미등록어 추정을 이용한 TAKTAG의 개선

차정원 이원일 이근배 이종혁
포항공과대학교 전자계산학과

Improvement of TAKTAG using unknown-word handling

Jeongwon Cha Wonil Lee Geunbae Lee Jong-Hyeok Lee
Dept. of Computer Science and Engineering
Pohang University of Science and Technology

요 약

본 논문에서는 음소단위의 bigram과 trigram 정보를 이용하여 어절내에서의 위치와 개수에 관계없이 미등록어를 추정하고, 미등록어용 형태소 패턴 사전을 도입하여 마치 등록어처럼 미등록어를 처리할 수 있는 방법을 제안한다. 제안된 미등록어 추정 모델은 조사나 어미와 같은 기능어에 의한 간접적인 추정방법이 아닌 미등록어 자체의 추정과 접속정보를 이용한 검사를 동시에 하여 정확도를 높였다. 본 미등록어 추정방법은 기존의 한국어 품사태깅모델인 TAKTAG에 적용하여 미등록어가 포함된 어절에 대해서 83.72%의 성능을 보였다..

1. 서 론

품사 태깅을 수행하는 품사 태거(tagger)는 어휘적 중의성으로 인한 구문 분석 단계에서의 과다한 부담을 줄이기 위한 파서(parser)의 전처리기로 사용되거나 [1] 정보 검색 시스템에서 높은 재현율 및 정확도를 갖는 색인어와 검색어 추출을 위한 작업 등 자연언어처리의 전 분야에 걸쳐 폭 넓게 사용될 수 있다.

그런데 이런 품사 태깅 시스템에서는 자료부족 문제, 미등록어 처리 문제, 중의성 해결 문제 등 해결해야 할 큰 문제가 여러 가지 있다. 그 중에서도 미등록어 처리는 유형이 다양하고 자주 발생하므로 형태소 분석 실패의 주원인이 된다[2].

지금까지 미등록어에 대한 연구는 기능어를 분리해 내고 이것에 접속 가능한 품사를 예측하는 방법과 유사어절 비교방식 등으로 크게 나누어 볼 수 있다 하지만 이러한 방법은 모두 다음과 같은 두 가지 가정하에서 만들어진 방법이다.

첫째는 미등록어는 어절내에서 하나만 존재한다. [3]에서는 미등록 어절에서 조사 사전을 이용하여 형태소 분석에 실패한 어절로부터 최장조사를 떼어내고 나머지를 미등록어로 추정한다. 따라서 "최진실신드롬은" 과 같이 어절내에서 두 개 이상의 마등록어가 있거나 "미등록어+명사+조사"의 어절은 "미등록어+조사"로 추정하여 정확도를 떨어뜨린다[4].

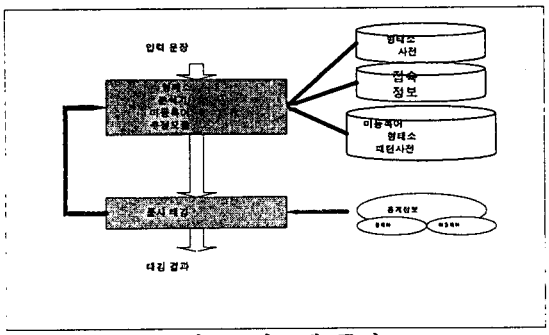
둘째는 미등록어는 어절의 앞부분에 존재한다. [5]에서는 미등록어절에서 뒤에서부터 기능어를 떼어내고 나머지 앞부분을 휴리스틱과 언어정보를 이용하여 미등록어를 추정한다. 이러한 방법에서는 "초현실주의파"와 같이 미등록어가 어절 뒷부분에 나올 경우에는 추정을 못한다.

