한국어 방송 음성 인식에 관한 연구

A Study on the Korean Broadcasting Speech Recognition

김 석 동*, 송 도 선**, 이 행 세***

(Suk Dong Kim*, Do Sun Song*, Heing Sei Lee*)

요 약

이 논문은 한국 방송 음성 인식에 관한 연구 이다. 여기서 우리는 대규모 어휘를 갖는 연속 음성 인식을 위한 방법을 제시한다. 주요 관점은 언어 모델과 탐색 방법이다. 사용된 음성 모델은 기본음소 Semi-continuous HMM 이고 언어 모델은 N-gram 방법이다. 탐색 방법은 음성과 언어 정보를 최대한 활용하기 위해 3단계의 방법을 사용하였다. 첫째로, 단어의 끝 부분과 그에 관련된 정보를 만들기 위한 순방향 Viterbi Beam탐색을 하였으며, 둘째로 단어의 시작 부분과 그에 관련된 정보를 만드는 역방향 Viterbi Beam탐색, 그리고 마지막으로 이들 두 결과와 확률적인 언어 모델을 결합하여 최종 인식결과를 얻기 위해 A* 탐색을 한다. 이 방법을 사용하여 12,000개의 단어에 대한 화자 독립으로 최고 96.0%의 단어 인식률과 99.2%의 음절 인식률을 얻었다.

ABSTRACT

This paper is a study on the korean broadcasting speech recognition. Here we present the methods for the large vocabuary continuous speech recognition. Our main concerns are the language modeling and the search algorithm. The used acoustic model is the uni-phone semi-continuous hidden markov model and the used linguistic model is the N-gram model. The search algorithm consist of three phases in order to utilize all available acoustic and linguistic information. First, we use the forward Viterbi beam search to find word end frames and to estimate related scores. Second, we use the backword Viterbi beam search to find word begin frames and to estimate related scores. Finally, we use A* search to combine the above two results with the N-grams language model and to get recognition results. Using these methods maximum 96.0% word recognition rate and 99.2% syllable recognition rate are achieved for the speaker-independent continuous speech recognition problem with about 12,000 vocabulary size.

I. 서 론

대규모 어휘 연속음성인식 시스템은 수만 단어의 어휘와 임의의 길이를 갖는 문장과 자연스러운 형태의 발성음을 다룬다. 이것은 인간의 음성 처리 능력과 비슷하나 구현하기에는 어려운 점이 많다. 실험실 수준일지라도 대규모 연속음성을 처리하기에 음성 인식기술은 아직도 속도가 느리고 넓은 영역에 걸친 많은 비용이 든다.

1980년대 후반부터 시작된 연속 음성에 대한 본격적인 연구는 1000개의 단어에서[1]부터 1995년도의 라디오 방송의 음성을 문자로 변환시키는 연구[2] 같은 무제한 단어까지 하고 있다. 대규모 어휘를 가진 연속음성의 인식은 상당히 어렵다. 그 이유는 첫째, 단어 구간이 불확실하다. 그 결과 많은 잘못 가정된 단어가 종종 만들어 진다. 그래서 의미 정보나 단어 문맥을 제공하는 복잡한 언어 모델을 사용하여 여러개의 가정 중에서 가장 그럴듯한 것을 선택하는 것이 필요하다. 둘째, 상호 조음 효과가 매우 강해서 어느 순간의 음성은 앞뒤의 음성에 영향을 많이 받는다. 이것들을 다루기 위해서는 Di-phone, Tri-phone과 같이 문맥 정보가 고려된 보다 정교한 음성 모델이 필요하다. 언어모델과 음성 모델이 복잡해질수록 작업의 난이도가 커지게 되어 컴퓨터의 계산능력이나 기억능력을 초과하기도 한다. 인식속도, 필요한 기억 장치의 자원과 인식률과 같은 세가지 종류의 성능은 서로 상충된다. 예를 들면 탐색 공간을 줄여 인식속도를 늘리고 간단한 음성과

* 호서대학교 컴퓨터 학부
** 우송 공업대학 전자정보 계열
*** 아주대학교 전자공학과
접수일자 : 1998년 9월 28일

언어 모델을 이용해 기억용량을 줄이면 인식률이 떨어진다. 즉 동시에 높은 인식률을 유지하면서 인식속도를 증가시키고 기억 용량을 줄이는 문제는 상당히 힘들다.

대규모 어휘 연속음성인식에서 언어모델은 탐색 과정에서 만들어진 많은 수의 가정된 단어열에서 가장 그럴듯한 단어열을 선택하는데 필요하다. 연속 음성 인식에서는 명확한 단어의 구간을 알 수가 없으므로 각 구간에 여러개의 단어들을 가정된 후보로 선정한다. 예를들어 "전해상이 파도가"와 "전 해상의 파도가"란 문장들은 음성학적으로 구분하기 어렵다. 이 여러가지 단어열에서 가장 그럴듯한 단어의 열을 선택하는 기준을 바로 언어모델이 제공한다. 언어 모델링 방법으로 인식 어휘나 문장이 소규모인 경우에는 Rule-based regular나 Context-free grammar를 이용할 수 있으나 대규모 어휘를 처리하는 경우에는 이것은 불가능하다. 대신 두 개 혹은 세 개의 단어의 전후 관계를 고려하는 구성되는 Bi-gram과 Tri-gram 언어 모델이 널리 이용된다.

