SVAPI 1.0 환경에서의 어구 종속 화자 확인 시스템

김유진, 조태현, 김지운, 정재호
인하대학교 전자공학과 DSP 연구실

Text-dependent Speaker Verification System in SVAPI 1.0 Environment

Yu-Jin Kim, Tae-Hyun Cho, Ji-Un Kim, Jae-Ho Chung
DSP laboratory, Dept. of Electronic Engineering, INHA Univ.
EuGene@cc.inha.ac.kr
Phone: +82-32-860-7420, Fax:+82-32-868-3654

Abstract

본 고는 SVAPI(Speaker Verification Application Programming Interface) 1.0 환경에서의 어구 종속 화자 확인(Text-dependent Speaker Verificaiton) 시스템에 대해 기술한다. 구현된 시스템은 궁극적으로 상용 전화망 응용이 가능한 실용 시스템을 목표로 개발되었으며 이를 위해 SVAPI 위원회에 의해 제안된 SVAPI 1.0을 개발 환경(framework)으로 사용하였다. SVAPI는 객체 지향 구조, 클라이언트-서버 및 telephony 환경의 지원 등이 특징이며 어플리케이션과 엔진을 독립적으로 개발할 수 있는 이점을 제공한다.

구현된 데모 시스템은 펜티엄 프로세서와 Windows 95/NT 4.0 운영 체제 그리고 Win16/Win32 API를 통해 제어 가능하며 음성 입력이 가능한 디바이스(일반적으로 사운드 블래스터 호환 카드)를 장착한 IBM 호환 PC 이다. 화자의 성문(聲紋, voiceprint) 등록은 화자가 동일한 어구를 3회 발성하여 이뤄지며 등록과 확인의 응답 속도는 모두 1초 이내이다.

소프트웨어의 구성은 크게 어플리케이션과 어구 종속 화자 확인 엔진으로 구분할 수 있으며 엔진은 끝점 검출 알고리즘, 음성 특징 추출 알고리즘 그리고 연속 HMM기반의 화자 성문 모델 등록 및 유사도(similarity) 계산 등을 포함한 확인 알고리즘으로 구성되어 있다. 화자의 성문은 이름과 같은 약 3음절 이상의 단어로 등록되고 테스트되었다.

엔진의 객관적인 평가를 위해 전화선을 통해 남자 6명, 여자 3명의 화자로부터 자신의 이름을 각각 40회 발성하여 구축된 음성 데이터 베이스를 사용하였으며 실험 결과 남자는 2.85%, 여자는 2.44%의 EER(Equal Error Rate)을 각각 얻었다.

1. 서론

개인의 정보 욕구가 증가하고 유무선 통신망, 인터넷과 같은 통신 인프라가 급속도로 보급되면서 다양한 망을 통한 음성 인터페이스 기술의 실용화 요구가 증대되고 있다. 특히 최근에는 음성 인식 기술과 더불어 다양한 망을 통한 데이터나 시스템의 접근을 쉽고 효율적으로 관리하기 위한 기술로서 화자 인식 기술에 대한 관심이 높아지고 있다.

화자 인식 기술은 Biometric 기술의 한 분야로 볼 수 있으며 이는 인간의 신체 특징을 사용하여 신원(identity)을 구별 또는 확인하는 기술이다. Biometric이란 물리적 또는 생리학적인 개인의 특성으로서 개인의 손재를 다른 사람과 구분할 수 있는 요소를 말한다. 이러한 화자 인식 기술은 지문, 망막, 얼굴 그리고 음성과 같은 신체의 일부를 사용한다. 특히 음성 신호 처리 기술과 집적된 화자 인식 기술은 음성을 개인의 고유한 특징으로 간주하여 화자를 인식하는 기술이다.[1]

이러한 화자 인식 기술 중에서 특히 선언된 화자의 음성과 초기에 등록된 동일 화자의 성문을 비교하여 본인임을 확인하는 기술인 화자 확인 기술은 작게는 개인 전자 문서의 관리로부터 크게는 자동화된 은행 업무 및 전자 상거래에 이르기까지 광범위하게 응용될 수 있으리라 사료된다.

