내부 및 외부 확률을 이용한 의존문법의 비통제 학습

*+ 장두성 ○ * 최기선
*한국과학기술원 첨단정보기술연구센터 전문용어언어공학연구센터
+한국전기통신공사 연구개발본부
{dschang, kschoi}@world.kaist.ac.kr

An unsupervised learning of dependency grammar Using inside-outside probability

*+ Du-seong Chang ○ * Key-Sun Choi
*KORTERM, AITRC, KAIST
+Korea Telecom R&D Group

요 약

구문태그가 부착되지 않은 코퍼스를 사용하여 문법규칙의 확률을 훈련하는 비통제 학습(unsupervised learning) 방법의 대표적인 것이 CNF(Chomsky Normal Form)의 CFG(Context Free Grammar)를 입력으로 하는 inside-outside 알고리즘이다. 본 연구에서는 의존문법을 CNF 로 변환하는 기법에 대해 논하고 의존문법을 위해 변형된 inside-outside 알고리즘을 논한다. 또한 이 알고리즘을 사용하여 실제 훈련한 결과를 보이고, 의존규칙과 구문구조 확률을 같이 사용하는 hybrid 방식 구문분석기에 적용한 결과를 보인다.

1. 개요

비통제 학습이란 훈련하고자 하는 목적이 되는 정보가 직접 부착되어 있지 않는 코퍼스를 이용하여 원하는 정보를 추출하는 방법으로, 필요한 정보를 부착하는 데 많은 노력이 필요한 경우에 사용되는 방법이다.[1]

문장을 구문분석하는 과정에서는 많은 가능한 분석 후보가 존재할 수 있으며, 이들 중 확률적으로 가장 적합한 결과나 일정 개수이하의 결과만을 선택하기 위해서는 각 구문규칙의 확률이 필요하다. 하지만, 이러한 확률을 직접 코퍼스에서 추출(통제 훈련, supervised learning)하기 위해서는 대량의 코퍼스를 직접 수동 분석하는 과정이 필요하나, 이러한 과정은 상당한 양의 노력이 필요하다.

이러한 이유로 구문규칙의 확률훈련에 비통제 학습 방법이 여러 가지 방법으로 시도되어 왔으며, 그 대표적인 방법이 inside-outside 알고리즘이다.[1][2] 이 알고리즘을 이용하면 품사가 부착되어 있는 대량의 코퍼스를 이용하여 구문규칙의 확률을 학습할 수 있다.

2 장과 3 장에서는 이 inside-outside 알고리즘과 한국어의 분석에 흔하게 쓰이는 의존문법에 대해 간략히 설명하고, 4 장에서 이 의존문법을 inside-outside 알고리즘을 이용하여 훈련하는 방법에 대해 논의한다.

5 장 및 6 장에서는 제안된 알고리즘을 이용하여 실제 훈련한 결과를 보이고, 의존규칙과 구문구조 확률을 같이 사용하는 hybrid 방식 구문분석기에 적용한 결과를 보인다.

2. Inside-outside 알고리즘

구문규칙 훈련을 위한 inside-outside 알고리즘은 품사

태그가 부착된 코퍼스와 그 코퍼스를 분석하기 위해 사용된 구문규칙들을 입력으로 구문규칙들의 확률을 확률재추정 (Expectation and Maximization) 방식으로 학습하는 방법이다.[1]

```
N = sentence length;
Compute Inside probabilities I:
For each word_i
  For each Rule A→w_i
    I(i,A,i) = 0;
For each length len, 1 to N
  For each start i, 0 to N-len
    j=i+len;
    For each k, I to j-1
      For each Rule: A→BC
        I(i,A,j) += I(i,B,k)*I(k+1,C,j)*P(A→BC);
      End
    End
  End
End

Compute Outside probabilities O :
For each Rule A→BC
  O(i,A,j) = 0;
For each length len, N-1 to 0
  For each start i, 0 to N-1
    j = i+1;
    For each k, I to j-1
      For each Rule : A→BC
        O(i,B,k) += O(i,A,j)*I(k+1,C,j)*P(A→BC);
        O(k+1,C,j) += O(i,A,j)*I(i,B,k)*P(A→BC);
      End
    End
  End
End
```

그림 1 내부 및 외부확률 계산 알고리즘

입력으로 사용되는 구문규칙들은 일반적으로 일정량의 구문분석된 코퍼스에서 자동으로 추출된 구문규칙을 사용하거나, 사람이 직접 손으로 작성한 규칙을 사용한다. 이러한 규칙은 알고리즘에서 사용되기 위해 CNF(Chomsky Normal Form)의 CFG(Context Free Grammar)로 표현되어야 한다.