음성 모델과 단어사전 그리고 언어 모델이 잘 주어졌다해도 마지막으로 이들을 사용하여 관찰된 음성에 대하여 가장 그럴듯한 단어의 나열을 찾는 탐색에 대한 문제가 남아있다. 이것을 Decoding이라 한다. Decoding은 탐색의 문제이다. 즉 미지의 음소열, 단어열들로 이루어진 커다란 탐색 공간에서 가장 최적의 경로를 발견하는 것을 말한다. 탐색은 일반적으로 깊이-우선(Depth-first) 탐색과 넓이-우선(Breath-first) 탐색으로 나뉜다[3]. 넓이-우선 방법은 탐색이 동시에 여러개가 병렬로 이루어지는 것으로 Bellman의 최적이론에 기초한 Viterbi decoding[4]이 대표적이다. 이 방법은 동적 프로그램 알고리즘을 이용하여 입력 음성에 가장 가까운 상태 열을 병렬적으로 찾아 나가는 것이다. 탐색공간은 음소 HMM들의 합성으로 많은 가정된 단어 HMM들이 생성되면서 구축된다. 탐색공간은 중간 규모의 어휘에 대해서도 매우 커지므로 그럴듯하지 않은 상태는 고려하지 않는 beam 탐색[5]방법으로 제한된 탐색을 하는 것이 보통이다. 이러한 결합을 간단히 Viterbi beam 탐색[6]이라 한다. 이 방법은 한 순간에 하나의 프레임만을 처리하므로 프레임 동기 탐색이라 불리우기도 하며 다음 프레임으로 이동하기 전에 그 프레임에 대해 그럴듯한 모든 상태들을 계산하며 전진한다. 깊이-우선 방법은 음성의 끝에 도달할 때까지 대기하고 있다가 마지막에 최종 결정하는 것으로 A*알고리즘[3]의 변형인 stack decoding[7,8]이 있다. 이 방법에서는 부분적 가정들이 저장되는 스택이 사용된다. 가정들은 가장 그럴듯한 것 순으로 정렬되어 저장된다. 여기서 '부분적 가정'이라 하는 것은 입력 음성의 초기부분에 대한 가정을 말하는 것이며 '완전한 가정'은 전체 입력 음성에 대한 것을 말하는 것이다. 각 프레임마다 스택에서 가장 그럴듯한 단어를 뽑아 탐색경로를 확장한다. 이렇게 한 프레임씩 음성 끝까지 전진하여 완전한 가정에 도달하면 출력이 이루어진다. 그렇지 않으면(음성의 끝에 도달하지 않는 경우) 다시 스택 저장된 가장 그럴듯한 단어가 모든 가능한 경로(단어)로 확장되고 이 경로에 대한 점수를 계산하여 스택의 내용을 갱신하는 과정이 반복된다. 이러한 방법으로 N-best 가정[9]을 만들어 낸다. 대규모 어휘에 대하여 그럴듯한 단어의 열이 기하 급수적으로 커지는 것을 피하기 위해 각 프레임에서 부분적인 가정을 몇가지로 제한하여 해당 후보 단어만을 확장하여 처리한다. 본 논문에서는 위에서 언급한 Viterbi Beam탐색과 Stack decoding을 결합한 다단계의 탐색 방법으로 인식을 하였고, 음성 모델은 Uni-phone단위로 계산 비용에 비하여 인식률이 높은 Semi-continuous HMM[10]방법을 사용하였으며, 언어 모델은 통계적인 방법인 N-grams[11]을 사용하였다.

Ⅱ. 음성 모델

음성은 단어들의 나열(sequence of words)로 되어있다. 나열된 단어들이 $W = w_1, w_2, .. w_n$일때, 관찰된 음성 신호 Y에 대하여 가장 그럴듯한 단어의 열 W'을 찾는 것이 음성인식 과정이다. 이를 위해 Bayes 규칙을 이용하면 요구되는 확률 P(W|Y)을 구할 수 있다. 즉

$$W' = \arg\max_{W} P(W|Y) = \arg\max_{W} \frac{P(W)P(Y|W)}{P(Y)} \qquad (2.1)$$

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 W가 관찰될 선험적 확률인데 이것은 언어 모델에서 계산되며 둘째항은 주어진 단어 열에 대하여 음성 벡터의 열 Y가 관찰될 확률로 이것은 음성 모델에 의해 계산된다.

