韓國 컴퓨터情報學會 2007하계학술발표논문집&학회지
第15卷 第1號, 2007. 6.

말뭉치 오류를 고려한 HMM 한국어 품사 태깅 시스템

설 용 수*, 김 동 주*, 김 규 상*, 김 한 우*

A Korean POS Tagging System with Handling Corpus Errors

Yong-Soo Seol*, Dong-Joo Kim*, Kyu-Sang Kim*, Han-Woo Kim*

요 약

통계 기반 접근 방법을 이용한 품사태깅에서 태깅 정확도는 훈련 데이터의 양에 좌우될 뿐 아니라, 말뭉치가 충분할지라도 수작업으로 구축한 말뭉치의 경우 항상 오류의 가능성을 내포하고 있으며 언어의 특성상 통계적으로 신뢰할만한 데이터의 수집에도 어려움이 따른다. 훈련 데이터로 사용되는 말뭉치는 많은 사람들이 수작업으로 구축하므로 작업자 중 일부가 언어에 대한 지식이 부족하다거나 주관적인 판단에 의한 태깅 실수를 포함할 수도 있기 때문에 단순한 저빈도와 관련된 잡음 외의 오류들이 포함될 수 있는데 이러한 오류들은 재추정이나 평탄화 기법으로 해결될 수 있는 문제가 아니다. 본 논문에서는 HMM(Hidden Markov Model)을 이용한 한국어 품사 태깅에서 재추정 후 여전히 존재하는 말뭉치의 잡음에 인한 태깅 오류 해결을 위해 비터비 알고리즘 적용 단계에서 데이터 부족과 말뭉치의 오류로 인해 문제가 되는 부분을 찾아내고 규칙을 통해 수정을 하여 태깅 결과를 개선하는 방안을 제안한다. 실험결과는 오류가 존재하는 말뭉치를 사용하여 구현된 HMM과 비터비 알고리즘을 적용한 태깅 정확도에 비해 오류를 수정하는 과정을 거친 후 정확도가 향상됨을 보여준다.

▸ Keyword : HMM 품사 태깅, 한국어 품사 태깅, HMM POS Tagging, Korean POS Tagging

Ⅰ. 서 론

컴퓨터를 이용하여 자연언어를 처리하기 위해서는 자연언어를 컴퓨터가 처리할 수 있는 형태로 분석해 내는 과정이 필수적이다. 형태소 분석, 구문 분석, 의미 분석 등이 그 예가 되겠다. 각각의 분석 단계에서 정확한 분석결과를 얻기 위해서는 언어의 기본 특성중 하나인 모호성을 해소해야만 한다.

품사 태깅은 어휘적 모호성 대한 해결책으로 사용(한국어의 품사 태깅은 한국어의 언어적 특성 때문에 어휘적 모호성뿐만 아니라 경계 모호성, 형태적 모호성도 해결해야 한다.)되며 구문 분석과 의미 분석의 전처리 과정으로 사용되기도 한다.

품사 태깅을 위해 접근하는 방식은 통계를 기반한 방법, 규칙을 기반한 방법, 이 두 가지를 통합한 통합

• 제1저자 : 설용수
*한양대학교 컴퓨터공학과

방법이 있다. 이 중 통계 기반 품사 태깅은 모든 어휘 중의성을 해결할 수 있으면서도 높은 태깅 정확도를 제공한다는 장점 때문에 많이 사용되는 태깅 방법이다.

그러나 통계 기반 품사 태깅이 좋은 성능을 내기 위해서는 충분한 양과 높은 품질의 말뭉치가 필요하다. 실세계 언어 현상을 충분히 대표할 수 있는 양과 질의 말뭉치는 현실적으로 존재하기 어렵다. 이 때문에 모든 통계 기반 품사 태깅 시스템은 시스템 설계자의 의도에 부합할 만큼의 성능을 내지 못한다.

말뭉치의 양이 부족하면 자료 부족 문제(data sparseness problem)가 발생하고 말뭉치에 오류가 있으면 잡음(noise)이 나타난다. 일반적으로 자료 부족 문제와 잡음 문제를 해결하기 위해 평탄화 방법이나 재추정 방법 등이 사용되지만 말뭉치에 기인한 태깅 오류를 해결하는 근본적인 방안이 되지 못한다.

본 논문에서는 통계 기반 품사 태깅에서 많이 사용되는 HMM과 Viterbi알고리즘을 이용한 품사 태깅에서 자료 부족 문제와 자료 오류로 인해 신뢰하기 힘든 부분을 검출하는 척도를 제안하고, 이를 사용하여 검출된 부분은 규칙을 적용하여 수정함으로써 태깅 정확도를 향상 시키는 과정을 보인다.