CNF 는 CFG 를 생성규칙이 A→BC 혹은 A→a 의 형태로만 구성될 수 있도록 변형한 것이다. (여기에서 A,B,C 는 문법의 비단말노드 즉 구문태그이며, a 는 단말노드 즉 단어이다.)

Inside-outside 알고리즘은 단어열 $w_{i,j}$가 A 로부터 생성되었을 확률 즉 내부(inside) 확률과 단어열 $w_{i,j}$를 A 로 둘러싼 문장의 확률 즉, 외부확률(outside)을 임의의 구문규칙 확률을 이용하여 계산하고, 이들 내부 및 외부확률을 이용하여 구문규칙 확률을 다시 추정하는 방법을 사용한다.

코퍼스내 모든 단어열과 모든 구문규칙에 대하여 그림 1 의 알고리즘을 적용하여 내부 및 외부확률을 구하고, 이들 내부 및 외부 확률을 이용하여 각 구문규칙의 확률을 재추정하는 알고리즘은 그림 2 와 같다.

```
For each rule A→a
    Compute P_1(A→a);
While iteration I is not converged
    For each Rule A→a
      C(A→a) = 0;
    For each sentences
      N = sentence length;
      Compute Inside probabilities I using P_i;
      Compute Outside probabilities O using P_i;
      For each length len, 1 to N
        For each start i, 0 to N-len
          J=i+len;
          For each k, I to j-1
            For each Rule: A→BC
              C(A→BC) += I(i,B,k)*I(k+1,C,j)*O(i,A,j)
              *P_i(A→BC)/I(0,S,N);
            End
          End
        End
      End
    End
    For each Rule A→a
      P_i+1(A→a) = C(A→a)/∑_b C(A→b)
    End
End
```

그림 2 내부 및 외부확률을 이용한 규칙 재추정 알고리즘

3. 의존문법

구문구조문법(Phrase-Structure Grammar)가 각 구문태그로부터 하위 구문태그가 생성되어지는 규칙을 기술하여 문장의 구조를 분석하는 것에 비해 의존 문법(Dependency Grammar)은 문장 내 각 단어간의 의존관

계를 기술하여 문장의 구조를 분석한다. 이러한 의존문법은 각 단어와 단어간의 관계만을 제약하고 단어간의 선후 순서는 고려하지 않으므로, 구문구조문법에 비해 한국어와 같은 부분자유어순의 언어를 분석하는데 자주 사용된다. 그림 3 은 의존규칙에 의해 분석된 문장의 구조를 보이고 있다.

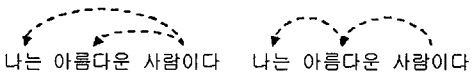

그림 3 의존문법에 의한 분석

일반적으로 의존규칙은 각 단어간의, 혹은 품사간의 의존관계를 기술하므로, (문법의 일반화를 위해서는 품사간의 관계를 주로 사용한다.) 의존규칙은 A←B 의 형태를 띤다. 여기에서 "A 는 B 에 의존한다." , 혹은 "B 는 A 를 지배한다."라고 풀이된다.

이러한 형태의 의존규칙을 inside-outside 알고리즘을 위해 CNF 로 변환할 필요가 있다. 의존규칙을 CNF 로 변환하는 것은 A←B 의 형태를 B→AB | B→BA 의 형태로 변환하는 방법을 사용할 수 있다. 그림 4 는 그림 3 의 문장을 CNF 형태로 변형된 의존규칙에 의해 다시 분석한 결과이다.

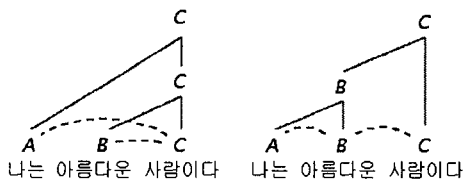

그림 4 CNF-의존문법에 의한 분석

한국어를 대상으로 하는 의존문법에서는 A←B 의 형태를 B→BA 의 형태 없이 B→AB 의 형태로 일관되게 변환할 수 있는데, 그 이유는 한국어에서 지배소는 자신의 지배하고 있는 의존소들의 뒤(문장내에서 오른쪽)에 위치한다는 지배소 후위의 원칙이 있기 때문이다.[3] 이러한 지배소 후위의 원칙은 적은 경우를 제외하고는 한국어에서 항상 적용되는 규칙이다.

