제15회 음성통신 및 신호처리 워크샵(KSCSP '98 15권1호)

확률 발음사전을 이용한 대어휘 연속음성인식

윤성진, 오영환
한국과학기술원 전산학과

Large Vocabulary Continuous Speech Recognition using Stochastic Pronunciation Lexicon Modeling

Seong-Jin Yun, Yung-Hwan Oh
KAIST CS Department

요 약

본 논문에서는 대어휘 연속음성인식을 위한 확률 발음사전 모델에 대해서 제안하였다. 제안된 확률 발음사전은 인속음성과 같은 자연스런 발성에서 자주 발생되는 단어의 변이를 확률적인 subword-state로 이루어진 HMM으로 모델화 함으로써 단어의 발음 변이를 효과적으로 표현할 수 있으며, 단위 인식 모델과 인식기의 특성을 반영하여 전체 인식 시스템의 성능을 보다 높일 수 있도록 구성되었다. 확률 발음사전의 생성은 음성자료와 음소 모델을 이용하여 단어 단위의 분할과 학습을 통해서 자동으로 생성되게 되며 음소와 같은 언어학적인 단위뿐만 아니라 PLU(phone like unit)이나 비언어학적인 인식 모델을 이용한 연속음성인식기에도 적용이 가능하다. 인속음성인식 실험 결과 확률 발음사전을 사용함으로써 표준 발음 표기를 사용하는 인식 시스템에 비해 단어 오류율은 39.8%, 문장 오류율은 24.4%의 큰 폭으로 오류율을 감소시킬 수 있었다.

I. 서 론

음성인식은 대상으로 하는 음성의 발성 방법에 따라 크게 구분발성음성과 연속음성으로 나눌 수 있다. 구분발성음성은 단어 단위로 띄워서 발성하는 방법으로 단어간의 경계가 확실하여 단어간 조음 현상으로 인한 변이가 작아 비교적 높은 인식 성능을 보인다. 그러나, 발성 방법의 제약으로 인하여 명령어 인식과 같이 단순한 형태의 인식시스템에 주로 이용된다. 이에 반해 연속음성은 문장의 형태로 자연스럽게 발성되므로 단어의 구분이 불분명하여 단어간 조음 현상으로 인한 변이가 심하다. 또한, 구분발성과 달리 같은 단어라도 문장 내에서의 발성 속도의 차나 단어내의 변이도 심하게 된다. 일반적으로 연속음성은 고립단어에 비해 발성의 변이가 크고 단어간의 조음 결합 현상과 단어 경계의 불분명성 등으로 인하여 인식이 훨씬 어렵다고 알려져 있으나 보다 자연스럽게 음성인식을 사용할 수 있기 위해서는 연속음성의 인식이 필수적이다.

대부분의 대어휘 연속음성인식 시스템에서는 기본 단위 모델로 단어(word)보다는 subword를 사용한다[1]. subword로는 주로 음소(phone)나 음소 유사 단위(PLU: phone like unit)등을 주로 사용하게 되며, 이러한 음성인식 시스템에서는 단어나 문장을 인식하기 위해서 subword 단위들의 조합으로 구성된 단어 발음 사전(pronunciation lexicon)을 필요로 하게 된다. 단어 발음사전은 일반적으로 단어에 해당하는 표준 발음표기를 인식 단위의 일련 나열함으로써 구성하거나 음운학적 지식을 가진 전문가에 의해서 만들어진다. 이렇게 구성된 발음사전은 몇 가지 문제점들을 갖게 된다.

첫째, 표준 발음 표기로 구성된 단어사전은 여러 화자들의 발성의 변이를 표현하기 힘들다는 문제점이 있다. 발성 변이는 크게 개인의 발성 습관, 사투리와 문장 내의 위치나 내용에 따라 발생 차이로 발생하는 단어내 변이와 단어의 단어의 경계에서 조음결합 현상에 의해 발생하는 단어간 변이로 나눌 수 있다. 발성 변이에 대한 해결 방법으로 많은 음성 인식 시스템에서는 복수개의 발음표기를 사용하고 있다[2]. 또 단어간 변이를 해결하는 방법으로 발음 규칙을 사용하고 있으며, 이 발음 규칙은 표준 발음 표기와 전문가가 음성을 듣고 작성한 발음 표기간의 차를 규칙화한 후 단어간 천이 시 적용시킨다[3]. 그러나 이러한 방법들은 초기에 전문가가 작성한 발음 표기 방법들을 필요로 하기 때문에 발음 사전의 구성이 쉽지 않고, 복수의 발음 표기를 정하는 방법과 표현 방법에 따라 인식률의 향상에 많은 영향을 미친다는 문제점이 있다.

