1998년도 한국음향학회 학술발표대회 논문집 제 17 권 제 1(s) 호

제한된 단어를 갖는 우리말 연속 음성 인식

김석동[0*], 조영훈[*], 송도선[**], 이행세[***]
*호서대학교 컴퓨터학부, **중경대학 전산기 공학과
***아주대학교 전자공학과

The Continuous Speech Recognition with Limited word

Suk-dong Kim[0*], Young-hoon Cho[*], Do-sun Song[**], Haing-sei Lee[***]
*Computer Science, Hoseo University,
**Computer Eng. ChungKyung University
***Electronics Eng. Ajou University

sdkim@dogsuri.hoseo.ac.kr, yhcho@dogsuri.hoseo.ac.kr,

요 약

이 논문에서 우리는 대규모 어휘를 갖는 연속 음성 인식을 위한 방법을 제시한다. 우리말은 영어와 구조적으로 달라서 대용량 어휘를 갖는 연속 음성을 인식하기 위한 언어모델을 만들기가 매우 어렵다. 언어 모델을 우리말 문장에 적용하기 위해 신문의 사설을 3-gram을 이용하여 처리하였다. 우리의 인식 시스템을 평가하기 위하여 시스템 공학 연구소에서 제공한 낭독 음성을 대상으로 인식률을 계산하였다. 589개의 문장을 대상으로 총 20명이 발음한 3,156개의 문장에 대하여 남자 92.2%, 여자 87.9%의 인식률을 얻었다. 발음사전은 낭독음성과 신문 사설에서 추출한 10K의 크기이며 uniphone의 음성모델을 사용하였다.

1. 서론

음성에 대한 연구[1]는 음성모델과 언어모델에 매우 밀접한 관계를 가진다. 1980년대 후반부터 시작된 연속 음성에 대한 본격적인 연구는 1000개의 단어에서[2]부터 1995년도의 라디오 방송의 음성을 문자로 변환시키는 연구[3] 같은 무제한 단어까지 하고 있다. 처리 방법이 복잡해지면 단어인식률은 증가되지만 요구되는 자원도 같이 증가된다. 더구나 대규모 어휘를 가진 연속음성의 인식은 상당히 어렵다. 그 이유는 첫째 단어 구간이 불확실하다. 현재순간이 단어 구간내 인지 아닌지 끊임없이 검사를 해야한다. 더구나 많은 잘못 가정된 단어가 종종 만들어진다. 그래서 의미 정보나 단어 문맥을 제공하는 복잡한 언어 모델이 여러 개의 가정 중에서 혼동되지 않는 것을 선택하는데 필요하다. 둘째 조음효과가 매우 강해서 어느 한순간의 음성은 앞뒤의 음성에 영향을 많이 받는 경우이다. 인식속도, 필요한 기억 장치의 자원과 인식률과 같은 세 가지 종류의 성능은 상호 충돌적이다. 예를 들면 탐색 공간을 줄여 인식속도를 줄이고 간단한 음성과 언어 모델을 이용해 기억용량을 줄이면 인식률은 다소 떨어진다. 그러나 높은 인식률을 유지하면서 인식속도를 증가시키고 기억 용량을 줄이는 문제는 상당히 힘들다. 수십 개의 단어를 갖는 작은 규모 시스템에서는 전 단어에 대한 모델을 만들 수 있으나 단어의 크기가 증가하면 이것은 쉽지 않다. 모든 단어모델을 만드는 충분한 학습 데이터를 얻기가 어렵다. 그래서 작은 단어의 형태로 단어를 표현하여 하면 새로운 단어에 대해서도 이미 학습된 작은 단어의 조합으로 처리할 수 있게 된다. 자연스러운 연속적인 음성에는 강한 조음 효과가 들어있다. 음소는 입이나 코에서 여러 조음 조직(혀나 입술같은)의 위치에 따라 만들어진다. 이들 조음 조직은 음성이 만들어질 때 서로 다른 음성사이를 부드럽게 이동하기 때문에 각각의 음소는 이웃한 음소의 영향을 받으며 특히 하나의 음소에서 다른 음소로 변화하는 동안에는 더욱 영향을 받게된다. 작은 규모의 단어에서는 큰 문제가 없지만 단어의 수가 커지고 복잡성이 증가하면 문제가 된다.