2. 본론

2.1 관련 연구

통계기반 품사 태깅 방법은 크게 어휘 확률만을 이용하는 방법, HMM의 자율 학습을 이용하는 방법, N-gram의 문맥 확률과 어휘 확률을 이용하는 방법, 신경망을 이용하는 방법, 퍼지망을 이용하는 방법 등이 있다[1]. 한국어 품사 태깅은 이중에서도 HMM을 이용한 태깅 방식이 많이 연구되었다. 그 중 많이 알려진 연구들을 살펴보면, [2]에서는 51개의 품사 분류, 지도학습, 형태소 단위로 태깅하여 93.6%의 정확도를 보였고, [3]에서는 어절단위 문맥을 고려할 수 있게 변형된 형태소 단위 HMM을 통해 태깅하는 시스템으로 52개의 품사분류, 지도학습으로 98.3%의 정확도를 보였고, [4]에서는 14개의 품사분류, 자율학습, 어절단위 태깅으로 93.7%의 정확도를 보였고, [5]에서는 13개의 품사 분류, 지도학습, 변형된 어절단위 태깅으로 98.8%의 정확도를 보였고, [6]에서는 25개의 품사 분류, 지도학습, 어절단위 HMM과 변형규칙을 이용하여 91.7%의 정확도를 보였으며 [7]에서는 59개의 품사 분류, 지도학습, Twoply HMM을 이용하여 94.4%의 정확도를 보였다. 이러한 연구들은 자료 부족 문제에 대한 해결책으로 평탄화 기법이나 재추정 기법을 사용하였을 뿐 말뭉치가 가진 오류의 가능성을 시스템 설계시에 고려하지는 않았다.

2.2 통계 기반 품사 태깅에서의 자료 부족 문제

학습 자료에서 관련된 확률정보가 부족하여 시스템의 정확도를 떨어뜨리는 문제를 자료 부족 문제(data sparseness problem)라고 한다.

품사 태깅 시스템에서 자료 부족 문제는 말뭉치의 양이 부족하거나 말뭉치가 언어의 모든 특성을 골고루 반영하고 있지 않아서 발생한다. 품사 태깅에 있어 자료 부족 문제는 태깅 정확도에 큰 영향을 미친다. 자료 부족 문제를 해결하기 위해 일반적으로 평탄화(smoothing) 방법이나 재추정(re-estimation) 방법을 사용한다. 그러나 이러한 방법들은 태깅의 정확도를 개선할 뿐 해결하지는 못한다. 대부분의 선행연구에서는 자료 부족 문제를 해결하기 위한 여러 가지 방법들을 사용하였다. 미등록어를 처리할 수 있도록 설계하거나 태깅 단위를 어절 단위 대신 형태소 단위를 사용하기도 하고, 구분 가능한 품사의 개수를 줄이거나, HMM의 파라미터를 변형하기도 하는 등 다양한 시도를 해 왔다. 그러나 자료 부족 문제는 충분한 양질의 자료가 없이는 근본적으로 해결할 수 없다.

2.3 HMM과 비터비 알고리즘을 이용한 품사태깅

통계를 기반으로 하는 품사 태깅 방법 중 HMM과 비터비 알고리즘을 이용하는 방법은 비교적 간단한 알고리즘이면서도 높은 정확도를 보여주는 방식으로 많이 이용되고 있다. 이는 크게 두 가지 과정으로 요약된다. 첫째는 HMM 학습단계이다. 대량의 말뭉치를 이용하여 태깅하고자 하는 언어에 맞는 상태전이확률, 심볼발생확률, 초기상태확률의 세 가지 파라미터를 학습한다. 둘째는 비터비 알고리즘을 이용하여 입력문장에 대해 최고 확률을 보이는 품사열을 찾아내는 단계이다. 앞 단계에서 학습된 HMM의 파라미터를 이용하여 품사열의 발생확률을 계산하여 가장 확률이 높은 품사열을 결과로 도출한다.

2.4 말뭉치의 오류가 태깅 정확도에 미치는 영향

말뭉치는 학습과정을 통해 통계모델의 파라미터를 조절하는데 사용된다. 따라서 말뭉치가 오류를 갖고 있다면 그

말뭉치로 만들어진 모델의 파라미터도 오류를 갖게 된다. 말뭉치는 자동으로 구축되기도 하지만 일반적으로 많은 인원과 노력이 동원되어 제작된다. 품사 부착된 말뭉치의 경우는 더 많은 인원과 노력이 필요하다. 이때 작업에 참여하는 사람들은 공통적인 관점으로 품사 태깅 작업을 해야 하며 태깅 대상 언어에 대한 이해가 충분해야 한다. 그러나 실제 말뭉치 제작시에는 언어의 기본적 특성인 모호성 때문에 태깅 후보 품사중 하나를 선택하는 경우가 빈번하게 일어나고 이때마다 다수의 사람들이 공통적인 관점으로 하나의 품사를 결정하는 것은 기대하기 힘들다. 또한 작업자 중 일부는 대상 언어에 대한 이해도가 충분하지 않아서 실수를 범하기도 한다. 이처럼 말뭉치는 오류의 가능성을 항상 내포하고 있기 때문에 이를 사용하여 만들어진 태깅 시스템은 완벽할 수가 없다.

