

이종의 통계정보를 이용한 품사 부착 기법

Part-Of-Speech Tagging using multiple sources of statistical data

조세형

Sehyeong Cho

명지대학교 컴퓨터소프트웨어학과

요 약

통계적 방식의 품사부착 문제는 보통 N-그램과 같은 단일 통계정보를 활용하지만 단순한 통계 정보라는 원천적인 한계가 있어 많은 오류가 발생한다. 따라서 다양한 정보를 활용하는 것이 정확도를 높일 가능성이 있다는 데는 이론의 여지가 없다. 그러나 다른 종류의 통계 자료는 배타적 자료가 아닌 한 상충되는 정보를 가질 수밖에 없으므로 이러한 정보들로부터 어떻게 종합적인 결론을 내는가 문제이다.

본 논문에서는 이러한 상이한 통계정보를 통합하는 방법으로 언어 모델의 구성에서 활용된 바 있는 최대엔트로피 모델의 한국어 품사 부착에의 사용 가능성을 제시한다. 여기서는 이종의 통계정보로서 N그램과, 트리거 쌍을 사용하게 된다. 이러한 트리거 쌍 통계정보를 N그램과 함께 최대엔트로피 모델링을 했을 경우 퍼플렉시티가 어떻게 변화하는지에 대한 실험 결과를 관찰하게 될 것이다. 트리거 쌍은 또한 다양하게 문맥사이즈를 변화할 수 있으며, N그램의 확률 모델도 다양하기 때문에 여러 종류의 실험을 통한 많은 향상을 예상 할 수 있다. 본 실험에서는 단일 모델 사용시 94.9 %의 정확도를 가진 3-그램 모델에 트리거 쌍을 최대 엔트로피 방식으로 추가한 경우 95.6%의 정확도를 보여 0.7% 포인트의 정확도 향상을 기록하였다. 따라서 향후 다양한 정보원을 개발하여 최대엔트로피 방식으로 통합할 경우 지속적인 정확도 향상을 가져올 수 있는 가능성을 보여준다고 할 수 있다.

키워드 : 최대 엔트로피, 품사, N-그램, 트리거 쌍

Abstract

Statistical POS tagging is prone to error, because of the inherent limitations of statistical data, especially single source of data. Therefore it is widely agreed that the possibility of further enhancement lies in exploiting various knowledge sources. However these data sources are bound to be inconsistent to each other.

This paper shows the possibility of using maximum entropy model to Korean language POS tagging. We use as the knowledge sources n-gram data and trigger pair data. We show how perplexity measure varies when two knowledge sources are combined using maximum entropy method. The experiment used a trigram model which produced 94.9% accuracy using Hidden Markov Model, and showed increase to 95.6% when combined with trigger pair data using Maximum Entropy method. This clearly shows possibility of further enhancement when various knowledge sources are developed and combined using ME method.

Key Words : Maximum Entropy, Part of speech, N-gram, trigger pair

1. 서 론

자연어 문장을 분석하기 위하여 흔히 사용하는 품사 부착 문제는 근본적으로 동일한 형태소가 여러 품사를 가질 수 있는 모호성을 해결하는 문제이며 그 해결 방식 중에서 가장 많이 쓰이는 방법 중의 하나는 통계적 정보를 이용하는 방법이다. 널리 쓰이고 있는 방법의 하나는 N-그램의 통계적 정보를 이용하여 가장 확률이 높은 품사열을 찾아내는 방법으로서 흔히 은닉 마르코프 모델[1]을 이용한다. 그러나 N-그램이라는 정보는 언어 현상의 복잡함을 지나치게

단순히 모델링하였기 때문에 근본적인 한계를 안고 있다. 또한 그 외의 트리거 쌍이나 상호정보(mutual information) 등의 통계적 정보 또한 근본적인 한계를 가지고 있다.