둘째, 인식의 기본 단위로 음소와 같은 언어적(linguistic)인 단위를 사용하지 않고 음소 유사 단위나 acoustic unit과 같이 음향학적으로 유사도에 기반한 단위를 쓰는 경우 언어적인 단위나 단어로의 사상이 필요하므로 발음사전을 만들기 위해서는 많은 노력을 필요로 하게 된다. 일반적으로 음향학적인 단위를 이용하여 발음 사전을 구성하기 위해서는 인식기에 의해 각 단어에 대한 최적의 단위열을 구한 후 발음사전을 구성하고 이 발음사전을 이용하여 다시 인식기에 사용되는 단위 모델을 학습을 반복하는 방법을 사용하게 된다. 이밖에 음향학적 모델과 음소간의 사상관계를 구한 후 표준 음소에 대한 발음 표기를 음향학적 단위에 대한 발음 표기로 바꾸는 방법을 사용하고 있으나 언어적인 단위와의 분일치로 정확한 발음사전을 구성하기가 어렵다.

셋째, 일반적으로 음성인식기에서 사용되는 기본 subword 단위들은 음향 모델링 특성상 발음사전에 사용되는 단위들과 정확히 일치하지 않기 때문에 연속음성 인식기의 성능을 최적화 할 수 있는 발음사전의 구성과는 거리가 있다. 이는 발음사전에서는 언어적으로 정확히 구분되는 자소(grapheme)에 기반한 단위를 사용하는데 비해 음성인식기에는 음소(phone)와 같은 음향학적 유사성에 의해 학습된 단위 모델들을 사용하는데 원인이 있다. 따라서 연속음성인식기의 성능 향상을 위해서는 발음사전과 음성인식기가 대상으로 하는 단위 모델 사이의 불일치를 해결한 방법이 필요하다. [4]에서는 음성인식기의 음소열 인식 결과를 이용하여 발음사전을 구성하는 방법을 제안하였다. 이것서 각 단어의 발음 표기는 인식기로부터 탐색과정을 거쳐 얻은 다수의 최적 음소열을 결합화 하여 하나 복수개의 표기를 얻고 있다. 그러나 이 방법은 인식기에 의존하여 각 음성에 해당하는 최적의 음소 표기열들을 구하기는 하지만 이를 인식에 사용하기 위해서 효과적으로 병합하거나 신뢰하는 과정이 요구된다.

이와 같이 많은 연구들이 실제 발음을 보다 정확히 표현할 수 있는 음소열이나 발음 규칙을 찾고자 노력

하고 있다. 본 논문에서는 앞서 제시한 발음사전의 문제점들을 극복하기 위해서 확률 발음 사전을 제안하였다. 논문의 구성은 다음과 같다. 2장에서는 확률 발음 사전의 구성과 학습 방법 인식시스템에 적용에 대해서 기술한다. 3장에서는 인식 시스템의 성능 평가 및 실험 결과를 분석하고 4장에서 결론을 맺는다.

II. 확률 발음사전 모델

표준 발음 표기만으로는 실제 음성 자료를 정확히 표현하지 못하기 때문에 각 단어의 발음 변이에 대한 고려가 있어야 한다. 그러나, 전문가 지식이나 음운 규칙을 적용 기반으로 하는 방법은 많은 노력을 필요로 할 뿐 아니라 실제 음성자료에서 나타나는 발성의 변이를 모두 표현하기가 힘들다. 또한 발성의 변이에 대한 모델링 및 인식기에서 사용되는 음소 모델보다는 언어적인 상위 단위에 대한 표현이기 때문에 인식 시스템의 최적화와는 거리가 멀게 된다. 이 장에서는 인식음성인식에 사용되는 확률 발음사전을 구성하는 방법과 학습 과정, 그리고 인식 과정에서 확률 발음사전을 이용하는 방법에 대해서 기술하고자 한다.

2.1 발음 사전 모델

각 단어는 자소열 $G(g_1, g_2, \ldots, g_I)$로 이루어져 있으며 발음 사전에서는 이를 발음 규칙에 의해 음소열 $P(p_1, p_2, \ldots, p_J)$로 변환한다. 발음 사전에서 각 단어는 구성하는 음소열을 baseform이라 하며 이 baseform은 단어 모델을 대신하게 된다. 단어 음성 자료 $X(x_1, x_2, \ldots, x_T)$이 주어질 때 음성인식 과정은 단어 모델과의 우도(likelihood) $\Pr(X|G)$을 구하는 과정이다. 먼저 각 단어에서 X와 P가 동시에 발생할 확률은 식(1)과 같이 음향 모델과의 우도 $\Pr(X|P,G)$와 해당 baseform에 대한 확률 $\Pr(P|G)$의 곱으로 나타낼 수 있다.