2.5 저신뢰도 구간 검사

본 논문에서는 말뭉치에 작업자의 언어지식부족이나 주관적견해가 포함된 오류가 포함될 가능성이 있음을 고려한 상태에서 자료 부족 문제와 말뭉치의 오류로 인한 잡음을 찾아내고자 한다. 비터비 알고리즘 적용 단계에서는 HMM의 파라미터를 이용하여 품사열의 발생 확률을 계산한다. 이때 참조되는 파라미터의 특성을 이용하면 자료 부족 문제나 말뭉치의 오류로 인해 태깅 결과를 신뢰하기 힘든 구간을 검출할 수 있다.

순수하게 HMM과 비터비 알고리즘만을 적용한 태깅 시스템에서 잘못 태깅된 구간들을 분석해 본 결과 세 가지 특성이 있었다. 저 빈도로 구해진 확률 값을 가지고 있거나 확률 분포의 엔트로피가 높은 경우, 그리고 품사전이확률과 어휘발생확률의 분포가 유사하지 않을 경우에 잘못된 태깅 결과가 나올 확률이 높았다.

HMM의 파라미터인 품사전이확률, 어휘발생확률, 초기 품사발생확률은 확률값들로 이루어져 있다. 그러나 같은 확률값이라 할지라도 표본의 크기에 따라 신뢰도는 다르다. 예를 들어 70%의 확률을 가진 품사전이확률이라 할지라도 말뭉치에서 총 10번 중 7번 관찰되어 도출된 확률보다는 1,000번 중 700번 관찰되어 도출된 확률의 신뢰도가 높을 것이다. 이처럼 말뭉치에서 저빈도로 관측된 확률값은 언어 특성상 저빈도로 관측되었거나 말뭉치의 오류로 인한 잡음일 수 있다. 따라서 참조되는 확률값이 저빈도 데이터로부터 도출된 확률이라면 신뢰도가 낮다고 판단할 수 있다.

태깅 후보 품사중 하나의 품사에 대해서만 높은 확률이 나왔다면 태깅 신뢰도가 높겠지만 후보 품사들이 비슷한 확률을 보인다면 최대 확률을 선택한다 할지라도 다음 등위의 품사가 올바른 선택일 가능성도 배재할 수 없다. 이 같은 특성을 표현하는 척도로 엔트로피가 있다. 엔트로피란 선택될 대상의 불확정성을 나타내는 척도로써 (1)와 같이 표현 한다.

$$H(M) = E\{I(m)\} = \sum_{m \in M} p(m) I(m) = -\sum_{m \in M} p(m) \log p(m) \quad \cdots\cdots (1)$$

- $H(M)$: 엔트로피
- $E\{\ \}$: $\{\ \}$의 기대값
- $I(m)$: m이 선택되지 않을 확률
- M: 선택 대상의 전체 집합
- m: 하나의 선택 대상
- $p(m)$: m이 선택될 확률

즉, 엔트로피의 값이 크면 불확정성이 크다는 의미로 선택의 기준이 되기 어려움을 뜻한다. 반대로 엔트로피의 값이 낮으면 불확정성이 낮다는 의미로 선택의 기준으로 적합함을 의미 한다. 따라서 품사 태깅에서는 엔트로피가 낮은 확률 분포일수록 태깅의 신뢰도를 높일 수 있다.

비터비 알고리즘에서는 최적 품사열을 찾기 위해 $\delta_{t+1}(j) = \max\left[\delta_t(i) a_{ij}\right] b_j(o_{t+1})$ 식 을 사용한다. $\delta_{t+1}(j)$는 $\delta_t(i)$와 품사전이확률(a_{ij})의 곱의 최대값과, 어휘발생확률($b_j(o_{t+1})$)에 비례한다. 즉, 이 두 값들이 모두 높은 값을 갖는다면 j품사가 선택될 확률이 높고 이 원리에 의해 품사가 결정된다. 그러나 품사전이확률은 $\delta_t(i)$와 i품사에서 j품사로의 전이확률을 곱한 값의 최대값에서의 품사전이확률이 선택된다. 즉, 선택된 품사 전이 확률의 i품사와 어휘발생확률의 j품사가 동일하지 않다면 두 확률이 모두 높게 나왔다 할지라도 각각 다른 품사를 대상으로 하여 높은 확률이 나왔기 때문에 $\delta_{t+1}(j)$만 높은 값이 나온 것일 뿐 j품사로 인해 전체가 높은 확률 값이 나왔다고 판단할 수 없다. 따라서 품사전이확률과 어휘발생확률의 확률 분포 성향이 다르다면 태깅의 신뢰도가 낮아진다. 본 논문에서는 두 확률분포간의 유사도를 측정하는 척도 중 상관계수를 사용하였다. 상관 계수는 공분산 $Cov(X, Y)$을 X와 Y의 표준편차의 곱으로 나누어 얻은 값으로, X와 Y의 단위와 무관한 척도를 얻기 위해 사용된다. 위의 특성들을 고려한 (2)의 척도를 이용하여 저신뢰도 구간을 찾는다.