본 연구의 목적은 두개 이상의 서로 다른 통계정보를 통합함으로써, 보다 예측의 정확도가 높은 확률 모델을 만드는 것이다. 본 연구에서는 21세기 세종계획에서 2001년도에 만든 말뭉치로부터 기존 방식에서 사용하던 이종의 통계 정보를 추출해 낸 다음, 이 두 가지의 상충되는 통계 정보를 이용하여 하나의 확률 모델을 생성해 냄으로써 좀 더 퍼플렉시티가 높은 모델을 생성해 내는 것을 목표로 한다. 이 이종의 확률 모델을 최대엔트로피[2]를 통해 통합함으로써, 향상된 정확도를 위한 품사부착기법을 제시한다. 일반적으로 최대 엔트로피 모델은 여러 종류의 확률모델을 결합하기에 적합한 확률 모델로 알려져 있으며 다방면에서 활용이 된 바 있다.

Rosenfeld 는 언어 모델에서 최대 엔트로피 방식을 이용

접수일자 : 2007년 9월 1일

완료일자 : 2008년 2월 20일

본 연구는 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 정보통신부의 유비쿼터스컴퓨팅및네트워크원천기술개발사업의 지원에 의한 것임.

하여 2종 이상의 정보원천을 이용하는 방법을 개발하였다.[2]

Ratnaparkhi는 자연어 처리중 애매성 해소를 위하여 최대 엔트로피 방식을 이용하였으며[6], 이 방식을 영어 품사 부착에 활용하여 96.5%의 정확도를 얻었다.[9]

한국어에서는 텍스트 단위화(chunking)를 위해 최대 엔트로피 모델이 사용된 바 있다.[10]

통계 정보의 원천 자체를 2원화 하여 사용하는 방법도 있으나[11] 본 연구에서는 원천 자체는 하나, 즉 하나의 말뭉치만을 사용하고 여기에서 다른 형태의 조건부 확률들을 추출하여 사용하게 된다.

2절에서는 단일 통계 정보를 사용하는 기존 방식들을 소개한다. 3절에서는 최대 엔트로피 방법을 소개하고 여기서 어떻게 품사 부착에 이용하는 방법을 소개한다. 4절에서는 실험 결과를 소개한다.

2. N그램 은닉 마코프 모델과 트리거쌍

2.1 N그램과 은닉 마코프 모델

상태 전이 모델을 이용하는 은닉 마코프 모델(hidden markov model)은 일반 마코프 모델과는 달리 숨겨진 상태 전이 모델로써, 외부의 관찰된 데이터(단어 또는 형태소)를 통해서 가장 확률이 높은 상태 전이 시퀀스, 즉, 품사열을 추정한다.[3]

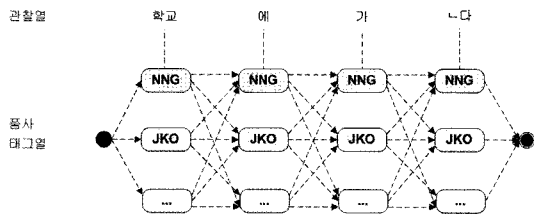


그림 1. 품사 전이의 상태 다이어그램
Fig. 1. State transition diagram of POS's

$$tag = \operatorname{argmax} P(tag|word) = \operatorname{argmax} P(word|tag)P(tag)$$

$$\hat{T} = \operatorname{argmax} P(T|W) \quad (2.1)$$

$$P(T|W) = \frac{P(W|T)P(T)}{P(W)} \quad (2.2)$$

$$\hat{T} = \operatorname{argmax} P(W|T)P(T) \quad (2.3)$$

\hat{T} 는 주어진 문장의 최대 확률을 갖는 품사열이다. Bayes rule에 의해서 수식 (2.2)로 표현하였으며, P(W)는 이미 W가 주어졌을 동일한 확률을 갖기 때문에 생략하여 최종적으로 수식 (2.3)으로 표현할 수 있다. 수식 (2.3)에서 P(W|T)는 보통 방출(emission)확률이라고 하며 이 경우에는 해당 품사로부터 특정 형태소가 발현될 확률이다. 그리고 P(T)는 해당 품사열이 나타날 확률로써, 은닉 마코프 모델에서는 상태 전이(transition)확률이라고 한다. 하지만 전체 이력(history)을 고려하여 품사 출현 확률을 계산하는 것은 현실적으로 불가능하기 때문에 마코프 가정을 도입하여 한정된 길이의 이전 문맥에 의존하게 된다. 여기서 N그램은 상태 전이 확률을 구하기 위한 통계적 데이터로 사용된다.

