최대 엔트로피 모델을 이용한 텍스트 단위화 학습

박성배° 장병탁
서울대학교 컴퓨터공학부

Learning Text Chunking Using Maximum Entropy Models

Seong-Bae Park° and Byoung-Tak Zhang
School of Computer Science and Engineering
Seoul National University

{sbpark,btzhang}@bi.snu.ac.kr

요 약

최대 엔트로피 모델(maximum entropy model)은 여러 가지 자연언어 문제를 학습하는데 성공적으로 적용되어 왔지만, 두 가지의 주요한 문제점을 가지고 있다. 그 첫번째 문제는 해당 언어에 대한 많은 사전 지식(prior knowledge)이 필요하다는 것이고, 두번째 문제는 계산량이 너무 많다는 것이다. 본 논문에서는 텍스트 단위화(text chunking)에 최대 엔트로피 모델을 적용하는 데 나타나는 이 문제점들을 해소하기 위해 새로운 방법을 제시한다. 사전 지식으로, 간단한 언어 모델로부터 쉽게 생성된 결정트리(decision tree)에서 자동적으로 만들어진 규칙을 사용한다. 따라서, 제시된 방법에서의 최대 엔트로피 모델은 결정트리를 보강하는 방법으로 간주될 수 있다. 계산론적 복잡도를 줄이기 위해서, 최대 엔트로피 모델을 학습할 때 일종의 능동 학습(active learning) 방법을 사용한다. 전체 학습 데이터가 아닌 일부분만을 사용함으로써 계산 비용은 크게 줄어들 수 있다. 실험 결과, 제시된 방법으로 결정트리의 오류의 수가 반으로 줄었다. 대부분의 자연언어 데이터가 매우 불균형을 이루므로, 학습된 모델을 부스팅(boosting)으로 강화할 수 있다. 부스팅을 한 후, 제시된 방법은 전문가에 의해 선택된 자질로 학습된 최대 엔트로피 모델보다 좋은 성능을 보이며 지금까지 보고된 기계 학습 알고리즘 중 가장 성능이 좋은 방법과 비슷한 성능을 보인다. 텍스트 단위화가 일반적으로 전체 구문분석의 전 단계이고 이 단계에서의 오류가 다음 단계에서 복구될 수 없으므로 이 성능은 텍스트 단위화에서 매우 의미가 깊다.

1. 서 론

텍스트 단위화(text chunking)는 비교적인 피상적인 분석을 바탕으로 문장을 문법적으로 서로 관련된 단어 집합으로 나누는 것을 말한다. Abney 가 서로 관련 있는 단어들의 집합을 사용한 구문분석의 가능성을 처음으로 제시한 후[1], 단위화는 자연언어 학습 분야의 주요 관심 분야가 되어왔다. 단위화를 함으로써, n 개의 단어로 이루어진 문장은 일반적으로 n 보다 훨씬 적은 m 개의 구문적으로 관련된 부분으로 나누어질 수 있다. 대부분의 자연언어 구문 분석기의 복잡도가 $O(n^3)$이기 때문에, 단위화는 구문 분석의 성능을 크게 높일 수 있다.

Ramshaw 와 Marcus 가 텍스트 단위화에 기계학습 기법을 처음으로 적용한 이래[14], 많은 연구자들이 텍스트 단위화의 최적의 가설을 찾는데 통계 기반 방법이나 기계학습 기법을 적용해 왔다[2,5,7,10]. 이중 최대 엔트로피(maximum entropy) 모델도 다른 여러 가지 자연언어 문제에서 좋은 성능을 보인 것처럼[3] 단위화에도 성공적으로 적용되었다. Koeling 은 지역적인 어휘 정보를 사용한 최대 엔트로피 모델이 단위화에서 높은 성능을 보임을 보였다[9]. 그러나, 자연언어 문제에 적용된 최대 엔트로피 모델은 두 가지의 주요한 문제를 포함하고 있다. 그 첫번째 문제는 이 모델이 대상 문제에 대한 많은 *사전 지식*(prior knowledge)을 요구한다는 것이다. 최대 엔트로피 모델에서는 이 사전 지식이 *자질*(feature)을 구축하는데 사용된다. 두 번째 문제는 이 모델의 계산량이 너무 많다는 것이다. 대부분의 자연언어 학습 문제에서는 수백만의 학습 예제를 사용하므로, 이 문제를 해결하지 못하면 최대 엔트로피 모델을 학습하지 못할 수 있다.

본 논문에서는, 사전 지식을 많이 필요로 하지 않으면서 텍스트 단위화에 최대 엔트로피 모델을 적용하는 방법을 제시한다. 위에서 언급한 두 가지 문제를 해결하기 위해서,

- 최대 엔트로피 모델의 자질을 결정트리로부터 자동으로 구축한다. 결정트리로부터 생성된 if-then 규칙 집합은 같은 중요도를 갖는 자질 집합

으로 생각될 수 있다. 그러므로, 최대 엔트로피 모델을 사용해서, 이 if-then 규칙을 강화할 수 있다. 결정트리는 주위 단어들의 품사, 구절 표지 등의 지역 어휘 정보를 사용해서 쉽게 학습될 수 있다.

- 최대 엔트로피 모델을 학습하는데 드는 계산 복잡도를 줄이기 위해, 능동 *학습*(active learning) 기법을 사용한다. 주어진 전체 학습 예제를 다 사용하기 보다는 지능적으로 중요한 일부분의 학습 예제를 선택한다. 따라서, 학습에 필요한 예제의 수를 크게 줄일 수 있다.

