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

연속 음성 인식 기법을 이용한 단어 음성 인식

조영훈°, 김석동
호서대학교 컴퓨터학부

The recognition of word by continuous speech recognition technic

Young-hoon Cho°, Suk-dong Kim
Computer Science, Hoseo University

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

요 약

우리말은 영어와는 달리 단어를 공백으로만 구분할 수 없다. 그러므로 대용량 어휘를 갖는 연속 음성을 인식하기 위한 언어모델을 만들기가 매우 어렵다. N-gram의 언어 모델을 우리말 문장에 적용하기 위해 하나의 문장을 한 단어로 구성하여 처리하였다. 우리의 인식 시스템을 평가하기 위하여 시스템 공학 연구소에서 제공한 음성을 대상으로 인식률을 계산하였다. 단어의 종류는 452개이며 한명이 이 단어들을 2번씩 발음하고 총 70명이 발음한 총 63,280개의 단어에 대하여 92.8%의 인식률을 얻었다. 일간지 사설로부터 추출한 단어를 대상으로 발음 사전을 10K 크기로 만들었다. 음성 모델은 uniphone을 사용하였다.

1. 서론

컴퓨터와의 대화 수단을 보다 인간과 가깝게 하기 위해서는 음성입력이 필요하다. 음성 처리 기술과 컴퓨터 과학의 발달로 컴퓨터에 의한 음성인식을 실생활에 응용하려는 시도가 현재 다양하게 이루어 지고 있다. 음성인식은 인식 대상에 따라 3가지로 나눌 수 있다[1]. 첫째 독립 단어 인식으로 각각의 단어 앞뒤에 묵음구간이 있어 음성 구간 검출 방법을 이용해 단어 구간을 신뢰성 있게 식별할 수 있게된다. 둘째로 제한된 연속 음성인식으로 소규모의 단어와 특별한 어구만을 이용한다. 마지막으로 대규모 어휘를 사용하는 연속음성 인식으로 수만 단어의 어휘와 임의의 길이를 갖는 문장과 자연스러운 형태의 음성을 다룬다. 이것은 사람들이 음성을 처리하는 것과 비슷하나 구현하기에는 어려운 점이 많다. 실험실 수준에서 높은 인식률을 얻을 수 있는 음성인식기술은 아직도 속도가 느리고 넓은 영역에 걸친 대규모 연속음성을 처리하기에는 많은 비용이 든다. 본 논문은 두번째 방법으로 연속 음성 인식 방법을 통한 우리말 단어 음성을 인식한다.

대용량 연속 음성 인식은 영어, 불어, 독일어, 이탈리아 및 일본어는 Wall Street Journal, Le Monde, Frunkfurter Rundschau, Sole 24 Ore, Nihon Keizai 신문[2-10]과 같은 것을 이용하여 처리하는 연구가 외국에서 많이 이루어지고 있다. 그러나 이 분야에 대하여 우리말에 대한 본격적인 연구가 이루어지고 있지 않는 실정이다. 우리말의 문장은 서양의 언어와 달리 단어 사이에 공백이 없어 문장에서 단어를 자동적으로 찾는 것은 어렵다. 그러므로 대용량 연속 음성 인식에 중요한 역할을 하는 N-grams 언어 모델을 사용하기가 어렵다. 본 연구에서는 문장단위의 음성을 인식할 수 있는 연속 음성 인식 기법을 이용하여 독립단어 63,280개를 대상으로 인식하였다. 인식에 사용한 음성은 시스템 공학 연구소에서 제공한 PBW로 단어의 종류는 452개이며 남자가 38명 여자가 32명이 각각 2번씩 발음한 것이다. 남자의 평균 인식률은 94.3%, 여자는 91.0%를 얻었다.

2. 인식 방법

오늘날의 연속음성인식에 주로 사용하는 것은 통계적인 패턴인식 방법이다. 이방법은 1970년대에 IBM의 Baker와 Jelinek가 제안한 이후 음성 인식의 기본 원리가 되었다.

인식 대상의 음성은 전처리 과정에서 음성 벡터인 $Y = y_1, y_2, \cdots, y_T$ 로 변환된다. 각 벡터들은 보통 10 msec동안의 단구간 음성 스펙트럼에 해당한다.

