통계적 결정 그래프 학습 방법을 이용한 한국어 품사 부착 오류 수정

류원호[U] 이상주 임해창
고려대학교 컴퓨터학과
{whryu, zoo, rim}@nlp.korea.ac.kr

Korean Part-of-Speech Tagging Error Correction Method Based on Statistical Decision Graph Learning

Won-Ho Ryu[U] Sang-Zoo Lee Hae-Chang Rim
Dept. of Computer Science and Engineering, Korea University

요 약

지금까지 한국어 품사 부착을 위해 다양한 모델이 제안되었고 95% 이상의 높은 정확도를 보여주고 있다. 그러나 4-5%의 오류는 실제 응용 분야에서 많은 문제를 야기시킬 수 있다. 이러한 오류를 최소화하기 위해서는 오류를 분석하고 이를 수정할 수 있는 규칙들을 학습하여 재사용하는 방법이 효과적이다. 오류 수정 규칙을 학습하기 위한 기존의 방법들은 수동 학습 방법과 자동 학습 방법으로 나눌 수 있다. 수동 학습 방법은 많은 비용이 요구되는 단점이 있다. 자동 학습 방법의 경우 모두 변형규칙 기반 접근 방법을 사용하였는데 어휘 정보를 고려할 경우 탐색 공간과 규칙 적용 시간이 매우 크다는 단점이 있다. 따라서 본 논문에서는 초기 모델에 대한 오류 수정 규칙을 효율적으로 학습하기 위한 새로운 방법으로 결정 트리 학습 방법을 확장한 통계적 결정 그래프 학습 방법을 제안한다. 제안된 방법으로 두 가지 실험을 수행하였다. 초기 모델의 정확도가 높고 말뭉치의 크기가 작은 첫 번째 실험의 경우 초기 모델의 정확도 95.48%를 97.37%까지 향상시킬 수 있었다. 초기 모델의 정확도가 낮고 말뭉치 크기가 큰 두 번째 실험의 경우 초기 모델의 정확도 87.22%를 95.59%로 향상시켰다. 또한 실험을 통해 결정 트리 학습 방법에 비해 통계적 결정 그래프 학습 방법이 더욱 효과적임을 알 수 있었다.

1. 서론

품사 부착이란 말뭉치 내의 각 어절을 올바른 형태소 열로 분리하고 각 형태소에 올바른 품사를 부여하는 것을 말한다. 품사 부착은 구문분석의 전처리 과정으로 간주되기도 하고, 정보 검색 시스템에서 높은 재현률(recall) 및 정확도(precision)를 갖는 색인어와 검색어 추출을 위해 사용되기도 한다. 또한 기계 번역, 용례 추출, 질의 응답, 철자 검사 및 수정, 사전 구축 등 자연어처리 제반 분야에서 필수적인 과정으로 인식되고 있다.

한국어는 교착어이면서 다양한 음운 현상을 가지고 있으며 한 어절 내에서 품사 중의성뿐만 아니라 형태소들의 분할 중의성까지 발생하므로 영어에 비해 다양한 형태론적 중의성을 가지고 있다. 품사 부착을 위해서는 이러한 다양한 형태론적 중의성을 해결해야 하는데 이를 위해 대량의 품사 부착된 말뭉치로부터 통계정보를 획득하여 사용하는 방법이 주로 사용되고 있으며 마르코프 모형에 기반한 방법, 최대 엔트로피 모델에 기반한 방법, 신경망에 기반한 방법 등을 예로 들 수 있다.

이러한 말뭉치 기반 접근방법은 모델을 현실적으로 구현하기 위해 문맥 정보의 크기를 제한하여 사용하게 되고 이로 인해 모든 언어현상을 고려할 수 없어 필연적으로 품사 부착 오류가 발생하게 된다. 이러한 품사 부착 오류를 최소화하기 위해서는 모델의 오류를 분석하고 이를 올바르게 수정할 수 있는 규칙을 학습하여 재사용하는 것이 가장 효과적인 방법이라 할 수 있다.

한국어 품사 부착 오류를 수정하고자 하는 기존 연구에는 다음과 같은 것들이 있다.