특히 음성을 이용한 화자 확인 기술이 선호되는 이유는 카드, 도장, 서명 그리고 신분증 등의 물리적인 수단이 가진 도난이나 위조의 문제점이 전혀 없으며, 다른 biometric인 지문 또는 망막을 입력받기 위해 고가의 스캐너 장비가 필요한 반면 음성은 상대적으로 저가인 마이크 또는 유무선 전화를 통해 원거리에서 쉽게 처리될 수 있기 때문이다.[1]

화자 인식 기술에 대한 연구는 이미 1960년대부터 음성 인식 기술과 함께 연구되어 왔으며 1970년대 중반부터 Texas Instruments의 소규모 화자 인식 시스템과 AT&T Bell Lab.의 시스템들을 통해 실용화에 대한 연구 결과가 선보이기 시작했다. 최근에는 음성 인식 기술의 연구와 함께 실험실 결과로서 EER(Equal Error Rate)이 약 1% 미만인 기술 수준이 발표되고 있으며 최근에는 실용성을 높이기 위해 유무선 전화망을 이용한 화자 확인 기술 및 실용화 연구가 한창 진행되고 있다.[2, 3, 4] 특히 미국의 경우 다수 고객 서비스에 응용된 Calling Card Service시스템이 미국의 Sprint 통신 회사에 의해 1995년부터 이미

선보이고 있다.

한편 화자 인식 기술의 신용화에 대한 요구가 증대되면서 1996년 관련 개발자, 업체 등이 구성한 컨소시엄은 화자 인식 기술을 구현하기 위한 표준 API(Application Programming Interface)를 발표했다. 일명 SVAPI라 불리는 이 표준은 엔진 개발자와 응용 프로그램 개발자 모두가 서로 다른 엔진 또는 응용 프로그램에 적응하기 위해 소모되는 시간과 노력을 없애는 동시에 그러한 결과로 발생 가능한 위험을 줄이고자 하는 것이 목적이다.[1]

본 고에서는 현재까지 제안된 화자 확인 기술 및 실용화 기술들을 토대로 구현된 화자 확인 시스템에 대해 기술한다. 2장에서 화자 확인 시스템의 종류에 대해 살펴보고 3장에서는 시스템 구현을 위해 이용한 SVAPI 환경을 설명한다. 4장과 5장에서는 시스템과 엔진에 대해 개괄적으로 설명하며 끝으로 6장에서 엔진의 평가 결과에 대해 기술한다.

2. 화자 확인 시스템의 종류

화자 확인 시스템은 사용하는 어구의 종류에 따라 어구 종속(Text-dependent), 어구 독립(Text-independent) 그리고 어구 종속 시스템의 변형으로서 어구 지시(Text-prompted) 시스템으로 구분할 수 있다.[3]

● Text-Independent SV(TI SV) 시스템

TI SV 시스템에서는 어구의 의미와는 무관한 음성의 특징을 추출하여 화자의 성문을 생성한다. 이러한 특징으로는 피치(pitch), 톤(tone) 그리고 음색 등이 있다. TI SV 시스템의 가장 큰 장점으로는 동일한 어구를 여러 번 발성해야 하는 등록 과정이 필요 없다는 점이다. 물론 한번의 등록 과정을 거치야 하지만 상대적으로 매우 간단하다. 또한 어구의 내용과 무관하므로 화자가 등록된 어구를 기억할 필요가 없다. 그리고 음성 인식과 같은 보조 기술이 필요 없다는 점도 장점으로 꼽을 수 있다. 하지만 녹음에 의한 사칭자를 막을 수 없는 치명적인 단점이 있다.

