

단어¹와 클래스 기반의 한국어 언어 모델링

김길연⁰ 최기선
한국과학기술원 전 산 학 부
[\[gykim, kschoi\]@world.kaist.ac.kr](mailto:[gykim, kschoi]@world.kaist.ac.kr)

Word and class-based language modeling for Korean

Kilyoun Kim⁰ Key-Sun Choi
Computer Science Division, KAIST

요 약

본 논문에서는 대량의 말뭉치를 바탕으로 한국어에 대해 단어 기반의 n-gram 언어 모델과 클래스 기반의 언어 모델을 구축하고, 이를 실험적으로 검증한다. 단어 기반의 n-gram 모델링의 경우 Katz의 백오프와 Kneser-ney의 스무딩(smoothing) 알고리즘에 대해 실험을 수행한다. 클래스 기반의 언어 모델의 경우에는 품사 태그를 단어의 클래스로 사용한 경우와 말뭉치로부터 자동으로 구축된 클래스를 사용한 경우로 나누어 실험한다. 마지막으로 단어 기반 모델과 클래스 기반 모델을 결합하여 각각의 모델과 그 성능을 비교한다. 실험 결과 단어 기반의 언어 모델의 경우 Katz의 백오프에 비해 Kneser-ney의 스무딩이 보다 좋은 성능을 나타내었다. 클래스 기반의 모델의 경우 품사 기반의 방법보다 자동 구축된 단어 클래스를 이용하는 방법의 성능이 더 좋았다. 또한, 단어 모델과 클래스 모델을 결합한 모델이 가장 좋은 성능을 나타냈다. 논문의 모든 알고리즘은 직접 구현되었으며 KLM Toolkit이란 이름으로 제공된다.

1. 서론

통계적 언어 모델링은 대량의 학습데이터로부터 단어열의 확률 분포를 추정하는 것이다. 자연언어 처리에서 이러한 통계 언어 모델링이 가장 성공적으로 적용된 부분은 음성인식과 품사 태깅이다. 품사 태깅의 경우 추정의 대상이 되는 확률 변수는 태그 열이 되며, 음성 인식의 경우는 단어 열이 된다. 본 논문에서는 현재의 단어를 이전 n개의 단어를 통해 통계적으로 모델링하여, 전체 단어열의 확률을 추정하는 n-gram 모델링(마르코브 가정)으로 정의한다.

이러한 N-gram 모델링에 있어 우리를 가장 괴롭히는 것은 자료 희귀(data sparseness)의 문제이다. 학습에 사용하는 말뭉치를 아무리 늘리더라도 빈도수가 작은 단어들과 미등록어는 나타나게 마련이며, 이 경우 추정된 확률 값이 믿을 수 없게 된다. 또한, N의 크기를 늘릴수록 언어 모델의 사이즈가 커지게 되는 문제가 발생한다. 예로, 1만 어휘 규모의 트라이그램(trigram) 언어 모델의 경우, 최고 $10000^3 = 10^{12}$ 확률 값을 저장

해야 하며, 이는 태깅이나 음성 인식 시스템의 메모리와 속도에 큰 영향을 끼친다.

단어 클래스 기반의 언어 모델링은 이러한 두 가지 문제를 해결하기 위해 제안되었다. 예로, ‘하이트, 카스’ 등은 ‘맥주’의 클래스로 묶일 수 있으며, 만약 ‘카프리’가 학습 말뭉치에 등장하지 않았다 하더라도 P(마시다 | 카프리)의 추정에 P(마시다 | 맥주)와 같이 단어의 클래스를 이용할 수 있다. 따라서, 미등록어나 희귀한 단어에 대해 클래스를 대신 사용할 수 있으므로 자료 희귀의 문제를 해결할 수 있으며, 추정해야 할 파라미터의 수가 단어의 수(|V|)보다 훨씬 작은 클래스의 수(|C|)로 줄어들게 되므로 언어 모델 크기를 줄일 수 있다.

본 논문에서는 대량의 말뭉치를 바탕으로 한국어에 대해 단어 기반의 N-gram 언어 모델과 클래스 기반의 언어 모델을 구축하고, 이를 실험적으로 검증하는 것을 목표로 한다. 단어 기반의 모델링의 경우 현재 제시된 여러 스무딩(smoothing) 알고리즘에 대해 실험을 수행하였다. 또한, 클래스 기반의 언어 모델의 경우에는 기존의 품사 태그를 클래스로 사용한 경우와 말뭉치로부터 자동으로 구축된 클래스를 사용한 경우로 나누어 실험하였다. 마지막으로 단어 기반 모델과 클래스 기반 모델을 결합하여 각각의 모델과 그 성능을 비교한다.