본 논문의 나머지 부분은 다음과 같이 구성되어 진다. 2 장에서는 텍스트 단위화를 위한 최대 엔트로피 모델에 대한 간단한 소개를 하고, 3 장에서는 최대 엔트로피 모델을 위한 자질이 결정트리로부터 자동으로 구축되는 방법을 설명한다. 4 장에서는 최대 엔트로피 모델을 위한 능동 학습을 기술하고, 5 장에서는 불균형 분포를 갖는 데이터를 처리하기 위한 방법을 제시한다. 6 장에서는 실험을 위한 데이터 집합과 실험 결과를 보인다. 최종적으로 7 장에서 결론을 맺는다.

2. 텍스트 단위화를 위한 최대 엔트로피 모델

최대 엔트로피 모델은 다양한 종류의 자연언어 학습 문제에 성공적으로 적용되어 왔다[2]. 이 모델은 한 해(solution)가 다른 것에 우선한다는 증거가 없으면 모든 다른 해는 같은 가능성을 가져야 한다는 직관을 구현한 지수 모델이다. 즉, 최대 엔트로피 모델은 다음과 같은 형태를 지닌다.

$$p(x,y)=\frac{\prod_i \mu_i^{f_i(x,y)}}{Z}$$

여기서, μ_i는

$$\mu_i=\exp(\lambda_i)$$

이고, λ_i는 실수 파라미터이다. 또한, Z 는 정규화 상수이고 $f(x, y)$들은 한 예제 (x, y)의 자질들이다.

각 자질 $f_i(x, y)$에 대해서, 학습 집합 $S = \{(x_1, y_1), \ldots, (x_N, y_N)\}$에 대한 분포 p 하의 기대값은 특정 값 K_i 로 규정된다.

$$E_p[f_i]=\sum_{j=1}^{N} p(x_j,y_j)f_i(x_j,y_j)=K_i$$

K_i의 값은 또한 S 의 경험적 분포 $\tilde{p}$ 하의 기대값으로도 설정된다. 따라서, 제약은

$$E_{\tilde{p}}[f_i]=\sum_{j=1}^{N} \tilde{p}(x_j,y_j)f_i(x_j,y_j)=E_p[f_i]$$

이 된다.

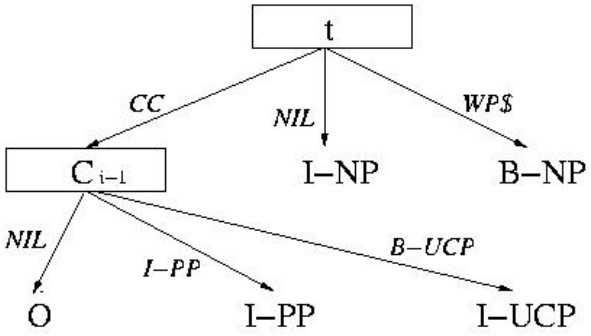


그림 1. 결정 트리의 예.

제약이 일관적이라면, $p(x, y)$를 만족하는 유일한 해 μ_i가 존재한다. 이 해는 Generalized Iterative Scaling (GIS) 알고리즘[6]과 같은 반복적 절차에 의해 구해질 수 있다. 이 알고리즘은 임의의 μ_i 로 시작한다. 각 반복에서, 현재 $\tilde{p}$ 하의 f_i 의 기대값을 K_i 와 비교한 후, μ_i 를 다음식에 의해 변경한다.

$$\mu_i=\mu_i\cdot\frac{K_i}{E_{\tilde{p}}[f_i]}$$

그 다음, 경험적 분포 $\tilde{p}$ 는 새로운 μ_i 를 가지고 변경된다.

$$\tilde{p}(x,y)=\frac{\prod_i \mu_i^{f_i(x,y)}}{Z}$$

이런 최대 엔트로피 모델에는, 두 가지 문제가 있다. 첫번째 문제는 자질 f_i 를 어떻게 선택할 것인가 하는 문제이다. 일반적으로, 최대 엔트로피 모델에서의 자질은 주어진 데이터의 특성을 잘 표현하도록 전문가에 의해서 결정된다. 따라서, 문제에 대한 충분한 지식이 없을 때에는 f_i 를 만들기가 매우 어렵다. 다른 문제는 최대 엔트로피 모델에 대한 학습은 모든 f_i 에 대한 $E_{\tilde{p}}[f_i]$를 추정해야 한다는 것이다. $E_{\tilde{p}}[f_i]$이 전체 학습 예제에 대한 덧셈을 요구하기 때문에, 학습 데이터의 수가 매우 많을 때에는 $E_{\tilde{p}}[f_i]$를 추정하는 것이 힘들다.

이 두 문제를 해결하기 위해서, 결정트리와 능동 학습을 이용한다. 결정트리는 if-then 규칙의 집합으로 표현될 수 있기 때문에, 자질은 결정트리를 if-then 규칙으로 변경함으로써 구축할 수 있다. 결정트리는 n-gram 과 같은 간단한 언어 모델로부터 학습될 수 있다. 계산 복잡도 문제를 위해서, 학습 예제를 지능적으로 선택하는 능동 학습 기법을 사용한다. 따라서, 훨씬 적은 수의 데이터로 최대 엔트로피 모델을 학습하는 데 필요한 정보를 제공할 수 있다.

3. 최대 엔트로피 모델을 위한 자질의 자동 구성

위에서 언급한 바와 같이, 결정트리는 if-then 규칙의 집합으로 쉽게 재표현될 수 있기 때문에 최대 엔트로피 모델의 자질을 구성하는데 사용될 수 있다. 예를 들어, 그림 1 과 같은 결정트리로부터 다음과 같은 5 개의 규칙을 구성할 수 있다.