Semi-continuous HMM은 Huang[10]에 의해 제안한 것으로 이산과 연속 HMM을 포함하는 일반적인 모델이다. 이것은 이산 HMM과 연속 HMM을 같은 확률 구조로 단일화 시키고 있다. 이산 HMM에서 벡터 양자화(VQ)는 관찰된 음성 신호를 모델링하기위해 비 파라미터인 이산 출력 확률 분포를 이용한다. 이산 HMM 구조하에서 VQ는 관찰된 음성에 대한 가장 가까운 codeword를 codebook에서 찾는다. 연속적인 음성 공간에서 양자화된 이산 공간으로의 사상(mapping)은 중요한 정보를 잃어버리게 할 수 있다. 이산 HMM의 또다른 단점은 VQ codebook과 이산 HMM은 개별적으로 모델링이 되므로 이들을 조합한 것에 대해서 최적화를 할 수 없다. 한편 연속 HMM은 추정된 연속 확률 밀도 함수를 이용해 관찰되는 음성을 직접 모델링한다. 이산 HMM과 비교해서 인식률을 향상시키기 위해서는 많은 수의 확률 밀도 함수를 혼합해야 한다[12]. 많은 수의 확률 밀도 함수를 혼합하면 계산상의 복잡

도뿐 아니라 신뢰할 수 있는 추정에 필요한 파라미터의 수가 상당히 증가한다.

Semi-continuous HMM은 연속 HMM에 필적하는 인식률을 보장하면서도 계산상의 이득을 가져다 준다. 본 논문은 문맥을 고려하지 않는 기본 음소(Uni-phoneme)를 인식 단위로 하며 Semi-continuous HMM방법으로 이들을 모델링 한다.

Ⅲ. 언어 모델

언어 모델은 선험적 확률 P(W), 즉 문장 W(혹은 단어의 열 $W=w_1, w_2,..,w_n$)의 발생 확률을 구하는데 쓰인다. 여기서 P(W)는

$$P(W)=P(w_1)P(w_2\mid w_1)P(w_3\mid w_1w_2)P(w_4\mid w_1w_2w_3)\cdots$$
$$P(w_n\mid w_1w_2,\cdots,w_{n-1}) \qquad =\prod_{i=1}^{n}P(w_i\mid w_1,\cdots,w_{i-1}) \tag{3.1}$$

로 주어진다. 수식 $P(w_i\mid w_1,\ldots,w_{i-1})$에서 $w_1,\ldots,w_{i-1}$는 과거 단어(word history)혹은 간단히 w_i의 과거라 한다. 현실적으로 주어진 임의 길이의 과거로부터 현재 단어의 발생확률을 신뢰성 있게 구하기는 어렵다. 왜냐하면 상당히 많은 학습 자료가 있어야 하기 때문이다. 그래서 대신에 다음과 같이 과거의 N-1개의 단어만 고려하여 식 (3.1)의 확률을 계산하는 N-gram(특히 bigram, trigram) 근사화의 방법들을 이용하는 것이 보통이다.

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

$P(w)$ = 단어 w의 확률
$P(w_i\mid w_j)$ = 단어 wi다음에 단어 wi가 나올 확률
$P(w_k\mid w_i, w_j)$ = 과거에 wi, wi가 나온 다음에 단어 wk가 나올 확률

학습자료의 부족에 대한 해결책은 discounting과 backing-off의 조합을 이용하는 것이다[11]. discounting은 많은 빈도가 발생하는 trigram에 대하여 그 빈도수를 줄여 다시 추정하는 방법이고, Backing-off는 추정하기에 너무 적은 trigram에 대하여 scaled bigram확률로 대체하여 추정하는 방법이다.

$$P'(w_k\mid w_{k-1},w_{k-2})=B(w_{k-1},w_{k-2})P(w_k\mid w_{k-1}) \tag{3.2}$$

여기서 B는 $P'(w_k\mid w_{k-1},w_{k-2})$을 적당히 정규화하게 하는 back-off 함수이다. 다른 high-order backoff N-gram도 위와 비슷하게 한다.

3.1 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 그림 3.1과 그림 3.2와 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 3.1. trigram 언어 모델의 계산
Fig. 3.1. The computation of trigram language model.

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

그림 3.2. bigram 언어 모델의 계산
Fig. 3.2. The computation of bigram language model.

또한 Discouting ratio값은 1-gram discounting ratio 은 0.814363, 2-gram discounting ratio은 0.535957, 3-gram discounting ratio 은 0.387617을 사용하였다.