¹ 본 논문에서는 언어 모델링에 초점을 맞추고 클래스와 대치되는 개념으로 단어라는 용어를 사용했으나 실제로는 한국어에 있어 형태소이다.

실험에 사용된 n-gram 언어 모델링과 자동 단어 클러스터링 알고리즘 및 평가 도구는 모두 직접 구현되었으며 추후 KLM Toolkit의 이름으로 공개 할 예정이다.

2. 이론적 배경

2.1 N-gram 언어 모델과 스무딩

단어 기반의 N-gram 언어 모델은 다음과 같이 이전의 N개의 단어를 기반으로 현재의 단어를 추정하는 것으로 정의된다.

$$P_w(W) = \prod_{i=1}^n p(w_i | w_{i-1} \dots w_{i-N+1})$$

이 때, N이 커질수록 발생하는 자료 희귀의 문제를 해결하기 위해 여러 가지 스무딩(smoothing) 기법들이 제안 되었다. 스무딩은 언어 모델의 성능에 미치는 영향은 막대하며, 비단 언어 모델링 뿐만 아니라 모든 통계적 모델링에 필수적으로 사용된다.

N-gram 언어 모델링의 경우 Good-turing discounting과 백오프(backoff) 방법을 결합한 Katz의 스무딩 방법이 널리 쓰이고 있다. Katz의 스무딩은 다음 수식과 같이 정의된다[1].

$$P_{Katz}(w_i | w_{i-1}) = \begin{cases} C(w_{i-1}w_i) / C(w_{i-1}) & \text{if } C(w_{i-1}w_i) > k \\ d_r C(w_{i-1}w_i) / C(w_{i-1}) & \text{if } k \geq C(w_{i-1}w_i) > 0 \\ \alpha(w_{i-1}) P_{Katz}(w_i) & \text{if } C(w_{i-1}w_i) = 0 \end{cases}$$

한편, 유니그램(unigram)의 확률에 이전 단어의 정보를 포함하여 스무딩을 수행하여 보다 일반화 시킨 Kneser-Ney의 방법이 Katz의 방법에 비해 좋은 성능을 보임이 실험으로 확인되었다[2].

$$P_{KN}(w_i | w_{i-1}) = \begin{cases} \frac{\max\{C(w_{i-1}w_i) - D, 0\}}{C(w_{i-1})} & \text{if } C(w_{i-1}w_i) > 0 \\ \alpha(w_{i-1}) P_{KN}(w_i) & \text{otherwise} \end{cases}$$

$$P_{KN}(w_i) = C(\bullet w_i) / \sum_w C(\bullet w_i), \# \text{ of uniq. words preceding } w_i$$

본 논문에서는 위의 두 가지 스무딩 기법을 한국어에 적용하여 그 효용성을 검증하는 실험을 수행한다. 한국어의 경우 단어 단위가 아닌 형태소 단위로 실험을 수행하며 앞으로 단어라 함은 형태소를 대치하는 개념으로 사용한다.

2.2 언어 모델의 평가

추정된 언어 모델의 평가는 잘 알려진 교차 엔트로피와 혼잡도(perplexity)를 사용한다. 교차 엔트로피는 학습된 통계 모델이 원래 내재된 언어 모델과 얼마나 가까운가의 통계적 척도이며, 혼잡도는 언어 모델을 통해 테스트 집합의 단어를 생성할 때 단어 분기의 정도를 의미한다.

$$H(W) = -\frac{1}{N_w} \log_2 P(W) = -\frac{1}{N} \sum_{i=1}^N \log_2 P(w_i | w_{i-1} \dots w_1)$$

$$PP(W) = 2^{H(W)} = \sqrt[N]{\prod_{i=1}^N \frac{1}{P(w_i | w_{i-1} \dots w_1)}}$$

대부분의 언어 모델 실험의 경우 혼잡도 뿐만 아니라 음성 인식 시스템에 적용하여 WER(Word Error Rate)를 계산하지만, 본 논문에서는 혼잡도만을 언어 모델 평가에 적용한다. 혼잡도의 감소가 반드시 음성 인식의 WER의 감소로 이어지는 것은 아니나, 혼잡도는 내재된 본래 언어 모델에의 근접도를 나타내므로 언어 모델 평가에 표준으로 사용된다.