본 논문에서 이러한 문제를 해결하기 위해 [6]을 개선한 시스템을 소개한다. 원래의 TAKTAG에서는 통계와 규칙을 혼합한 hybrid approach를 사용하였으나 본 논문에서는 전반부인 통계에 기반한 품사 태깅 방법의 성능향상에 촛점을 맞추어, 형태소 분석기, 미등록어 추정 모듈, 접속 검사 모듈, 태깅 모듈로 구성되어 있다. 입력 문장에 대해서 형태소 분석기는 어절의 왼쪽에서부터 읽어 들여 모든 가능한 형태소 분석 결과를 내고 접속 검사 테이블을 이용하여 형태소 그래프로 만든다. 이 때 미등록어 추정 모듈이 함께 동작하여 미등록어 패턴 사전을 참조하여 등록어, 미등록어 모두에 대하여 가능한 후보를 내어 미등록어도 마치 등록어처럼 처리하게 된다. 이렇게 함으로써 미등록어가 항상 앞부분에 있다는 가정도 필요없고 하나라는 가정도 필요없이 처리할 수 있다. TAKTAG의 개선 모델에서는 기존에 형태소 분석에서 실패한 어절에 대해서만 미등록어를 추정하던 방법과는 달리 형태소 분석에서 미등록어를 함

께 처리한다.

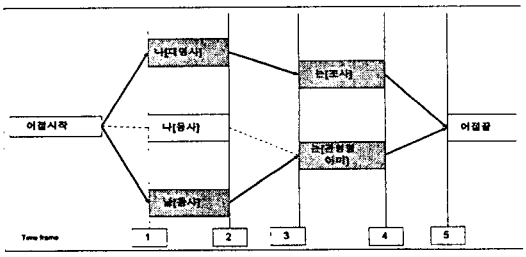
본 논문의 구성은 다음과 같다. 2 장에서는 형태소 분석 및 태깅에 대해 기술하고 3 장에서는 미등록어 추정모듈에 대해 설명하고 4 장에서는 실험및 결과, 5 장에서 결론을 맺는다.

2. 형태소 분석 및 태깅



[그림 1] 시스템 구성도

TAKTAG 개선 시스템의 형태소 분석기는 주어진 어절을 왼쪽에서부터 CYK 알고리즘으로 형태소 사전 뿐만이 아니라 미등록어용 형태소 패턴 사전을 동시에 찾아서 분석 결과를 하나의 형태소 그래프로 표현한다. 이렇게 형태소 별로 그래프를 만들면서 Viterbi 알고리즘을 이용하여 최적의 n 경로를 찾고 나머지 노드는 제거한다.



[그림 2] "나는"의 형태소 그래프

예를 들어 어절 "나는"을 분석할 경우 "나[대명사]+는[조사]", 나[동사]+는[관형형어미]", "날[동사]+는[관형형어미]" 그리고 미등록어용 형태소 패턴 사전을 찾아서 나온 후보들로 그래프가 만들어진다. [그림 2]에서 Time frame 2 에 오면 "나[대명사]", "나[동사]", "날[동사]"와 형태소 패턴 사전에서 후보들로 그래프가 만들어 진다. 여기서 Viterbi 알고리즘을 적용하여 각각의 확률값을 계산하여 "나[대명사]"와 "날[동사]" (n=2 라고 가정)가 선택되면 나머지 노드는 제거하여 그래프의 경로를 줄여 후보를 줄여줌으로써 경로의 과다 생성으로 인해 정확도를 떨어뜨리는 일은 없어진다. [그림 2]에서 Time frame 4 에 오면 "는[조사]", "는[관형형어미]"와 패턴 사전에서 나온 후보들이 나타나는데 경로가 줄어들어 정확도를 높일 수 있다.

그리고 미등록어가 아닌 경우에도 미등록어 모듈에서 후보를 내는 이유는 하나 이상의 경로를 만들었다고 하더라도 이 경로가 항상 올바른 경로라고 확신할 수 없기 때문이다. 예를 들어 "나는 새"의 경우에 만약 "날[동사]"가 사전에 등록이 되어 있지 않다면 항상 그릇된 결과를 낼 수 있다.