[4][5]에서는 사용자가 품사 부착 오류를 수정하는 과정에서 사용하는 정보를 규칙화하여 저장해 둠으로써 동

일한 오류가 반복적으로 발생하지 않도록 하였다. 이 방법의 경우 사용자의 수작업에 의존하기 때문에 비용이 많이 들고 모든 품사 부착 에러에 대한 규칙을 획득하기 어려운 단점이 있다.

[8]에서는 Brill의 변형 규칙 기반(Transformation-Based Error-Driven Learning) 학습 방법을 사용하여 품사 부착 오류를 수정할 수 있는 규칙들을 자동으로 학습하였다. 규칙은 어절태그 단위로 오류를 수정할 수 있도록 하였고 특정 어절에만 적용될 수 있는 세부 변형 규칙을 추가로 학습하여 사용하였다.

[6][9]에서는 [8]방법과 마찬가지로 변형 규칙 기반 학습 방법을 사용하여 오류 수정 규칙을 자동으로 학습하였다. [6]에서 오류 수정 규칙은 형태소 태그 단위로 오류를 수정할 수 있고 100여 개의 규칙틀을 사용하여 약 400여 개의 규칙을 학습하였으며 1.4%의 정확도 향상(90.4%->91.8%)을 보여주고 있다.

오류 수정 규칙을 자동으로 학습하는 [6][8][9] 방법의 경우 모두 Brill의 변형 규칙 기반 접근 방법[1]을 사용하고 있다. 이 방법은 먼저 규칙틀을 정의한 후 규칙틀로 생성 가능한 모든 규칙들 가운데 평가 점수가 가장 높은 규칙을 하나 선택하여 말뭉치를 수정한다. 그리고 수정된 말뭉치에 대해 다시 평가 점수가 가장 높은 다음 규칙을 탐색하는 과정을 반복함으로써 말뭉치 전체에 대해 순차적으로 적용되어야 하는 규칙들을 추출한다. 초기 품사 태거를 효과적으로 보완하기 위해서는 반드시 어휘 정보를 참조해야 하고 광범위한 문맥을 참조할 수 있는 규칙을 학습해야 하는데 변형 규칙 기반 학습 방법의 경우 규칙 탐색 공간이 커지면 그에 따라 학습 시간이 커지는 단점이 있다. 따라서 오류 수정에 필요한 많은 수의 규칙을 효율적으로 획득하기 어려운 단점이 있다. 또한 획득된 오류 수정 규칙들은 전체 말뭉치에 대해 모든 규칙들이 순차적으로 적용되어야만 하기 때문에 오류 수정 규칙을 적용하는 과정에 많은 부담을 준다.

본 논문에서는 오류 수정 규칙을 학습하기 위한 새로운 학습 방법으로 통계적 결정 그래프(Statistical decision graph) 학습 방법을 제안한다. 제안된 방법은 결정 트리 학습 방법을 기반으로 하고 있기 때문에 어휘 정보를 속성으로 사용하여도 비교적 적은 시간 내에 많은 수의 규칙을 학습할 수 있으며 품사 정보와 함께 어휘 정보를 속성으로 사용하기 때문에 매우 높은 정확도를 갖는 규칙들을 자동으로 생성해 낼 수 있다. 또한 결정 트리 학습 알고리즘을 확장한 결정 그래프 학습 방법을 사용함으로써 결정 트리를 통해 생성된 규칙들보다 훨씬 더 많은 수의 규칙을 생성할 수 있고 이로 인해 실험 데이타에 대한 규칙 적용률과 정확도를 높일 수 있다.

본 논문의 구성은 다음과 같다. 2절에서는 통계적 결정 규칙과 결정 그래프 학습 방법에 대해 기술한다. 3절에서 통계적 결정 그래프 학습 방법을 이용하여 품사 부착 오류를 수정하는 과정에 대해 기술하고 4장과 5장에서 실험 결과와 결론 및 향후 연구 방향에 대해 기술한다.

2. 통계적 결정 그래프

2.1 통계적 결정 규칙의 정의

통계적 결정 규칙은 다음과 같이 정의된다[7].