4. 의존문법을 위한 inside-outside 알고리즘

CNF 로 변환된 한국어의 의존문법을 사용하여 내부 및 외부확률을 구하는 알고리즘의 일부는 그림 5 와 같다. 그림 5 는 그림 1 에서 C 를 A 로 치환한 것이며, 이는 그림 1 에서의 A→BC 형식의 규칙을 의존문법의 CNF 의 일반적인 규칙인 A→BA 에 대응하여 C 를 A 로 치환한 결과이다.

```
For each k, i to j-1
    For each Rule : A→BA
        I(i,A,j) += I(i,B,k)*I(k+1,A,j)*P(A→BA);
        O(i,B,k) += O(i,A,j)*I(k+1,A,j)*P(A→BA);
        O(k+1,A,j) += O(i,A,j)*I(i,B,k)*P(A→BA);
```

그림 5 CNF-의존문법을 위한 내부 및 외부 확률 계산 수정 부분

그림 5 는 다시 그림 6 과 같이 품사정보를 생략하여 표현할 수 있는 데, 그 이유는 모든 문장열 $w_{i,j}$ 에 대해 그 문장열을 생성하는 루트노드의 품사는 지배소 후위의 원칙에 의해 문장열의 마지막 단어 w_j 의 품사로 결정되어지기 때문이다. 그림 4 의 예제에서 보면, 분석 내용과는 상관없이 항상 각 부분 트리의 루트 노드의 품사는 2 개의 하위 노드 중 오른쪽 노드의 품사와 동일하게 분석되는 것을 볼 수 있다. 이러한 원칙에 의해 그림 6 의 알고리즘은 그림 1 과 그림 5 의 알고리즘에 비해 모든 규칙을 검토해야 하는 한단계의 계산시간을 줄일 수 있는 장점이 있다.

```
For each k, i to j-1
    I(i, j) += I(I, k)*I(k+1, j)*P(Tk←Tj);
    O(i, k) += O(i, j)*I(k+1, j)*P(Tk←Tj);
    O(k+1, j) += O(i,j)*I(i, k)*P(Tk←Tj);
```

그림 6 한국어 의존문법을 위한 내부 및 외부 확률 계산 수정

5. 훈련

4 장에서 제시된 알고리즘을 이용하여 의존문법의 비통제학습을 하기 위해 사용된 의존규칙은 그림 7 과 같은 한국어 의존구조 분석을 위해 수동 작성된 의존 문법에서 각종 하위범주화 제약 및 기타 제약을 제거하여 얻었다. 그림 7 에 표현된 문법의 의미는 보통명사와 의존명사 간의 관형어 수식 의존관계에 대해 기술한 것이며, 두 단어가 연이어 나와야 한다는 제약과, 의존명사의 하위범주가 명사관형어선행 의존명사이어야 한다는 제약을 가지고 있다.

```
Mv ← Me
    { CON_WORD , HSM= MegM } { Adn }
```

그림 7 수동작성된 의존문법의 일부

그림 7 과 같이 수동 작성된 의존문법을 훈련하기 위해서 제약규칙만 다른 같은 품사를 가지는 규칙들은 하나의 규칙으로 생각하여 같은 확률로 학습하고. 6 장에 기술할 구문분석과정에서는 두 규칙이 같은 확률을 가지도록 하는 방법을 사용하였다. 이러한 방법이 전체 분석결과에 미치는 영향은 특별히 측정되지 않았다.

수동작성된 의존규칙의 수는 209 개이며, 제약규칙을 고려하지 않은 이유로 중복되는 규칙들을 제거하고 실제 훈련에 사용된 규칙의 수는 175 개이었다.

훈련에 사용된 코퍼스는 한국통신 HANSORI 실험문장[4]으로 136 개 문장, 1632 어절로 이루어져 있다(12 어절/문장). 이 코퍼스는 음성합성기의 음질평가를 위해 작성된 것으로, 대량의 코퍼스에서 가능한 많은 구문현상을 포함하도록 추출된 코퍼스로서 적은 시간에 구문규칙을 가장 효율적으로 훈련할 수 있을 것이라는 이유로 선택되었다.

코퍼스를 훈련할 때 수동작성된 구문규칙을 이용한 이유로, 작성된 구문규칙으로 분석실패하는 문장이 발생하였다. 이러한 경우 훈련된 결과의 정확도를 보장할 수 없으므로, 훈련시 이러한 문장은 제거되었다. 또한 코퍼스의 품사태깅 오류도 동일한 문제를 야기하기 때문에 훈련 코퍼스 내 품사태그부착의 오류도 수정되었다.