$$\Pr(X, P|G) = \Pr(X|P,G)\ \Pr(P|G) \tag{1}$$

단어 모델과의 우도 $\Pr(X|G)$은 식(2)와 같이 가능한 모든 음소열에 대해 식(1)을 합함으로써 구할 수 있다.

$$\begin{aligned}\Pr(X|G) &= \sum_{all\,P} \Pr(X, P|G) \\ &= \sum_{all\,P} \Pr(X|P,G)\ \Pr(P|G)\end{aligned} \tag{2}$$

대부분의 음성인식 시스템에서는 각 단어에 대해 하나의 표준 발음표기로 구성된 발음사전을 사용하기 때문에 baseform에 대한 확률은 $\Pr(P|G) = 1$로 볼 수 있다. 이때의 단어 모델과의 우도는 다음과 같다.

$$\begin{aligned}\Pr(X|G) &= \Pr(X|P,G) \\ &= \Pr(X|P)\end{aligned} \tag{3}$$

실제 발음에서 발생할 수 있는 변이를 고려한 경우는 각 단어에 대해 복수개의 baseform을 사용하기도 한다. 이때 각 baseform에 대한 확률은 전문가의 지식이나 발음 변이 규칙, 실제 음성 자료에서 발생되는 빈도 등으로부터 구하게 된다. 몇몇 인구에서는 표준 발음 표기를 이용하여 음소열을 결정하지 않고 음성 자료로부터 최적의 음소열 P^*나 복수개의 최적 음소열의 집합을 구하고 있다. 이때 최적화 기준으로는 식(4)을 주로 사용한다[4].

$$P^* = \arg\max_{P} \left\{\Pr(X|P,G)\ \Pr(P|G)\right\} \tag{4}$$

즉, 확률 발음 사전에서는 단어의 발음을 표현하기 위해 최적의 음소열을 찾는 것이 아니라 발음의 변이를 음소 단위 수준에서 모델화 하는 것을 목적으로 하고 있다. 제안된 방법에서는 음성상의 변이를 음소 상위에서 모델화 하기 위해서 ***subword-state***를 도입하였다. subword-state란 음소열을 대체하면서 실제 발음에서 음소의 실현을 나타낸다. 따라서 식(2)에서의 각 음소열에 대해 전개된 우도는 식(5)과 같이 subword-state인 $S(s_1, s_2, \ldots, s_T)$로 대체할 수 있다.

$$\begin{aligned}\Pr(X|G) &= \sum_{all\,S} \Pr(X, S|G) \\ &= \sum_{all\,S} \Pr(X|S,G)\ \Pr(S|G)\end{aligned} \tag{5}$$

식(5)의 확률을 구하는 과정은 HMM과 같은 1차 Markov 가정과 output-independent 가정을 도입하면 보통 사용된 상태열은 다음과 같이 계산한다. 먼저 임의의 상태열 S에 대해서 상태열에 대한 확률은 식(6)과 같다.

$$\Pr(S|G) = a_{s_0 s_1}\, a_{s_1 s_2} \cdots a_{s_{T-1} s_T} \tag{6}$$

여기서 $a_{ij} = \Pr(s_t = j | s_{t-1} = i)$는 subword-state들 간의 천이 확률을 나타낸다. 한편 음성 자료 X대한 조건 확률은 식(7)과 같이 나타낼 수 있다.

$$\Pr(X|S,G) = b_{s_1}(x_1)\, b_{s_2}(x_2) \cdots b_{s_T}(x_T) \tag{7}$$

여기서 $b_i(x_t)$은 각 상태에서의 출력 확률을 나타낸다. 출력 확률은 식(8)과 같이 음소 모델에의 우도 $\Pr(x_t|p_m)$과 각 음소들에 대한 가중치인 $w_i(m)$의 조합으로부터 구한다. 이때 N은 단위 음소 모델의 개수를 의미한다.

$$\begin{aligned}b_i(x_t) &= \Pr(x_t|s_t = i) \\ &= \sum_{m=1}^{N} \Pr(x_t|p_m) \Pr(p_m|s_t = i) \\ &= \sum_{m=1}^{N} w_i(m) \Pr(x_t|p_m)\end{aligned} \tag{8}$$

마지막으로, 식(6)~(8)을 식(5)에 대입하여 정리하면 식(9)과 같이 우도를 얻을 수 있다.

$$\Pr(X|G) = \sum_{all\,S} \left[\prod_{t=1}^{T} a_{s_{t-1} s_t}\, b_{s_t}(x_t)\right] \tag{9}$$

결과적으로 확률 발음 사전은 식(9)에서 보듯이 음소에 대한 확률 분포를 포함하고 있는 subword-state들의 Markov 연결로 구성된 HMM으로 모델화 될 수 있다. 이는 단위 음소 모델들이 음향 파라미터 수준에서의 HMM인 반면 확률 발음 사전은 subword 수준에서의 HMM임을 의미한다. 따라서 확률 발음 사전을 구성하는 모델 파라미터 λ는 아래와 같이 상태들 간의 천이 확률과 각 상태에서의 음소들에 대한 확률 분포로 구성된다.