1. $f_1(t, C_{i-1}, C) = \begin{cases} 1 & \text{if } t = WP\$ \text{ and } C = B - NP \\ 0 & \text{otherwise} \end{cases}$
2. $f_2(t, C_{i-1}, C) = \begin{cases} 1 & \text{if } t = NIL \text{ and } C = I - NP \\ 0 & \text{otherwise} \end{cases}$
3. $f_3(t, C_{i-1}, C) = \begin{cases} 1 & \text{if } C_{i-1} = B - UCP \text{ and } t = CC \text{ and } C = I - UCP \\ 0 & \text{otherwise} \end{cases}$
4. $f_4(t, C_{i-1}, C) = \begin{cases} 1 & \text{if } C_{i-1} = I - PP \text{ and } t = CC \text{ and } C = I - PP \\ 0 & \text{otherwise} \end{cases}$
5. $f_5(t, C_{i-1}, C) = \begin{cases} 1 & \text{if } C_{i-1} = NIL \text{ and } t = CC \text{ and } C = O \\ 0 & \text{otherwise} \end{cases}$

그러므로, 일단 결정트리를 학습하고 나면 이 결정트리로부터 최대 엔트로피의 자질을 자동으로 구축할 수 있다.

결정트리의 관점에서 보면, 자질은 학습 집합을 동등한 하위 집합으로 나누는 것이다. 만약 자질들이 전체 학습 예제를 완벽하게 구분할 수 있다면, 나누어진 하위 집합은 최소한 학습 예제에 대해서는 최적이다. 이런 경우에는, 더 이상 자질을 변경할 필요가 없다. 자질이 학습예제를 완벽하게 구분하지 못할 때, 잘못 구분되는 학습 예제는 하위 집합이 중복을 가지지 않기 때문에 생기는 것이다. 따라서, 하위 집합들 사이의 중복을 허락함으로써 자질을 향상시킬 수 있다.

최대 엔트로피 원리에 의해 찾아진 최고의 모델 p^*는

$$p^* = \arg\max_{p \in C} H(p)$$

이다. 여기서, $H(p)$는 분포 p 의 엔트로피이고 C 는 모든 i 에 대해서 제약 $E_p[f_i] = E_{\tilde{p}}[f_i]$ 을 만족하는 분포들의 집합이다. 만약 각 예제 (x_j, y_j)가 독립이라고 가정하면, 경험적 분포 $\tilde{p}$ 의 조건부 로그 가능도(log-likelihood)는 다음과 같이 정의된다.

$$L_{\tilde{p}}(p) = \sum_{(x,y)} \tilde{p}(x,y) \log p(y \mid x) = \log \prod_{(x,y)} p(y \mid x)^{\tilde{p}(x,y)}$$

분포 p 에 대해서 지수 모델만 고려하므로,

$$p^* = \arg\max_{p \in C} H(p) = \arg\max_{p \in Q} L_{\tilde{p}}(p)$$

여기서, $Q = \{p \mid p(x) = \pi \prod_i \alpha_i^{f_i(x)},\ 0 < \alpha < \infty\}$ 이다.

즉, 선형 제약의 최대 엔트로피 문제는 지수 모델의 최대 가능도 문제와 같은 문제이다. 예제의 수가 늘어날수록, 최대 가능도 추정값은 비편향 최소 분산(variance) 추정값이 된다. 따라서, 주어진 자질과 데이터로 최대 엔트로피 모델을 학습하는 것은 자질들의 최적의 중요도를 찾는 것과 같다.

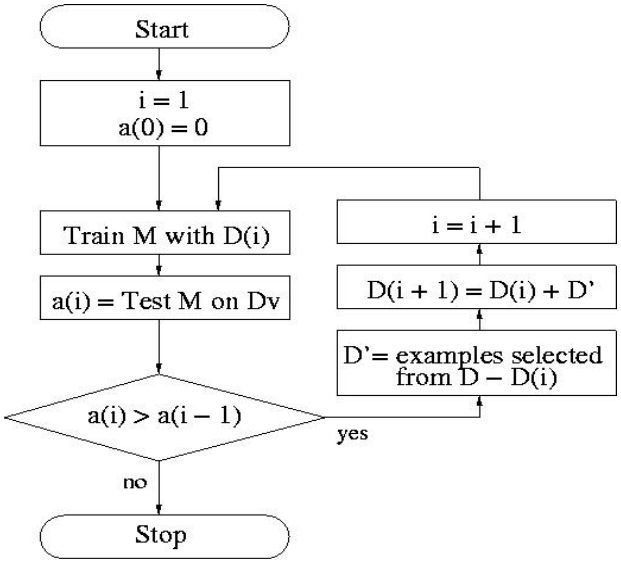


그림 2. 능동 학습 과정. M 은 학습 모델이고, Dv 는 검증 집합이고, D 는 전체 학습 예제이다.

최대 엔트로피 모델의 관점에서 보면, 자질은 결정트리로부터 자동적으로 생성된 자질은 같은 중요도 μ_i 를 갖는 것으로 해석될 수 있다. 이 자질들이 중복이 없이 학습 예제의 공간을 나누려 하기 때문에, 각 자질은 같은 중요도를 갖는 것으로 생각될 수 있다. 그러므로, 결정트리로 잘못 결정되는 학습 예제들은 정확하게 구분하려면, μ_i 의 값을 최적으로 변경하여야 한다. 텍스트 단위화 문제를 최대 엔트로피 문제로 표현하기 때문에, μ_i 의 최적값은 GIS 알고리즘에 의해 찾아질 수 있다.

또한, 결정트리는 최대 엔트로피 모델을 위한 1 차 자질(first-order feature)에서 고차원 자질(high-order feature)을 생성하는 모델로 생각할 수 있다. 예를 들어, 그림 1 에는 세 개의 1 차 자질 t, C_{i-1}, C_i가 주어져 있다. 각 가질이 각각 100 개의 값을 가질 수 있다고 가정하면, 고차원 자질의 공간은 10^6 개의 원소로 이루어진다. 이 그림에서 결정트리를 학습함으로써 단지 5 개의 가능한 규칙만 만들어졌다. 따라서, 공간의 크기가 크게 줄었다.

