspired Parse." in Proceedings of NAACL-2000, pp 132--139.
- [4] Slav Petrov and Dan Klein, "Improved Inference for Unlexicalized Parsing." In proceedings of HLT-NAACL 2007, pp. 404-411.
- [5] Eugene Charniak and Mark Johnson. "Coarse-to-fine n-best parsing and MaxEnt discriminative reranking." In ACL 2005, pp. 173-180.
- [6] Masakazu Fujio and Yuji Matsumoto. "Japanese Dependency Structure Analysis based on Lexicalized Statistics." In Proceedings of EMNLP '98, 1998, pp. 87-96.
- [7] Msahiko Haruno, Satoshi Shirai, and Yoshifumi Ooyama. "Using Decision Trees to Construct a Practical Parser." Machine Learning, 34:131 - 149. 1999.
- [8] Kiyotaka Uchimoto, Satoshi Sekine, and Hitoshi Isahara. "Japanese Dependency Structure Analysis Based on Maximum Entropy Models." In Proceedings of the EACL, pp. 196 - 203. 1999.
- [9] Kiyotaka Uchimoto, Masaki Murata, Satoshi Sekine, and Hitoshi Isahara. Dependency model using posterior context. In Proceedings of Sixth International Workshop on Parsing Technologies. 2000.
- [10] Taku Kudo and Yuji Matsumoto. "Japanese Dependency Structure Analysis based on Support Vector Machines." In Empirical Methods in Natural Language Processing and Very Large Corpora, pp. 18 - 25. 2000.
- [11] Geum, J. C. and G. Kim. "Implementation of HPSG parsing mechanism for Korean syntactic structure analysis." In Proceedings of the Spring Conference of Korea Information Science Society, pp. 139 - 142, 1998.
- [12] Jung, H.-S., J.-H. Kim, J.-S. Lee, S.-Y. Chun, and M.-J. Park "Design of Korean-English machine translation system (KoEng)." In Proceedings of the 1st Workshop of Machine Translation, pp. 87 - 96. 1989.
- [13] Yang, J. "A study on the Korean analyzer based on HPSG." Master's thesis, Dept. of Computer Engineering. Seoul National University. 1990.
- [14] Yoon, D. H. and Y. T. Kim "Analysis techniques for Korean sentence based on Lexical Functional Grammar." In Proceedings of the International Parsing Workshop '89, pp. 369 - 78. 1989.
- [15] Jeongwon Cha, Geunbae Lee, Jong-Hyeok Lee, Morpho-syntactic categorial modeling of Korean, computers and the humanities journal, vol 36, No. 4, page 431-453. 2002.
- [16] Hoojung Chung, "Statistical Korean Dependency Parsing Model based on the surface Contextual Information", Ph.D. dissertation, 2004.
- [17] Yong-Hun Lee, Jong-Hyeok Lee, "Korean Parsing using Machine Learning Techniques", KCC 2008, pp. 285-288.
- [18] Steven Abney. "Parsing By Chunking." In Principle-Based Parsing. Kluwer Academic Publishers. 1991.
- [19] Kudo, T. and Y. Matsumoto. "Japanese Dependency Analysis using cascaded Chunking." coling02. 2002.
- [20] Zhou, H., T. Yu, et al. "Japanese Dependency Analysis Based on SVMs and CRFs."

- INTERNATIONAL JOURNAL of MATHEMATICS  
AND COMPUTERS IN SIMULATION 1(3):  
233-237. 2007.
- [21] J. Lafferty, A. McCallum, F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data", Proceedings. 18th International Conference on Machine Learning, pp. 282-289, 2001.
- [22] S. Della Pietra, V. Della Pietra, J. Lafferty, "Inducing features of random fields", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.19, No.4, pp.380-393, 1997.
- [23] A. L. Berger, V. J. Della Pietra, S. A. Della Pietra, "A maximum entropy approach to natural language processing", Computational Linguistics, Vol.22, No.1, pp.39-71, 1996.
- [24] <http://crfpp.sourceforge.net/>
- [25] 세종계획 21, <http://www.sejong.or.kr/>
- [26] 홍진표, 차정원, "어절패턴 사전을 이용한 새로운 한국어 형태소 분석기", 한국 컴퓨터 종합 학술대회 논문집 제 35권, pp. 279-284, 2008
- [27] Yamada, H., Matsumoto, Y., "Statistical dependency analysis with support vector machines," In Proceedings of IPWT, 2003.
- [\*] 우연문, 송영인, 박소영, 임해창, "지배가능 경로 문맥을 이용한 의존 구문 분석의 수식 거리 모델", 정보과학회논문지 : 소프트웨어 및 응용 제 34권 제2호, pp.140-149, 2007