여기에서 사용된 태깅 모델은 bigram 확률 모델이고 지도학습을 하였으며 Viterbi 알고리즘에서 수치적 안정성을 위해 log 함수를 이용하여 scaling 한다[7].

형태소 분석기가 n 개의 경로를 만들어 주면 문장의 마지막에 도달했을 때 역으로 최적 경로를 찾아가게 된다.

3. 미등록어 추정

3.1 미등록어용 형태소 패턴 사전

미등록어는 추정 형태소에 품사를 할당할 방법이 없다. 기존의 연구에서는 조사나 어미와 같은 기능어로부터 접속 가능한 품사를 예측하는 방법을 사용했다. 그러나 본 논문에서는 이러한 방법을 사용하지 않고 미등록어에 보다 많은 정보를 줄 수 있게 모든 형태소의 패턴을 분류해 만든 형태소 패턴 사전을 만들어 사용한다. 미등록어용 형태소 패턴 사전은 형태소에서 접속검사시 중요한 정보를 패턴별로 분류하여 여기에 품사를 할당하여 만든 사전이다. 즉 형태소의 패턴을 보고 품사와 접속정보를 할당해 줄 수 있는 사전이다.

예를 들어 "소묘는"을 분석한다고 하자. 그리고 "소묘"가 미등록어라고 하자. 그러면 형태소 분석기에서 "소묘"라는 입력 형태소를 사전에서 찾아본다. 미등록어이므로 사전에 없다. 이럴 경우 'æ쇼'È쿋나는 형태소가 가질 수 품사를 정의해 놓은 [그림 3]과 같은 미등록어용 형태소 패턴 사전을 찾아서 품사를 할당한다.

```
(*쇼)
<M0>[#MORPH {*m} #CTRL{SE} #LCI {MC*m[hdcp]} #RCI {MC*m[hdcp]}]
<M1>[#MORPH{*m} #CTRL{SE} #LCI {MC*m[HLS]}  #RCI {MC*m[HLS]}]
<M2>[#MORPH{*m}#CTRL{SE} #LCI {MP*m[hdcp]}  #RCI {MP*m[hdcp]}]
<M3>[#MORPH{*m}#CTRL{SE}#LCI {MP*m[HLS]}  #RCI {MP*m[HLS]}]
(*쇼)
<M0>[#MORPH  {*m}  #CTRL  {SE}  #LCI {T*y[p]}  #RCI {T*y[p]}]
```

```
(*ㅛ)
<M1>{#MORPH {*m} #CTRL {SE} #LCI {S*m[]} #RCI {S*m[]}}
(*ㅛ)
<M0>{#MORPH{*m} #CTRL {SE} #LCI {G*m[CK]} #RCI {G*m[CK]}}
(*ㅛ)
<M0>{#MORPH{*m} #CTRL{SE} #LCI {BWm[jCjSjO]} #RCI
{BWm[jCjSjO]}}
<M0>{#MORPH{*m} #CTRL{SE} #LCI {BOm[jCjSjO]} #RCI {Bom[jCjSjO]}}
(*ㅛ)
<M0>{#MORPH {*m} #CTRL {SE} #LCI {DRm[a]} #RCI {DRm[a]}}
(*ㅛ)
<M0>{#MORPH {*m} #CTRL {SE} #LCI {HRm[a]} #RCI {HRm[a]}}
```

[그림 3] 미등록어용 형태소 패턴 사전

그림에서 보면 'ㅛ'로끝나는 "보통명사(MC),"고유명사(MP)", "대명사(T)", "수사(S)", "관형사(G)", "의문부사(BW)", "그외 부사(BO)", "규칙 형용사(HR)", "규칙동사(DR)"에 대해서 추정을 하는 것을 알 수 있다.

현재 미등록어용 형태소 패턴 사전에는 51 개의 패턴이 있다. 이러한 방법은 기능어로 예측하는 방법이 미등록어의 위치나 수를 제한하고 접속정보를 정확히 줄 수 없는데 반해 이러한 문제를 해결할 수 있다.