언어 모델을 우리말 문장에 적용하기 위해 인터넷에 올라와 있는 on-line 신문의 사설 10개월 분량을 기본 언어 모델로 구성하고 인식 대상의 문장과 함께 N-gram중의 하나인 3-gram을 이용하여 언어 모델을

구축하였다. 우리의 인식 시스템을 평가하기 위하여 시스템 공학 연구소에서 제공한 낭독 음성을 대상으로 인식률을 계산하였다. 총 589개의 서로 다른 문장을 20명이 부분적으로 발음한 3,156개의 문장에 대하여 남자 92.2%, 여자 87.9%, 평균 인식률이 90%를 얻었다. 발음사전은 낭독음성과 신문 사설에서 추출한 10K의 크기이며 uniphone의 음성모델을 사용하였다.

2. 연속 음성 인식

그림1에 연속 음성 인식에 사용하는 기본 개념을 나타내었다. 발음한 음성, W = "이것은 음성 입니다"라고 할때 언어모델에서 이 단어들이 나타날 확률인 $P(W)$를 계산한다. 다음 순서는 각 단어를 발음사전을 이용하여 음소들의 조합 형태로 바꾼다. 각 음소는 hidden Markov model(HMM)이라 하는 통계 모델에 대응된다. 전체의 문장을 HMM의 나열로 나타내어 하나의 복합 모델을 만들고 이 모델이 관찰되는 열인 Y를 만들어 낼 확률을 구한다. 이것이 요구되는 확률 $P(Y|W)$ 이 된다. 이러한 처리 과정은 모든 가능한 단어의 열에 적용하여 가장 비슷한 열을 인식 결과로 선택한다. 음성은 단어들로 나열(sequence of words)로 되어있다. 나열된 단어들이 $W = w_1, w_2, \cdots, w_n$ 인때, 관찰된 음성 Y 에 대하여 가장 그럴듯한 단어의 열 W^*을 찾는 것이 우리의 목표이다. 이를 위해 Bayes 규칙을 이용하면 요구되는 확률 $P(W|Y)$을 구할 수 있다. 즉

$$W^* = \arg\max_W P(W|Y) = \arg\max_W \frac{P(W)P(Y|W)}{P(Y)} \quad (2.1)$$

이 방정식은 가장 그럴듯한 단어의 열, W^*을 찾기위해서 $P(W)$와 $P(Y|W)$의 곱이 최대가 되는 단어의 열을 찾아야 된다는 것을 보여주고 있다. 첫째항은 관찰된 신호와는 무관한 W 가 관찰될 priori확률인데 이것은 언어 모델에서 계산되며 둘째항은 주어진 단어 열에 대하여 음성 백터의 열 Y 가 관찰될 확률로 이것은 음성 모델에서 계산된다.

2.1 음성 모델

음성 모델의 목적은 주어진 단어 W 에 대하여 임의의 벡터 열 Y의 비슷한 정도를 계산하는 것이다. 원칙적으로 확률분포는 각각의 W 와 그에 대한 백터 열로 계산할 수 있으나 대용량의 어휘시스템에는 불가능하므로 대신에 단어의 열은 음소라 불리는 기본 음성으로 나누어 처리하며 각 음소는 HMM으로 나타낸다.