$$R = \frac{\log\left\{\alpha\left(\sum_{i=1}^{N} f_{Ai} \cdot b_i(k)\right) + \beta\left|CORR(A,B)\right|\right\}}{H(A,B)} \quad \cdots\cdots (2)$$

- f_{Ai}: i품사로의 품사전이확률의 빈도수
- $b_i(k)$: i품사에서 k 어휘심볼이 발생할 확률
- $CORR(A,B)$: 품사전이확률(A)와 어휘발생확률(B)의 상관계수
- $H(A,B)$: 품사전이확률(A)와 어휘발생확률(B)의 엔트로피
- α: 품사전이확률의 빈도수의 가중치
- β: 상관계수의 가중치

세 가지 특성 중 상관 계수의 절대값과 빈도는 신뢰도와 비례하고, 엔트로피는 반비례하는 특성을 갖는다. 따라서 빈도와 상관계수는 분자에 위치하고 엔트로피는 분모에 위치한다. 각 특성을 나타내는 부분은 아래와 같다.

- $\alpha\sum_{i=1}^{N} f_{Ai} \cdot b_i(k)$: 빈도

i품사로의 품사 전이확률의 빈도수를 모두 합하되 현재 관측된 어휘가 가질 수 없는 품사를 걸러내기 위해 어휘발생확률을 곱해준다. α는 빈도가 신뢰도에 미치는 영향을 조절하기 위한 가중치이다.

- $\beta|CORR(A,B)|$: 품사전이확률과 어휘발생확률의 확률분포의 성향

품사전이확률과 어휘발생확률의 상관계수를 구하여 양 또는 음의 상관도를 구할 수 있다. 하지만 우리는 크기에만 관심이 있기 때문에 절대값을 이용한다. β는 확률분포 성향이 신뢰도에 미치는 영향을 조절하기 위한 가중치이다.

- $H(A,B)$: 품사전이확률과 어휘발생확률의 엔트로피로써 다시 말하면 각 품사가 선택될 확률 분포의 엔트로피이다.

분자에 log를 취하여 빈도와 상관계수의 합이 극히 낮을 때에만 척도의 값이 급감하도록 하였고, 분모에 엔트로피를 넣어 엔트로피가 0에 가까울수록 척도 값이 급격하게 커질 수 있도록 하였다(구현 시 엔트로피가 0이 나오는 경우가 있을 수 있는데 이 경우에는 미리 정의해 놓은 척도의 최대값으로 결과를 대체한다).

2.6 저신뢰도 구간의 태깅 수정

비터비 알고리즘 적용단계에서 저신뢰도 검출 척도를 구하고 임계값 미만의 구간을 저신뢰도 구간으로 검출한다. 검출된 구간은 수정단계에서 수정 목표를 찾을 수 있도록 비터비 알고리즘에서 역추적을 하기위해 사용되는 ψ에 unknown 표기를 한다.

δ 테이블

	나무	는	푸르	다
ncn	0.5			
jco		0.06		
ef	0.2	0.002		0.000003
paa		0.01	0.00024	

태그 저장 배열

$\Psi_1(1)$	$\Psi_2(2)$	$\Psi_3(4)$	$\Psi_4(3)$
null	ncn	unknown	paa

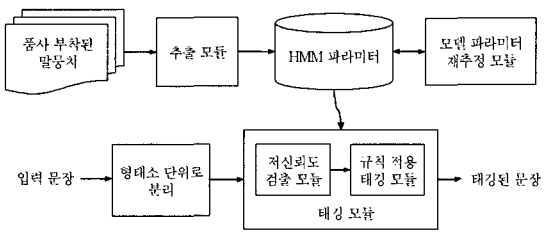
<그림 1> 저신뢰도 구간 검출 후, 수정 전처리의 예

그림 1은 신뢰도 측정 결과 '는' 형태소 구간에서 임계값보다 낮은 신뢰도가 나온 경우 ψ에 unknown 표기가 된 예를 나타낸다. 문장 내의 모든 구간의 신뢰도 측정 및 unknown 표기 단계가 끝나면 수정 단계로 넘어간다.

