한국어 음성인식 플랫폼 개발현황

권오욱*, 권석봉**, 장규철***, 윤성락***, 김용래*,
장광동*, 김회린**, 유창동***, 김봉완****, 이용주****
*충북대학교, **한국정보통신대학교, ***한국과학기술원, ****음성정보기술산업지원센터

Status Report on the Korean Speech Recognition Platform

Oh-Wook Kwon*, Sukbong Kwon**, Gyucheol Jang***, Sungrack Yun***, Yong-Rae Kim*,
Kwang-Dong Jang*, Hoi-Rin Kim**, Changdong Yoo***, Bong-Wan Kim****, Yong-Ju Lee****
*Chungbuk National University, **ICU, ***KAIST, ****SiTEC

owkwon@chungbuk.ac.kr

Abstract

This paper reports the current status of development of the Korean speech recognition platform (ECHOS). We implement new modules including ETSI feature extraction, backward search with trigram, and utterance verification. The ETSI feature extraction module is implemented by converting the public software to an object-oriented program. We show that trigram language modeling in the backward search pass reduces the word error rate from 23.5% to 22% on a large vocabulary continuous speech recognition task. We confirm the utterance verification module by examining word graphs with confidence score.

I. 서론

음성인식의 상용화에 따라서 음성인식에 관심이 있는 학생 및 응용프로그램 개발자들이 늘어나고 있다. 기존에 공개된 음성인식 플랫폼으로서 HTK (Hidden Markov Toolkit) [1], Sphinx [2], Mississippi 대학 음성인식기[3], Julius [4], ezCSR [5] 등이 있으나, 관련 정보가 충분하지 않아 내부구조를 파악하여 자신의 아이디어를 구현하는 것이 어렵다. 이에 쉽게 이해할 수 있고 문서화가 잘된 교육 및 연구를 위한 한국어 음성인식 플랫폼을 개발하고 있다.

개발중인 음성인식 플랫폼인 ECHOS (Easy Compact Hangeul Object-oriented Speech recognizer) [6]는 쉽고 작으면서 한글 처리가 가능한 객체기반의 구조를 갖는다. 플랫폼은 모듈구조로 설계되어 재사용이 용이하며, 각 모듈에 대한 사용 예제 프로그램을 제공한다. 잡음제거 기능이 전처리부에 추가되고, 한국어 발음생성과 같은 한글 처리 기능이 보완되었다. 본 논문은 최근에 새롭게 추가된 ETSI 특징추출, 트라이그램을 적용한 후방향 탐색, 발화검증을 소개한다.

II. ECHOS

1. 특징

한국어 음성인식 플랫폼은 쉽고 문서화가 잘되어 초보자도 쉽게 접근할 수 있다. 또한 최근의 연구동향을 파악하여 성능향상에 필수적인 모듈을 기본으로 제공한다. 사용자가 쉽게 자신의 알고리듬을 치환하여 검증할 수 있도록 객체 지향의 프로그램 구조를 갖도록 한다. 플랫폼의 일부 모듈을 음성인식 이외의 다른 용도를 위하여 쉽게 가져다 쓸 수 있도록 각 모듈마다 독립적으로 사용할 수 있는 응용예제 프로그램을 제공한다. 고수준의 표준 라이브러리인 standard template library (STL) [7]를 사용함으로써 프로그램의 가독성을 높이고 알고리듬의 본질을 구현하는데 주력할 수 있도록 한다. 문서화를 위하여 C++ 언어에 적합한 unified modeling language (UML) [8]을 채택한다.

2. 구조및기능

ECHOS는 고립단어 인식, 연속음성인식, 음성 분할 기능을 수행할 수 있다. 사운드카드로부터 직접 입력되는 음성을 인식하는 온라인 인식과 파일에 저장된 음성을 인식하는 오프라인 인식을 지원한다. 응용 프로그램 개발을 위한 라이브러리, 음성인식 실험을 위한 도구, 음성 파일의 음소단위 분할기로서 사용 가능하다. 플랫폼은 그림 1과 같이 신호처리 및 특징추출로 이루어진 전처리부, 음향모델, 발음사전, 언어모델, 탐색 모듈, 후처리 모듈을 가진다.

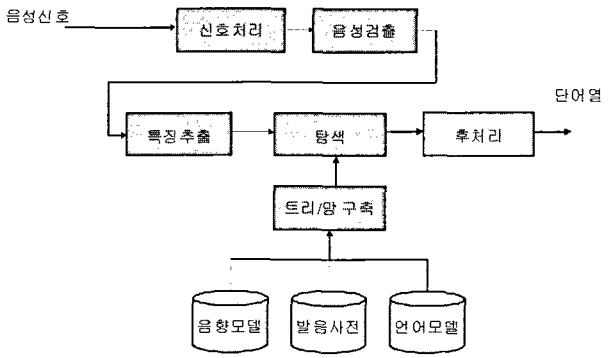

그림 1 ECHOS 구조

■ 신호처리 및 음성검출

배경잡음과 채널잡음을 제거하기 위하여 spectral subtraction, Wiener filtering 잡음제거 알고리듬을 지원한다. 음성부분을 찾기 위하여 에너지 기반의 음성검출 알고리듬을 제공한다.