Bigram 모델의 경우 하나의 품사는 이전 품사에 의해 확률이 결정된다고 가정한다.

$$P(T_1) = P(T_1)P(T_2|T_1)...P(T_n|T_{n-1})$$

Trigram 모델의 경우 하나의 품사는 이전 두 개의 품사에 영향을 받아서 발생한다고 가정한다.

$$P(T_1) = P(T_1)P(T_2|T_1)P(T_3|T_{1,2})...P(T_n|T_{n-2,n-1})$$

일반적으로 n의 값이 클수록 더 성능이 우수할 것이라고 생각할 수 있다. 하지만 현재 품사에 영향을 주는 문맥 사이즈가 클수록 계산량이 많아질 뿐더러 충분한 통계 정보를 얻기가 힘들다. 따라서 훈련 말뭉치가 클 경우에는 trigram 이상을 사용하며, 적을 경우에는 bigram이 주로 사용된다. 음성인식을 위한 언어 모델에서는 4또는 5-gram도 종종 사용된다.

2.2 트리거 쌍

트리거 쌍(trigger pair)은 문장의 이력으로부터 정보를 추출하는 하나의 통계적 방법을 이용한 확률 모델이다. 만약에 단어 A가 다른 단어 B와 연관성이 크다면, 즉 A가 이력에 있을 경우 B가 출현하는 확률이 보통의 경우보다 크다면 이것을 (A->B)라고 나타낼 수 있다. 여기서 A는 트리거이며, B는 피 트리거(triggered sequence), 이 쌍을 트리거 쌍이라고 말한다[4].

예를 들어, 이력에 “은행”이라는 단어가 있을 경우 “대출”이라는 단어가 출현할 확률은 평상시보다 높을 수 있다. 이러한 두 단어 간의 연관성을 이용하는 것이 바로 트리거 쌍이다.

그렇다면 이러한 트리거 쌍을 어떻게 선택할 것인지가 문제이다. 해당 트리거에 대해 피 트리거가 연관성이 높은 순으로 나열함으로써, 우리는 통계정보로 활용할 수가 있다. 단어 쌍의 연관성을 측정하는 좋은 방법으로 상호정보(mutual information)[5]를 계산한다.

$$MI(A,B) = P(A,B) \log \frac{P(B|A)}{P(B)} \quad (2.4)$$

모든 단어 쌍에 대해서 상호정보를 계산하여, 가장 높은 값을 갖는 상위 3개 또는 6개 등을 선택하여 사용하게 된다.

본 논문에서는 실제 품사 부착의 문제를 다루기 때문에 피 트리거는 품사가 된다. 이력 내에 특정 트리거가 나올 경우 해당 품사(피 트리거)가 나올 확률이 높다고 이해할 수 있다.

트리거는 문맥상의 정보를 말하는 것이며, 여기서는 이력 내의 품사나 형태소가 이에 속한다. 물론 품사결정문제에서 트리거 쌍이 얼마나 많은 유용한 통계정보를 제공하게 될지는 모르나 N그램으로 다루지 못한 문맥의 또 다른 정보로 활용이 가능할 것이다.

그리고 트리거는 두 가지 방법이 가능하다. 문맥상에 나타난 위치정보까지 다루는 방법과 그렇지 않는 방법이 있다. 최대 엔트로피 모델에 쓰이게 될 트리거 쌍은 문맥상의 위치정보는 포함하지 않았다. 따라서 문맥 사이즈를 정해서 위치와는 상관없이 해당 이력내의 트리거의 발생 빈도로 트리거 쌍을 선택하게 되며, 문맥 크기의 변화에 따라서도 트리거 쌍의 변화가 생길 가능성 또한 존재한다. 따라서 알맞은 문맥사이즈를 정하는 것 또한 하나의 중요한 과제이다. 고려되는 문맥사이즈의 변화가 실험 결과에 어떠한 영향을

주게 되는지 4절의 실험결과를 통해서 관찰할 수 있다.