2.3 클래스 기반의 언어 모델

클래스 기반의 언어 모델에서는 모델 추정에 단어 자체를 사용하는 것이 아니라 다음과 같이 단어가 속한 클래스를 사용한다[3]².

$$P_c(W) \cong \prod_{i=1}^n p(w_i | c_i) p(c_i | c_{i-N+1} \dots c_{i-1})$$

클래스 기반의 언어 모델은 단어 기반의 언어 모델에 비해 몇 가지 장점을 가진다. 첫째, 미등록어나 희귀한 단어에 대해 클래스를 대신 사용할 수 있으므로 자료 희귀의 문제를 해결하는 스무딩 방법의 하나로 볼 수 있다. 둘째, 추정해야 할 파라미터의 수가 단어의 수(|V|)보다 훨씬 작은 클래스의 수(|C|)로 줄어들게 되므로 단어 기반에 비해 언어 모델 크기를 줄일 수 있다. 마지막으로, 가능한 문맥의 종류가 작으므로 학습 말뭉치의 부족에 대처할 수 있다.

이 때, 단어를 클래스로 묶는 방법은 사람이 언어 정보를 바탕으로 미리 나누어 놓은 범주를 이용할 수도 있고, 대량의 말뭉치에서 공기 정보를 바탕으로 자동화 해서 클러스터링 할 수 있다.

2.3.1 클래스의 수동 구축 - 품사 기반

단어를 어떠한 클래스로 묶는데 있어 우리가 기존에 사용하고 있는 문법이나 의미정보를 사용할 수 있다. 가장 대표적으로 품사(POS) 태그를 들 수 있으며, 문법 정보를 언어 모델에 응용할 수 있다는 장점이 있다. 다음과 같은 수식을 통해 언어 모델링을 수행한다.

$$P_t(W) \cong \prod_{i=1}^n p(w_i | t_i) p(t_i | t_{i-N+1} \dots t_{i-1})$$

품사 기반의 모델을 자동 품사 태깅에 응용하면 90% 이상의 정확률을 얻을 수 있음이 보고되어 있으나, 음성인식의 경우에는 단어 기반의 언어 모델에

² 클래스 기반의 언어 모델의 식은 Predictive 방식, Index 방식 등 여러 형태가 제안되었으나, IBM에서 제안된 위의 식이 좋은 성능을 보이며 널리 사용된다.

비해 오히려 10% 정도 혼잡도가 증가하는 결과를 보인다[4]. 이는 품사 태그가 단어의 어휘 정보를 너무 추상화 시키기 때문인데, 특히 명사의 대해서는 분류기준이 작기 때문에 의미적 차이로 생기는 문제에 대해서는 해결책이 될 수 없다.

본 논문에서는 55개의 태그 집합으로 세분화 된 KAIST 태그 셋과 품사 태깅 말뭉치를 바탕으로 혼잡도의 증감을 실험한다[5].

2.3.2 클래스의 자동 구축 \ 단어 클러스터링

단어를 어떠한 클래스로 묶는 또 하나의 방법은 대량의 말뭉치에서 단어간의 공기 정보(Mutual information)를 통해 자동적으로 클러스터링 하는 것이다.

단어 클러스터링 알고리즘에 대해서는 많은 연구가 있었다. 모든 단어를 하나의 클래스로 보고 가장 유사한 단어끼리 계층적으로 묶어나가는 상향식(Bottom-up) 방식이 처음 제시되었으며(Brown,92), 이때 국소 최적(local optima) 문제를 해결하기 위해 simulated annealing 방법이 사용되었다. 또한, 하나의 클래스로부터 단계적으로 분리해 나가는 하향식 방식도 제안되었으나, 실제 클러스터링 방법의 차이에 따른 언어 모델 성능의 차이는 미미하다[6].