Ⅳ Decoding

Decoding이란 관찰된 음성에 대하여 가장 그럴듯한 단어의 나열을 찾는 것이다. 즉 음성 대한 특성을 학습에 의해 음성 모델(Acoustic model)로 만들어져 있고, 언어 특성은 언어 모델(Language model)로 주어지고, 단어는 음소들의 결합에 의해 사전으로 주어져 있을때 인식할 음성에 대하여 가장 그럴듯한 단어의 열을 발견하는 것을 음성 인식의 마지막 단계이다. 위에서 언급한 모델들을 λ라 하고 인식할 음성을 Y라 할때, 이 음성을 만들어 낼 수 있는 음성 모델들의 상태 열(State sequence,Q)중에서 가장 확률이 높은 상태의 열(Q^*)을 찾는 것이다. 즉

$$Q^* = \arg\max_{Q} P(Y,Q\mid\lambda) \tag{4.1}$$

인식 대상의 음성은 HMM 음소 모델로 나열된다. 각각의 음성 프레임마다 시작되는 음소가 보통 수십개씩 나오며 음소의 길이는 같기도 하고 다르기도 한다. 달리 말하면 임의의 프레임은 음소의 시작점이 되기도하고 음소의 중간이기도하며, 음소의 마지막 부분

일 수도 있다. 이 모든 경우는 사전에 있는 단어들과 비교하여 단어 모델로 바꾸어진다. 또한 만들어진 단어들은 음소와 마찬가지로 어느 한순간에 여러개의 다른 단어가 시작과 끝이 다르게 나타난다. 이 모든 경우는 언어 모델을 이용해 하나의 연속 문장이 만들어 진다. 예를 들어 *"지금 보시는 것처럼 중국 내륙과 남해상 그리고 일본에 걸쳐서 동서로 장마 구름 대가 자리를 잡고 한반도 지역에는 소나기구름들이 곳곳에 있어서 오늘밤과 내일 오후에 소나기가 예상됩니다"* 를 인식한 경우 인식된 단어의 갯수는 26개이었으나 음소모델의 총 갯수는 25,809개이고, 단어 모델의 총 갯수는 4,967개 이었다. 이때 총 프레임 수는 1,719개 이었다. 음소나 단어 격자의 형태는 같으나 음소 격자의 밀도가 단어 격자의 것보다 많다.

4.1. 다 단계 탐색

앞서 말한바와 같이 수만개의 음소모델과 수천개의 단어모델에서 우리가 원하는 단어들의 나열을 찾는 문제는 쉬운 문제가 아니다. 그래서 본 논문에서는 이를 위해 다음과 같이 3단계의 구조를 가진 탐색 방법을 사용했다.

1) 순방향 Viterbi beam 탐색[4, 5]. 이것은 Semi-continuous 음성 모델과 bigram혹은 trigram 언어 모델들을 이용하여 사전에 있는 모든 단어를 완전히 탐색한다. 이 탐색의 결과는 인식된 모든 후보 단어가 들어있는 단어 격자가 나온다. 격자에는 단어의 구간과 경로 점수에 대한 정보가 들어있다. 즉, 각 단어의 끝부분을 결정하고 음성의 시작에서 그 곳까지(현재까지의 비용) 도달하는 여러 경로중 하나를 경로 점수에 따라 선택한 뒤 관련 정보를 저장한다.

2) 역방향 Viterbi beam 탐색. 이 방법은 순방향에서 식별된 단어에 대해서만 한정을 두므로 매우 빠르다. 첫번째 방법과 마찬가지이나 방향이 반대로 이번에는 각 단어의 시작 부분을 결정하고 그 곳에서 음성의 끝까지의 여러 경로중 하나를 선택한 뒤 필요한 정보를 저장해 둔다. 역방향 viterbi탐색의 주요 역할은 임의의 중간지점(any point in the utterance)에서 그 마지막 지점(the end of utterance)까지(남은 거리)의 가장 좋은 경로 점수를 계산하여 그 값을 세 번째 단계로 넘겨주기 위해 저장해 두는 것이다.

3) A*혹은 stack decoding. 이 단계에서는 앞선 두 단계에서 얻은 경로 점수와 언어 모델을 적용하여 N-best 목록[13]을 만들어 낸다. 이 알고리즘은 Best-first 방법[7,8,14]의 확장으로서 부분 가정들을 저장하는 스택을 가지고 있다. 부분 가정들이란 단어열들을 뜻하며 위 1, 2번 탐색의 결과로 얻어진 경로 점수에 따라 정렬되어 저장된다. 탐색 과정은 스택 맨 위의 가정부터 '완전한 가정' 인지 판별하며 그렇다면 N-best 목록으로 출력하고 그렇지 않으면 결합 가능한 다음 단어들로 부분 가정을 확장하여 점수를 Up-date하고 스택을 재정렬 과정을 N-best 목록이 채워질 때까지 반복하게 된다. 비록 첫번째 탐색에서 좋은 인식 결과를 얻을 수 있다해도 역방향 탐색과 A*탐색을 다시 거치는 이유는 순방향 Viterbi탐색에서는 단어의 구간이 결정되지 않은 상태에서 전진하게 되므로 복잡한 문법을 적용하기가 어렵다는 것이다. A* 탐색에서 부분적 가정들이란 단어열들을 가리키며 따라서 임의의 언어모델을 적용하기가 수월하고 보다 높은 인식률을 얻을 수 있다. Stack decoding은 가장 좋은 것 하나를 선택하기 보다 그럴듯한 N개의 가정을 출력하도록 하고 이 중 가장 최고의 것을 인식결과로 결정한다.