■ 특징추출

특징추출 모듈은 MFCC [11], PLP[12], ETSI [13] 특징을 지원한다.

■ 음향모델

ECHOS에서는 continuous HMM[9][10]을 사용하며, HTK 호환 포맷의 음향모델을 읽을 수 있다. State-tying과 decision tree를 지원한다.

■ 발음사전

발음사전 생성기를 통해 한글 발음사전을 자동으로 생성하고, 한 어휘에 대한 다중 발음을 허용한다.

■ 언어모델

ECHOS는 소규모 태스크를 위한 FSN(Finite State Network)과 대어휘 연속음성인식을 위한 통계적 언어모델인 바이그램과 트라이그램을 지원한다.

■ 탐색

ECHOS는 FSN 탐색과 트리기반 탐색 알고리듬을 모두 지원한다. 인식률 향상을 위하여 2단계 탐색 기법을 지원한다. 먼저 bigram으로 일차적인 탐색을 하고 재차 trigram으로 stack decoding을 하여 보다 정확한 인식결과를 얻는다.

ECHOS는 1-best 인식결과와 워드그래프 형태로 인식결과를 제공한다. 워드그래프 인식결과로부터 backward tracking을 통해 N-best 인식결과를 제공한다.

인식결과로 주어지는 워드그래프를 처리하여 그래프의 단어 링크의 confidence score를 추출하여 제공함으로써, 화자적응 및 핵심어 검출에 활용될 수 있는 발화검증 기능을 가진다.

■ 후처리

현재 버전에서는 별도의 기능이 없지만 향후에 보완될 계획이다.

3. EAPI

응용프로그램 작성을 위하여 독자적인 응용 프로그램 인터페이스(API) 규격을 제공한다. EAPI (ECHOS Application Program Interface)는 ECHOS와 응용프로그램의 인터페이스를 나타낸다. EAPI 규격은 사용자 수준에 따라서 저수준과 고수준의 두 단계로 제공된다.

4. 클래스구조

ECHOS는 그림 2와 같이 구성된다. SpeechRecog는 사용자 프로그램과의 인터페이스를 담당한다. SearchBase는 탐색모듈. 인식에 필요한 모든 모듈을 관리하고, 탐색 알고리듬에 따라서 해당하는 탐색객체를 호출한다. SearchTree는 대어휘를 위한 lexical 트리를 탐색하고, SearchNetwork는 소규모 또는 중규모 어휘를 갖는 음성인식을 위한 FSN을 탐색한다. AudioIO는 사운드카드 또는 파일로부터 음성을 읽어 들인다. Feature는 입력신호로부터 잡음을 제거하고 특징을 추출한다. AModel은 음향모델로서 입력된 특징에 대한 로그확률을 계산한다. LModel은 언어모델로서 이전의 단어열이 주어질 때 현재단어의 로그확률을 계산한다. PostProc는 후처리 모듈로서 Lattice입력에 대하여 다른 지식원을 사용하여 향상된 인식결과를 제공한다.

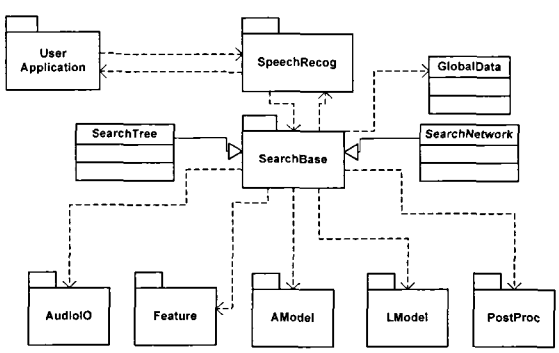

그림 2 ECHOS 클래스 다이어그램

III. 추가 기능

1. ETSI특징추출

ETSI특징추출은 분산음성인식을 위한 전처리기로서 ETSI에서 표준으로 정한 알고리듬이다[14]. 이 알고리듬은 잡음제거와 MFCC 특징 계산으로 구성된다. 잡음제거 기법은 2단계 mel-warped Wiener filtering을 거친 후 SWP(SNR-dependent waveform processing) 과정을 거친다. 8kHz, 11kHz, 16kHz의 샘플링 주파수를 처리할 수 있다[14].

