신경망을 이용한 구문패턴의 자동 학습

임희석*, 한군희**
*한신대학교 컴퓨터정보소프트웨어학부
e-mail:spknn@korea.ac.kr
**백석대학교 정보통신학부
e-mail:hankh@cheonan.ac.kr

A Automatic Learning of Syntactic Patterns by using Artificial Neural Network

Heui Seok Lim*, Kun-Hee Han**
*Div. of Computer, Information & Software, Hanshin Uiversity
**Div. of Information and Communication, Baeseok University

요 약

구문분석 말뭉치를 구축하는 작업은 문법 전문가의 많은 시간과 노력을 필요로 하기 때문에, 문법 전문가의 수작업을 감소시켜 줄 수 있는 방법이 연구되고 있다. 기존 방법 중 하나로 구문패턴을 사용하는 방법이 있는데, 이 방법은 두 개의 구문패턴이 완벽하게 일치하는 경우에만 구문패턴을 적용하는 방법이다. 본 논문은 신경망을 사용하여 구문패턴을 학습하고, 다시 구문분석 말뭉치를 구축하는데 학습된 신경망을 적용하는 방법을 사용한다. 소량의 말뭉치에서 실험한 결과, 본 논문에서 사용한 방법이 기존의 방법보다 12%이상의 수작업 감소율을 나타냈다.

1. 서론

말뭉치란 실세계에서 사람들이 사용하는 자연어를 기계가 읽을 수 있는 형태로 컴퓨터에 저장해 놓은 언어 정보를 말한다[1]. 이와 같은 말뭉치에는 다른 부가정보가 없이 단어나 문장만이 있는 원시 말뭉치, 어절을 분석하여 각 형태소에 품사를 부착한 형태소 분석 말뭉치, 문장을 분석하여 문장의 구조를 부착한 구문분석 말뭉치, 각 단어의 의미를 분석한 의미분석 말뭉치 등이 있다.

언어를 분석하는 작업은 언어학적인 이론이나 가설을 바탕으로 수행할 수 있지만, 이런 이론이나 가설은 언어학자의 주관적인 문법적 판단에 영향을 받을 수 있으며, 시대의 흐름에 따른 언어의 변화를 적절히 반영할 수 없다. 이들 말뭉치는 이런 주관적일 수 있는 언어가설을 검증할 수 있는 객관적인 정보를 제공한다[2]. 이런 이유로, 말뭉치는 언어학뿐만 아니라 자연어 처리의 여러 분야에서 유용하게 사용되고 있으며 이는 구문분석 분야에서도 마찬가지이다. 즉, 구문분석 말뭉치에서 어떤 단어나 품사의 발생 빈도, 문법의 사용 빈도, 단어 혹은 품사들의 상호정보 등을 추출하고 이를 활용하면 효과적으로 구문분석을 수행할 수 있다[3,4]. 이를 위해서는 구문분석 말뭉치의 구축이 선행되어야 한다.

구문분석 말뭉치를 구축하는데 있어서 중요한 점은 그 크기가 충분히 커야하고, 포함되어 있는 정보가 정확해야 한다는 것이다. 그러나, 정확한 말뭉치의 구축은 아직 컴퓨터가 자동으로 처리할 수 없는 문제이므로, 사람이 각 문장을 보고 수동으로 구문분석을 수행한다. 이처럼 사람이 수동으로 말뭉치를 구축하는 작업은 많은 시간과 인력을 필요로 하는 작업이고, 그렇기 때문에 사람의 수작업을 줄여줄 수 있는 방법이 필요하다.

기존에 연구된 방법 중, 구문패턴을 사용하여 문법전문가의 수작업을 감소시키는 방법이 있다[5]. 하지만, [5]는 구문패턴이 완벽하게 일치하는 경우에만 적용하였기 때문에, 높은 수작업 감소율을 나타내지

못하였다. 본 논문에서는 신경망을 통해서 구문패턴을 학습하고 적용하는 방법이 더 많은 양의 수작업을 감소시킬 수 있음을 보이려고 한다.

2. 기존연구

구문분석 말뭉치를 구축할 때 수작업을 감소시킬 수 있는 방법에 대해서 몇 가지 연구가 있었다([5],[6],[7]). 이 가운데 구문패턴을 사용하여 사용자의 수작업을 감소시킬 수 있음을 보여준 연구는 [5]이다.