IF [대상 속성][조건] THEN [통계적 결정]
where [조건] := $(a_1\_v_1)\wedge(a_2\_v_2)\wedge\ldots\wedge(a_n\_v_n)$
[통계적 결정] := $(t_1\_f_1)\vee(t_2\_f_2)\vee\ldots\vee(t_m\_f_m)$
a_i : 규칙이 검사할 i번째 속성
v_i : i번째 속성에 대한 값
n : 규칙이 검사해야 할 속성의 수
t_i : 대상 속성에 대한 i번째 값
f_i : i번째 대상 속성값의 빈도
m : 대상 속성이 가질 수 있는 값의 수

[그림 1] 통계적 결정 규칙의 정의

이 규칙은 [조건]이 만족되면 대상 속성을 가장 높은 빈도의 값으로 결정한다는 것을 나타낸다. 이 규칙을 품사 부착 오류 수정에 적용한다면 대상 속성은 오류 수정의 대상인 (형태소/품사)가 될 것이고 대상 속성의 값은 오류 수정 대상에 가능한 모든 수정 후보들이 될 수 있다. 조건으로는 오류 수정 대상의 좌우에 위치한 형태소, 품사 등의 문맥 정보가 될 수 있다. 오류 수정 규칙의 자세한 예는 3.4절에서 기술한다.

2.2 통계적 결정 그래프 학습 방법

본 논문에서는 앞서 정의된 통계적 결정 규칙을 학습하기 위한 알고리즘으로 통계적 결정 그래프 학습 방법을 사용한다. 통계적 결정 그래프 학습 방법은 결정 트리(Decision Tree) 학습 알고리즘인 ID3를 확장한 것으로 [그림 2]와 같다.

본 논문에서는 주어진 사례 집합을 가장 잘 분류할 수 있는 속성을 선택하기 위해 정보이득률(GainRatio)을 사용한다[3].

한 노드에서 여러 개의 속성을 동시에 선택하기 위해 다음과 같은 방법을 사용한다. 먼저 각각의 속성에 대해

정보 이득률을 계산한 후 정렬한다. 이때 가장 높은 정보 이득률을 갖는 속성을 제외한 나머지 속성들 가운데 가장 높은 정보 이득률과의 값의 비율이 λ% 이내인 모든 속성들을 선택함으로써 한 노드에서 여러 개의 속성을 동시에 선택할 수 있도록 한다.

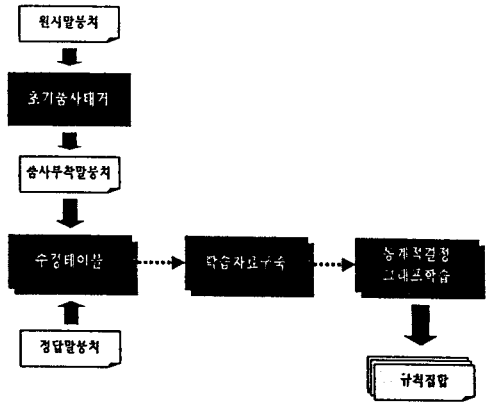
```
통계적결정그래프학습(대상속성 T, 조건집합 C, 학습자료집합 E, 속성집합 A)
{
1. 조건집합 C와 학습자료가 갖는 대상값과 빈도를 이용하여 규칙 r을 생성 (r:=[T][(ai_vj)*]->[(ti_fi)'])
2. 규칙집합 R에 r을 추가
3. 학습자료가 모두 동일한 대상값을 갖거나 A가 공집합인 경우 종료
4. 그렇지 않을 경우
 4.1 속성집합 A의 속성들 가운데 가장 높은 분별력을 갖는 속성들을 선택하여 B를 구성
 4.2 for B에 속한 각 속성 ai에 대해
      for 속성 ai가 갖는 모든 값 vj에 대해
      {
         E의 학습자료 가운데 속성 ai의 값이 vj인 학습 자료들로 E'를 구성
         A에서 ai를 제거하여 A'를 생성
         조건집합 C에 (ai_vj)를 추가하여 C'를 생성
         통계적결정그래프학습(T,C',E',A')
      }
}
```