자동 단어 클러스터링의 문제는 알고리즘의 방법론보다는 시간 복잡도의 문제이다. 가장 기본적인 Brown의 상향식 방법은 $O(V^3)$ 의 복잡도를 가지므로, 단어의 수가 늘어날수록 학습에 걸리는 시간은 기하급수적으로 늘어난다³. 본 논문에서는 상향식 방법을 시간복잡도 면에서 개선한 Martin의 exchange 알고리즘을 사용한다. 개략적인 알고리즘은 다음과 같으며, 복잡도는 $O(V * C^2)$ 이다.

```

start randomly mapping w → cw
for each w of the vocabulary do
  for each class ck do
    move w to ck and compute perplexity
  move w to ck with minimum perplexity
do until stopping criterion is met
  
```

2.4 단어 기반 모델과 클래스 기반 모델의 결합

클래스 기반의 언어 모델은 단어 기반의 모델에 비해 모델이 간결하고 새로운 단어에 대해서 좋은 성능을 보일 수 있다. 그러나, 대용량의 말뭉치를 사용할 경우에는 세부 어휘의 정보를 살리지 못해, 단어 기반의 모델에 비해 성능이 오히려 떨어짐이 보고되어 있다[7].

이에, 단어 기반 모델의 정확성과 클래스 기반 모델

³ 본 논문의 실험에 의하면 100만 형태소를 256개의 클래스로 클러스터링 할 때 1회의 반복이 6시간 소요되었다(Pentium III 700/128MB/Linux).

1. N-gram counting

2. Smoothing
 - A. Discounting
 - I. Good-turing / Absolute / Written-bell
 - B. Backoff
 - I. Katz's backoff
 - II. Kneser-Ney backoff
 - C. Interpolation

3. Class-based Modeling
 - A. Automatic Word Clustering
 - I. Bottom-up
 - II. Exchange

4. Evaluation
 - A. Perplexity / Entropy

표 1. KLM Toolkit의 기능

의 범용성을 함께 살리기 위해 두 모델을 결합하는 방식이 많이 사용된다[8]. 본 논문에서는 여러 개의 통계 모델을 결합할 수 있는 방법으로 보간법(Interpolation)을 사용하고, 모델별 가중치를 달리하여 실험한다.

$$P_{INT}(W) = r_w P_w(W) + r_c P_c(W)$$

3. KLM Toolkit

실험에 사용된 n-gram 언어 모델링과 자동 단어 클러스터링 알고리즘 및 평가 도구는 모두 직접 구현되었으며 <표 1>과 같은 기능을 제공한다.

현재는 언어 독립적으로 제작되어 한국어의 특성을 반영하지 못하는 단점이 있다. 한국어의 특성상 언어 정보는 두 어절에 걸쳐 나타나며, 앞 어절의 실질어와 뒤 어절의 실질어, 앞 어절의 기능어와 뒤 어절의 기능어를 중심으로 분포한다. 예를 들어, “미소/nc+ 를/j 짓/pv+ 다/ef”에서 (미소/nc, 짓/pv)는 높은 연관성을 가지며, 기존의 바이그램 모델로는 표현이 어렵다. 따라서, 향후에는 한국어의 특성을 반영하여, skip n-gram 및 원거리의 n-gram 모델링의 기능을 추가할 예정이다.

4. 실험

4.1 실험 준비

본 논문에서는 KAIST 품사 태깅 말뭉치를 바탕으로 한국어에 대해 단어 기반의 n-gram 언어 모델과 클래스 기반의 언어 모델을 구축하고, 이를 실험했다. 먼저, 말뭉치가 특정 도메인에 국한되지 않도록 2000만 어절 규모의 KAIST 품사 태깅 말뭉치에서 신문, 방송, 소설, 경제 등 7개 분야에서 걸쳐 100만 어절 규모의 균형 코퍼스를 구축하였다. 그 후, 정확

성을 위해 두 번에 걸쳐 수동으로 오류 제거 작업을 수행하여 '100만 어절 품질관리 코퍼스'를 제작하였고 이를 실험에 사용하였다. 말뭉치는 다시 80:20 규모로 나누어 80을 학습에 사용하고, 20을 테스트에 사용하는 5-fold 교차 실험을 수행했다. <표 2>에 실험집합의 어절 수와 형태소 수를 나타낸다.

말뭉치	100만 어절 품질관리 코퍼스	
문장수	어절수	형태소수
72,659	1,021,508	1,883,330

표 2. 실험집합

4.2 실험 결과

4.2.1 단어 기반의 언어 모델링

가장 먼저 2.1절에서 소개한 Katz의 스무딩과 Kneser-ney의 스무딩 기법을 비교 실험했다. 바이그램과 트라이그램에 대해 스무딩을 하지 않을 때와 각각의 스무딩 기법을 적용했을 때 실험한 결과는 <표 3>과 같다.