V. 실험 및 결과

5.1 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 $x_0(t)$로 나타낸다. 우선 이 캡스트럼 벡터와 power를 정규화시키고 1차와 2차 미분을 하여 각 프레임 마다 4가지 종류의 특징 벡터들을 구한다.

$x(t)$ = 정규화된 캡스트럼 벡터

$$\Delta x(t) = x(t+2) - x(t-2), \Delta_1 x(t) = x(t+4) - x(t-4)$$
$$\Delta\Delta x(t) = \Delta x(t+1) - \Delta x(t-1)$$
$$x_0(t) = x_{0(t)},$$
$$\Delta x_0(t) = x_0(t+2) - x_0(t-2),$$
$$\Delta\Delta x_0(t) = \Delta x_0(t+1) - \Delta x_0(t-1)$$

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

5.2 음소 HMM 모델

여기서 사용한 음소 모델은 기본 음소이다. 모든 HMM은 5개의 상태를 갖는 Bakis 형태를 갖는다. Semi-continuous 음성 모델은 각 codebook에 대하여 256개의 성분 밀도를 가지고 있다. 상태는 각각의 특징 열에 대하여 256개의 weighting codebook의 혼합 계수를 갖는다. 계산 비용을 줄이기 위해 각각의 특징 codebook에서 가장 가능성이 있는 순으로 몇 개의 성분 밀도(보통 4개)만 각 프레임에서 상태 출력 확률을 구하기 위해 계산한다. 이러한 방법은 각 프레임에서 출력 확률을 구하는데 있어서 혼합 weight의 비용을 상당히 줄인다. 256개를 모두 계산하는 대신에 가장 좋은 4개의 요소만 처리하였다.

5.3 발음 사전

발음사전은 모든 단어에 대하여 발음을 음소의 선형적인 형태로 나열하였다. 표 5.1은 여기서 사용한 대표적인 기본 음소에 대한 발음 사전을 보여준다. 종성의 "ㅇ"은 중성 모음에 따라 여러개로 구분하였다. 이중 모음은 표에서 보는 바와 같이 W와 Y를 모음 앞에 연결하여 처리하였다. 예를들어 "여"는 "Y + 어" --> "Y AO"로, "왜"는 "W + 애" --> "W AE"와 같이 사용하였다. 실험에 사용한 발음 사전의 크기는 11,951개로 언어 모델과 인식에 사용할 모든 단어를 대상으로 하였다.

표 5.1. 발음 사전의 예
Table 5.1. The example of dictionary.

음소	단어	발음사전
AA	까닭은	KK AA D AA L G EU N
AE	폐내	PP AE N AE
AO	따라서	TT AA R AA S AO
B	어업은	AO AO B EU N
CH	엄청난	AO M CH ON N AA N
D	영덕군	Y ON D AO KC G U
EH	영향에	Y ON HH Y AN EH
G	여쭤보고	Y AO JJ W AO B OW G OW
HH	열흘	Y AO L HH EU L
IY	열린	Y AO L IY N
JH	위주의	W IY JH UW EU IY

5.4 음성 자료 및 언어 자료

5.4.1 학습과 인식을 위한 음성 자료

Semi-continuous HMM 음소 모델 학습에 사용한 음성 데이타는 162명이(남자:92명, 여자:70명) 각기 30분에서 2시간 정도로 발음한 것으로 읽은 자료는 우리말의 음소가 모두 들어있는 음성이 되도록 국내의 신문사 Web site에서 무작위로 발췌하였다. 우리말 음성 모델을 만들기 위한 학습 시간은 약 20시간 정도였고 학습에 사용한 컴퓨터는 시간을 단축시키기 위해 여러대의 컴퓨터를 이용해 처리하였다. 예를 들면 특징 추출된 입력 음성과 언어모델은 Sun machine에, 음성 모델과 중간 결과는 Alpha machine에, 처리와 최종 결과는 Linux machine으로 처리하였다. 인식 데이타로는 학습에 참여하지 않은 남성이 KBS 9시 뉴스의 여름(5월- 10월까지)의 일기예보를 발췌하여 읽은 음성을 사용하였다. 인식에 사용한 음성은 두가지 종류로 나누어 언어 모델링에 반영된 음성 40문장과 (6월 1일부터 9월 30일까지에서 발췌), 언어모델링에 반영되지 않은 그러나 큰 변화는 없을 것이라 생각되는 5월 31일과 10월 1일의 일기예보 중 40개의 문장을 택하였다.