$$\lambda = \{A, W\}$$

$$A = \{a_{ij} \mid a_{ij} = \Pr(s_{t+1} = j | s_t = i)\}$$

$$W = \{w_i(m) \mid w_i(m) = \Pr(p_m | s_t = i)\}$$

그림 1은 기존의 발음 사전 모델과 제안한 모델을 비교하고 있다. 그림 1 (a), (b)에서 보듯이 기존의 발음 사전에서는 baseform의 형태가 결정적(deterministic)인 반면 그림 1 (c)와 확률 발음 사전에서는 baseform의 형태가 결정적이지 않고 인식시 확률적(stochastic)으로 적용된다. 확률 발음 사전의 subword-state는 일반적인 발음 사전 모델과 비교해서 각 음소에 대응한다고 볼 수 있다. 그러나 여기서의 subword-state들은 서로간에 천이 확률이 존재하며, 각 상태는 하나의 음소만을

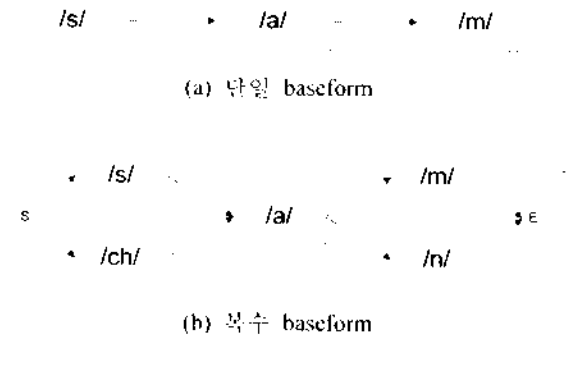
나타내는 것이 아니라 식(8)에서 보는 바와 같이 각 상태에서의 음소는 발생 확률 $w_i(n)$에 의해 결정된다. 확률 발음 사전의 baseform을 단일 baseform과 비교하면 단일 baseform은 각 단어에 대해 하나의 음소열만을 가지며, 식(10)과 같이 음소열에 해당하는 음소에 대해서만 $w_i(n)$이 값을 갖는 경우에 해당한다.

$$w_i(n) = \begin{cases} 1 & : \text{if } p_i = n \\ 0 & : \text{otherwise} \end{cases} \tag{10}$$

복수개의 baseform을 사용하는 경우 역시 식(8)은 동일한 형태인 식(11)을 사용하는 경우와 유사하다. 그러나 확률 발음사전에서는 subword-state들 간의 상태 천이 확률을 포함함으로써 같은 음소라도 문맥 내에서의 개별적인 지속 시간과 실제 발음에서의 음소의 삭제 현상을 표현할 수 있게 된다. 또한 subword-state에서의 출력 확률은 발음상에서 발생되는 음소의 치환이나 삽입 등을 보다 자세히 모델화 할 수 있게 된다.

$$b_i(x_t) = \max_n \left[w_i(n) \Pr(x_t | p_n) \right] \tag{11}$$

음성 인식에 있어서 음소 보다 작은 단위 모델의 사용을 고려해 볼 수 있다. 예를 들어 senone[5]과 같은 단위는 음소 HMM을 구성하는 각 상태에 해당하는 단위로써 음소 보다 작은 세세한 부분을 모델화한 단위라 볼 수 있다. 원래 senone은 음소 HMM의 상태 공유(state sharing)를 위해 제안되지만 각 senone은 음소 모델을 구성하는 기본 단위뿐 아니라 단어를 모델화 하기 위한 기본 단위로 사용 할 수 있다. 따라서 확률 발음 사전에서는 식(8)에서 음소 단위 대신 senone 단위를 사용하여 단어 발음 사전을 구성할 수 있으며, 음소 보다 작은 단위 모델을 사용함으로써 보다 자세한 단어 모델링이 가능하게 된다. 또한 음향학적 유사도에 의해 모델링 된 비연어학적인 인식 단위에 대해서도 같은 방법으로 발음사전에 적용할 수 있다.