[그림 2] 통계적 결정 그래프 학습 알고리즘

일반적인 결정 트리 학습 방법의 경우 한 노드에서 최고의 정보 이득률을 갖는 하나의 속성만을 고려함으로써 학습자료와 일치하는 가장 짧은 트리를 선택하게 된다[3]. 이 경우 선택된 트리가 새로운 데이타에 대해 항상 최고의 성능을 보인다고 확신할 수 없다[2]. 만일 한 노드에서 비슷한 분별력을 갖는 속성이 n개 존재한다고 가정하자. 이때 이들을 모두 선택하여 확장한다면 유사한 분별력을 갖는 다수의 결정 트리를 학습할 수 있다. 하나의 트리만을 생성하였을 경우, 만일 분류하고자 하는 사례가 해당 노드에서 선택한 속성에 대한 값을 가지고 있지 않는 경우 더 이상 하위 노드로 내려갈 수 없는데 비해 여러 개의 트리를 동시에 가지고 있는 경우에는 다른 속성의 값을 동시에 고려할 수 있으므로 새로운 자료에 대한 적용률을 높일 수 있게 된다. 특히 속성으로 어휘 정보를 사용할 경우에는 이러한 경우가 빈번히 발생하게 된다.

3. 통계적 결정 그래프 학습 방법을 이용한 품사 태거의 오류 수정

3.1 품사 태거의 오류 수정 규칙 자동 학습 방법



[그림 3] 오류 수정 규칙 학습 과정

[그림 3]은 초기 품사 태거의 오류를 수정하기 위한 규칙들을 자동으로 학습하는 과정을 나타낸다. 먼저 정답(품사 부착된 말뭉치)이 존재하는 원시 말뭉치를 초기 품사 태거를 이용하여 품사 부착한다. 품사 부착된 결과와 정답을 비교하여 잘못 품사 부착된 어절들로부터 오류 수정 대상들을 추출하고 이들의 집합인 수정 테이블을 구축한다. 각각의 수정 대상에 대해 학습 자료(training example)를 구축한 후 결정 그래프 학습 알고리즘을 이용하여 규칙 집합을 생성한다. 각각의 수정 대상에 대해 하나씩의 규칙집합이 생성된다. 각 단계에 대해 좀더 자세히 설명하면 다음과 같다.

3.2 오류 수정 대상 선정

초기 품사 부착 결과와 정답 말뭉치를 비교하면 다음과 같이 잘못 품사 부착된 어절들을 찾아낼 수 있다.

```
31 : 하/VX+ㄴ다/EFF+./SS. -> 하/VV+ㄴ다/EFF+./SS.
35 : 정당/NNCG+이/PS -> 정당/NNCG+이/PC
70 : 관계/NNCV+부처/NNCG+와/PN -> 관계/NNCV+부처/NNCG+와/PA
102 : 지난/NNCG -> 지나/VV+ㄴ/EFD
590 : 형사/NNCG+소송법/NNCG -> 형사소송법/NNCG
1691 : 그러기에/AC+는/PX -> 그러/VV+기/EFN+에는/PA
.....
```

[그림 4] 품사 부착 오류 어절의 예

가장 왼쪽은 품사 부착 오류가 발생한 어절의 번호이고 화살표를 기준으로 왼쪽은 초기 품사 태거의 잘못된 품사 부착 결과, 오른쪽은 올바른 품사 부착 결과이다. 이를 분석해 보면 형태소/품사를 단위로 다음과 같은 형태의 품사 부착 오류가 발생하고 있음을 알 수 있다.

- **[1:N] 오류** : 하나의 형태소/품사를 수정해야 하는 경우
  예) 하/VX -> 하/VV, 이/PS -> 이/PC,
  와/PN -> 와/PA, 지난/NNCG -> 지나/VV+ㄴ/EFD
- **[M:N] 오류** : 둘 이상의 형태소/품사를 수정해야 하는 경우
  예) 형사/NNCG+소송법/NNCG -> 형사소송법/NNCG,
  그러기에/AC+는/PX -> 그러/VV+기/EFN+에는/PA

[그림 5] 품사 부착 오류 유형

품사부착 오류를 수정하기 위해서는 가장 먼저 어떤 단위로 수정할 지를 정의해야 한다. 가능한 단위로는 품사태그 단위, 형태소/품사 단위, 어절태그 단위 등을 고려할 수 있다. 수정의 단위를 어절태그 단위로 할 경우 '와/PA'로 품사 부착되어야 할 "업무/NNCG+와/PN", "실수/NNCG+와/PN", "아내/NNCG+와/PN"와 같이 '와/PN'를 포함한 모든 품사 부착 결과들에 대해 각각을 수정할 수 있는 규칙 집합을 하나씩 생성해야 하기 때문에 매우 많은 수의 규칙이 필요하게 된다. 하나의 형태소/품사쌍만을 수정의 단위로 할 경우 어절 태그 단위에 비해 적은 수의 규칙을 학습할 수 있지만 [M:N] 오류를 수정할 수 없게 되는 단점이 있다. 따라서 본 논문에서는 오류 수정의 기본 단위를 "형태소/품사쌍"으로 하되 [M:N] 오류를 수정하기 위해 "형태소/품사+형태소/품사+.."도 하나의 단위로 정의한다.