목적 언어와 텍스트 단위화에 대한 충분한 언어학적 지식이 없을 때에는, 결정트리를 학습시키기 위해 n-gram 과 같은 간단한 언어 모델을 사용할 수 있다. 본 논문에서는 Quinlan 의 C4.5 release 8[13]을 결정트리로 사용하였다.

4. 학습데이터를 줄이기 위한 능동학습

GIS 알고리즘에서 $E_{\tilde{p}}[f_i]$의 복잡도는 M 을 자질의 수, N 을 학습 예제의 수라고 할 때 O($M \cdot N$)이므로, N 이나 M 이 증가함에 따라 $E_{\tilde{p}}[f_i]$를 계산하는 것이 불가능해질 수 있다. M 을 줄이기 위한 여러 연구가 있었지만

[3,12], N 을 줄이기 위한 연구는 많지 않았다. 본 논문에서 설정한 문제 설정에서는 자질이 학습예제 공간을 나누도록 생성되기 때문에 M 을 줄일 수 없다. 따라서, N 을 줄이기 위해 $E_{\tilde{p}}[f_i]$를 추정하는데 능동 학습 기법을 사용한다. 즉, $E_{\tilde{p}}[f_i]$를 경험적 기대값

$$E_{\tilde{p}}[f_i] \approx \frac{1}{n}\sum_{j=1}^{n} f_i(x_j, y_j)$$

으로 추정한다. 여기서 (x_1, y_1), …, (x_n, y_n)은 알려지지 않은 분포 p 로부터의 임의표본(random sample)이다. 이런 가정하에서, $E_{\tilde{p}}[f_i]$의 계산 복잡도를 줄이는 방법은 크게 두 가지로 나누어진다. 첫번째 방법은 통계적 접근법이고 두번째 방법은 정보 이론적 접근법이다.

통계적 추출 방법으로는 Monte Carlo Markov Chain (MCMC) 방법이 미지의 분포 p 에서 표본을 추출하기 위해 가장 널리 쓰인다. Gibbs sampling, Metropolis sampling, perfect sampling 등이 모두 이 부류에 속한다. 이 접근법의 가장 큰 단점은 표본들이 실질적으로는 분포 p 를 모르기 때문에 p 가 아니라 추정 분포 q 에서 추출된다는 점이다. 만약 p 와 q 사이의 거리가 멀다면, 이 방법들은 매우 느리게 수렴할 것이다. 따라서 MCMC 에서는 새로운 표본을 생성하기 위한 분포 q 가 실제 분포 p 와 가능한 한 비슷해야 한다. 그러나, 이런 분포를 찾기가 쉽지 않다.

대부분의 자연언어 자원들이 매우 중복되어 있기 때문에, 능동 학습도 널리 쓰인다. 즉, 정보 이론적 접근법의 관점에서 보면, 전체 데이터 다 쓰는 것보다 정보량이 많은 부분만을 쓰는 것이 매우 효율적이다.

그림 2 는 최대 엔트로피 모델을 사용하는 능동 학습의 과정을 보이고 있다. 주어진 N 개의 학습예제를 갖는 D 에 대해서, 첫 학습예제 $D(0)$가 $D(0) \subset D$ 로 초기화된다. Dv 라는 독립된 검증 집합이 있다고 가정한다. 각 i 번째 반복에서, 모델이 $D(i)$로 학습된 후, 학습 집합은 학습된 모델이 Dv 에 대해 이전 모델보다 더 좋은 성능을 보일 때에만 확장된다. 학습 집합을 확장하기 위해서, $D - D(i)$ 에서 λ 개의 예제를 선택해서 현재 $D(i)$에 더해 $D(i+1)$을 구성한다. 만약, 학습된 모델의 성능이 Dv 로 검사해 보았을 때 이전 모델보다 좋지 않다면, 학습을 중단한다.

$D - D(i)$에서 에제를 선택하는 기준으로는 무엇을 사용해야 할까? Zhang 은 평균 제곱 오차에 기반해서 중요한 예제를 선택하는 능동 학습이 학습 속도와 일반화 성능을 개선할 수 있음을 증명하였다[16]. Seung et al.은 'Query by Committee'(QBC)의 이론적 분석을 제시하였다[15]. Freund et al.은 어떤 조건 하에서 임의의 n 예제를 검사하는데 필요한 질의의 수가 n 의 로그 형태로 표현됨을 증명하였다[8]. 이 때 질의가 전체 예제에 대해 행해진다면 일반화 오류는 $O(1/n)$으로 줄어든다. 그러므로 질의의 수라는 면에서 일반화 오류는 매우 빨리 줄어든다. 그러나, QBC 의 효과는 분류기를 version space 에서 임의로 추출할 수 있을 때에만 유효하다.

여러 개의 분류기를 유지하기가 힘들고 하나의 분류기가 예제의 불확실성(uncertainty)을 측정할 수 있는 경우에는, 불확실성 추정값이 예제를 선택하는데 사용될 수 있다. 예제 e 의 불확실성은 사후분포(posterior distribution) $P(C \mid e)$와 *균등 분포*(uniform distribution) 사이의 거리를 사용해 측정할 수 있다. 두 확률분포 사이의 거리를 측정하는 자연스러운 단위는 *상대 엔트로피*(relative entropy) 혹은 *Kullback-Leibler divergence*(KL-divergence)이다. 두 분포 p 와 q 의 거리는

$$KL(p \,\|\, q) = \sum_{c_i \in C} p(c_i) \ln \frac{p(c_i)}{q(c_i)}$$

로 정의되며, C 는 학습 예제가 속할 수 있는 클래스의 집합이다. 그러므로, 가장 불확실한 예제 e^*는 다음과 같이 정해진다.

$$e^* = \arg\max_{e_j \in D - D(i)} KL(U \,\|\, p(C \mid e_j))$$

여기서, U 는 균등 분포이다.