[그림 1] baseform의 비교 (단어: /s a m/)

2.2 확률 발음사전의 학습

확률 발음 사전의 학습은 주어진 단어 음성 자료와 모델로부터 식(9)의 확률을 최대화 할 수 있는 모델 파라미터를 추정하는 것이다. 본 논문에서는 HMM의 학습에 주로 사용되고 있는 EM(expectation maximization) 알고리즘인 Baum-Welch 재추정 방법을 확장하여 확률 발음 사전의 학습에 적용하였다. 먼저, 확률 a_{ij}의 의미는 상태 i에서 상태 j로의 천이 확률을 의미하므로 새로운 천이 확률의 추정은 식(12)과 같이 구할 수 있다.

$$a_{ij} = \sum_{t=1}^{T-1} \gamma_t(i,j) \Big/ \sum_{t=1}^{T-1} \gamma_t(i) \tag{12}$$

마찬가지로 확률 $w_i(n)$의 의미는 상태 i에서의 음소 p_n의 발생 확률을 나타내며, 이를 계산하기 위해서는 식(13)과 같이 각 상태에서 모든 발생 가능한 음소에 대한 음소 p_n의 상대적인 출현 빈도를 이용한다.

$$w_i(n) = \sum_{t=1}^{T} \zeta_t(i,n) \Big/ \sum_{t=1}^{T} \gamma_t(i) \tag{13}$$

where,

$$\gamma_t(i,j) = \Pr(s_t = i, s_{t-1} = j | X, \lambda) = \frac{\alpha_{t-1}(i)\, a_{ij}\, b_j(x_{t+1})\, \beta_{t+1}(j)}{\sum_{i=1}^{M} \alpha_t(i)\beta_t(i)} \tag{14}$$

$$\gamma_t(i) = \Pr(s_t = i | X, \lambda) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{M} \alpha_t(i)\beta_t(i)} \tag{15}$$

$$\zeta_t(j,n) = \Pr(s_t = j, p' = p_n | X, \lambda) = \frac{\sum_{i=1}^{M} \alpha_{t-1}(i)\, a_{ij} \left[w_j(n) \Pr(x_t | p_n) \right] \beta_t(j)}{\sum_{i=1}^{M} \alpha_t(i)\beta_t(i)} \tag{16}$$

$\alpha_t(i) = \Pr(x_1, \ldots, x_t, s_t = i | \lambda)$: forward probability

$\beta_t(i) = \Pr(x_{t+1}, \ldots, x_T | s_t = i, \lambda)$: backward probability

확률 발음 사전의 학습과정을 정리하면 다음과 같다.

1) 초기화(Initialization)

확률 발음사전의 초기 모델을 만든다. 확률 발음사전은 HMM으로 표현되기 때문에 이를 구성하기 위해서는 subword-state들의 천이 확률과 각 상태에서의 확률분포 함수를 추정해야 한다. 먼저 HMM의 상태 수와 형태는 표준 발음 표기를 기반으로 하여 표준 발음 표기에서 나타나 음소의 수를 상태의 수로 정하며 이 음소들이 [그림 1] (c)와 같이 left-to-right 형태로 연결된 형태로 구성하며 상태간 skip을 허용할 수 있다. 또, 각 상태에서의 확률분포 함수는 임의로 초기화하거나 음성 인식기에 의해 음소 단위로 자동으로 분할된 각각의 음성 자료들과 음소 모델을 이용하여 음소 모델들 사이의 유사도를 구한 후 표준 발음 표기의 음소에 해당하는 상태의 확률 분포를 초기화한다.

2) 분할(Segmentation)

확률 발음사전을 학습하기 위해서는 단어로 분할 된 음성 자료가 필요하다. 따라서 음소 모델들과 표준 발음 사전으로부터 구성된 각 단어의 음소 표기열로부터 문

장을 단어 단위로 분할한다. 분할 과정은 문장에 대한 표기(transcription)를 이용하여 음소 모델의 network을 구성하고 단어 단위의 Viterbi 정합을 수행함으로써 구할 수 있다.

3) 추정(Estimation)

확률 발음사전의 모델 파라미터는 단어 단위로 분할된 음성자료를 이용하여 앞에서 언명한 식(10)-(14)의 재추정 방법을 이용한다. 계산량의 감소와 저장 공간을 줄이기 위해서 모든 음소 모델에 대해 음소 발생 확률 $w_i(m)$을 구하는 대신에 발생 확률이 높은 상위 K개의 음소만을 사용 할 수 있다. 재추정 과정에서 식(8)의 출력 확률을 구하기 위해서는 모든 음소를 대상으로 하기 때문에 계산량이 많게 된다. 특히 문맥 의존 음소 모델인 triphone과 같이 단위 모델의 개수가 많을 경우 더욱 많은 계산이 필요하게 되며 음소 발생 확률을 저장하기 위한 공간도 매우 크게 된다. 또한 다수의 음소 발생 확률 파라미터를 추정을 위해서는 많은 학습 자료가 필요로 하게 된다. 그러나 실제로 $w_i(m)$의 분포가 몇 개의 유사한 단위 모델 위주로 분포하기 때문에 적절한 K를 사용함으로써 학습 상의 문제를 해결할 수 있다.