실험결과 바이그램에 비해 트라이그램 모델이 월등한 성능을 보이며, 스무딩을 사용하지 않을 때보다 스무딩을 사용할 때 200%이상 복잡도가 감소한 것으로 나타났다. 가장 성능이 좋은 모델은 트라이그램 모델의 Kneser-ney 스무딩으로 가장 널리 사용되는 Katz의 스무딩 방법에 비해 상대적으로 약 8% 정도 복잡도가 감소했다.

스무딩 방법		복잡도(Perplexity,5-fold)	
		문장복잡도	단어복잡도
Bigram (n=2)	no-smooth	1163.24	1531.04
	Katz	291.36	363.37
	Kneser-ney	294.27	367.14
Trigram (n=3)	no-smooth	214.39	264.21
	Katz	90.05	107.28
	Kneser-ney	82.46	97.91

표 3. 단어 기반 언어 모델의 실험 결과

4.2.2 클래스 기반의 언어 모델링

단어 모델에 대한 대안으로 2.1절에서 소개한 클래스 기반의 언어 모델에 대한 실험 결과는 <표 4>와 같다. 수동 구축된 클래스인 품사와 클러스터링 알고리즘을 통해 자동으로 구축된 클래스의 경우를 나누어서 실험했다. 클래스의 자동구축의 경우 클래스의 크기에 따라 학습 시간이 늘어나는 문제가 있어 성능이 다시 감소하는 1024개의 클래스까지만 실험을 수행했다.

실험결과 수동 구축된 품사 기반의 클래스 언어 모델 보다는 자동 학습된 클래스에 기반한 모델이 클래스의 수에 상관없이 더 좋은 성능을 보임을 확인할 수 있다. 또한, 클래스의 수가 512개일 때 가장 좋은 성능을 보이며, 그보다 작거나 더 큰 경우에는 복잡도가 오히려 증가한다.

클래스 기반 모델은 바이그램 단어 모델에 비해서는 성능이 더 좋으나, 트라이그램 기반의 단어 모델에 비해서는 성능이 좋지 않았다. 이는, 트라이그램 이상의 단어 자체에 포함된 정보가 많다는 뜻으로 해석할 수 있다. 하지만, 언어 모델의 크기에 있어서는 단어 모델에 비해 그 크기가 1/3수준으로 감소한다(트라이그램 스무딩 모델: 12MB, 512 클래스 모델: 4M)

클래스 방법		복잡도(Perplexity,5-fold)	
IVI = 23,000		문장복잡도	단어복잡도
품사기반		257.20	318.37
자동학습	ICI = 64	186.34	227.81
	ICI = 256	140.99	170.54
	ICI = 512	137.47	166.12
	ICI = 1024	140.13	169.46

표 3. 클래스 기반 언어 모델의 실험 결과

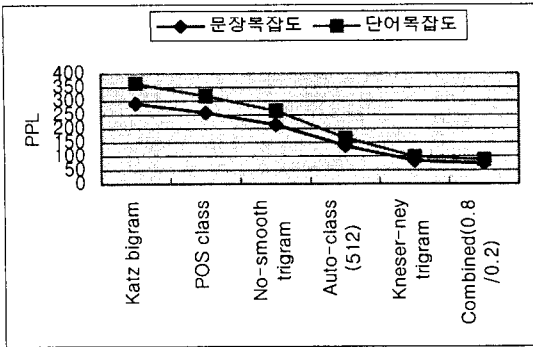
4.2.3 단어 모델과 클래스 모델의 결합

단어 기반 모델의 정확성과 클래스 기반 모델의 범용성을 함께 살리기 위해 두 모델을 보간법(Interpolation)으로 결합한 실험 결과를 <표 4>에 나타낸다.

실험 결과 결합 모델이 각각의 언어 모델에 비해 우수한 성능을 나타내었다. 단어 모델과 클래스 모델에 대해 각각 11% 및 47%의 복잡도가 감소했다. 따라서, 단어 모델과 클래스 모델의 결합의 효율성이 검증되었다.