5. 불균형 데이터를 위한 AdaBoost

CoNLL-2000 데이터 집합과 같은 대부분의 자연언어 자원들은 매우 불균형적이다. 예를 들어, CoNLL-2000 데이터 집합에서는 12 개의 단위 종류 중 단지 네 개의 주요 단위(NP, PP, VP, O)가 전체 데이터의 95.10%를 차지한다. 이런 고도의 불균형은 모델의 재현도(recall)을 낮춤으로써 학습 모델의 성능을 떨어뜨린다. 데이터 불균형 문제를 해결하기 위해, O 와 Zhang 은 AdaBoost 알고리즘을 사용하여 문서 분류의 실험에서 AdaBoost 가 불균형 데이터의 재현도를 높일 수 있음을 보였다[18].

AdaBoost 에서는 이전의 약 분류기(weak classifier)에 의해 쉽게 분류되는 예제는 중요도가 낮아지고, 분류하기 어려운 예제의 중요도는 높아진다. 따라서, AdaBoost 는 약 분류기에게 가장 어려워 보이는 예제들에 점점 집중하게 된다. 드물게 나타나는 단위화 종류의 예제들은 루프(loop)의 처음에 잘 분류가 되지 않으므로, 이런 예제들은 반복회수가 많아질수록 자주 나타나는 단위화 종류보다 높은 확률로 다시 샘플링된다. 또한, AdaBoost 가 일종의 위원회 모델(committee model)로 생각될 수 있기 때문에 재현도 뿐만 아니라 정확도도 높아질 수 있다.

6. 실 험

6.1 데이터 집합

데이터 집합으로 *CoNLL-2000 Shared Task* 데이터와 [11]에서 사용된 한국어 단위화 데이타를 사용하였다[5]. CoNLL-2000 데이터 집합은 Wall Street Journal(WSJ) 말뭉치에서 추출한 학습 집합과 테스트 집합으로 구성되었다. 학습 집합은 WSJ 말뭉치의 section 15-18 까지이며 211,727 단어로 구성되었고, 테스트 집합은 WSJ 말뭉치의 section 20 에서 추출한 47,377 단어로 구성되었다. 학습 집합과 테스트 집합의 각 단어는 두 개의 부가 표지를 가지는데, 이들은 각각 품사와 단위 표지이다(그림 3). 품사는 Brill tagger 에 의해 결정된 것이다. 단위 표지는

He	PRP	B-NP
reckons	VBZ	B-VP
the	DT	B-NP
current	JJ	I-NP
deficit	NN	I-NP
will	MD	B-VP
narrow	VB	I-VP
to	TO	B-PP
only	RB	B-NP
#	#	I-NP
1.8	CD	I-NP
billion	CD	I-NP
in	IN	B-PP
September	NNP	B-NP
.	.	O

그림 3. CoNLL-2000 데이터집합의 예.

단위 종류의 이름을 포함한다. 대부분의 단위 종류는 두 가지의 단위 표지를 가진다. 예를 들어, B-NP 는 명사구 절의 첫번째 단어를 나타내며, I-NP 는 명사구 단위의 다른 단어에 주어진다. 이 데이터 집합에서는, 11 개의 구의 종류와 하나의 추가적인 단위 표지, O 를 가정한다. 따라서, 전체 단위 표지는 23 개이다. O 단위 표지는 어떤 단위에도 포함되지 않는 단어들을 위해 존재한다. CoNLL-2000 데이터 집합을 능동 학습에 사용하기 위해, 학습 집합을 두 하위 집합으로 나누었다. 작은 학습 집합 D 는 148,181 단어를 포함하고, 검증 집합 Dv 는 63,546 단어를 포함한다.

[11]에서 사용된 한국어 데이터는 인터넷 홈페이지에서 추출한 문장들로 이루어져 있다. 이 데이터집합은 1,273 개의 문장으로 이루어진 학습 데이터와 638 개의 문장으로 이루어진 테스트 데이터로 구성된다. 각 문장의 평균 어절 수는 8.9 개이다.

6.2 일차 자질

영어 문장을 단위화하기 위해서, 품사를 위한 *trigram* 모델과 단위를 위한 *bigram* 모델의 조합을 사용하였다. 표 1 은 결정트리를 학습하기 위한 언어학적 자질을 보이고 있다. 단어 W_i 의 단위 표지를 결정하기 위해, 주의 단어들의 품사와, W_i 자체, 그리고 앞 두 단어의 단위 표지가 사용되었다. 문맥 단어의 어휘 정보는 전혀 사용되지 않았다. 어휘 정보를 많이 사용할수록 단말의 수가 엄청나게 많은 결정트리가 만들어진다. 따라서, 최대 엔트로피 문제를 계산하기가 점점 더 힘들어 진다. 또한, W_i 의 문맥정보만 사용할 때 성능이 더 좋아짐을 실험적으로 알았다.

한국어 문장을 단위화하기 위해서는 [11]에서 *k*-NN 을 학습시킬 때 사용한 자질과 같은 일차 자질을 사용하였다(표 2). 앞서 나타난 두 어절과 자신의 어휘 정보(w_{i-2}, w_{i-1}, w_i) 및 품사(POS_{i-2}, POS_{i-1}, POS_i), 앞 단어(w_{i-1})의 조사나 어미(E_i), 그리고 앞의 두 단위화 표지(C_{i-2}, C_{i-1})가 사용되었다. E_i를 제외하면, 영어에서 사용된 자질과 큰 차이가 없다. 이는 사용할 수 있는 일차 자질이 주로 지역적 정보이기 때문이다. 물론, 표 2 에서 제시한 자질이 한국어 문장 단위화를 위한 최적의 자질 집합은 아닐 수 있지만, 이 자질들은 실험을 통해서 좋은 성능을 보이는 것들로 선택된 것이다.