3.2 미등록어 확률 추정 모듈

이렇게 그래프를 구성하면서 태깅을 하면 미등록어를 등록어와 동일한 방법으로 태깅을 할 수 있다. 다만 경로가 많아진다. 실험에 의하면 어절의 평균 경로수는 5.07 개이다.

여기서 미등록어 각각에 대한 $P(w|t)$는 다음과 같은 공식에 의해서 구해진다[8]. 만약 $C = c_1 c_2 \ldots\ldots c_n$이 품사 t를 가지는 단어 w를 구성하는 n개의 음소열이라고 하자.

$$P(w|t) = P_t(C)$$
$$= P_t(c_1|\#,\#)P_t(c_2|\#,c_1)\prod_{i=3}^{n} P_t(c_i|c_{i-2},c_{i-1})P_t(\#|c_{n-1},c_n)$$

여기서 '©#'©은어의 처음과 끝을 표시하는 문자이다. 음소 trigram 확률은 품사 t가 나타나는 bigram 과 trigram 음소의 상대적 빈도수를 학습 코퍼스에서 평가하게 된다.

$$P_t(c_i|c_{i-2},c_{i-1}) = f_t(c_i|c_{i-2},c_{i-1}) = \frac{N_t(c_{i-2},c_{i-1},c_i)}{N_t(c_{i-2},c_{i-1})}$$

여기서 $N_t(c_{i-2},c_{i-1},c_i)$는 학습 코퍼스에서 품사 t로써 나타나는 trigram 음소열 $c_{i-2}c_{i-1}c_i$의 총 수를 나타낸다.

학습 코퍼스의 량이 많지 않으므로 상대적으로 "보통명사"의 빈도가 많아지는 약점이 있다.

만약 "소묘"의 "보통명사"에 대한 추정 관측확률 값은 다음과 같이 구해진다

P(소묘|MC)
$= P_{MC}(ㅅ|\#,\#)P_{MC}(ㅗ|\#,ㅅ)P_{MC}(ㅁ|ㅅ,ㅗ)P_{MC}(ㅛ|ㅗ,ㅁ)P_{MC}(\#|ㅁ,ㅛ)$

미등록어는 모든 품사를 예측할 수도 없고 또 할 필요도 없다. 그래서 개방단어에 속하는 보통명사(MC), 고유명사(MP), 대명사(T), 수사(S), 관형사(G), 의문부사(BW), 그외 부사(BO), 감탄사(K), 규칙동사(DR), 불규칙동사(DI), 규칙형용사(HR), 불규칙형용사(HI) 에 대해서만 추정을 한다.

4. 실험 및 결과

4.1 학습

태깅 정확율을 평가하기 위하여 총 40,000 여 형태소 코퍼스를 사용하였다. 학습 방법은 올바르게 태깅된 학습 코퍼스에서 나타난 어휘가 가지는 품사나 품사사이의 공기관계를 이용하여 전이확률과 관측확률을 추정하는 지도학습을 이용하였다. 추정 공식은 아래와 같다.

$$P(w_i|t_i) = \frac{f(w_i,t_i)}{f(t_i)}$$
$$p(t_i|t_{i-1}) = \frac{f(t_{i-1},t_i)}{f(t_{i-1})}$$

여기서 $f(t_i)$는 학습 데이터에 나타난 품사 태그 t_i를 가진 형태소의 수, $f(w_i,t_i)$는 품사 태그 t_i를 가진 형태소 w_i의 수, $f(t_{i-1},t_i)$는 품사 태그t_{i-1}과 t_i가 연이어 나오는 빈도수를 나타낸다.

또한 미등록어 처리를 위한 음소별 trigram, bigram 빈도수를 동일한 학습 코퍼스를 이용하여 추출하였다. 코퍼스의 수가 많지 않은 관계로 보통명사의 출현 빈도가 높아 추정에서 보통명사로 추정될 확률이 높은데 이는 더 많은 코퍼스를 사용하고 사전을 확충하면 실제 상황을 더 잘 모델링 할 수 있을 것으로 본다.