비터비 알고리즘 적용단계가 끝나면 unknown 표기를 한 구간을 수정한다. 수정은 규칙기반 품사 태깅에서 사용되는 다양한 방법들을 적용할 수 있을 것이다. 본 논문에서는 수정 규칙 방법의 좋고 나쁨에 따른 시스템의 정확도 변화가 논지가 아니고 신뢰도가 낮다고 판단되는 태깅 결과를 얼마나 잘 검출할 수 있는가를 논지로 한다. 따라서 본 논문에서는 수정 알고리즘에 의한 시스템의 태깅 정확도 향상보다는 저신뢰도 검출 척도의 오류 검출 성능이 얼마나 좋은가에 관심을 둔다. 이러한 관점에서 수정을 위한 규칙 적용을 위해 시스템의 크기를 키우기 보다는 간단한 수정 규칙을 사용하여 수정하는 방식을 택하였다.

수정 방식은 대용량의 품사 부착된 말뭉치로부터 품사열의 패턴을 추출한 뒤 unknown 표기가 되어 있는 부분의 앞과 뒤의 태깅 결과와 비교한 후 적합한 패턴의 품사열을 참조하는 방식을 사용하였다. 상세한 품사열 패턴을 이용한 규칙 생성및 적용 과정은 아래와 같다. 품사 부착된 말뭉치에서 길이가 2~4인 품사열 패턴을 추출하여 데이터베이스에 저장해 둔다. 검출 척도에 의해 검출된 신뢰도가 낮은 품사 태깅 부분이 정해지면 수정과정을 수행한다. 품사를 선택해야할 형태소는 비터비 알고리즘 적용 단계에서 unknown 표기가 되어 있다. unknown 표기를 기준으로 앞뒤 품사들과 저장되어 있는 품사열 패턴을 매칭시켜본다. 이렇게 추출된 패턴들로부터 길이와 빈도 우선순위에 따라 패턴을 적용하여 unknown 표기 부분의 품사를 결정한다.

2.7 저신뢰도 구간 검출 및 수정 시스템



<그림 2> 저신뢰도 검출 결과를 수정하는 HMM 태깅 시스템

시스템의 구성도는 그림 2와 같다. 시스템은 추출모듈, 재추정 모듈, 태깅 모듈로 이루어져 있다. 추출 모듈은 품사 부착된 말뭉치로부터 품사 전이 빈도, 품사별 어휘 발생 빈도를 추출한 후 추출된 발생 빈도를 바탕으로 HMM의 파라미터인 품사 전이 확률, 품사별 어휘 발생 확률, 품사별 발생 확률을 생성한다. 추출모델이 생성한 HMM 파라미터는 데이터베이스에 저장한다. 재추정 모듈은 Baum-Welch 알고리즘[8]을 사용하여 데이터베이스에 저장된 HMM 파라미터가 태깅 시에 최대 확률 값을 구할 수 있도록 파라미터의 값들을 조절한다. 태깅 모듈은 재추정 과정까지 거친 HMM의 파라미터 값들을 이용하여 입력된 문장의 각 형태소를 태깅 한다. 입력 문장에 대한 단일 최적 품사열을 찾기 위해 비터비 알고리즘을 사용한다. 저신뢰도 구간 검출 모듈은 비터비 알고리즘 수행 시에 실행된다. 비터비 알고리즘 수행과정을 마친 후 규칙적용 태깅 모듈은 검출 모듈에 의해 검출된 태깅 구간을 규칙을 통해 수정한다.

3.실험 및 평가

3.1 실험 환경

KAIST에서 구축한 '대한민국 국어정보 베이스[9]'의 품사 부착된 말뭉치를 사용하여 HMM 파라미터를 학습하였다. 실험에는 한국 십진 분류 체계(KDC)의 각 분야들을 균등하게 1만 1천 어절 내외로 발췌하여 사용한 약 10만 어절을 HMM의 파라미터 학습 데이터로 사용하였다. 태깅 단위는 형태소 단위이며 54개의 품사를 사용하였다. 사용한 말뭉치에는 작업자의 언어지식 부족으로 인한 태깅 오류들이 다수 포함되어 있다. 학습데이터로 사용한 10만 어절의 말뭉치 중, 표 2의 각 분류당 약 1천개의 어절을 임의로 추출하였다. 모아진 약 9천 어절 크기의 말뭉치를 분석하여 말뭉치 제작시 작업자가 범한 오류들의 유형을 정리하였다.

〈표 5〉 말뭉치에 포함되어 있는 오류 유형

유형 번호	오류 유형 설명	빈도	비율(%)
1	주격조사와 보격 조사를 혼동한 경우	9	3.0
2	동작성명사와 상태성명사를 혼동한 경우	43	14.7
3	동작성명사와 일반명사를 혼동한 경우	87	29.8
4	상태성명사와 일반명사를 혼동한 경우	124	42.5
5	접속격조사 공동격조사를 혼동한 경우	12	4.1
6	종결보조사와 종결어미를 혼동한 경우	3	1.0
7	말뭉치 입력시에 오타가 포함된 경우	2	0.7
8	동형이의어를 혼동한 경우	12	4.1
합계		292	100.0

오류검출정확도 실험에서 오류양에 따른 오류검출정확도 변화를 알아보기 위해 인위적으로 오류를 포함시키게 되는데 이때 표 1의 각 오류 유형을 각각의 발생 비율에 맞게 생성하였다.