가중치		복잡도(Perplexity,5-fold)	
단어모델	클래스모델	문장복잡도	단어복잡도
0.2	0.8	85.98	102.04
0.4	0.6	77.25	91.31
0.6	0.4	73.75	87.01
0.8	0.2	73.71	86.97
0.9	0.1	75.58	89.25

표 4. 결합 모델의 실험 결과



5. 결론

통계적 언어 모델은 대어휘 음성인식에 필수적으로 사용된다. 영어의 대어휘 음성인식 시스템은 1990년대 중반 Philips, IBM, Dragon을 시작으로 2001년 Microsoft Office XP의 음성 인식 시스템에 이르기까지 꾸준히 보완 발전해 왔다. 이러한 시스템들은 5~10만 단어 이상의 통계적 언어 모델을 포함하고 있으며, 대용량의 말뭉치(Brown, WSJ)를 바탕으로 10년 이상 언어 모델의 성능 향상에 대한 연구가 이루어져 왔다. 한국어의 경우 서강대와 ETRI에서 신문 및 방송 뉴스를 대상으로 언어 모델을 구축하고 대어휘 음성 인식에 적용하고 있으나 각각의 언어모델은 Katz의 스무딩 방식의 트라이그램 언어 모델이다 [9][10].

본 논문에서는 대용량의 한국어 품사 태깅 말뭉치를 바탕으로 통계적 언어 모델의 기본이 되는 단어(형태소) 기반 모델과 클래스 기반의 모델을 구축하고 그 성능을 평가했다. 언어 모델링의 가장 기본이 되는 n-gram 단어 모델링과 여러 스무딩 기법들을 실험했으며, 더 나아가 단어 클래스 기반의 언어 모델에 대해서도 실험을 수행했다.

실험 결과(<그림 1>) 단어 모델의 스무딩 방법은 이제까지의 Katz의 스무딩보다 Kneser-ney의 스무딩 방법이 우수함이 검증되었다. 클래스 기반의 모델의 경우 품사 기반의 방법보다 자동 구축된 단어 클래스를 이용하는 방법의 성능이 더 좋았다. 가장 좋은 성능을 나타내는 경우는 단어기반의 언어 모델과 클래스 기반의 모델이 결합된 경우였다. 이 연구 결과를 기존의 대어휘 음성인식 시스템에 적용할 경우 좋은 성능이 기대된다.

본 논문의 모든 알고리즘은 KLM Toolkit이라는 이름으로 개발하여 공개하였다[11]. 하지만, 이는 언어를 컴퓨터로 모델링 하려는 일련의 노력의 일부분에 불과하다. 한국어 어절 특성을 반영한 노력(skip n-gram), 말뭉치의 분야가 달라지는 경우에 대한 해결책(topic adaptation), 문법정보를 고려한 언어 모델(dependency grammar), 문맥 및 화용 정보까지 포괄할 수 있는 기법(whole-sentence model) 등 보다 많은 연구가 지속적으로 필요하다.

6. 참고 문헌

- [1] S. M. Katz. "Estimation of probabilities from sparse data for the language model component of a speech recognizer," Proc. ICASSP 1987, ASSP-35(3):400-401 (1987).
- [2] S. Chen; J. Goodman. "An empirical study of smoothing techniques for language modeling," TR-10-98, Harvard University (1998).
- [3] P.F. Brown, et al. "Class-based n-gram models of natural language," Computational Linguistics, 18(4):467-479 (1992)
- [4] T.R.Niesler; P.C.Woodland. "Comparison of POS and automatically deried category-based language models for speech recognition", Proc. ICASSP 1998, Vol. 1, 2003: 177-180 (1998)
- [5] "코퍼스 품사 태깅 매뉴얼" 한국과학기술원 TR, <http://morph.kaist.ac.kr/~morph/>
- [6] J. Goodman. "A bit progress in language modeling," MSR-TR-2001-72, Microsoft Research (2001).
- [7] T.R.Niesler; P.C.Woodland. "Combination of word-based and category-based language models," In Procs. ICSLP 96, 220-223 (1996)
- [8] C. Samuelsson; W. Reichl. "A class-based language model for LVCSR extracted from POS statistics," Proc. ICASSP 1999, paper no. 1781 (1999)
- [9] Oh-Wook Kwon, K. Hwang, J. Park. "Korean large vocabulary continuous speech recognition using pseudomorpheme units" Proc. EUROSPEECH 99, 483-486, (1999)
- [10] 신영숙, "한국어 대어휘 연속음성 인식기의 성능 향상을 위한 토픽 클러스터링 기반의 언어모델" 석사학위 논문. 서강대학교. (2000)
- [11] KLM Toolkit, <http://morph.kaist.ac.kr/~gykim>