6. Hybrid 구문분석기

Inside-outside 알고리즘으로 훈련된 의존규칙의 확률을 기존의 제약조건을 포함하는 의존규칙과 같이 사용하기 위하여 Hybrid 방식의 구문분석기를 구현하였다. 이 구문분석 방식에서는 문장을 오른쪽에서 왼쪽으로 분석해 나가면서 각 단어와 그 단어에 오른쪽에 인접한 단어의 지배가능 경로에 있는 단어들간의 의존규칙을 검사하는 방식을 사용하였다.[3] 또한 중간분석 결과들 중 가장 확률이 높은 중간분석결과부터 확장해 나가는 방식으로 전체 문장에 대해 하나의 최적분석결과를 추출하는 방법을 사용하였다. 전체적인 분석 알고리즘은 그림 8 과 같다.

```
N = sentence length;
Insert Tree(N,N) to Tree list;
While not parsed
    Select Tree(i,N) which has the highest probability;
    For each Words Wj on the headable path of Wi
        For each rule
            If dependecy (Wi-1, Wj) exist then
                add Tree(I-1,N) to Tree list.;
        End
    End
    Remove Tree(I,N);
End
```

그림 8 구문분석알고리즘

구문 트리의 확률은 수식 1 과 같이 트리를 구성하고 있는 의존 규칙들의 확률의 곱을 사용하였으며, 실제 분석 알고리즘에서는 계산상의 편의를 위해 확률의 로그합을 이용하였다.

$$P(T) = \prod_{T_i \leftarrow T_j \in T} P(T_i \leftarrow T_j) \quad \text{(수식 1)}$$

표 1 은 확률추출 코퍼스에 포함되지 않은 20 문장(169 어절)에 대해 Hybrid 구문분석기의 정확도를 평가한 결과이다. 표에서 규칙기반 구문분석기는 각 단어에서 지배가능 경로에 있는 단어 중 의존규칙에 의해

의존관계가 성립할 수 있는 가장 가까운 단어를 유일한 지배소로 선택하는 규칙을 사용하여 구문분석의 애매성을 제거하는 분석방법으로, 애매성 해소를 위한 몇 개의 추가 규칙이 포함되어 있다.

구문분석방식	정확한 의존관계수	단어 정확도	단어 오류율
규칙기반 구문분석	160	94.6%	5.4%
Hybrid 구문분석	162	95.8%	4.2%

표 1 Hybrid 구문분석기의 성능비교

구문분석기의 정확도는 문장내 각 의존소와 지배소, 의존관계의 3 개항으로 이루어지는 3-tuple 들 중 모든 항목을 정확하게 찾은 비율로 구했으며, 규칙과 확률을 같이 사용한 Hybrid 방식의 정확도가 규칙기반 방식보다 22%의 오류율의 감소를 보였다. 반면, 문장이 길어질수록 분석시간이 규칙기반에 비해 큰 비율로 많이 필요하였다. Hybrid 구문분석기는 반면에 규칙기반 구문분석 방식에 비해 그 분석시간이 많이 소요되었으며, 문장 내 단어의 수가 많아짐에 따라 그 분석시간이 현저히 늘어남을 볼 수 있었다. 이 실험에 사용된 구문분석기는 아래 URL 에서 사용할 수 있다.

http://lplab.kaist.ac.kr/~dschang/hansori/plp.html

7. 결론

한국어의 특성으로 인해 의존문법을 훈련하기 위해 inside-outside 알고리즘의 계산량은 일부 줄어들 수 있으며, 한국어 의존문법에 쉽게 적용가능함을 보였다. 또한 여러가지 제약규칙을 포함하고 있는 의존규칙을 의존규칙의 확률과 같이 사용함으로서 규칙만을 사용한 구문분석기보다 정확도를 22%의 오류율을 감소시킨 결과를 보였다.

보다 정확한 구문분석기를 위해서는 좀더 많은 양의 코퍼스를 이용하여 훈련할 필요가 있으며, 구문분석기의 계산시간을 줄일 수 있는 기법을 좀더 도입할 필요가 있다.

감사의 글

본 연구의 일부는 전문용어언어공학연구센터에서 수행한 과학기술부와 KISTEP 의 핵심소프트웨어사업 중 "대용량 국어정보 심층처리 및 품질관리 기술개발" 과제의 일환으로 수행되었으며, 첨단정보기술연구센터를 통하여 과학재단의 지원도 받았습니다.

References

[1] C.D.Manning, H.Schütze, Foundations of Statistical Natural Language Processing, pages 232, 398, MIT Press, 1999

[2] J.T.Goodman, Parsing Inside-Out, Ph.D. thesis, Harvard University, 1998

[3] C.H.Kim, et.al. "A Right-to-Left Chart Parser for Dependency Grammar using Headable Paths," Proceeding of the 1994 International Conference on Computer Processing of Oriental Language, pages 175~179, 1994

[4] D.S.Chang, et.al. "HanSoRi97: The Unlimited Korean Text-To-Speech System," Proceeding of 4th Conference on Natural Language Processing Pacific-Rim Symposium, 1997