4.2 결과

실험에서는 다음과 같이 총 10946 형태소를 이용하였다.

- ●국민교육헌장
- ●소설 운명의 힘

●쓰기교과서
●자연교과서 1
●자연교과서 2

결과는 미등록어의 추정 정확도를 알아보기 위해 미등록어 추정 정확도를 따로 분리해서 결과를 내어 보았다.

	형 태 소		정 확 도	TAKTAG의 전처리 부분
국민교육헌장	총 형태소	263	96.57	95.2
	미등록어	0		NA
	등록어	263	96.57	NA
소설운명의 힘	총 형태소	718	88.30	84
	미등록어	40	60.00	NA
	등록어	678	89.97	NA
쓰기교과서	총 형태소	2073	91.79	91.6
	미등록어	20	90.00	NA
	등록어	2053	91.81	NA
자연교과서 1	총 형태소	4155	91.98	88.6
	미등록어	203	89.16	NA
	등록어	3952	92.13	NA
자연교과서 2	총 형태소	3737	91.00	91.6
	미등록어	118	81.35	NA
	등록어	3619	91.32	NA
총 계	총 형태소	10946	91.48	89.9
	미등록어	381	83.72	NA
	등록어	10565	91.76	NA

5. 결론

본 논문에서는 미등록어를 고려해 TAKTAG를 개선한 품사 태깅 시스템을 소개하였다. 본래 TAKTAG는 Hybrid Approach를 이용하는 시스템이므로 전처리기로 확률을 이용하고 후처리기로 Eric Brill 형식의 규칙을 이용하여 태깅을 한다. 이번에 소개된 시스템은 바로 이 전처리기를 개선하려는 시도에서 만들어진 시스템이다. 현재 전체적인 정확도는 91.48%, 미등록어 추정 정확도는 83.72%로 TAKTAG의 전처리기의 전체 정확도 89.9%보다 약 2% 정도의 향상을 보인다. 미등록어의 추정 정확도가 좋지 않지만 이것은 방법론적인 문제가 아니라 학습량의 문제라고 생각된다. 앞으로 보다 많은 학습 코퍼스를 사용하여 학습하고 규칙 부분이 완성되면 보다 정확도에서 우수하고 견고한 시스템이 되리라고 확신한다.

6. 참고문헌

1. E Charniak, G. Carroll, J. Adcock, A. Casandra, Y, Gotoh, J. Katz, M. Littman, and J. McCann, "Taggers for parsers", Technical Report CS-94-06, Dept. of Computer Science Brown University, 1994.
2. R. Weischedel, R. Schwartz, J. Ralmucci, M. Meteer, L. Rawshaw "Coping with ambiguity and unknown words through probabilistic model", Computational Linguistics Vol. 19, No.2, pp. 359-382, 1993
3. 강승식,"음절 정보와 복수어 단위정보를 이용한 한국어 형태소 분석", 서울대학교 컴퓨터 공학과 박사학위 논문, 1993
4. 양장모, 김민정, 권혁철 "언어 정보를 이용한 한국어 미등록어 추정" 한국정보과학회 봄 학술발표논문집 Vol.23, No.1, pp.957-960,1996.
5. 이상호, 서정연, 오영환, "KTS : 미등록어를 고려한 한국어 품사 태깅 시스템", 음성통신 및 신호처리 워크샵 논문집(제 SCAS-12 권 1 호), pp.195-199,1995
6. 신상현, TAKTAG : 통계와 규칙에 기반한 혼합형 한국어 품사 태깅 시스템, 포항공대 전자계산학과 석사학위 논문, 1996
7. Dong Cutting, Julian Kupiec, Jan Pedersen and Penelope Sibun, "A practical part-of-speech tagger", Proceedings of the 3rd conference on applied natural language processing, pp.133-140, 1992.
8. Masaaki, "A stochastic Japanese Morphological Analyzer Using a Forward-DP Backward-A* N-Best Search Algorithm" Proceedings of 15th International Conference on Computational Linguistics Vol1. pp.201-207, 1994.