오류 수정 대상은 [그림 4]의 품사 부착 오류 어절의 예에서 화살표의 왼쪽에 위치한 형태소/품사열 가운데 오른편의 정답과 일치하지 않는 첫 번째 형태소/품사(오류 시작)부터 일치하지 않는 마지막 형태소/품사(오류 끝)까지가 하나의 수정 대상이 된다. 이 때 수정 대상에 대응되는 화살표의 오른쪽에 위치한 형태소/품사열이 수정 대상의 값이 된다.

[그림 6]은 이러한 방법으로 선정된 수정 대상들의 집합인 수정 테이블의 일부이다.

```
31 : 하/VV -> 하/VX
35 : 이/PC -> 이/PS
61 : 주무/NNCG+부서/NNCG+이/I+ㄴ/EFD -> 주무부/NNCG+서인/NNCG
70 : 와/PA -> 와/PN
93 : 대하/VV+ㄴ/EFD -> 대한/NNP
102 : 지나/VV+ㄴ/EFD -> 지난/NNCG
120 : 지나치/VV -> 지나치/VJ
125 : 해내/VV -> 하/VV+어/EFC+내/VX
126 : 하/VX+ㄹ/EFD -> 할/NNCG
.....
```

[그림 6] 수정 테이블

수정 테이블에서 화살표의 왼편이 오류 수정의 기본 단위인 오류 수정 대상이 되고 오른편이 수정 대상의 값이 된다.

3.3 학습자료 구축

본 논문에서는 수정 테이블에 포함된 각각의 수정 대상에 대해 올바른 품사 부착 결과를 찾아내는 과정을 기계학습의 관점에서 접근한다. 각각의 수정 대상을 대상 속성으로, 수정 대상이 가질 수 있는 수정 후보들을 대상 속성의 값으로 정의할 수 있다. 예를 들어 수정 대상 '나/NPP'의 경우 수정테이블을 검사해 봄으로써 '내/NNCG', '나/VV+어/EFC', '나/NNCG', '나/NPP'(오류가 아닌 경우)의 네 가지 수정 후보를 가질 수 있음을 알 수 있다. 이 경우 수정 대상 '나/NPP'를 대상 속성으로, 가능한 네 개의 수정 후보를 대상 속성의 값으로 정의할 수 있고 주어진 문맥에서 어떤 대상 속성값을 가질 것인지로 분류하는 문제로 수정 대상에 대한 오류 수정 문제를 정의할 수 있다. 올바른 수정 후보를 선택하기 위한 속성으로 다음과 같은 정보를 사용한다.

1. 어절의 분석 결과 가운데 수정 대상의 좌우에 인접한 형태소와 품사 정보 (4개의 속성)
2. 수정 대상을 포함하고 있는 어절의 왼쪽에 위치한 두 어절의 첫 번째(어두)와 마지막(어말) 형태소, 품사 정보 (8개의 속성)
3. 수정 대상을 포함하고 있는 어절의 오른쪽에 위치한 두 어절의 첫 번째와 마지막 형태소, 품사 정보 (8개의 속성)