4) 반복(Iteration)

모델 파라미터의 추정은 재추정 방법에 의해서 반복적으로 갱신된다. 과정 1에서 3은 표준 발음 표기를 기반으로 하여 모델을 추정한 것이기 때문에 보다 정확한 모델을 구하기 위해서 모델 파라미터의 추정과 확률 발음 사전을 이용한 단어 음성의 분할 과정을 반복하게 된다. 음성의 단어 단위 자동 분할과 모델 학습을 위해서는 segmental K-means 학습 방법을 사용한다[6]. 이 학습 방법은 주어진 단위 모델, 음성 자료와 발음사전으로부터 음성을 최적의 단어 분절(segment)들로 분할한다. 이렇게 분할된 분절들은 다시 단어의 발음 모델을 학습하는데 이용하는 과정을 반복하게 된다. 따라서 발음사전은 segmental K-means 학습방법을 이용한 단어 분할과 확률 발음 사전의 모델 파라미터 재추정의 반복에 의해 자동으로 생성된다.

2.3 확률 발음사전을 이용한 인식

본 논문에서 사용되는 기본적인 탐색 알고리즘은 1-pass DP(dynamic programming)[7]에 기반한 tree-trellis 탐색 알고리즘[8]을 사용한다. tree-trellis 탐색 알고리즘은 복수개의 최적 후보 문장을 찾는데 있어서 1개의 최적 문장을 찾는데 비해 거의 메모리와 시간의 증가 없이 효율적으로 찾을 수 있는 탐색 방법중 하나이다. tree-trellis 탐색 알고리즘의 구성은 시간 동기 trellis 탐색에 해당하는 전방향 탐색과 시간 비동기 tree 탐색에 해당하는 역방향 탐색으로 이루어 진다. 먼저 전방향 탐색에서는 1-pass DP 알고리즘을 이용하여 부분 경로에 대한 확률 값을 저장한 후에 역방향 탐색에서 A* 알고리즘을 이용하여 최적의 부분 경로들부터 확장해 가면서 N개의 최적 문장을 찾게 된다. 이때 A* 알고리즘에서 사용되는 부분 경로들의 평가 함수(evaluation function) 값은 1-pass DP에서 구해진 전방향 부분 경로 확률 값과 현재까지의 역방향 부분 경로 확률 값의 합을 사용하여 부분 경로 상에서 전 경로에 대한 정확한 확률 값을 알 수 있게 되므로 불필요한 부분 경로의 확장이 줄게 된다. 따라서 tree-trellis 탐색은 일반적인 stack 알고리즘에 비해 메모리의 증가가 적으면서도 빠르게 여러 개의 최적 문장들을 찾을 수 있게 된다.

전방향 탐색에서 탐색 공간은 단어와 구성하고 있는 음소 모델들을 노드로 하는 network으로 표현되며, 단어의 수가 늘어남에 따라 network을 구성하는 음소의 노드 수가 증가하게 된다. 만약 복수개의 baseform을 사용하는 경우라면 음소 노드가 늘어나게 되므로 network의 상대 크기는 더욱 커지게 되며, 이는 인식의 속도와 정확성에 중요한 영향을 미치게 된다. 실제로 복수개의 baseform을 사용하는 경우 발음 변이를 보다 자세히 표현하기는 하나 완전히 탐색 공간의 증가로 인하여 인식률의 향상을 얻지 못하는 경우가 많다. 그러나 확률 발음사전에서는 하나의 발음 표기를 사용하는 경우와 같은 수의 상태를 갖게 되므로 탐색 공간의 증가는 없게 된다. 확률 발음 사전을 탐색 과정에 적용하기 위해서는 전방향 탐색의 계산 과정을 확장할 필요가 있다. 탐색 과정에서는 상태의 경로를 알 수 없기 때문에 매 시간마다 단어의 경계들을 가정한 확률 값을 구해야 한다. 각 상태에서의 확률을 구하기 위해서는 Viterbi 알고리즘을 이용하여 식(17)과 같이 각 노드 H_{ij}에 대한 음성 파라미터 벡터 $x_1, \ldots, x_t$의 확률을 구한다.

$$Q_{it}(t) = p(x_1, \ldots, x_t | H_{ij}) = \max_k \left[D_{ik}(x_t)\, Q_{ik}(t-1) \right] \quad (17)$$

$$D_{ik}(x_t) = \Pr(x_t | s_{t-1} = k, s_t = j, H_{ij}) = \begin{cases} \Pr(x_t | p_r) & : \text{conventional lexicon} \quad (18) \\ a_{ik} \cdot b_j(x_t) & : \text{stochastic lexicon} \quad (19) \end{cases}$$