검출 척도에는 가중치 α, β와 임계값을 정해 주어야 한다. 가중치값 α, β의 최적값을 찾기 위해 대한민국 국어정보 베이스의 임의의 1,000문장(15,921 어절)을 대상으로 α와 β 그리고 임계값를 각각 0.00001부터 10000까지 변화시켜 가며 오류 검출 성능을 실험하였다. 빈도, 엔트로피, 상관계수가 오류 검출에 미치는 영향을 처음부터 판단할 근거가 없어 수치 범위를 아주 적은 수부터 아주 큰 수까지 폭넓게 실험해 보았고 그 결과 α : β : 임계값의 비율에 의해 오류 검출성능이 좌우됨을 알 수 있었다. 또한 α와 β와 임계값 중 α의 비율이 높은 것이 높은 검출정확도를 보였다. 실험결과에 따르면 빈도와 상관계수와 임계값의 비율이 930 : 310 : 1일 때 92.6%의 최대 오류 검출 정확도를 나타내었다. 이 같은 특성과 실험결과를 바탕으로 하여 , α, β 그리고 임계값를 정하였다. 상대적인 비율에 의해 오류 검출 정확도가 정해지므로 오류 검출 정확도를 최대로 하는 가중치와 임계치의 집합은 여러 집합이 있을 수 있다. 이 가중치는 대상언어에 따라 최적값이 달라질 수 있다. 언어에 따라 품사의 종류와 특성이 다르며 품사전이확률과 어휘발생확률이 태깅에 미치는 영향력

도 다르기 때문이다. 포괄적으로 말하면 HMM의 학습데이터로 사용된 말뭉치에 따라 가중치의 최적값은 달라질 수 있다.

3.2 실험 방법

실험은 신뢰성 검출 척도의 오류 검출 성능, 그리고 검출된 태깅 결과를 수정했을 때의 시스템의 태깅 성능을 테스트 하였다. 실험은 공통적으로 '대한민국 국어정보 베이스'의 세 부분을 입력 문장으로 사용하였다. 각 그룹은 말뭉치 내에서 성경, 사설, 문학 분야의 일부분을 선택하였으며 각각의 주제와 크기는 표 2와 같다. 이는 주제와 문장표현 방식이 다른 문장, 그리고 실험 문장의 크기가 실험 결과에 영향을 미칠 수 있음을 고려한 것이다.

〈표 6〉 실험에 사용된 말뭉치 그룹 특성

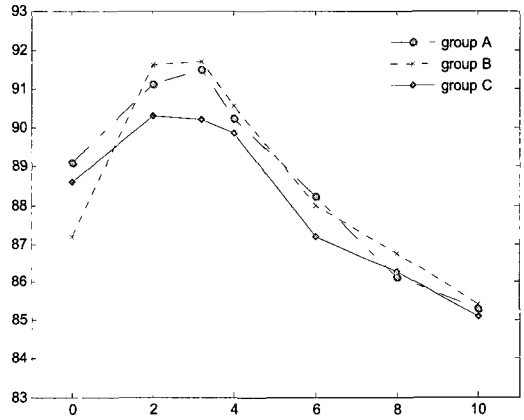
실험그룹	분야	주제	크기
A	종교	성경	8,048어절 (12,759 형태소)
B	사설	성문화	14,662어절 (28,925 형태소)
C	문학	남극기지	17,887어절 (33,495 형태소)

구현한 시스템은 미등록어를 고려하지 않았기 때문에 미등록어에 대응하지 못한다. 미등록어는 저빈도 문제의 원인으로 작용하여 시스템 성능에 미치는 영향력을 무시할 수 없다. 그러나 실험으로 확인하고자 하는 것은 저빈도 문제가 아니라 작업자의 오판으로 인해 발생된 오류를 검출하는 성능이기 때문에 미등록어는 유사한 등록어로 사전에 수정하여 실험에 영향을 미치지 않도록 했다. 태깅 결과가 오류인지 여부를 판단할 기준으로 삼을 A그룹, B그룹, C그룹의 정답 말뭉치를 수작업으로 만들었다. 각 정답 말뭉치에 오류의 비율을 2%, 4%, 6%, 8%, 10%로 인위적으로 늘려가며 오류 검출 정확도를 비교해 보았다. 오류는 표 2의 각 오류 유형을 각각의 상대적 발생 비율에 맞게 만들었다.