3. 최대 엔트로피 모델

3.1 최대 엔트로피 개요

둘 이상의 다른 통계적(즉 경험적) 정보가 같은 사상에 대해 다른 확률을 보이는 것은 당연하다. 예를 들어 텍스트에서 다음 단어가 “대출”이라는 단어가 나올 확률을 생각해 보자. 만일 이력을 고려하여 같은 문단 안에 ‘은행’이라는 단어가 있을 경우 ‘대출’의 출현 확률을 P1이라고 하고 또 다른 통계 정보에 의하면 이전 단어가 “신규” 인 경우 ‘대출’의 확률은 P2 라고 할 때 두 가지 조건이 모두 만족되는 경우의 동 단어의 출현확률은 얼마인가라는 문제이다. 이 두 가지의 정보를 가장 잘 통합하는 방법은 결과 확률이 높은 것도 아니고 낮은 것도 아니며 실제의 확률에 가까울수록 좋은 통합 방법이라고 할 수 있다.

최대 엔트로피 방법[2][4][6]이란 이러한 경우 제약 조건들 (이 경우 두 확률 정보)을 만족하는 무수한 확률 분포 가운데 엔트로피가 가장 높은 분포, 즉 bias가 가장 작은 분포를 채택하는 것이다.

제약 함수 f 는 아래와 같이 정의 된다.

$$f_j(a,b) = \begin{cases} 1 & \text{만약 입력 } a \text{가 feature } j \text{의 } a' \text{와 같고,} \\ & b \text{가 해당 문맥에 고려될 때} \\ 0 & \text{otherwise} \end{cases}$$

앞서 말했듯이, 우리는 모든 확률 분포들 중에서 엔트로피가 최대인 확률 분포 p*를 찾아야만 한다. 한가지의 방법은 Maximum likelihood이다.

$$p(ab) = \frac{1}{Z(b)} \prod_{j=1}^k \alpha_j^{f_j(a,b)} \quad (3.1)$$

$$Z(b) = \sum_a \prod_{j=1}^k \alpha_j^{f_j(a,b)} \quad (3.2)$$

여기서 Z(b)는 k개의 feature가 주어졌을 때 $\sum_a p(ab) = 1$ 을 보장하기 위한 정규화 과정이다. 현재 품사의 가중치의 총 합으로 나뉘므로써, 확률의 합을 1로 만들기 위함이다. 여기서 α 가 바로 해당 feature의 가중치가 되며 고려되고 있는 해당 feature의 가중치를 곱하여 확률분포를 찾는다.

$$Q = \left\{ p \mid p(ab) = \frac{1}{Z(b)} \prod_{j=1}^k \alpha_j^{f_j(a,b)} \right\} \quad (3.3)$$

$$L(p) = \sum_{a,b} \tilde{p}(a,b) \log p(ab) \quad (3.4)$$

$$p^* = \operatorname{argmax} L(q) \quad , \quad q \in Q \quad (3.5)$$

L(p)은 엔트로피를 계산하는 방법으로써, 확률 분포 p의 집합 Q중에서 엔트로피가 최대인 확률 분포가 바로 찾고자 하는 p*이다. $\tilde{p}(a,b)$ 는 훈련 말뭉치에서 나타난 a와 b의 동시 출현 확률이다.