문장의 나열이 $W = w_1, w_2, \cdots, w_n$이라 할때 음성 인식의 최종 목표는 관찰된 음성 신호 Y와 가장 확률이 높은 단어의 나열 $W^{\wedge}$를 결정하는 것이다. 이를위해 Bayes규칙을 이용하여 $P(W|Y)$를 두개의 구성 요소로 나누면 다음과 같다.

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

이 식의 의미는 인식할 문장 W를 찾기위해서 $P(W)$와 $P(Y|W)$의 곱이 최대가 되는 단어들을 발견해 내야 한다. $P(W)$는 관찰된 음성 신호와는 무관하며 오직 일반 문장과 연관이 있다. 이 확률값은 언어 모델에 의해서 결정된다. 두번째 항인 $P(Y|W)$은 어떤 특정한 단어의 나열인 W가 주어졌을 경우 그 나열이 관찰된 음성 신호 벡터 Y인 확률을 의미하는 것으로 이 값은 음성 모델에서 결정된다.

그림1에 이들관계를 이해하기 위한 개념도를 나타내었다. 단어의 나열 W = "이것은 음성 입니다"라고 가정하고 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)이라하는 통계 모델에 대응된다. 전체의 문장을 HMM의 나열로 나타내어 하나의 복합 모델을 만들고 이 모델이 관찰되는 열인 Y를 만들어 낼 확률을 구한다. 이것이 바로 요구되는 확률 $P(Y|W)$ 이 된다. 이러한 처리 과정은 모든 가능한 단어의 열에 적용하여 가장 비슷한 열을 인식 결과로 선택한다.

3. 음성 모델

음성은 단어들로 나열(sequence of words)로 되어있다. 나열된 단어들이 W - w1,w2,..wn일때, 관찰된 음성 신호 Y에 대하여 가장 그럴듯한 단어의 열 W'을 찾는 것이 음성 인식 과정이다. 이를 위해 Bayes 규칙을 이용하면 요구되는 확률 P(W|Y)을 구할 수 있다. 즉

$$W = \arg\max_{W} P(W|Y) = \arg\max_{W} \frac{P(W)P(Y|W)}{P(Y)} \quad (3.1)$$

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 priori확률인데 이것은 언어 모델에서 계산되며 둘째항은 주어진 단어 열에 대하여 음성 벡터의 열 Y가 관찰될 확률로 이것은 음성

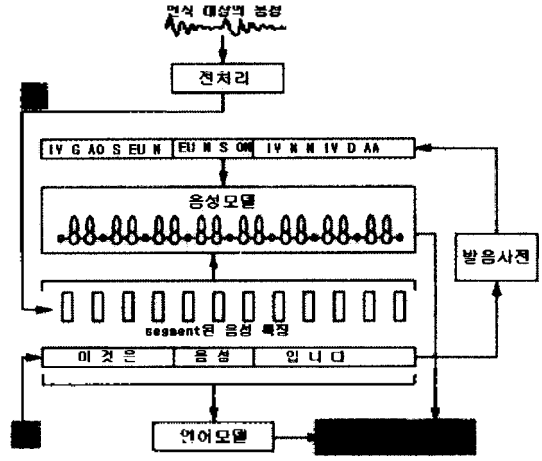

그림 1 통계적인 음성 인식 방법

모델에서 계산된다.

codeword는 연속 확률 밀도 함수으로 나타내며 혼합(mixture) 확률 밀도 함수의 조합으로 혼합 확률 밀도 함수 codebook 혹은 VQ codebook을 나타낸다고 가정한다. HMM에서 상태 st에 대하여 st가 벡터 x를 만드는 확률 밀도 함수는 다음과 같이 쓸 수 있다.

$$b_{s_t}(x) = \sum_{j=1}^{L} f(x|v_j, s_t)\Pr(v_j|s_t) \quad (3.2)$$

여기서 L은 혼합 확률 밀도 함수 codebook 크기이고 vk는 k번째 codeword를 나타낸다. 확률 밀도 함수 $f(x|v_j, s_t)$는 markov 상태 st와 무관하다고 가정한다. 그러면 주어진 상태 I에 대하여 (3.2)식은

$$b_i(x) = \sum_{j=1}^{L} f(x|v_j)b_i(j) \quad (3.3)$$

여기서 $b_i(j)$는 $\Pr(v_j|s_t = i)$이다. (3.3)식은 Parzen estimator, fuzzy VQ, multilabeling VQ와 같이 다른 비 파라미터 혹은 heuristic 방법과 유사하다. 그러나 연속 확률 밀도 함수를 사용하여 다른 heuristic 기술보다 통합된 구조로 보다 편리하게 확장할 수 있다.