[5]는 크게 구문패턴 후보 추출, 구문패턴 선정, 구문패턴 적용이라는 3가지 단계로 사용자의 수작업을 감소시켰다. 첫 번째 구문패턴 후부 추출 단계에서는 기존에 수동으로 구축한 구문분석 말뭉치에서 구문패턴 후보들을 추출한다. 두 번째 구문패턴 선정 단계에서는 추출한 구문패턴 후보들에 대해서 통계정보를 사용하여, 일정 수준이상의 신뢰도를 가지는 구문패턴 후보만을 구문패턴으로 선정한다. 세 번째 구문패턴 적용 단계에서는 선정된 구문패턴을 구문분석 말뭉치를 구축할 때 적용한다.

첫 번째 단계로 구문패턴 후보를 추출하기 위해서는 구문패턴을 구성하게 될 자질집합을 선택해야 한다. 좌우 구문-기능 범주와 좌우 중심어절의 품사열을 자질집합으로 사용할 경우, 그림1의 구문분석 결과에 대해서는 그림2와 같은 구문패턴 후보들을 추출한다. [5]에서 사용한 구문분석 부착도구는 기본적으로 묶기/이동(Reduce/Shift)의 LR연산을 사용하기 때문에 구문패턴의 결과는 묶기 또는 이동만을 값으로 가진다.

```
(VP (NP_MOD  철수/NNP + 에게/JKB)
    (VP  (NP-OBJ  악수/NNG + 를/JKO)
         (VP  청하 /VV + 았 /EP + 다 /EF)))
```

그림 1. 입력문장 예제

두 번째 단계로 구문패턴을 선정하기 위해서 [5]에서는 각 구문패턴 후보가 이항분포를 따른다고 가정하고, 이항분포의 값이 사용자가 명시한 신뢰도보다 높은 경우에만 구문패턴으로 선정한다. 세 번째 단계로 새로운 구문분석 말뭉치를 구축할 때, 현재 상태와 완벽하게 일치하는 구문패턴이 있는지 검사하고, 있다면 이를 적용한다.

[5]에서는 위와 같은 방법으로 구문패턴을 사용하여 수작업을 감소시켰다. [5]와 같이 구문패턴이 완벽하게 일치하는 경우에만 적용하는 방법은 언어학자가 수동으로 구문패턴을 만들어 기존의 구문패턴 집합에 추가하는 일이 가능하다는 장점을 가지고 있다. 하지만, 완벽하게 일치하는 경우에만 적용하였기 때문에, 상대적으로 낮은 수작업 감소율이 나타났다.

본 논문에서는 구문패턴에 신경망을 적용하여 더 높은 정확률, 재현율, 수작업 감소률을 얻을 수 있음을 보이고자 한다.

3. 구문패턴 학습

신경망을 사용하여 구문패턴을 학습하기 위해서는 다음과 같은 세 가지 사항을 고려해야 한다. 첫 번째는 구문패턴에 있는 태그 값들을 신경망이 학습할 수 있는 자료형으로 변환해야 한다는 점이다. 두 번째는 태그가 품사열 혹은 어휘열인 경우 n개의 태그 혹은 단어 값을 가진다는 점이다. 즉, 각 태그 혹은 단어를 각각 하나의 unit에 대응시킬지 아니면, 전체 열을 하나의 unit에 대응시킬지 등을 결정해야 한다. 세 번째는 묶기/이동으로 계산되는 결과 값을 어느 정도 신뢰할 수 있는지에 대한 정보도 알 수 있어야 한다는 점이다.

첫 번째로 품사태그, 구문태그, 어휘를 자질로 사용한 구문패턴의 경우 해당하는 값은 태그 값이기 때문에, 이를 불린 값의 벡터(boolean-valued vector)로 변환하여 사용한다. 즉, 하나의 태그 값 t는 식(1)과 같은 벡터 값을 가진다. 그림 3은 NP_MOD라는 구문-기능태그 값을 가지는 태그의 입력 예이다.