$$p^* = \operatorname{argmax} H(p) \quad , \quad p \in P$$

$$H(p) = - \sum_{a,b} \tilde{p}(a,b) p(ab) \log p(ab) \quad (3.6)$$

$$P = \{ p \mid E_p f_j = E_{\tilde{p}} f_j \quad \{j=1 \dots k\} \}$$

$$E_{\tilde{p}} f_j = \sum_{a,b} \tilde{p}(a,b) f_j(a,b) \quad (3.7)$$

$$E_p f_j = \sum_{a,b} \tilde{p}(b) p(ab) f_j(a,b) \quad (3.8)$$

여기서 $\tilde{p}(b)$ 는 훈련 말뭉치에서 관측된 해당 문맥에 대한 확률이다. 그리고 $E_{\tilde{p}} f_j$ 는 훈련 말뭉치에서 관측된 해당 feature의 기대치이며 $E_p f_j$ 는 모델 feature의 기대치이다. 우리가 원하는 확률분포 p(ab)중 $E_{\tilde{p}} f_j$ 와 $E_p f_j$ 가 같은 확률분포들 중 엔트로피가 최대인 확률분포를 찾는 것이 conditional maximum entropy model이다. 위와 같은 몇 가지 이유 때문에 일반적으로 Maximum likelihood와 maximum entropy model은 서로 다른 방법이라고 얘기하지만 궁극적으로 엔트로피가 높은 확률분포 p*를 찾는 것은 같다.

$$p^* = \operatorname{argmax} L(q) = \operatorname{argmax} H(p)$$

따라서 maximum entropy model 또한 maximum likelihood와 같은 형태를 가져야만하고 이것이 바로 최대 엔트로피의 기본 이론이다.

bigram, tri-gram, 트리거 쌍의 경우 제약함수는 각각 다음과 같이 정의한다.

$$f_{(t_1, t_2)}(h, t) = \begin{cases} 1 & \text{문맥의 마지막 품사가 } t_1 \text{ 이고} \\ & t = t_2 \text{ 일 때} \\ 0 & \text{otherwise} \end{cases}$$

$$f_{(t_1, t_2, t_3)}(h, t) = \begin{cases} 1 & \text{이전 품사가 } t_1, \text{ 전전 품사가 } t_2 \text{ 이고} \\ & t = t_3 \text{ 일 때,} \\ 0 & \text{otherwise} \end{cases}$$

$$f_{(A \rightarrow B)}(h, t) = \begin{cases} 1 & \text{문맥내에 } A \text{가 존재하고 } t = B \text{ 일 때} \\ 0 & \text{otherwise} \end{cases}$$

로 필요한 제약함수(constraint function)를 정의 할 수 있다. 이러한 제약함수는 뒤에 나올 GIS 알고리즘에 쓰이게 된다.

3.2 파라미터 추정(Parameter Estimation)

최대엔트로피 모델에 필요한 파라미터를 추정하기 위해서는 잘 알려진 GIS(Generalized Iterative Scaling[8]) 알고리즘을 사용하였다. 이것은 말뭉치를 통해서 얻은 자질의 기대치와 최대 엔트로피를 통해서 얻은 자질의 기대치가 근사 화 될 때까지 엔트로피를 최대로 유지하면서 원하는 확률 분포 p*를 찾는 알고리즘이다.

Algorithm (Generalized Iterative Scaling (GIS) 절차는 다음과 같다.

$$\alpha_j^{(0)} = 1 \quad \text{최초 } \alpha \text{ 값 초기화}$$

$$\alpha_j^{(n+1)} = \alpha_j^{(n)} \left[\frac{E_p f_j}{E_{\tilde{p}} f_j} \right]^{\frac{1}{C}} \quad (3.9)$$

$$E_{\tilde{p}} f_j = \sum_{a,b} \tilde{p}(b) p^{(n)}(ab) f_j(a,b) \quad (3.10)$$

$$p^{(n)}(ab) = \frac{1}{Z(b)} \prod_{j=1}^l (\alpha_j^{(n)})^{f_j(a,b)} \quad (3.11)$$

위에 n에 대해서 반복하여, 모든 자질을 만족하고 엔트로피가 최대가 되는 확률 분포 p*를 찾는다.