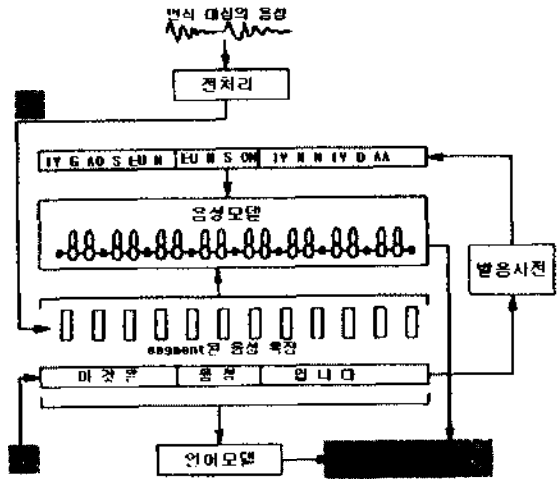

그림 1 통계적인 음성 인식 방법

2.2. 언어 모델

언어 모델은 단어의 열에 대하여 priori 확률로 정의된다. 문장(즉 단어의 열인 w1, w2,...wn)에 대한 언어 모델은

$$P(w_1)P(w_2|w_1)P(w_3|w_1w_2)P(w_4|w_1w_2w_3)\ldots$$

$$P(w_n|w_1w_2,\ldots,w_{n-1}) = \prod_{i=1}^{n} P(w_i|w_1,\ldots,w_{i-1})$$

로 주어진다. 수식 $P(w_i|w_1,\ldots,w_{i-1})$에서 $w_1,\ldots,w_{i-1}$는 과거 단어(word history)혹은 간단히 Wi의 과거라 한다. 실제로 주어진 임의의 길이를 가진 과거의 추정을 신뢰성 있는 확률로 구하기는 어렵다. 왜냐하면 상당히 많은 학습 자료가 있어야 하기 때문이다. 그래서 대신에 다음과 같은 근사화의 방법들을 이용하는 것이 보통이다.

* 문맥 자유 문법이나 정규 문법 : 이 언어 모델은 문법적으로 잘 만들어진 문장이나 구에 이용된다. 미리 만들어진 구조 이외의 다른 변형은 허용하지 않는다. 이러한 형식적인 문법은 너무 제한적이기 때문에 대규모 어휘 시스템에는 사용하지 않는다.

* 단어에 대한 unigram, bigram, trigram문법 : 이것은 다음과 같이 정의 된다.

$P(w)$ 단어 w의 확률

$P(w_j|w_i)$ 단어 wi다음에 단어 wj가 나올 확률

$P(w_k|w_i, w_j)$ 과거에 w_i, w_j 가 나온다음에 단어 w_k가 나올 확률 Bigram문법은 모든 가능한 단어 쌍에 대한 확률을 가질 필요는 없다. 실제로 우리는 소규모 단어만을 가지고 있다. 대신에 가장 자주 발생하는 bigram만을 저장하고 backoff방법을 이용한다. 이것은

원하는 bigram을 발견하지 못할 때 unigram으로 되돌아 가는 방법이다. 달리말하면 만약 $P(w_j|w_i)$를 발견하지 못하면 $P(w_j)$로 대신한다. 그러나 w_j는 w_i의 다음 단어가 아니라는 것을 나타내기 위해 backoff weight를 적용한다. 다른 high-order backoff n-gram도 위와 비슷하게 한다.

* Class n-gram 문법 : 이것은 n-gram 문법과 비슷하나 token이 숫자, 달 이름, 특정한 이름등 모든 단어 class를 대상으로 한다는 것이 다르다. class 문법을 만들고 이용하기 위해서는 단어들이 각 class에 여러번 나타날 수 있으므로 여러 기법을 동원해야 한다.

* 먼 거리 문법 : n-gram언어 모델과 달리 이것은 어떤 거리만큼 떨어진 단어들과 관계를 정한다. 예를들어 trigger-pair 구조가 하나의 예이다. 이문법은 과거의 decoding에서 n-best 가정을 다시 계산 할 때 사용한다.

간단하면서 효율적인 방법인 N-grams(bigram, trigram)은 w_k는 앞의 n-1개의 단어에 관계가 있다고 가정하는 것이다. 즉

$$p(w_k|W_1^{k-1}) = p(w_k|W_{k-n+1}^{k-1})$$

더구나 N-grams확률 분포는 text data에서 바로 구할 수 있으므로 명확한 언어 규칙이 필요없다. 이론적으로 N-grams은 간단히 빈도를 세어 추정할 수 있고 look-up table에 저장할 수 있다. 예를들어 trigram(N=3)인 경우

$$P(w_k|w_{k-1}, w_{k-2}) = \frac{t(w_{k-2}, w_{k-1}, w_k)}{b(w_{k-2}, w_{k-1})} \quad (3.1)$$

여기서 t(a,b,c)는 학습 data에 있는 trigram a,b,c가 나타나는 빈도수이고 b(a,b)는 bigram a,b가 나타나는 빈도수이다. 물론 문제는 V개의 단어에 대하여 V3 개의 trigram이 있는 것이다. 보편적인 단어 10,000개의 경우에도 많은 trigram이 존재한다.많은 trigram이 학습자료에서는 나타나지 않고 다른 것들도 한두번정도밖에 나타나지 않는다. 그러므로 식(3.1)에 의해 추정되는 값은 너무 부족하다.