ETSI의 Aurora-3 프로젝트에서 표준으로 정한 특징추출 프로그램은 C언어로 작성 되었지만 ECHOS에서는 객체지향언어인 C++로 변환되어 Feature 패키지 내에 그림 3과 같이 삽입되었다. FeEtsi는 입력신호로부터 특징추출까지 모든 작업을 관리한다. NoiseSupX는 Wiener 필터에 의한 잡음을 제거하고, WaveProcX는 피치 신호를 강조해 준다. CompCepsX는MFCC를 계산하고, PostProcx 버퍼에 남은 데이터를 처리한다. VAD는 음성 여부를 판단하고, DataFor16kProc는 16kHz 데이터 처리를 위한 모듈이다.

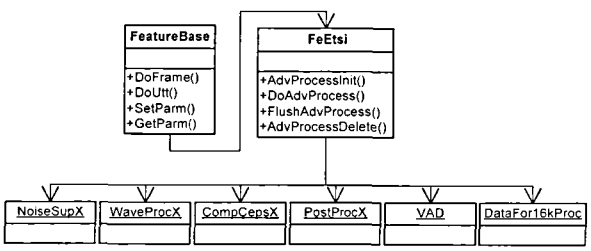

그림 3 ETSI 특징추출 클래스 다이어그램

2. 트라이그램 적용한 후 방향탐색

ECHOS는 인식률 향상을 위하여 2단계 탐색 기법을 제공한다. 1단계에서는 bigram을 적용한 정방향 프레임 동기식 Viterbi 탐색을 하여 워드그래프 형태의 인식결과를 얻고, 2단계에서는 1단계의 결과로부터 trigram 적용한 후방향 A* 탐색을 한다. 연속음성인식에서 A* 탐색에 사용되는 탐색공간이 정방향 탐색공간보다 매우 작고, 이미 저장되어 있는 음향모델과의 likelihood 값을 이용하기 때문에 A*탐색 속도는 정방향에 비해 무시할 정도로 작다.

ECHOS의 연속음성인식 검증을 위해 SiTEC의 Dict01 (낭독문장 음성 DB)로부터 훈련된 음향모델과 2단계 탐색으로 위해 두 가지 언어모델(bigram, trigram)을 사용하고, 인식 테스트는 8,670개의 단어로 구성된 발음사전과 훈련에 사용되지 않은 테스트 음성 105개를 사용하였다.

표 1을 보면 ECHOS에서 하나의 렉시컬트리를 사용하여 언어모델을 적용[6]하기 때문에 플랫렉시콘에서의 인식결과 보다 40% 상당의 인식성능 저하를 보이고 있음을 알 수 있다. Bigram 언어모델을 사용하여 정방향 탐색만으로 인식할 경우 76.5%의 인식률을 보이고 있고, 정방향 탐색결과인 워드그래프에서 trigram을 적용한 후방향 탐색을 하였을 경우 인식률이 80.1%로 15%정도의 인식성능 향상을 보여주고 있다. 또한 후방향 탐색을 추가로 적용하여도 정방향 탐색만을 사용한 경우의 인식속도와 거의 차이를 보이지 않고 있다.

표 1 2단계 탐색기법을 적용한 인식률(%) 비교분석 (렉시컬 트리에서 정방향 탐색만 했을 경우의 인식속도를 1로 함)

탐색방법	인식률	속도
정방향 bigram Viterbi (플랫렉시콘)	85.8	5
정방향 bigram Viterbi (렉시컬트리)	76.5	1
정방향 bigram Viterbi 탐색(렉시컬트리) + 후방향 trigram A* (워드그래프)	80.1	1

3. 발화검증

ECHOS에서 발화 검증을 위하여 사용한 confidence measure는 워드그래프를 사용한 단어의 사후확률 값이다[15].

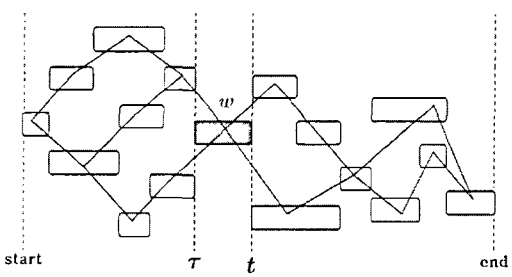

그림 4 워드그래프 예제

앞의 그림에서 보듯이 워드 그래프를 사용하여 모든 가능한 문장에서 특정 구간 τ와 t사이에 해당하는 단어 ω가 나올 사후확률을 모두 구하고 이들의 총합을 confidence measure로 사용하였다. 이를 식으로 나타내어 보면 아래와 같다.