$$\lim_{n \rightarrow \infty} p^{(n)} = p^*$$

C는 상수 값으로 해당 문맥에 최대 몇 개의 자질(feature)이 참조되는지를 나타낸다. 아래의 수식을 통해서 간단히 구할 수 있다. 보통 이종의 동등한 조건의 통계정보일 경우 C=2의 값을 가진다.

$$C = \max \sum_{j=1}^k f_j(a,b)$$

또한, 여기서 k개의 자질(feature)이 주어질 때 GIS 알고리즘은 말뭉치를 통해서 얻은 자질의 기대치 $E_p f_j$ 와 모델을 통해서 얻은 자질의 기대치 $E_{p^{(n)}} f_j$ 가 필요하다.

$$E_p f_j = \sum_{a,b} \tilde{p}(a,b) f_j(a,b) = \frac{1}{N} \sum_{i=1}^N f_j(a_i, b_i) \quad (3.12)$$

$$E_{p^{(n)}} f_j = \sum_{a,b} \tilde{p}^{(n)}(a,b) f_j(a,b) \quad (3.13)$$

여기서 N은 훈련 말뭉치에서 나타나는 형태소의 수가 된다. 즉 token의 전체 수이다.

4. 실험 결과

어떠한 품사 부착 알고리즘을 사용하는가에 따라서 정확도는 달라지지만 우선은 알고리즘이 사용할 확률 모델 자체가 얼마나 정확한지와 이 모델을 활용하여 실제 정확도가 높은 품사 부착기를 생성할 수 있을 만큼 의미가 있는지, 이 두 문제를 분리하여 고찰할 필요가 있다. Perplexity는 전자를 잘 나타내는 수치이므로 Shannon-McMillan-Breiman 정리에 의해 근사치를 구하여 표 1에 나타내었다. 순수한 trigram일 경우 5.93의 값을 보였으며 이것은 쉬운 말로 하자면 실제 43개의 품사가 있지만 예측 모델은 43개 중 하나를 임의로 고르는 것에 비해 5.93개중에 하나를 임의로 고르는 정도의 정확성을 가졌다는 뜻이 된다. 최대엔트로피 모델을 통해서 얻은 perplexity는 문맥 크기 5와 상위 6개 트리거 쌍을 활용한 경우 4.16 으로 낮아졌다. 즉, 이만큼 다음 품사 예측이 좀 더 정확해진 것이다.

이러한 perplexity의 향상이 실제로 의미가 있는지를 확인하기 위하여 실제 HMM 기반 품사 부착기를 활용하여 실험을 하였다. 표 2는 이 실험의 결과이다. Perplexity가 낮을수록 일반적으로 이 확률 모델을 기초로 만든 태거는 정확도가 증가한다고 볼 수 있다. (물론 구체적인 품사부착 방식에 따라서 정확도는 달라질 것이다.) 여기서는 HMM 방식을 이용하되 위에서 GIS 알고리즘에 의해 구한 확률 모델을 HMM의 state-transition 확률로 이용하였으며 Emission 확률(품사⇒형태소)은 훈련 말뭉치에서 채집한 통계치에 의해 계산하였다. 프로그램은 C언어로 구현되었다.

표 1. 각 실험별 perplexity
Table 1. Perplexity result

tag set	세종 말뭉치 내 43개 품사	
훈련 말뭉치	8000000 형태소	
실험 말뭉치	964055 형태소	
bigram (baseline)	6.50318	6.50318
trigram (baseline)	5.93611	5.93611
트리거쌍(TP) best n	top 3	top 6
bigram + TP (문맥 사이즈 = 3)	5.33069	5.42762
bigram + TP (문맥 사이즈 = 5)	4.81024	4.59653
trigram + TP (문맥 사이즈 = 3)	4.32604	4.57875
trigram + TP (문맥 사이즈 = 5)	4.28079	4.15792

표 2. 각 실험별 품사부착 정확도
Table 2. Accuracy of tagging

tag set	세종 말뭉치 내 43개 품사
훈련 말뭉치	8000000 형태소
실험 말뭉치	964055 형태소
TP 선택수	top 6
문맥 사이즈	5
trigram	94.93%
trigram + TP	95.62%

기존 trigram태거의 경우 94.93%의 정확도를 보였으며, 최대엔트로피의 경우 95.62%의 정확도를 보였다. 이것은 0.69%p 증가한 정확도로서 에러 감소율은 13%에 해당한다. 형태소에 관해서 좀 더 올바른 품사부착 결과를 얻어낸 것이다. 약 4%가량의 말뭉치 오류를 감안한다면 0.69%포인트는 작지 않은 향상임을 알 수 있다.