학습자료의 부족에 대한 해결책은 discounting과 backing-off의 조합을 이용하는 것이다. discounting은 많은 빈도가 발생하는 trigram에 대하여 그 빈도수를 줄여 다시 추정하는 방법이고, Backing-off는 추정하기에 너무 적은 trigram에 대하여 scaled bigram확률로 대체하여 추정하는 방법이다.

$$P(w_k|w_{k-1}, w_{k-2}) = B(w_{k-1}, w_{k-2})P(w_k|w_{k-1})$$

여기서 B는 $P(w_k|w_{k-1}, w_{k-2})$을 적당히 정규화하게 하는 back-off 함수이다.

2.3. Decoding

Decoding은 탐색의 문제이다. 커다란 탐색 공간에서 가장 최적의 경로를 발견하는 것이다. 가장 일반적인 방법에는 깊이-우선(depth-first) 탐색과 너비-우선(Breath-first) 탐색이 있다[4]. 너비-우선 방법은 탐색이 동시에 여러개가 병렬로 이루어지는 것으로 Bellman의 최적이론에서 나온 것으로 Viterbi decoding[5]이 있다. 이 방법은 동적 프로그램을 이용한 알고리즘으로 입력 음성에 가장 가까운 상태 열을 탐색 공간에서 찾는 것이다. 탐색공간은 음소로부터 만들여진 단어 HMM 모델이 생성되면서 구축되고 모든 단어 HMM 모델은 병렬로 탐색된다. 탐색공간은 중간 규모의 단어에 대해서도 매우 커지므로 비슷하지 않은 상태는 고려하지 않는 방법인 beam 탐색[6]으로 제한된 탐색을 하는 것이 보통이다. 이러한 결합을 간단히 Viterbi beam 탐색[7]이라 한다. 이방법은 한 순간에 하나의 프레임만을 처리하는 시간에 따른 동기적인 탐색으로 다음 프레임으로 이동하기 전에 그 프레임에 대하여 모든 상태를 계산한다. 깊이-우선 방법은 음성의 끝에 도달할 때까지 대기하고 있다가 마지막에 최종 결정하는 것으로 A*알고리즘[4]의 변형인 stack decoding[8]이 있다.

3. 실험및 결과

3.1 신호 처리

음성을 16 KHz, 16-bit로 sampling하고 이것을 12개의 mel-scale 주파수 캡스트럼벡터와 하나의 power 계수를 매 10 msec 프레임 마다 구한다. 시간 t에서의 캡스트럼 벡터를 x(t)로 (즉 개별적인 요소는 xk(t), 1 <= k <=12). power계수는 간단히 x0(t)로 나타낸다. 우선 이 캡스트럼 벡터와 power를 정규화시키고 1차와 2차 미분을 하여 각 프레임마다 4가지 종류의 특징 벡터들을 구한다.

$\boldsymbol{x}(t)$ = 정규화된 캡스트럼 벡터

$$\Delta\boldsymbol{x}(t) = \boldsymbol{x}(t+2) - \boldsymbol{x}(t-2), \quad \Delta_1\boldsymbol{x}(t) = \boldsymbol{x}(t+4) - \boldsymbol{x}(t-4)$$

$$\Delta\Delta\boldsymbol{x}(t) = \Delta\boldsymbol{x}(t+1) - \Delta\boldsymbol{x}(t-1)$$

$$\boldsymbol{x}_0(t) = x_{0(t)},$$

$$\Delta x_0(t) = x_0(t+2) - x_0(t-2),$$

$$\Delta\Delta x_0(t) = \Delta x_0(t+1) - \Delta x_0(t-1)$$

그러므로 각 프레임마다 4종류의 특징 벡터 51개(12개, 24개,12개,3개)를 사용하였다.

3.2 언어 모델의 구조

모든 확률과 back-off 값 (bo_wt) 는 log10 형태로 계산을 하였고 각각의 N-gram형태는 다음과 같다.