5.4.2 언어 모델을 만들기 위한 기초 자료 및 언어 모델

언어 모델을 만들기 위한 기초 자료 역시 인터넷을 통해 한국의 신문사 및 방송사의 자료를 수집하였다. 우선 기본 언어 모델을 만들기 위해 일반적인 단어가 들어 있는 신문기사와 KBS 9시 뉴스의 일부분을 포함, 총 8,555개의 어휘로 구성된 1,348 문장(형태 1)을 수집하였고, 인식할 음성의 문장이 일기예보 이므로 6월 1일부터 9월 30일까지의 KBS9시 뉴스의 일기예보 부분만을 발췌한 3,306개 어휘로 구성된 2,327개의 문장(형태 2)을 수집하였다. 언어 모델을 만들기 위한 기초 자료와 이 자료를 기반으로 구축한 언어 모델의 종류는 다음 표와 같다.

표 5.2. 언어 모델을 만들기 위한 문장
Table 5.2. The sentence for making language model.

형태	문장 종류	문장 구성 방법	문장의 수
1	기본 문장	일반적인 문장	1,348 문장
2	일기 예보	6월 1일부터 9월 30일까지의 실일기예보	2,327 문장

표5.3. 언어 모델의 종류
Table 5.2. The kind of language model.

언어 모델의 종류	문장의 갯수	단어의 종류	언어 모델의 구성 방법
1	3,675	11,188	형태 1 + 형태 2 정상적인 문장
2	3,675	994	형태 1 + 형태 2 음절 형태의 문장
3	7,350	11,951	형태 1 + 형태 2 정상적인 문장 + 음절 형태 문장

표 5.3에서 '음절 형태의 문장'이란 '정상적인 문장'에서 모든 단어를 음절단위로 나누어 마치 음절 각각을 단어로 간주하여 문장을 구성한 것이다. 그 이유는 우리말을 단어 대신 음절을 기본으로 하여 인식해 보고자 함이다. 또한 문장의 수는 3,675개인데 단어가 11,188개밖에 안되는 이유는 이 숫자가 총 단어의 수가 아니라 서로 다른 단어의 종류를 의미하기 때문이며 음절 문장에서의 단어의 종류가 문장의 수보다 작은 이유도 마찬가지 때문이다.

5.5 인식률의 계산

고립된 단어의 인식률의 계산은 단지 인식된 단어가 잘못된 경우만 고려하면 되지만 연속 음성의 경우는 오인식(error)은 3가지 종류가 있을 수 있다 : 1) 고립 단어 인식의 경우와 같이 다른 단어로 인식된 경우, 2) 인식된 문장에 단어가 누락된 경우, 3) 인식된 문장에 다른 단어가 추가된 경우. 1)과 2)의 경우는 확실히 오인식이지만 3)의 경우는 애매할 경우가 많다. 특히 우리말의 경우에는 복합명사를 각 명사마다 띄어 쓰기도 하고 때로는 붙여 쓰기도 하므로 인식률의 계산이 복잡하다. 예를 들어 원래의 문장이 "전해상이 파도가" 이고 인식된 문장이 "전 해상이 파도가"일 경우 원래의 문장은 2개의 단어로 되어있지만 인식된 문장에는 3개의 단어로 이루어지기 때문에 "해상이"라는 단어가 추가되었다고 볼 수 있다. 사실 이러한

경우 오인식인지의 판단이 쉽지가 않다. 그래서 초기 인식률의 계산은 3)의 경우를 오인식에서 배제하였으나 현재는 다른 시스템들과의 비교를 위해 이 경우도 오인식으로 취급하고 있다. 본 실험에서는 두가지 방법으로 인식률을 계산하였다. 하나는 단어 인식률로 위에서 언급한 3가지 모두 오인식으로 취급하여 인식률을 계산하였고 다른하나는 음절 인식률로 모든 문장을 음절로 나타내어 ("전 해상이 파도가" --> "전 해 상 이 파 도 가") 인식률을 계산하였다. 위 문장의 경우 단어 인식률로 계산한 경우 오인식이 발견되나 음절 인식률로는 오인식이 없는 것으로 처리하였다. 인식률을 계산하기 위해서는 인식된 결과와 원래의 문장을 시간적으로 정렬을 해야 하는데 이러한 정렬을 미국 표준성(National Bureau of Standard"에서 제공한 동적인 프로그램[15]으로 처리하였다.