여기서 인자 t는 단어를 j는 subword-state를 나타내며, a, $b(x)$는 각각 식(9)에서의 천이 확률과 출력 확률을 나타낸다. 만약 단일 발음 표기를 사용하는 경우 식(18)과 같이 각 음소 노드에 해당하는 음소에 대해서만 우도를 계산하게 된다. 그러나 확률 발음 사전을 사용하는 경우는 식(19)에서와 같이 천이 확률과 출력확률의 곱으로 계산된다. 이때 식(19)은 식(18)을 사용할 때보다도 출력 확률을 계산하는데 있어서 모든 음소와의 우도를 계산해야 하므로 연산량이 많아지게 된다. 특히 이산 분포 HMM보다는 연속 분포 HMM에서는 이 부분의 계산량이 차지하는 비중이 크기 때문에 모든 음소에 대해 우도를 계산하지 않고, 학습 과정에서 결정된 상위 K개의 $w_i(m)$에 대해서만 계산을 수행함으로써 연산량을 줄일 수 있다. 또 현재 프레임의 음성 파라미터와의 유사도가 높은 음소 모델만을 대상으로 확률을 계산함으로써 더욱 많은 연산을 줄일 수 있다.

본 논문에서는 전방향 탐색 시 전체 탐색 공간에 대한 연산량으로 생기는 계산량의 비효율성을 개선하기 위해 beam 탐색 기법을 사용하였다. beam 탐색에서는 매 입력 프레임마다 모든 후보의 경로들을 확장하지 않고 확률이 높은 일부 후보 경로들만을 확장하게 된다. 이때 beam 탐색의 임계값을 작게 하면 탐색의 정확도는 감소하나 계산량이 줄어들게 되며, 크게 하면 그 반대가 된다. 따라서 beam 탐색에서는 정확도를 유지하면서 계산량을 줄이기 위해서는 임계값을 적절하게 선택해서 매 시간 마다 탐색 공간을 적당하게 유지시켜야 한다. 안정한 탐색 공간을 유지시키기 위한 방법으로는 현재 상태들의 확률 값을 고려하여 임계값을 조정하는 방법과 활성 상태의 수를 현재의 확률 값에 따라 제한하는 방법이 있으나 본 논문에서는 임계값과 활성 상태 수를 제한하는 방법을 동시에 사용하였다.

III. 실험 및 결과

3.1 실험 환경

제안한 방법의 성능평가를 위해서 한국과학기술원 음성 인공지능에서 제작한 무역상담용 연속음성 데이터

베이스를 사용하였다[9]. 표 3의 예와 같이 문장은 품사에 의해 분류된 3016개의 어휘로 구성되어 있으며, 남성 100명, 여성 50명이 평균 98 문장씩을 자연스럽게 발성하였다. 이중 음소 모델과 발음 사전의 학습을 위해서 남성 75명, 여성 25명이 발성한 문장을 사용하였으며, 평가를 위해서 나머지 남성 25명, 여성 25명이 발성한 문장을 사용하였다. 음성은 비교적 조용한 상태에서 16kHz로 샘플링 되었으며 매 10msec 마다 20msec 구간으로 분석되었다. 실험에 사용한 음성 파라미터는 14차 멜켑스트럼 계수와 14차 델타 멜켑스트럼 계수, 에너지, 델타 에너지를 함께 사용하였다. 인식에 사용된 기본 단위는 무음 모델을 포함하여 37개의 문맥 독립 음소 (monophone)와 3018개의 문맥 의존 음소(triphone)를 사용하였다. 음소를 모델링 하기 위해 사용한 방식은 left-to-right 형태의 HMVQM(hidden Markov VQ model)이며 각 음소 모델은 3개의 state로 구성되며 무음 모델은 1개의 state로 구성되었다[10,11].

3.2 실험 결과 및 검토

연속음성의 인식률은 단어의 인식률과 문장 인식률로 나타내었다. 단어의 오류는 치환, 첨가, 삭제에 의한 오류를 포함하여 식(20)과 같이 구하며, 문장 인식률은 단어오류가 포함되지 않은 인식 결과만을 이용하여 구하게 된다.

$$Word\ Error = \left(1 - \frac{Correct - Ins}{Correct + Sub + Del}\right) \times 100(\%) \qquad (20)$$