결과	좌구문-기능	좌품사열	우구문-기능	우품사열	비고
이동	NP_MOD	NNP +JKB	NP_OBJ	NNG +JKO	철수에게 악수를
묶기	NP_OBJ	NNG +JKO	VP	VV +EP +EF	악수를 청하였다

그림 2. 추출된 구문패턴 후보

$$t=\langle t_1, t_2, \ldots t_n \rangle$$
$$(\ n = \#\ of\ all\ possible\ tag.\ )$$
$$t_i = \begin{cases} 1 & \text{if } t_i = current\ tag \\ 0 & otherwise \end{cases} \quad (1)$$

0	0	0	0	0	1	.....	0	0	0	0
AP	...	...	...	...	NP_MOD	...	...	...	...	

그림 3. NP_MOD 입력 예

두 번째로 품사열, 어휘열과 같은 n개의 태그 값을 가지는 값은 하나의 태그 혹은 어휘 값에 대응시킨다. n개에 대해서 n개의 unit을 만들면, 각 값이 가지는 위치정보까지 고려할 수 있지만 추정해야 하는 가중치 값이 늘어나기 때문에, 태그열 혹은 어휘열 내에서의 위치 정보를 사용하지 않고, 하나의 unit에 그 값을 대응시킨다. 하나의 태그 열에 해당하는 입력 값은 식(2)와 같다. 그림 4는 NNP + JKB 품사열에 대한 입력 예이다.

$$t=\langle t_1, t_2, \ldots t_n \rangle$$
$$(\ n = \#\ of\ all\ possible\ tag.\ )$$
$$t_i = sequence \begin{cases} 1 & \text{if } t_i = member\ of\ current\ tag\ sequence \\ 0 & otherwise \end{cases} \quad (2)$$

0	0	1	0	0	1	.....	0	0	0	0
...	...	JKB	...	...	NNP	...	...	...	...	...

그림 4. NNP+JKB 품사열 입력 예

세 번째로 계산된 결과 값에 대한 신뢰도는 출력 unit을 두 개로 만듦으로써 알 수 있다. 첫 번째 unit은 묶기에 해당하는 값이고, 두 번째 unit은 이동에 해당하는 값을 출력한다. 그리고, 이 두 값의 차이를 결과에 대한 신뢰도로 사용한다. 즉, 두 값 사이의 차이가 작으면 작을수록 입력된 구문패턴이 묶기인지 이동인지 판단하기 힘든 것이고, 두 값의 차이가 크면 클수록 묶기인지 이동인지 판단하기 쉬운 경우라고 가정한다. 그래서, 사용자에게 임계값을 입력받고, 두 결과 값이 차이가 그 값보다 작으면 결과를 적용하지 않고, 두 값의 차이가 임계값보다 크면 결과를 적용하는 방법을 사용한다.

4. 실험 및 평가

4.1 실험 환경

사용한 말뭉치의 분석단위는 어절단위이다. 실험에 사용한 문장은 [10]에서 정의한 태그 집합과 구문구조를 사용하여 수작업으로 구축한 말뭉치로서, 소설에서 추출한 853문장과 신문에서 추출한 544문장으로 총 1,397문장이고 15,148어절로 구성되어 있다. 그리고, 전체 문장 중의 90%는 학습집합으로 사용하고 나머지 10%를 실험집합으로 사용하였다. 각각에 대한 문장 수는 표 1과 같다.

표 1. 실험집합

	학습집합	실험집합	총합
말뭉치	1,256문장	141문장	1,397문장

실험은 신경망을 이용한 구문패턴의 학습이 얼마나 효과적인지를 알아보는 목적으로 수행되었다. 이 실험을 수행하기 위해서 표 2와 같은 세 가지 자질 집합을 설정하였다.

표 2. 자질집합

자질집합	자질 조합
1	좌우 범주
2	좌우 범주+좌우 품사열
3	좌우 범주+좌우 품사열+좌우 어절수

실험에 대한 정확한 평가를 하기 위해서는 적용된 구문패턴이 올바른 것인지 틀린 것인지 확인하는 작업까지 고려하여야 하지만, 이는 수치적으로 측정하기 힘들기 때문에 고려대상에서 제외하였다. 이 실험에서 사용한 평가방법은 패턴 정확률, 패턴 재현율, 수작업 감소율이다. 패턴 정확률은 적용된 구문패턴들에 대해 정답의 비율을 나타낸다. 패턴 재현율은 정답구문구조에 대해 구문패턴의 올바른 적용 비율을 나타낸다. 수작업 감소율은 기존의 수동 구축도구의 수작업 수에 대해 구문패턴이 수작업을 감소시킨 비율을 타나낸다. 이때, 구문패턴이 틀리게 적용되면 다시 수작업으로 구축할 뿐만 아니라 취소하는 수작업도 부가적으로 필요하므로 이를 고려하여 수작업 감소율을 계산한다. 이 세 가지 평가방법에 대한 수식은 다음과 같다.