언어 속성	설명
POS_{i-2}	단어 W_{i-2}의 품사
POS_{i-1}	단어 W_{i-1}의 품사
POS_i	단어 W의 품사
POS_{i+1}	단어 W_{i+1}의 품사
POS_{i+2}	단어 W_{i+2}의 품사
W_i	단어 W
C_{i-2}	단어 W_{i-2}의 단위 표지
C_{i-1}	단어 W_{i-1}의 단위 표지

표 1. 영어 단위화를 위한 최대 엔트로피 모델의 자질 함수를 구성하기 위해 사용된 언어학 속성.

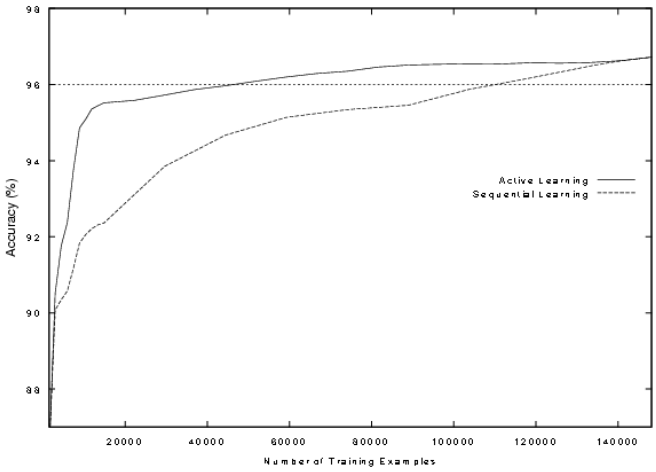
언어 속성	설명
W_{i-2}	단어 W_{i-2}
W_{i-1}	단어 W_{i-1}
W_i	단어 W
POS_{i-2}	단어 W_{i-2}의 품사
POS_{i-1}	단어 W_{i-1}의 품사
POS_i	단어 W의 품사
C_{i-2}	단어 W_{i-2}의 단위 표지
C_{i-1}	단어 W_{i-1}의 단위 표지

표 2. 한국어 단위화를 위한 최대 엔트로피 모델의 자질 함수를 구성하기 위해 사용된 언어학 속성.

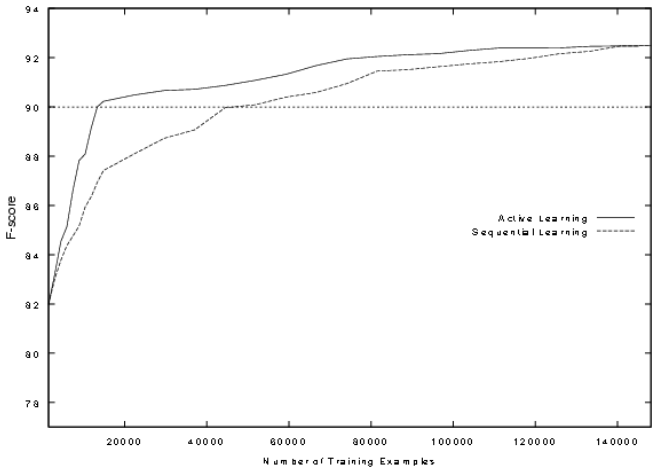
6.3 실험 결과

표 3 은 CoNLL-2000 데이터 집합의 텍스트 단위화에 대한 제시된 방법의 실험 결과를 보인다. 제시된 방법은 96.72%의 정확도와 92.48 의 F-score 를 보이는데, 이는 C4.5 보다 정확도와 F-score 에서 2.34%와 2.28 만큼 향상된 결과이다. 향상 정도는 미미해 보이지만, 오류의 수를 보면 2,663 개에서 1,554 개로 줄어들어 41.64% 줄어들었다. 텍스트 단위화가 주로 구문 분석의 전 단계이고 이 단계에서의 오류가 다른 단계에서 복구될 수 없으므로 이 정도의 향상도 매우 의미가 있다. 또한, 결정트리 자체가 강력한 분류기이므로, 향상된 결과는 매우 중요하다. 따라서, 제시된 방법이 기존의 결정트리를 강화하는데 유용하게 사용될 수 있다.

표 4 는 한국어 데이터에 대한 제시된 방법의 실험 결과이다. 이 표에서 기본 방법은 [20]과 [21]에서 제시된 규칙 기반의 방법을 사용하였을 때에 얻은 결과이고, 'Combined *k*-NN'은 [11]에서 제시한 규칙 기반의 방법과 *k*-Nearest Neighor 알고리즘을 섞어서 학습한 방법이다. 제시된 방법은 규칙만 사용했을 때보다는 좋은 성능을 보이지만, 규칙을 *k*-NN 과 함께 사용했을 때보다는 성능이 약간 떨어진다. 최대 엔트로피 모델이 *k*-NN 보다 훨씬 계산론적으로 비싼 모델이어서, 한국어에 대해서는 효용성이 높지 않았다. 하지만, 사용된 데이터집합이 크지 않은 비표준 집합이므로, 대용량의 표준 데이터집합을 사



(a) 정확도



(b) F-score

그림 4. 제시된 방법의 성능. 두 그림 모두 능동 학습과 순차 학습을 비교한다. 그림 (a)는 정확도 커브이고 (b)는 F-score 커브이다.

용하여 실험적으로 이를 다시 검사해 볼 필요가 있다.

방법	정확도	F-score	오류의 수
제시된 방법	**96.72%**	**92.48**	**1,554**
C4.5	93.03%	90.20	2,663

표 3. 텍스트 단위화의 실험 결과.

방법	정확도
기본 방법	95.5 %
Combined *k*-NN	97.8 %
제시된 방법	97.2 %

표 4. 한국어 단위화 실험 결과.