실제로 (3.3)식은 인식률에 큰 영향을 주지 않으면서 각 x에 대하여 $f(x|v_j)$의 가장 영향력이 있는 값인 M으로 간략히 할 수 있다. 이것은 양자화하는 과정에서 양자화 출력을 정렬시켜 영향력이 있는 값을 가지고만 있으면 된다. $\eta(x)$를 $f(x|v_j)$의 영향력이 있는 값의 codeword의 집합이라 하면 (3.3)식은

$$b_i(x) = \sum_{v_j \in \eta(x)} f(x|v_j)b_i(j) \quad (3.4)$$

$\eta(x)$에 있는 codeword의 수는 codebook의 크기보다 작으므로 (3.4)식은 (3.3)식보다 많은 계산량을 줄일수

있다. 또한 $b_i(x)$는 $f(x|v_i)$의 합으로 정규화하는 것은 매우 중요하다.

(3.4)식으로 나타나는 semicontinuous 출력 확률은 연속 HMM과 이산 HMM사이의 교량 역할을 한다. 만약 $y(x)$가 $f(x|v_i)$의 가장 영향력이 있는 값 하나만을 (즉 x에 가장 가까운 codeword 하나만)나타낸다면 SCHMM은 이산 HMM을 의미한다. 한편 공유 확률 밀도 함수는 각 상태에서 자기 자신의 확률 밀도 함수를 가질 수 있다. 상태 i에서 이산 출력 확률 $b_i(j)$는 그 상태에 대한 weight로 볼 수 있다. 그러므로 상태 I에서 semicontinuous 출력 확률을 결정하는 유일한 것은 상태와 관련이 있는 확률 밀도 함수이다.

4. 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[11]. 너비-우선 방법은 탐색이 동시에 여러개가 병렬로 이루어지는 것으로 Bellman의 최적이론에서 나온 것으로 Viterbi decoding이 있다. 이 방법은 동적 프로그램을 이용한 알고리즘으로 입력 음성에 가장 가까운 상태 열을 탐색 공간에서 찾는 것이다. 탐색공간은 음소로부터 만들어진 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 탐색으로 제한된 탐색을 하는 것이 보통이다. 이러한 결합을 간단히 Viterbi beam 탐색이라 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*알고리즘의 변형인 stack decoding이 있다. 이 방법에는 부분적인 가정이 있는 스택이 있다. 이 가정에는 가장 비슷한 것 순서적으로 정렬이 되어있다. 여기서 부분적이라 하는 것은 입력 음성의 초기부분을 설명하는 것이며 완전한 가정은 모든 입력 음성을 말하는 것이다. 각 순간마다 스택에서 가장 비슷한 것을 뽑아낸다. 완전한 가정이 되면 출력이 이루어진다. 그렇지 않으면(완전한 가정에 도달하지 않는 경우) 하나의 단어씩 확장되고 모든 가능한 단어를 시도하여 입력 음성에 대한 부분적인 가정을 계산하고 그것을 정렬된 스택에 다시 넣는다. 이러한 방법으로 임의의 수인 N-best 가정를 만들어 중간 규모나 대규모 어휘에 대하여 가능한 단어의 열이 기하 급수적으로 커지는 것을 피하고 각 순간에서 부분적인 가정을 몇가지로 제한된 후보 단어만으로 확장하여 처리한다.

5. 실험및 결과

5.1 전처리

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.

$x(t)$ = 정규화된 캡스트럼 벡터

$$\Delta x(t) = x(t+2) - x(t-2),\quad \Delta, x(t) = x(t+4)\quad x(t-4)$$

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

$$x_0(t) = x_{0(t)}$$

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

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

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

5.2 발음 사전

발음사전은 모든 단어에 대하여 발음을 음소의 선형적인 형태로 나열하였다. 다음그림은 우리가 사용한 36개의 기본 음소에 대한 발음 사전을 보여준다. 실험에 사용한 발음 사전의 크기는 9,941개로 인터넷에 올라있는 일간지 사설을 대상으로 무작위로 선정하였다.