[그림 7] 속성의 정의

수정 대상이 사용된 모든 어절들로부터 위에서 정의된 20개 속성에 대한 값을 추출하여 학습자료를 구축한다. 만일 특정 속성의 값을 추출할 수 없는 경우, 예를 들어 문장의 첫 어절에 수정 대상이 자리한 경우 왼쪽에 위치한 어절들이 존재하지 않는데 이 경우 해당 속성은 모두 'NULL'이라는 값을 갖도록 한다. 만일 왼쪽이나 오른쪽에 위치한 어절이 하나의 '형태소/품사'만으로 구성되어 있다면 해당 어절의 마지막 형태소와 품사정보는 없는 것으로 간주하여 학습자료를 구축한다. 다음은 '가/PS'에 대해 추출된 학습자료의 예이다.

```
20770 : 그나마 AA NULL NULL 국산 NNCG NULL NULL 기계 NNCG NULL NULL 아니 VJ 라 EFC 외국 NNCG 에서 PA -> 가/PC
21146 : 정식이 NNP NULL NULL 형편 NNCG 도 PX 이해 NNCV NULL NULL 되 VV 고 EFC 오죽하 VJ 면 EFC -> 가/PC
112839 : 장사 NNCV 가 PS 되 VV ㄹ EFD 수 NNB NULL NULL 없 VJ . SS. NULL NULL NULL NULL -> 가/PS
```

[그림 8] 수정 대상 '가/PS'에 대한 학습자료의 예

가장 좌측의 번호는 어절의 번호이고 콜론의 오른쪽부터 각 속성들에 대응되는 값이다. 화살표 우측에는 대상 속성값이 표시된다.

[그림 8]의 예 가운데 첫 번째 학습 자료는 "그나마 국산 기계가 아니라 외국에서"라는 문맥으로부터 추출된 것으로서 형태소 분석 결과는 "그나마/AA 국산/NNCG 기계/NNCG+가/PS 아니/VJ+라/EFC 외국/NNCG+에서/PA"이다. 왼쪽 두 번째와 첫 번째 어절인 "그나마"와 "국산"이 하나의 형태소/품사로 이루어져 있기 때문에 마지막 형태소와 품사 정보에 대한 값이 모두 'NULL'이 되었고 '가/PS'의 오른쪽에 인접한 형태소/품사가 없기 때문에 해당 속성 역시 'NULL'값을 갖게 되었다.

이와 같이 모든 수정 대상에 대해 학습자료를 구축한 후 해당 수정 대상에 대한 대상 속성값을 가장 잘 분류할 수 있는 분류기로서 2.2절에 기술한 통계적 결정 그래프 학습 방법을 사용하여 다음과 같은 오류 수정 규칙 집합을 학습한다.

3.4 오류 수정 규칙

품사 부착 오류를 수정하기 위한 규칙들은 다음과 같이 정의된다.

```
[수정대상][(속성_값)*]
                -> [학습자료의수][(수정후보_빈도)*]
```

[그림 9] 오류 수정 규칙들의 정의

수정 대상은 오류 수정의 대상이 되는 것으로 '가/PS'의 예와 같이 하나의 형태소/품사가 될 수도 있고 '대하/VV+ㄴ/EFD'와 같이 두 개 이상의 형태소/품사가 될 수도 있다. '속성_값'은 수정 대상을 수정 후보로 변환시킬 수 있는 조건을 나타낸다. 수정 대상의 좌우에 위치한 형태소 또는 품사 정보가 속성으로서 정의되어 사용될 수 있다. 수정 후보는 수정 대상에 대한 올바른 분석 결과를 의미한다.

다음은 오류 수정 규칙의 예로서 '가/PS'에 대해 학습된 규칙들 가운데 일부이다.

```
[가/PS][] -> [3204][가/PC_206, 가/PS _2998]
[가/PS][rh1m_되] -> [143][가/PC_143]
[가/PS][rh1m_안되] -> [2][가/PC_2]
[가/PS][rh1m_아니] -> [51][가/PC_51]
[가/PS][rh1m_결코] -> [2][가/PC_1, 가/PS _1]
```

[그림 10] '가/PS'에 대한 오류 수정 규칙의 예

속성 'rh1m'은 수정 엔트리 '가/PS'가 사용된 어절의 오른쪽(r)에 자리한 첫 번째(1) 어절의 분석 결과 가운데 첫 번째(h) 형태소(m)를 의미한다. 속성 'lt2t'는 왼쪽(l)에 위치한 두 번째(2) 어절의 마지막(t) 품사(t)를 의미한다[1].