저신뢰 구간으로 검출된 태깅 결과를 규칙으로 수정했을 때의 시스템의 태깅 성능 평가 실험은 다음 두 시스템의 태깅 성능 비교를 통해 진행한다. 첫 번째 시스템은 일반적인 HMM을 사용한 태깅 시스템이다. 10만 어절의 학습 말뭉치로부터 HMM 파라미터를 추출하고 Baum-Welch 재추정 알고리즘을 통하여 파라미터를 재추정 한다. 이렇게 만들어진 HMM 파라미터를 사용하여 비터비 알고리즘으로 A그룹, B그룹, C그룹의 문장들을 태깅한다. 두 번째 시스템은 2.7절에서 설계한 시스템으로, 첫 번째 시스템에 본 논문에서 제안한 저신뢰도 구간 검출 및 오류 수정 방안을 적용한 시스템이다. 첫 번째와 마찬가지로 A그룹, B그룹, C그룹의 문장들을 태깅하고 두 시스템의 결과를 비교한다.

3.3 실험 결과 및 평가

그림 4는 오류비율을 0%, 2%, 4%, 6%, 8%, 10%으로 증가시켰을 때 A그룹, B그룹, C그룹의 오류검출정확도의 변화를 나타낸 것이다. 여기서 0%는 모든 오류가 제거된 말뭉치의 오류검출정확도를 말한다.



〈그림 3〉 포함된 오류의 비율에 따른 오류검출 정확도

오류가 전혀 포함되어 있지 않은 정답 집합을 입력한 경우 신뢰성 척도가 잘못 검출하는 비율이 약 12% 정도로 나타났다. 오류의 양이 극히 적을 때에는 오류의 영향력이 크지 않아 검출이 잘 되지 않기 때문이다. 오류의 비율이 약 3% 내외일 때 최고 검출 정확도를 보였다. 오류의 비율이 3%정도일 때 검출 척도가 검출하고자 하는 특성이 잘 나타나고 있음을 알 수 있다. 그러나 오류의 양이 더 많아지면 오류로 인한 잡음이 잡음의 수준을 넘어 하나의 언어현상으로 인정되기 때문에 잘못 검출 하는 비율도 계속적으로 증가하였다. 이는 제안한 검출 척도가 저빈도라고 할 수 없을 만한 양이지만 하나의 언어현상으로 생각하기에는 부족한 양의 오류. 즉, 말뭉치 제작시 작업자의 오판으로 인해 발생한 오류를 검출해 내는데 적합하다는 것을 보여준다. 또한 3%에서 가장 높은 검출 정확도를 보인

이유로 검출 척도의 가중치를 정하는 실험에 사용한 말뭉치가 3.2%의 오류를 포함하고 있었기 때문에 유사한 오류량에 좋은 검출성능을 보였다고 볼 수 있다. 이러한 사실은 검출 척도의 가중치의 조절을 통해 다른 언어나 다른 말뭉치를 사용한 경우에도 해당 환경에 검출 성능을 최대화 시킬 수 있음을 짐작할 수 있게 한다.

실험 결과를 통해 말뭉치에 포함된 작업자의 오판으로 인한 오류에 대한 해법으로 신뢰성 검출 척도가 유용하게 사용될 수 있고, 또한 검출 척도의 가중치값을 조절하여 검출 성능을 높일 수 있음을 알 수 있었다.

전체 태깅 시스템의 성능 평가는 A그룹, B그룹, C그룹의 데이터를 입력으로 하여 태깅을 수행한 결과와 수작업을 통해 만든 태깅 정답과 비교하여 각 그룹별 정확도를 구하였다. 표 3은 순수한 HMM을 이용하여 태깅한 결과와 본 논문에서 제안하는 저신뢰도 태깅 구간 검출 및 수정 방법을 적용한 결과를 나타낸 것이다. 그리고 두 시스템의 태깅 결과를 분석하여 품사열 패턴을 이용한 규칙의 오류 수정 정확율을 구하여 표 4에 나타내었다.

〈표 7〉 저신뢰도 검출 및 수정 적용 전후 정확도

실험 그룹	A	B	C
적용 전 태깅 정확도	87.2	88.0	89.4
적용 후 태깅 정확도	90.1	89.7	90.8

HMM과 비터비 알고리즘만으로 구현된 태깅 시스템의 성능은 80% 후반으로 기존의 한국어 품사 태깅 시스템에 비해 낮은 성능을 보였다. 이는 비교적 적은양의 말뭉치, 오류를 다수 포함한 말뭉치를 사용했기 때문에 어느 정도 예상되는 수치이다. 문장들을 상세히 분석해 본 결과, 비터비 알고리즘 적용 단계에서 저신뢰도 구간으로 검출된 부분 중 일부가 말뭉치의 오류로 인해 잘못 태깅이 된 부분을 포함하고 있었고 해당 구간을 수정하자 정확도가 높아지는 것을 확인할 수 있었다.