$$p([w;\tau,t]|x_1^T) = \sum_{\substack{[w;\tau,t]_1^M: \\ \exists n\in\{1,\cdots,M\}: \\ [w_n;\tau_n,t_n]=[w;\tau,t]}} \frac{\prod_{m=1}^{M} p(x_{\tau_m}^{t_m}|w_m)p(w_m|w_1^{m-1})}{p(x_1^T)}$$

x_1^T는 시간 1에서 T까지의 음성 특징 벡터를 뜻하고, $[\omega;\tau,t]_1^M$ 은 M개의 단어 가설 $[\omega_1;\tau_1,t_1]$,..., $[\omega_M;\tau_M,t_M]$ $(\tau_1=1, t_M=T)$을 뜻한다. 여기서 단어 가설이란 시간 τ와 t사이에 워드 ω가 나오는 것을 말한다.

렉시컬 트리 탐색 방법을 사용하여 워드 그래프를 생성하여 위의 방법대로 사후확률을 계산하여 confidence measure를 구하였다. 하나의 테스트 음성에 대하여 2-best 인식결과를 뽑아 각각의 confidence measure를 구해봄으로써 계산된 값을 확인해보았다. 확률값은 log likelihood값으로서 비교하기 좋도록 적당히 스케일하였다.

표2 2-best 결과에 대한 confidence measure

1-best	그냥	그런	느낌이	들어서	해	본	겁니다
measure	2.197	1.994	1.168	1.027	0.427	0.159	0.409
2-best	그냥	그런	느낌이	들어서	해	본	뭡니까
measure	2.197	1.994	1.168	1.027	0.427	-0.432	-1.077

위의 표에서 1-best 결과는 테스트 음성과 일치하였다. 2-best 결과는 마지막 단어하나가 틀리게 나왔는데, 이 때의 confidence measure값이 1-best보다 작게 나옴을 알 수 있다.

IV. 결론

한국어 음성인식 플랫폼에서 최근에 추가된 기능 및 실험결과를 소개하였다. 플랫폼은 쉽고 작으면서 한글 처리가 가능한 객체기반의 구조를 가진다. 8000단어 연속 음성인식 태스크에 대하여 공개 음성인식기인 HTK와 성능을 비교하였다. 이 플랫폼은 국내 음성인식 분야의 저변을 확대하고, 연구자들이 알고리듬 연구에 전념할 수 있는 토대를 마련하며, 음성인식기술의 비교 기준의 역할을 할 것으로 기대된다.

감사의 글

이 논문은 음성정보기술산업지원센터의 연구비 지원으로 휴먼인터페이스연구조합을 통하여 "한국어 음성인식 플랫폼 개발" 과제에서 수행한 내용입니다.

참고문헌

[1] HTK Home page. http://htk.eng.cam.ac.uk

[2] CMU Sphinx: Open Source Speech Recognition. http://www.speech.cs.cmu.edu/sphinx/Sphinx.html

[3] Automatic Speech Recognition: Software. http://www.isip.msstate.edu/projects/speech/software/

[4] Multipurpose Large Vocabulary Continuous Speech Recognition Engine Julius. http://www.ar.media.kyoto-u.ac.jp/members/ian/doc

[5] ezCSR. http://speech.chungbuk.ac.kr/~owkwon/srhome/index.html

[6] 권오욱, 김회린, 유창동, 김봉완, 이용주, "한국어 음성인식 플랫폼의 설계," 말소리, 제51호, 2004.9.

[7] Standard Template Library Programmer's Guide. http://www.sgi.com/tech/stl/

[8] Practical UML: A Hands-On Introduction for Developers- by Randy Miller.

[9] L. Rabiner and B.-H. Juang, Fundamentals of Speech Recognition, Prentice-Hall, 1993.

[10]F. Jelinek, Statistical Methods for Speech Recognition (Language, Speech, and Communication), MIT Press, 1999.

[11]S.B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," IEEE Trans. ASSP, vol. 28, pp. 357-366, Aug. 1980.

[12]H. Hermansky, "Perceptual linear predictive (PLP) analysis of speech," Journal of the Acoustical Society of America, vol. 87, no. 4, pp. 1738-1752, 1990.

[13]Aurora, Distributed Speech Recognition. http://portal.etsi.org/stq/kta/DSR/dsr.asp

[14]ETSI Standard, Final Draft ETSI ES 202 050 v1.1.3 (2003-11), Speech Processing, Transmission and Quality Aspects (STQ); Distributed speech recognition; Advanced front-end feature extraction algorithm; Compression algorithms.

[15]F. Wessel, R. Schluter, K. Macherey, and H. Ney, "Confidence measures for large vocabulary continuous speech recognition," IEEE Trans. Speech Audio Processing, vol. 9, pp. 288-298. Mar. 2001.