3.5 오류 수정 규칙의 적용 방법

초기 품사 태거의 결과에 포함된 오류를 수정하기 위해 먼저 [M:N] 형태의 다중 형태소/품사 오류를 수정하기 위한 수정 대상들 가운데 현재 어절의 품사 부착 결과와 일치하는 것이 있는지 검색한다. 만일 일치하는 수정 대상이 존재한다면 수정 대상의 규칙 집합에서 일치하는 규칙들을 검색하여 수정 대상을 수정한다. 만일 [M:N] 형태의 수정 대상이 발견되지 않았다면 결과에 포함된 각각의 형태소/품사에 대해 [1:N] 오류 수정을 시도한다. 형태소/품사와 일치하는 규칙이 발견되면 해당 규칙을 이용하여 형태소/품사쌍을 수정하거나 정답일 경우 그대로 유지한다.

1) 'r'은 right, 'l'은 'left', 'h'는 'head', 't'는 'tail', 'm'은 'morph', 't'는 'tag'를 의미한다.

		학습말뭉치 크기	초기태거의정확도 (오류어절수)	획득된오류수정 규칙의수	실험말뭉치 크기	초기태거의정확도 (실험말뭉치)	규칙이적용된 어절의수	실험말뭉치정확도
실험1	결정트리	150,325	95.57%(6,646)	19,924	16,790	95.48%(758)	569	97.31%(450)
	결정그래프	150,325	95.57%(6,646)	32,846	16,790	95.48%(758)	605	97.37%(441)
실험2	결정트리	450,062	87.22%(47,484)	74,052	50,009	87.22%	4921	94.54%(2,728)
	결정그래프	450,062	87.22%(47,484)	124,239(λ =0.99) 153,102(λ =0.97) 189,623(λ =0.95)	50,009	87.22%	5506(λ =0.99) 5524(λ =0.97) 5524(λ =0.95)	94.55%(2,725) 94.56%(2,719) 94.59%(2,704)

[표 1] 오류 수정 규칙을 적용한 실험 결과

결정 그래프를 이용하여 규칙 집합을 학습할 경우 하나의 수정 대상에 대해 두 개 이상의 규칙이 적용될 수 있다. 이 때 어떤 규칙을 적용할 것인지를 결정해야 하는데 다음과 같은 원칙에 따라 규칙을 선택한다.

1. 규칙의 정확도가 높은 규칙을 우선적으로 선택한다. (정확도=수정후보의빈도/학습자료의수)
2. 규칙의 정확도가 동일할 경우, 수정 후보가 가장 높은 빈도를 갖는 규칙을 선택한다.
3. 규칙의 정확도가 동일하고 수정 후보의 빈도 역시 같을 경우, 검사하는 조건의 수가 적은 규칙을 선택한다.

[그림 11] 규칙 선택 방법

4. 실험 및 평가

제안된 방법으로 두 가지 실험을 수행하였다. 첫 번째 실험은 초기 품사 태거의 정확도가 매우 높은 작은 크기의 말뭉치에 대해 오류 수정 규칙을 학습하였다. 두 번째 실험은 초기 품사 태거의 정확도가 낮은 큰 크기의 말뭉치에 대해 실험을 수행하였다.

첫 번째 실험은 형태소 태그 44개와 기호 태그 17개의 품사 집합으로 품사 부착된 167,115 어절 크기의 말뭉치에 대해 수행하였다. 이 가운데 학습 말뭉치로 150,325 어절을, 실험 말뭉치로 16,790 어절을 사용하였다. 학습 말뭉치에 대한 초기 품사 태거의 정확도는 95.57%(오류수: 6,646개)이고 품사 부착 오류 어절로부터 수정 엔트리 1,396개를 추출하였다. 결정 트리 학습 알고리즘을 이용하여 학습 말뭉치로부터 19,924개의 오류 수정 규칙을 학습하였고 결정 그래프 학습 알고리즘을 이용하여 32,846개의 오류 수정 규칙을 학습하였다. 실험 말뭉치에 대한 초기 태거의 정확도는 95.48%(오류수: 758개)이다. 결정 트리 학습 알고리즘을 통해 학습된 규칙을 적용한 결과 569개의 어절에 오류 수정 규칙이 적용되었으며 정확도는 97.31%로 향상되었다. 결정 그래프 학습 알고리즘을 통해 학습된 오류 수정 규칙을 사용한 결과 605개의 어절에 적용되었고 정확도는 97.37%로 향상되었다.