[표 1]은 하나의 baseform만을 사용하는 기존의 발음사전과 제안된 확률 발음사전을 연속음성인식에 사용했을 경우를 비교 실험한 결과이다. 발음사전에 따른 인식 결과는 제안된 확률 발음 사전을 사용했을 경우 단어의 오류는 27.9%(monophone), 39.8%(triphone)을 문장의 오류는 14.4%(monophone), 24.4%(triphone)의 큰 폭으로 줄일 수 있었다. 다음으로 문맥 독립 음소인 monophone 모델과 문맥 의존 음소인 triphone 모델을 사용한 경우를 비교하면 기존의 발음 사전의 경우 triphone을 사용하면 단어 오류는 74.2%, 문장 오류는 46.8%를 줄일 수 있었다. 비슷한 경향으로 확률 발음사전을 사용한 경우에 있어서도 triphone의 경우 단어 오류는 78.5%, 문장 오류는 53.3%를 줄일 수 있었다. 이는 제안된 발음 사전은 조음 결합 특성을 잘 수용하고 있는 문맥 의존 음소인 triphone에 있어서도 비슷한 성능 향상을 보임으로써 문맥을 반영한 음소 모델만으로는 단어 단위의 발음 변이를 충분히 수용 할 수 없음을 알 수 있었다. 실험 결과에서 보듯이 연속음성인식에서는 인식 단위의 모델링 못지 않게 단어 모델링에 해당하는 발음사전의 구성이 인식 성능에 중요한 영향을 미침을 알 수 있다.

[표 1] 연속음성인식 결과

lexicon	subword unit	word error (%)	sentence error (%)
conventional lexicon	monophone	41.9	74.0
	triphone	10.8	39.4
stochastic lexicon	monophone	30.2	63.8
	triphone	6.5	29.8

IV. 결 론

본 논문에서는 대어휘 연속음성인식을 위한 발음사전의 구성에 대해서 기술하였다. 제안된 확률 발음 사전은 단어내 변이와 단어간 변이를 모두 효과적으로 표현할 수 있었으며, 인식 모델과 인식기의 특성을 반영함으로써 전체 인식 시스템의 성능을 보다 높일 수 있었다. 실험 결과 확률 발음 사전을 사용함으로써 단어 오류율은 39.8%, 문장 오류율은 24.4% 까지를 감소시킬 수 있었으며, 문맥 독립 음소뿐만 아니라 문맥 의존 음소 모델과 같이 조음 결합 특성을 많이 포함하는 단위에서도 비슷한 성능 향상을 얻을 수 있었다. 그러나 제안된 발음사전 모델은 인식 시 기존의 방법에 비해 계산량이 증가되는 단점이 있다. 앞으로 시스템 실시간 구현을 위해서 tree 형태의 확률 발음사전 구조를 만드는 방법과 더불어 확률 발음 사전과 결합하여 계산을 줄일 수 있는 탐색 알고리즘 등에 관하여 계속 연구해 나아갈 것이다

참고 문헌

[1] L.R.Bahl, P.F.Brown, P.V.de Souza, R.L. Mercer, M.A. Picheny, "A method for the Construction of Acoustic Markov Models for Words," IEEE Tans. Speech and Audio Processing, Vol.1, No.4, pp. 443-452 Oct. 1993.

[2] Chuck Wooters, Andreas Stolcke, "Multiple-Pronunciation Lexical Modeling in a Speaker Independent Speech Understanding Syatem," Proc. ICSLP'94, pp. 1364-1366, Yokohama, 1994

[3] Nick Cremelie, Jean-Pierre Martens, "On the Use of Pronunciation Rules for Improved Word Recognition," Proc. Eurospeech'95, pp. 1747-1750, Madrid, 1995

[4] Torbjörn Svendsen, Frank K. Soong, Heiko Purnhagen, "Optimizing Baseforms for HMM-Based Speech Recognition," Proc. Eurospeech'95, pp. 783-785, Madrid, 1995

[5] Mei-Yuh Hwang, Xuedong Huang, "Shared-Distrubution Hidden Markov Models for Speech Recognition," IEEE Trans. on Speech and Audio processing, Vol.1, No.4, pp. 414-420, Oct. 1993

[6] Lawrence Rabiner, Biing-Hwang Juang, Fundamentals of Speech Recognition, Prentice-Hall, 1993

[7] H. Ney, D. Mergel, A. Noll, A.Paeseler, "Data Driven Organization of the Dynamic Programming Beam Search for Continuous Speech Recognition," IEEE Trans. on Signal Processing, Vol 40, No.2, pp. 272-281, Feb. 1992

[8] Frank K Soong, Eng-Fong Huang, "A Tree-Trellis Based Fast Search for Finding the N Best Sentence Hypotheses in Continuous Speech Recognition," Proc. ICASSP'91, pp. 705-708, 1991

[9] 최인정, 권오욱, 박종환, 박용규, 김도영, 정호영, 은종관, "대용량 한국어 연속음성인식 시스템 개발," 한국음향학회지, 14권, 5호, pp. 44-50, 1995

[10] Seong Jin Yun and Yung Hwan Oh, "Performance Improvement of Speaker Recognition System for Small Training Data," Proc. ICSLP'94, pp. 1863-1866, Yokohama, 1994

[11] 윤성진, 최환진, 오영환, "확률 발음사전을 이용한 대어휘 연속음성인식," 한국음향학회지, 제 16권, 제 2호, pp 49-57, 1997