〈표 8〉 수정 규칙 성능 분석

실험 그룹	A	B	C
총 형태소 개수	12,759	28,925	33,495
검출된 태그 개수	1,554	3,484	3,604
수정 전 오류 개수	1,629	3,450	3,539
수정 후 오류 개수	1,263	2,974	3,072
수정 정확도(%)	27.9	22.6	22.7

수정 전 오류 개수와 수정 후 남아 있는 오류 개수를 비교하여 수정 규칙의 성능을 분석해 표 4와 같은 결과를 얻었다. 말뭉치로부터 자주 발생하는 품사열의 패턴만 가지고 만든 규칙은 성능이 매우 낮음을 알 수 있었다. 그러나 더 좋은 정확도의 수정 규칙을 사용한다면 시스템의 성능을 더 높일 수 있을 것이다.

4. 결론 및 향후 연구 과제

통계 기반 품사 태깅에서 말뭉치의 양과 질은 태깅 성능에 큰 영향을 미치기 때문에 말뭉치에 포함된 잡음이나 데이터 부족 문제는 필수적으로 고려해야할 부분이다. 일반적으로 잡음이나 데이터 부족문제에 대한 해결책으로 재추정 알고리즘이나 평탄화 알고리즘을 사용하지만 말뭉치 제작시 작업자의 언어 지식 부족이나 주관적 오판으로 말뭉치에 포함되는 태깅 오류들로 인한 태깅 정확도 저하는 재추정 알고리즘이나 평탄화 알고리즘으로 해결 할 수 없다. 이런 약점을 극복하기 위해 본 논문에서는 말뭉치의 오류가 포함되어 태깅 결과를 신뢰하기 어려운 구간을 검출하고 규칙을 적용하여 수정하는 방안을 제안하였다. 실험 결과 제안한 검출 방식은 92%의 오류 검출 정확도를 보였고, 검출된 태깅 구간을 수정하여 태깅 시스템의 정확도가 1~3%향상하는 것을 확인 할 수 있었다.

제안한 방안은 이상적인 말뭉치, 즉, 충분한 양의 오류가 없는 말뭉치를 가진 시스템에 적용할 시에는 오히려 성능 저하를 가져오는 원인이 될 수 있다. 그러나 존재하는 대부분의 말뭉치는 완벽하지 않은 것이 사실이고, 사용하고자 하는 말뭉치가 오류를 일정량 포함한 경우라면 유용한 성능 향상 방안이 될 수 있다. 또한 매우 높은 성능을 가진 태깅 시스템에서도 부가적인 오류 판단 근거로 사용시에는 성능 향상을 기대 할 수도 있다. 실험 결과에서도 나타났지만 본 논문에서 제안한 신뢰성 검출 척도는 최대 92%의 검출 정확도를 보여 그보다 높은 태깅 정확도를 보이는 시스템에서는 효용성이 없다. 앞으로 검출 척도를 보완하여 검출 정확도가 95% 이상 된다면 수정시의 정확도 저하를 감안하더라도 최종적인 태깅 정확도가 90% 중반정도를 기대할 수 있을 것이다.

참고문헌

[1] Julian Kupiec, "augmenting a Hidden Markov Model for Phrase-Dependent Word Tagging," Proc. of Darpa Speech and Natural Language Workshop, pp.92-98, 1989.

[2] Julian Kupiec, "Robust Part-of-Speech Tagging Using a Hidden Markov Model," Computer Speech and Language, pp.225-242, 1992.

[3] Bernard Merialdo, "Tagging Text with a Probabilistic Model," Proc. of ICASSP, pp.809-812, 1991.

[4] Ian Marshall, "Choice of Grammatical Word-Class without Global Syntactic Analysis : Tagging Words in the LOB Corpus," Computers in Humanities, Vol.17, pp.139-150, 1983.

[5] Steven J. DeRose, "Grammatical Category Disambiguation by Statistical Optimization," Computational Linguistics, Vol.14, No.1, pp.31-39, 1988.

[6] Kenneth W. Church, Robert L. Mercer, "Introduction to the Special Issue on Computational Linguistics Using Large Corpora." Computational Lingustics, Vol.19, No.1, pp. 1-24, 1993.

[7] Kjell Elemus, "Comparing a Connectionist and a Rule Based Model for Assigning Parts-of-Speech," Proc. of Int. Conf. on Acoustic, Speech and Signal Processing(ICASSP-90), pp.597-600, 1990.

[8] L. E. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains," Ann. Math. Statist., vol.41, no.1, pp.164-171, 1970.

[9] 윤준태, 최기선, 한국어 품사 부착 말뭉치에 대한 고찰, 기술보고서, KAIST, 1999