5.3 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 정도이다.

음소	단어	발음 사전
AA	까닭은	KK AA D AA L G EU N
AE	빼내	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 UW N
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
K	육해공군	Y UW K AE G ON G UW N
L	음식물	EU M S IY NG M UW L
M	아침	AA CH IY M
N	아니고	AA N IY G OW
NG	일등상	IY L TT EU NG S AN
OW	일곱개	IY L G OW PC KK AE
P	발표에	B AA L P Y OW EH

표 5.1 발음 사전의 예

인식에 사용한 음성은 시스템 공학 연구소에서 제공한 낭독음성(PBW)으로 연구소에서 선정한 452종류의 단어를 대상으로 하였다. 이 단어를 한사람이 2회씩 발음하여 한 사람당 904개의 음성을 발음하였다. 남자 38명, 여자 32명으로 총 70명이 발음한 63,280개의 음성에 대한 인식률을 계산하였다. 다음 표는 인식에 사용한 단어의 예를 나타낸다.

001	청화대	002	컴퓨터
003	그에게	004	위대한
005	당뇨병	006	그야말로
007	예컨대	008	분야에서
009	어두운	010	소프트웨어
	...		

표 5.2 음성단어의 예

표 5.3에 최종 인식 결과를 보이고 있다. 전체 평균 인식률은 92.8%이며, 남자 음성의 인식률이 여자의 음성보다 3 %정도 높게 나타냈다. 서울과 지방간의 인식률의 차이는 1%정도로 별 차이가 없다. 여기서 지방의 경우는 12살 이하의 성장지를 가지고 구분하여 큰 차이가 없음을 알수 있다.

남자	38명	94.3 %	서울	62명	92.9 %
여자	32명	91.0 %	지방	8명	91.8 %
전체평균	70명	92.8 %			

표 5.3 전체 인식률

6. 결론

본 연구는 음소를 기본으로 한 연속 음성 인식 방법을 이용하여 452종류의 독립 단어를 인식하여 약 93%의 비교적 높은 인식률을 얻었다. 높은 인식률을 얻을 수 있는 이유는 음질이 좋은 음성을 대상으로 하였기 때문이다. 앞으로 보통 환경에서 발생하는 음성을 대상으로 하여 실생활에 사용할 수 있는 시스템을 구축할 필요가 있다.

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] D. B. Paul and J. M. Baker, 'The Design for the Wall Street Journal-based CSR corpus," Proc. ICSLP-92, pp. 899-902

[3] J. Gauvain, L. F. Lamel, and M. Eskenazi, "Design considerations and text selection for BREF, a large French read-speech corpus," Proc. ICSLP-90, pp. 1097-1100

[4] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals, "WSJCAM0: A British English speech recognition," Proc. ICASSP-95, pp. 81-84

[5] H. J. M. Steeneken and D. A. van Leenwen, "Multilingual Assessment of speaker independent large vocabulary speech-recognition systems: SQUALE Project," Proc. EUROSPEECH-95, pp. 1271-1274

[6] P. C. Woodland, C. J. Leggetter, J. J. Odell, V. Valtchev, and S. J. Young, 'The 1994 HTK Large Vocabulary Speech Recognition System," Proc. ICASSP-95, pp. 73-76

[7] D. Pye, P. C. Woodland, and S. J. Young, "Large Vocabulary Multilingual Speech Recognition using HTK," Proc. EUROSPEECH-95, pp. 181-184

[8] L. Lamel, M. Adda-Decher, and J. L. Gauvain, "Issues in Large Vocabulary, Multilingual Speech Recognition," Proc. EUROSPEECH-956, pp. 185-188

[9] D. B. Paul, and B. F. Necioglu, 'The Lincoln Large Vocabulary Stack-Decoder HMM CSR," Proc. ICASSP93, pp. 660-663

[10] L. Lamel and R. De Mori, "Speech recognition of European languages," Proc. IEEE Automatic Speech Recognition Workshop, pp. 51-54, Snowbird, Dec. 1995

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