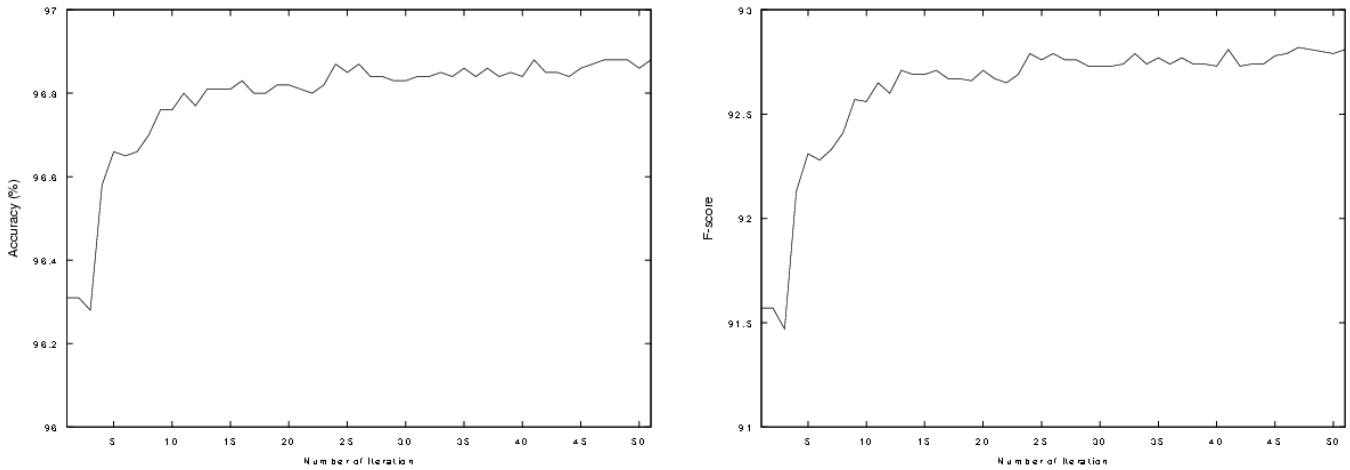
능동 학습이 학습 예제의 수를 줄이는데 얼마나 유용하였는지 알아보자. 그림 4 는 능동 학습이 학습 예제의 수를 줄이는데 얼마나 효과적인지 보인다. 이 그림에서 '*Sequential Learning*'은 학습 예제가 순차적으로 추가됨을 나타내고, '*Active Learning*'에서는 우선순위에 따라 추가된다. X-축은 학습 예제의 수를 나타내고, Y-축은 텍스트 단위화의 정확도와 F-score 를 나타낸다. 두 직선 사이의 차이가 능동 학습을 사용함으로써 얻을 수 있는 성능의 향상이다.

96%의 정확도를 얻기 위해, 능동 학습은 거의 50,000 개의 학습 예제를 사용한다. 이는 순차적 학습에서 사용된 예제의 절반보다 훨씬 적으며, 전체 학습 예제의 34%에 불과하다. 전체 학습 예제의 54%인 80,000 개 이상의 학습 예제가 사용된 후에는 정확도가 거의 증가하지 않지만, 이 정확도는 전체 데이터를 다 사용한 경우의 96.46%와 거의 비슷한 결과이다. F-score 에 대해서도, 학습 커브는 거의 비슷한 결과를 보인다. F-score 90 을 얻기 위해 능동학습에서는 순차학습에서 필요한 예제의 1/4 정도만 필요하다. 정확도에서처럼 학습 커브가 평평해진 80,000 개의 학습 예제로 92.05 의 F-score 를 얻었다. 이 값은 전체 학습 예제를 다 쓴 것과 거의 비슷한 결과이다.

6.4 AdaBoost 의 효과

그림 5 는 CoNLL-2000 데이터집합에 대한 텍스트 단위화에 대한 AdaBoost 의 효과이다. 13 번 반복까지는 정확도와 F-socre 모두 반복할 때마다 점점 더 향상된다. 그 이후에는 두 측정단위 모두에서 약간의 변화는 있지만 성능의 변화가 거의 없다. 따라서, boosting 으로 정확도와 F-score 모두 향상되고 F-score 가 정확도보다 훨씬 더 좋아진다. 이는 boosting 으로 재현도 외에도 정확도도 함께 좋아졌기 때문이다.

표 5 는 AdaBoost 로 가장 좋은 성능을 보일 때의 결과이다. 이 결과에 따르면, 96.88%의 정확도와 92.82 의 F-score 를 보인다. 표 3 과 비교해 보면, 정확도가 0.16% 좋아지는데 비해 F-score 는 0.34 정도 좋아진다. 이 성능은 Koeling 의 최대 엔트로피 모델[9]보다도 훨씬 더 좋은 결과이다. 비록 이 결과가 CoNLL-2000 데이터 집합을 사용한 결과 중에서 가장 좋은 것[19]은 아니지만, 제시된 방법은 매우 조심스럽게 선택된 자질로 학습된 최대 엔트로피 모델보다도 더 좋은 성능을 낼 수 있음을 보였다.



(a) 정확도 (b) F-score

그림 5. 제시된 방법에 대한 AdaBoost 의 효과. 그림 (a)는 정확도를 측정한 것이고, (b)는 F-score 를 측정한 것이다.

Type	Precision	Recall	F-score
ADJP	62.23%	65.07%	63.62
ADVP	74.48%	78.87%	76.61
CONJP	40.00%	66.67%	50.00
INTJ	100.00%	50.00%	66.67
LST	0.00%	0.00%	0.00
NP	92.49%	94.75%	93.61
PP	96.63%	97.71%	97.17
PRT	73.74%	68.87%	71.22
SBAR	90.89%	87.66%	89.25
VP	92.67%	93.58%	93.12
All	91.96%	93.69%	92.82

표 5. CoNLL-2000 데이터집합에 대한 최종 결과. 정확도는 96.88%이다. [9]의 F-score 는 91.97 이다.