표 3. 구문패턴 학습 및 적용결과

자질집합	임계값	패턴 정확률	패턴 재현율	수작업 감소율
1	0.4	36.3%	56.2%	-42.3%
	0.45	51.8%	59.3%	4.28%
	0.5	65.7%	54.8%	26.2%
	0.53	77.9%	47.2%	33.8%
	0.56	88.4%	36.6%	31.8%
	0.6	94.1%	24.0%	22.5%
2	0.3	46.3%	66.3%	-10.5%
	0.4	61.7%	66.7%	25.4%
	0.5	75.8%	62.4%	42.5%
	0.6	82.8%	48.0%	38.0%
	0.7	91.8%	26.2%	23.9%
	0.8	55.5%	0.9%	0.1%
3	0.3	47.3%	66.6%	-7.5%
	0.4	58.5%	67.4%	19.6%
	0.5	76.1%	60.2%	41.3%
	0.6	87.0%	43.2%	36.8%
	0.7	91.1%	31.6%	28.5%
	0.8	96.7%	2.2%	2.1%

$$\text{패턴정확률} = \frac{\text{맞은적용수}}{\text{맞은적용수} + \text{틀린적용수}}$$

$$\text{패턴재현율} = \frac{\text{맞은적용수}}{\text{수동 구축도구에서의 수작업수}}$$

$$\text{수작업감소율} = \frac{\text{맞은적용수} - \text{틀린적용수}}{\text{수동 구축도구에서의 수작업수}}$$

4.2 실험 결과

실험 결과는 표 3과 같다. 실험 결과를 살펴보면 최고 42.5%까지 수작업 감소율이 나타났음을 알 수 있다. 표 4는 [5]에서 같은 자질집합에 대해서 각 자질집합 별로 가장 높은 성능을 나타낸 것이다. 표 3의 결과와 표 4의 결과를 비교하여 보면 전체적으로 성능이 향상했음을 알 수 있고, 가장 높은 성능 값을 비교했을 때는 12% 정도 향상되었음을 알 수 있다.

표 4. 기존 구문패턴 사용방법의 성능

항목 / 자질,신뢰도		패턴 정확률	패턴 재현율	수작업 감소율
1	95%	33.9%	60.0%	-56.5%
2	70%	69.8%	52.0%	29.6%
3	50%	81.7%	39.5%	30.6%

표 3의 결과에서 임계값과 각 평가방법 사이의 관계를 살펴보면 다음과 같다. 각 자질집합에서 임계값이 올라갈수록 더 신뢰할 수 있는 학습 결과가 적용되어 패턴 정확률이 향상되지만, 적용되는 경우의 수가 줄어들어 패턴 재현율이 감소함을 알 수 있다. 수작업 감소율의 경우는 패턴 정확률과 패턴 재현율 양쪽 모두에 영향을 받기 때문에, 임계값이 증가함에 따라 증가하다가 다시 감소함을 알 수 있다.

마지막으로, 구문패턴을 학습하는 것이 몇 %의 문장에 대해서 틀리게 적용되지 않고 맞게만 적용되었는지 실험하여 보았다. 실험결과는 표5와 같다. 표5에서 알 수 있듯이, 60%이상의 문장에 대해서 틀린 적용수가 없고 맞은 적용수만 가진다는 것을 알 수 있다. 그리고, 그 문장들에 대해서 55%이상의 패턴재현율을 나타냄을 알 수 있다.

결론적으로, 신경망을 사용하여 구문패턴을 학습하고 적용하는 것이 수작업 감소율을 높이는데 도움이 된다는 것을 알 수 있다.