1-grams: p_1 wd_1 bo_wt_1
2-grams: p_2 wd_1 wd_2 bo_wt_2
3-grams: p_3 wd_1 wd_2 wd_3

여기서 p_i는 i-gram의 확률을 나타내며 bo_wt_i는 i-gram의 back-off 값이다.

trigram과 bigram을 이용한 확률값의 계산은 각각 다음과 같이 수행하였다.

```
p(wd3|wd1,wd2)= if(trigram exists)
                  p_3(wd1,wd2,wd3)
                else if(bigram w1,w2 exists)
                  bo_wt_2(w1,w2)*p(wd3|wd2)
                else  p(wd3|w2)
```

그림 3.2 trigram 언어 모델의 계산

```
p(wd2|wd1)= if(bigram exists)
                  p_2(wd1,wd2)
          else    bo_wt_1(wd1)*p_1(wd2)
```

그림 3.3 bigram 언어 모델의 계산

3.4 인식 결과

학습에 사용한 음성은 162명이(남자:92명, 여자:70명) 각기 30분에서 2시간 정도로 발음한 것으로 읽은 자료는 우리말의 음소가 모두 들어있는 음성이 되도록 한국내의 신문사 Web site에서 무작위로 발췌하였다. 각 자료는 그 내용에 따라 여러개의 그룹으로 나누었고 한사람이 다른 그룹의 자료를 여러번 발음하기도 하여 총 자료의 수는 243개(남자: 156개, 여자:87)의 집합이었다. 우리말 음성 모델을 만들기 위한 학습 시간은 한 iteration당 약 20시간 정도였고 학습에 사용한 컴퓨터는 여러종류로 네트워크를 이용해 처리하였다.

언어 모델을 만들기 위한 기초 자료는 인터넷을 통해 우리나라 신문사의 자료를 수집하였다. 우선 기본 언어 모델을 만들기 위해 일반적인 단어가 들어있는 신문기사중 10개월분량의 사설을 수집하여 처리하였다. 최종 언어모델을 구축하기 위해서 인식 대상의 문장도 같이 포함하여 총 1,177개의 문장을 대상으로 언어 모델을 만들었다.

인식에 사용한 음성은 시스템 공학 연구소에서 제공한 낭독음성(PBS)을 대상으로 하였다. 문장의 종류는 589개로 이중에서 50문장은 공동 집합으로 하고 나머지 539개의 문장을 5개의 개별 집합으로 나누어 인식 실험을 하였다. 인식에 참여한 사람들은 남,여 각각 10명씩 총 20명이다. 아래 표에 성별 인식률을 나타내었다.

성별 인식률		전체 인식률
남자	여자	
92.2%	87.9%	90.0%

4. 결론

본 연구는 우리말 연속음성 인식 결과 90%의 비교적 높은 인식률을 얻었다. 비록 음질이 양호한 음성을 대상으로 한 결과이지만 앞으로 보통 환경에서도 높은 인식률을 얻기위한 다각적인 연구가 필요하다.

참고 문헌

[1] Rabiner, L.R. "Applications of Voice Processing to Telecommunications." In Pro ceedings of the IEEE, Vol. 82, No. 2, Feb. 1994, pp. 199-228.

[2] Price, P ., Fisher, W.M., Bernstein, J. and Pallet, D.S. "The DARPA 1000-Word R esource Management Database for Continuous Speech Recognition." In IEEE In ternational Conference on Acoustics, Speech, and Signal Processing, 1988.

[3] John S. Garofolo, Jonathan G. Fiscus,William M. Fisher "Design and prtparation of the 1996 HUB-4 Broadcast News Benchmark Test Corpora." DARPA Speech Recognition Workshop, Feb. 1997, pp. 15 - 21.

[4] Nilsson, N.J. "Problem Solving Methods in Articial Intelligence." McGraw-Hill, New York, 1971.

[5] Viterbi, A.J. "Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm." In IEEE Transactions on Information Theory , vol. IT-13, Apr. 1967, pp. 260-269.

[6] Lowerre, B. "The Harpy Speech Understanding System." Ph.D. thesis, Computer Science Department, Carnegie Mellon University , Apr 1976.

[7] R.Haeb-Umbach, H.Ney. "Improvements in Time-synchronous Beam Search for 10000-Word Continuous Speech Recognition." IEEE Trans Speech and Audio Processing, Vol 2, 1994, pp. 353-356.

[8] Bahl, L.R., Jelinek, F. and Mercer, R. "A Maximum Likelihood Approach to Continuous Speech Recognition." In IEEE Transactions on Pattern Analysis and Mac hine Intelligence, Vol. PAMI-5, No. 2, Mar. 1983, pp. 179-190.