두 번째 실험은 500,071 어절 크기의 말뭉치에 대해 실험을 수행하였다. 이 가운데 450,062 어절을 학습 말뭉치로, 50,009 어절을 실험 말뭉치로 사용하였다. 학습 말뭉치에 대해 초기 태거는 87.22%의 정확도를 나타내었다. 결정 트리 학습 알고리즘을 이용하여 74,052 개의 오류 수정 규칙을 학습하였다. 결정 그래프 학습 알고리즘을 이용한 경우에는 λ값을 변화시켜 가면서 규칙을 학습하였다. λ값이 99%, 97%, 95%인 경우, 각각 124,239, 153,102, 189,623개의 규칙이 학습되었다. 실험 말뭉치에 대한 초기 태거의 정확도는 87.22%이고 결정 트리 알고리즘을 통해 학습된 규칙을 적용한 경우 94.54%로 정확도가 향상되었다. 결정 그래프의 경우 λ값에 따라 94.55%, 94.55%, 94.59%로 정확도가 향상되었다.

제안된 방법으로 두 가지 실험을 수행한 결과, 제안된 방법이 두 가지 경우 모두에 효과적임을 알 수 있었고 특히 초기 모델의 정확도가 낮고 말뭉치의 크기가 클 경우에 더욱 효과적임을 알 수 있었다. 또한 실험을 통해 결정 트리 학습 방법에 비해 통계적 결정 그래프 학습 방법이 더욱 효과적임을 알 수 있었다.

5. 결론 및 향후연구

본 논문에서는 품사 부착 오류를 최소화하기 위한 새로운 오류 수정 규칙 학습 방법을 제안하였다. 제안된 방법은 기존의 변형 규칙 기반 접근 방법에 비해 규칙의 탐색 공간이 작고 학습 속도 및 적용 속도가 빠르다는 장점을 가진다. 실험 결과 제안된 방법이 초기 모델의 정확도를 효과적으로 향상시킬 수 있음을 알 수 있었다. 그리고 결정 트리 학습 알고리즘에 비해 결정 그래프 학습 알고리즘이 보다 많은 수의 규칙을 획득함으로써 적

용률과 정확도를 보다 향상시킬 수 있음을 알 수 있었다.

현재 제안된 방법에서는 불필요한 규칙들을 줄일 수 있는 가지치기(prunning) 알고리즘이 사용되지 않았는데 추후 가지치기 알고리즘을 도입하여 실험을 수행할 계획이다. 그리고 제안된 방법은 품사 부착 문제뿐만 아니라 구묶음(chunking), 구문분석, 의미 중의성 해결 등의 다양한 분야에 적용이 가능하다. 따라서 이러한 분야에도 적용하여 실험을 수행할 계획이다.

참고문헌

[1] Brill, E. "Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging", Computational Linguistics, vol. 21, no. 4, pp. 543-564. 1995.

[2] Murphy, P. M. and Pazzani, M. J. "Exploring the Decision Forest: An Empirical Investigation of Occam's Razor in Decision Tree Induction", Journal of Artificial Intelligence Research, vol. 1, pp. 257-275. 1994.

[3] Quinlan, J. R. C4.5: Programs for Machine Learning, Morgan Kaufman, 1993.

[4] 류원호, 어휘규칙과 통계모델을 이용한 한국어 품사 부착말뭉치 구축도구, 석사학위논문, 고려대학교, 1998.

[5] 박영찬, 김남일, 허욱, 남기춘, 최기선, "품사태그부착 코퍼스 구축을 위한 한국어 품사태깅 워크벤치", 제9회 한글 및 한국어 정보처리 학술대회 발표논문집, pp. 94-101, 1997.

[6] 신상현, 이근배, 이종혁, "통계와 규칙에 기반한 2단계 한국어 품사 태깅 시스템", 한국정보과학회 논문지(B), 제24권, 제2호, pp. 160-169, 1997.

[7] 이상주, 자동 품사 부착을 위한 새로운 통계적 모형, 박사학위논문, 고려학교, 1999.

[8] 임희석, 김진동, 임해창, "한국어 특성을 고려한 변형 규칙 기반 품사 태깅", 춘계 인공지능연구회 학술대회 발표논문집, pp. 36-57, 1996.

[9] 차정원, 이원일, 이근배, 이종혁, "일반화된 미등록어 처리와 오류 수정규칙을 이용한 혼합형 품사태깅", 제9회 한글 및 한국어 정보처리 학술대회 발표논문집, pp. 88-93, 1997.