7. 결 론

본 논문은 최대 엔트로피 모델을 텍스트 단위화에 적용하는 새로운 방법을 제시하였다. 결정트리에서 추출된 자질로 학습된 최대 엔트로피 모델은 능동 학습을 함으로써 실용성과 성능 향상을 보인다. 최대 엔트로피 모델을 위한 자질은 지역 어휘 정보를 사용해서 학습된 결정트리에서 자동으로 생성되었다. 그러므로, 텍스트 단위화에 최대 엔트로피 모델을 사용하기 위해 목적 언어에 대한 많은 사전 지식이 없어도 된다.

한국어 데이터집합에 대한 테스트 결과, 제시된 방법은 97.2%의 정확도를 보여, 규칙만 사용한 경우보다 높은 정화도를 보였다. 한편, CoNLL-2000 데이터집합에 대해서, 제시된 방법은 92.48 의 F-score 를 보였는데, 이는 결정 트리는 전문가에 의해 선택된 자질로 학습된 최대 엔트로피 모델보다 좋은 결과이다. 이 향상은 매우 미미해 보이지만, 단위화가 구문 분석의 전 단계이고 이 단계의 오류가 자연언어 처리의 하위 단계에서 복구될 수 없으므로 매우 의미 있는 향상이다. 또한, 제시된 방법은 능동 학습으로 제공된 전체 예제의 54%만 사용하고도 모든 예제를 다 사용한 것과 비슷한 결과를 보였다.

또한, 텍스트 단위화 학습에 AdaBoost 가 효과적임을 보였다. AdaBoost 를 통해서 정확도와 재현도 모두 향상되었다. 최종적으로, 본 논문에서 제시된 방법으로 92.82 의 F-score 를 얻었으며, 이는 텍스트 단위화에 적용된 여러 가지 기계 학습 알고리즘 중 가장 좋은 방법과 비슷한 성능이다.

감사의 글

본 연구는 한국과학재단(KOSEF)의 첨단정보기술연구센터(AITrc)와 교육부 BK 21 사업에 의하여 지원되었음.

참고문헌

[1] S. Abney, “Parsing by Chunks,” In *Principle-Based Parsing*, Kluwer Academic Publishers, 1991.

[2] S. Argamon, I. Dagan, and Y. Krymolowski, “A Memory-based Approach to Learning Shallow Natural Language Patterns,” In *Proceedings of COLING/ACL 1998*, pp. 67-73, 1998.

[3] A. Berger, S. Pietra, and V. Pietra, “A Maximum Entropy Approach to Natural Language Processing,” *Computational Linguistics*, Vol. 22, No. 1, 1996.

[4] S. Chen and R. Rosenfeld, “Efficient Sampling and Feature Selection in Whole Sentence Maximum Entropy Language Models,” In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 549-552, 1999.

[5] CoNLL, *Shared Task for Computational Natural Language Learning (CoNLL)*, http://lcg-www.uia.ac.be/conll2000/chunking, 2000.
[6] J. Darroch and D. Ratcliff, "Generalized Iterative Scaling for Log-linear Models," *The Annals of Mathematical Statistics*, Vol. 43, No. 5, pp. 1470-1480, 1972.
[7] R. Florian, J. Henderson, and G. Ngai, "Coaxing Confidences from an Old Friend: Probabilistic Classification from Transformation Rule Lists," In *Proceedings of EMNLP/VLC-2000*, pp. 26-34, 2000.
[8] Y. Friend, S. Seung, E. Shamir, and N. Tishby, "Information, Prediction, and Query by Committee," In *Proceedings of NIPS-92*, pp. 483-490, 1992.
[9] R. Koeling, "Chunking with Maximum Entropy Models," In *Proceedings of CoNLL-2000 and LLL-2000*, pp. 139-141, 2000.
[10] G. Ngai and D. Yarowsky, "Rule Writing or Annotation: Cost-efficient Resource Usage for Base Noun Phrase Chunking," In *Proceedings of ACL-2000*, pp. 547-554, 2000.
[11] S.-B. Park and B.-T. Zhang, "Combining a Rule-based Method and a *k*-NN for Chunking Korean Text," In *Proceedings of ICCPOL 2001*, pp. 225-230, 2001.
[12] S. Pietra, V. Pietra, and J. Lafferty, "Inducing Features of Random Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 4, pp. 380-393, 1997.
[13] R. Quinlan, *C4.5: Programs for Machine Learning*, Morgran Kaufmann Publishers, 1993.
[14] L. Ramshaw and M. Marcus, Text Chunking Using Transformation-based Learning," In *Proceedings of VLC-95*, pp. 82-94, 1995.
[15] S. Seung, M. Opper, and H, Sompolinsky, "Query by Committee," In *Proceedings of COLT-92*, pp. 287-294, 1992.
[16] B.-T. Zhang, "Accelerated Learning by Active Example Selection," *International Journal of Neural Systems*, Vol. 5, No. 1, pp. 67-75, 1994.
[17] G. Zhou and J. Su, "Error-driven HMM-based Chunk Tagger with Context-dependent Lexicon," In *Proceedings of EMNLP/VLC-2000*, pp.71-79, 2000.
[18] J.-M. O and B.-T. Zhang, "Boosting Linear Perceptrons for Unbalanced Data," In *Proceedings of International Conference on Neural Information Processing*, pp. 642-645, 2000.
[19] T. Zhang, F. Damerau, and D. Johnson, "Text Chunking Using Regularized Winnow," In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, 2001.
[20] 신효필, "최소자원 최대효과의 구문분석," 제 11 회 한글 및 한국어정보 처리 학술 대회 논문집, pp. 242-247, 1999.
[21] 김미영, 강신재, 이종혁, "단위분석과 의존문법에 기반한 한국어 구문분석," 제 27 회 정보과학회 봄 학술대회 논문집, pp. 327-329, 2000.