$$\text{오인식률}=100\cdot\frac{\text{다른단어로인식된단어의수}+\text{삭제된단어의수}+\text{추가된단어의수}}{\text{문장 길이}}$$

$$\text{인식률}=100\cdot\frac{\text{인식이 잘된 단어의 수}}{\text{문장 길이}}$$

그림 5.1과 그림 5.2는 각각 정상적인 문장의 언어 모델을 이용하여 인식한 결과와 음절로 만 구성된 언어 모델을 이용한 인식 결과를 보여주고 있다. 여기서 REF는 기준 문장이고 HYP는 인식된 문장이다. 각각의 문장 별로 인식률이 나오고 전체 인식률은 모든 문장에 대한 평균 인식률로 계산하였다.

```
REF: 전해상이  안개  끼는  곳이 많겠고 물결은 앞내지 이미터로 비교적 낮게 일
겠습니다
HYP: 전        해상이  안개끼는 곳에 많겠고 물결은 앞내지 이미터로 비교적 낮게 일
겠습니다

SENTENCE 2  (12)
Recognition     = 72.7%   (8)
Errors          = 27.3%   (3)
```

그림 5.1. 정상적인 문장으로 만든 언어 모델에 의한 인식 결과

Fig. 5.1. The recognition result for the language model which made by normal sentence.

```
REF: 지 금 도 풍 부 와 일 부 영 남 지 방 에 비 구 름 이 낮 아 져 서 오 늘 밤 에
도 오 내 지 십 억 니 정 도 의 비 가 더 내 리 고 내 일 도 대 기 가 불 안 정 한 상 태
에 서 주 로 풍 부 지 역 으 로 소 나 기 구 름 이 나 타 나 겠 습 니 다

HYP: 지 ** 도 풍 부 와 일 부 영 남 지 방 에 비 구 름 이 낮 아 져 서 오 늘 밤 에
도 오 내 지 십 ** 이 정 도 의 비 가 더 내 리 고 내 일 도 대 기 가 불 안 정 한 상 태
에 서 주 로 풍 부 ** 으 로 소 나 기 구 름 이 나 타 나 겠 습 니 다

SENTENCE 34  (o34)
Recognition     = 88.3%   (68)
Errors          = 11.7%   ( 9)
```

그림 5.2. 음절로 구성된 문장으로 만든 언어 모델에 의한 인식 결과

Fig. 5.2. The recognition result for the language model which only made by syllable sentence.

5.6. 인식 결과

언어 모델에 따른 단어 및 음절별 인식률을 표 5.5에 나타내었다. 표 5.4에서 언어 모델은 표 5.3의 언어 모델 종류에 따른 것이고, 음성 종류에서 A는 6월~9월 사이의 문장을 읽은 음성이고 B는 5월과 10월의 문장을 읽은 것이다. 즉 언어 모델링에 '음성 A'의 문장들은 반영이 된 것이고 '음성 B'는 반영이 안된 것이다. 언어 모델 1에서 음절 인식률의 결과에 복잡도가 없는 이유는 '음절 형태의 인식'을 하므로 '음절 형태의 문장으로 구축된 언어 모델'이어야 복잡도를 계산할 수 있는데 이들 1은 '정상적인 문장으로 구축된 언어 모델'이기 때문이다. 또한 언어 모델 2에서 단어 인식률을 구할 수 없는 이유는 음절 형태의 문장으로 구축된 언어 모델을 가지고 인식을 해야하기 때문에 단어 형태로 인식 결과가 나올 수 없기 때문이다 (그림 5.2 참조).

인식률을 관찰하여 보면 인식기는 B군 음성 보다 A군 음성에 대해서 모든 경우에서 있어서 월등한 인식률을 보임을 알 수 있다. 이는 언어 모델링의 중요성을 나타내는 것으로 A군의 문장들은 언어 모델의 구축에 반영된 반면 B군의 문장들은 그렇지 못하였기 때문이다. 또한 단어 인식률 보다는 음절 단어 인식률이 높은 결과를 나타낸다. 이는 단어 단위로 인식률을 계산할 경우 삽입이나 삭제에 의한 에러가 음절 단위로 계산하면 일부 무마되기 때문이다. 최고 인식률은 단어 인식률의 경우 A군에 대해 96.0%, B군에 대해서는 79.0%를, 음절 인식률의 경우 A군에 대해서는 99.2%, B군에 대해서는 92.6%를 얻었다. A군, B군을 모두 고려하여 인식률을 조사해 보면 언어 모델 1번에 trigram 모델을 사용하는 것이 언어 모델링에 최적임을 알 수 있다.

언어 모델이 unigram보다는 bigram 혹은 trigram이 인식률이 훨씬 좋음을 알 수 있고 본 연구에서 대상으로 한 일기 예보와 같이 제한된 경우에는 bigram과 trigram의 차이가 별로 없다. 우리말을 단어 대신 음절을 기본으로 하여 인식을 시도한 '언어 모델 2와 5'의 경우 결과에서 보듯이 인식률이 높지 않았다. 그 이유는 비록 언어 모델이 많은 정보는 가지고 있으나 복잡도가 너무 커(1,000이상) 단어 인식률이 좋지가 않았다.

표 5.4. 언어 모델에 따른 단어 및 음절별 인식률

Table 5.4. The word and syllable recognition result for various language model.