5. 결 론

트리거 쌍은 다음 단어를 예측하는 순수 언어모델링에 주로 사용된다. 따라서 품사 결정 문제와 같은 품사를 다룰 경우에는 얼마나 유용한 정보를 제공할지를 N그램과 최대엔트로피 모델을 통해서 통합하여 실험하였다. 결과는 순수 N그램에 비해서 Perplexity가 낮게 나타났다.

또한, 트리거 쌍의 트리거를 형태소를 가지고 했을 경우 생각보다 낮은 향상을 보였다. 10만개의 형태소 트리거가 있기 때문에 세분화된 확률이 향상에 도움이 안 되는 것으로 나타났다. 현재로서는, trigram과 POS 태그 트리거를 최대엔트로피로 통합했을 경우가 가장 좋은 개선을 보이고 있다.

좀 더 나은 성능을 위해서는 신뢰성 있는 트리거 쌍의 선정과 문맥정보를 좀 더 활용함으로써 가능하다. 하지만 최대엔트로피의 가장 큰 단점으로써 자질의 수가 많아질수록 계산 량의 부하는 급격히 증가하게 된다. 따라서 다음 품사

를 예측하기 위한 최적의 자질을 선택하는 것이 최대 엔트로피를 이용한 품사부착문제의 관건이라 할 수 있다.

참 고 문 헌

- [1] L.E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state Markov chains," *Ann. Math. Stat.*, vol.37, pp.1554-1563
- [2] E. T. Jaynes, "Information Theory and Statistical Mechanics," *Physical Review* 1957
- [3] Daniel Jurafsky and James H. Martin, *Speech and Language Processing*, Prentice-Hall, 2000
- [4] Ronald Rosenfeld, "Adaptive Statistical Language Modeling: A Maximum Entropy Approach," Ph.D. thesis, School of Computer Science Carnegie Mellon University Pittsburgh, April 19, 1994
- [5] Ken Church and Patrich Hanks, "Word Association Norms, Mutual Information, and Lexicography," *Computational Linguistics*, Volume 16, number 1, pages 22-29, March 1990
- [6] Adwait Ratnaparkhi, "Maximum Entropy Models For Natural Language Ambiguity Resolution," Ph.D. thesis, University of Pennsylvania, 1998
- [7] A.Berger, S.A. Della Pietra, and V.J. Della Pietra, "A Maximum Entropy Approach to Natural Language Processing," *Computation Linguistics*, 22(1):39-71 1996
- [8] J. Darroch and D. Ratcliff, Generalized iterative scaling for log-linear models. *Ann. Math. Statistics*, 43:1470-1480, 1972
- [9] Adwait Ratnaparkhi, "A Maximum Entropy Model of Part-of-speech tagging," *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp.133-142, 1996
- [10] 박성배, 장병탁, "최대 엔트로피 모델을 이용한 텍스트 단위화," 제13회 한글 및 한국어 정보처리 학술대회 논문집, pp. 130-137, 2001.
- [11] Sehyeong Cho, "Improvement of language models using dual-source backoff," *Lecture Notes in Artificial Intelligence*, vol.3157, pp.892-900, Springer, 2004

저 자 소개



조세형(Sehyeong Cho)

1981년 : 서울대학교 공학사

1983년 : 서울대학교 계산통계학석사

1992년 : 펜실베니아주립대 전산학박사

1984년~2000년 : 한국전자통신연구원

2000년~현재 : 명지대학교

관심분야 : 기계학습, 자연어처리, planning

E-mail : shcho@mju.ac.kr