표 5. 문장단위 분석 결과

자질 집합	임계값	정확률 (%)	재현율 (%)	패턴이 적용된 문장 %	틀린 적용수가 0인 문장 %	틀린 적용수가 0인 문장들의 재현율	패턴이 한 번도 적용되지 않은 문장 %
1	0.0	10.8	40.9	92.8	11.5	85.7	7.1
	0.1	25.2	57.0	92.8	15.0	80.1	7.1
	0.2	66.2	56.9	89.2	24.2	57.7	10.7
	0.3	70.7	54.0	89.2	32.1	55.1	10.7
	0.4	75.9	44.4	89.2	42.4	45.3	10.7
	0.45	81.0	41.1	88.8	52.7	40.4	11.1
	0.5	81.5	40.1	88.4	53.1	39.9	11.5
	0.53	81.6	40.0	88.4	53.1	39.7	11.5
	0.56	81.6	40.0	88.4	53.1	39.7	11.5
	0.6	82.4	39.3	87.3	52.3	39.0	12.6
	0.7	84.7	34.5	87.3	53.9	34.2	12.6
2	0.0	9.3	47.1	92.8	12.6	87.2	7.1
	0.1	33.5	65.1	92.8	15.0	82.5	7.1
	0.2	54.3	68.9	92.8	25.0	77.3	7.1
	0.3	68.4	66.1	92.8	32.9	73.8	7.1
	0.4	77.3	62.6	92.8	39.2	67.7	7.1
	0.5	85.7	55.4	91.6	53.9	59.0	8.3
	0.6	90.2	52.6	90.4	61.1	55.6	9.5
	0.7	93.0	46.6	88.8	64.2	48.3	11.1
	0.8	99.6	6.0	59.1	58.7	8.1	40.8
3	0.0	10.7	49.5	92.8	11.5	87.1	7.1
	0.1	33.9	66.4	92.8	15.8	80.5	7.1
	0.2	52.1	69.5	92.8	20.6	77.6	7.1
	0.3	63.5	67.5	92.8	25.0	75.3	7.1
	0.4	74.0	65.9	92.8	32.5	70.0	7.1
	0.5	80.7	62.3	90.8	38.0	67.6	9.1
	0.6	87.1	55.9	90.0	47.2	60.3	9.9
	0.7	91.4	49.3	89.6	57.5	52.5	10.3
	0.8	95.7	17.2	80.9	68.6	17.4	19.0

5. 결론

본 논문에서는 신경망을 사용한 구문패턴 학습 방법이 기존의 구문패턴을 사용한 방법보다 더 높은 수작업 감소율을 나타낼 수 있음을 보였다.

참고문헌

[1] 류원호, 이상주, 임해창, "어휘 문맥 의존 규칙과 통계 모델을 이용한 한국어 품사 부착 말뭉치 구축 도구", 정보과학회 논문지, 제25권, 1호, pp.396-398, 1998.

[2] 박소영, 곽용재, 정후중, 황영숙, 임해창, "한국어 구문분석의 효율성을 개선하기 위한 구문 제약규칙의 학습", 한국정보과학회논문지:소프

트웨어 및 응용, 제 29권, 10호, pp.755-765, 2002.

[3] Charniak, Eugene, "Tree-bank grammars". AAAI/IAAI Vol. 2, pp.1031-1036, 1996

[4] Michael Collins, "Head-Driven Statistical Models for Natural Language Parsing", PhD Dissertation, University of Pennsylvania, 1999.

[5] 임준호, 박소영, 곽용재, 임해창, 김의수, 강범모, "구문패턴을 이용한 반자동 구문분석 말뭉치 구축도구", 한글 및 한국어 정보처리 학회, pp.343-350 2002

[6] Mitchell P. Marcus, B. Santorini, and M. A. Marcinkiewicz, " Building a large annotated corpus of English : the Penn Treebank", Computational Linguistics, Vol.19, No.2, pp.313-330, 1993

[7] 장병규, 이공주, 김길창, "대량의 한국어 구문 트리 태깅 코퍼스 구축을 위한 구문 트리 태깅 워크 벤치의 설계 및 구현", 제 9회 한글 및 한국어 정보처리 학술 발표 논문집, pp.421-429, 1997.

[8] Tom M. Mitchell. “Machine Learning", McGraw-Hill, 1997.

[9] 김홍규 외, "제8장 구문 분석 방법론 및 표지의 권장 표준안 연구", 21세기 세종계획 국어 기초자료 구축 학술용역 과제 보고서, pp.377-403, 2001.