		단어 인식률						음절 인식률					
언어 모델	음성 종류	unigram		bigram		trigram		unigram		bigram		trigram	
		복잡도	인식률	복잡도	인식률	복잡도	인식률	복잡도	인식률	복잡도	인식률	복잡도	인식률
1	A	703	72.2	16	94.2	10	96.0		87.7		98.6		99.2
	B	634	66.0	68	78.7	69	78.5		85.5		91.8		91.7
2	A							130	32.8	10	76.6	4	89.4
	B							127	33.1	11	68.7	5	82.2
3	A	2383	61.1	13	92.2	7	88.9	167	82.4	11	98.0	5	97.5
	B	2131	58.2	100	70.5	106	66.9	163	81.4	12	90.0	6	89.6

V. 결 론

약 12,000여개의 대규모 어휘의 일기예보 방송음에 대해 문맥을 고려하지 않은 기본 음소 모델을 사용하여 단어 인식률은 최고 96.0%, 음절 인식률은 99.2%까지 얻었다. 연속 음성 인식의 디코딩을 위해 3단계의 탐색 방법을 사용하였는데 인식에 걸리는 시간은 많이 필요하였으나 높은 인식률을 얻을 수 가 있었다. 음절 단위로 구성된 언어 모델을 이용할 경우 인식률이 높지 않음을 확인 할 수 있었고 또한 이 경우에는 음절들을 단어로 변환하는 후처리 과정이 필요하므로 이 과정에서도 오류가 발생할 수 있기 때문에 단어 기반의 문장으로 만든 언어 모델이 바람직하다고 생각한다. 앞으로의 과제는 보다 우리말에 가까운 발음사전의 구축과 언어모델에 대한 연구가 필요하다.

참 고 문 헌

1. 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.
2. 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.
3. Nilsson, N.J. "Problem Solving Methods in Articial Intelligence." McGraw-Hill, New York, 1971.
4. 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.
5. Lowerre, B. "The Harpy Speech Understanding System." Ph.D. thesis, Computer Science Department, Carnegic Mellon University , Apr 1976.
6. 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.
7. 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.
8. Paul, Douglas B. "An Effcient A* Stack Decoder Algorithm for Continuous Speech Recognition with a Stochastic Language Model." In Proceedings of DARPA Speech and Natural Language Workshop, Feb. 1992, pp 405-409.
9. Schwartz, R. and Chow, Y.L. "The Optimal N-Best A lgorithm: An Effcient Proce dure for Finding Multiple Sentence Hypotheses." In IEEE International Conference on Acoustics, Speech, and Signal Processing, Apr. 1990.
10. X.D. Huang and M.A. Jack "Semi-continuous hidden Markov model for speech signals" Computer Speech and Language, Vol. 3, 1989, pp. 238-251.
11. H. Ney, U.Essen, R.Kneser. "On Structuring Probabilistic Dependences in Stochastic Language Modelling." Computer Speech Language, Vol. 8, 1994, pp. 1-38.
12. L.R. Rabiner, J.G. Wilpon, and F.K. Soong."High Performance connected digit recognition using hidden markov models." in Proc. ICASSP-88, 1988.
13. R.Lau, R.Rosenfield, S.Roukos."Trigger-based Language Models: a Maximum Entropy Approach." Proc ICASSP 93, Vol 2, pp. 45-48, 1993.
14. Allev a,F., Huang, X., and Hw ang, M. "An Improved Search Algorithm for Continuous Speech Recognition." In IEEE International Conference on Acoustics, Speec h, and Signal Processing, 1993.
15. D.Pallett,"Test procedure for the March 1987 DARPA benchmark tests," Proc. DARPA Speech Recog. Workshop, Mar. 1987, pp. 75-78

▲김 석 동(Suk-Dong Kim)

1957.8.23일 생
1982년 2월 : 아주대학교 전자공학과 졸업(공학사)
1984년 2월 : 아주대학교 대학원 전자공학과 졸업(공학석사)
1993년 2월 : 아주대학교 대학원 전자공학과 졸업(공학박사)
1985년 3월 - 현재 : 호서대학교 컴퓨터학부 교수

▲송 도 선(Do-Sun Song)

1956.4.14일 생
1977년 2월 : 영남대학교 전자공학과 졸업(공학사)
1981년 8월 : 고려대학교 대학원 전자공학과 졸업(공학석사)
1995년 2월 : 아주대학교 대학원 전자공학과 졸업(공학박사)
1981년 3월 - 현재 : 우송 공업 대학 전자정보 계열 교수
*주관심 분야 : 음성인식및 신호처리

▲이 행 세(Heing-Sei Lee)
1943.8.29일 생
1966년 2월 : 전북대학교 전기공학과 졸업(공학사)
1972년 2월 : 서울대학교 대학원 전자공학과 졸업(공학석사)
1984년 2월 : 고려대학교 대학원 전자공학과 졸업(공학박사)
1973년 2월 - 현재 : 아주대학교 전자공학과 교수