● Text-dependent SV(TD SV) 시스템

TD SV 시스템에서는 암호, 카드 번호 또는 PIN(Personnel Identity Number)번호와 같은 특별한 어구를 발성하게 하여 화자의 신분을 확인한다. 기술적인 관점에서 TD SV 시스템은 가장 강력하다고 할 수 있다. 하지만 음성 인식 기술과 결합되어야만 하며 TI SV 시스템에 비해 좀 더 어렵고 복잡한 등록 과정이 요구된다. 한편 TI SD 시스템과 같이 녹음에 의한 사칭자를 막기 어렵다.

● Text-Prompted SV(TP SV) 시스템

TD SV 시스템 역시 화자의 특징 단어 또는 어구를 녹음한 사칭자를 막기 어려우므로 매번 다른 암호를 발성하도록 하는 보다 강력한 보안이 필요하다. 이것은 미리 선정된 여러 단어 중에서 임의로 선택된 단어 또는 어구의 발성을 요구하여 화자 확인 과정에 이용하는 것으로 구현될 수 있다. 보통 일련의 숫자 조합으로 이러한 시스템을 구성할 수 있다. 하지만 화자 확인 기술과 함께 지시된 어구를 알려주는 음성 합성 기술 그리고 지시된 어구를 정확히 발성한지를 알아내는 인식 기술 등이 조합되어야 한다.

어구 종속 또는 어구 독립 화자 확인 시스템은 고립 단어 음성 인식 기술과 같이 언어에 대한 보조 지식이 없어도 구현 가능한 기술, 즉 언어 독립적인 기술이라 할 수 있다. 한편 어구 종속 화자 확인 시스템은 기술적인 면에서 가장 간단한 시스템이면서 동시에 가장 응용 분야는 광범위하다고 할 수 있다. 반면 어구 지시 화자 확인 기술은 언어에 종속적인 지식들과 함께 합성, 인식 등과 같은 여러 기술의 결합이 필요하므로 기술적으로 가장 복잡한 시스템이라 할 수 있다. 하지만 보다 강력한 보안이 필요한 신용 거래, 홈뱅킹 같은 어플리케이션에 반드시 요구되는 SV 기술이라고 할 수 있다.

3. SVAPI(Speaker Verification Application Programming Interface)

SVAPI는 화자 확인 기술을 구현하기 위한 개방형, 플랫폼 독립적, 객체 지향의 프로그래밍 인터페이스이다. 다시 말해 SVAPI는 화자 확인 시스템을 구현하고 운용하기 위한 종합적인 틀(framework)이라고 할 수 있다. 따라서 SVAPI는 엔진과 어플리케이션 개발자가 공통의 표준을 통해 서로 독립적인 개발이 가능하도록 한다.

또한 SVAPI는 인터넷을 포함한 다양한 망과 데이터를 거치면서 사용되는 클라이언트-서버 환경에서 동작하도록 지원하고 있다. SVAPI는 native 버전과 java 버전으로 나눌 수 있는데, native 버전은 C++을 이용하여 구현된 API를 지칭하며 엔진과 어플리케이션은 C++(또는 C)를 통해 구현되어진다고 가정한다. 하지만 java로 구현된 엔진 또는 어플리케이션과 상호 동작(interoperable)이 가능하도록 설계되어 있다.[1]

SVAPI를 이용한 SV system은 크게 어플리케이션 계층, API 계층 그리고 마지막으로 engine 공급자에 의해 제공되는 engine 계층의 3부분으로 나눌 수 있다. 이들 계층은 객체 지향 언어의 클래스 특성을 이용하여 객체들간의 메세지에 의해 제안된 방법으로 연결되어 있다. 어플리케이션 계층은 API계층을 통해 DLL(Dynamic Linking Library)의 형태로 엔진 공급자가 제공한 엔진으로의 접속을 시도하며, 접속된 엔진과 어플리케이션은 발성, 모델, 스코어 등의 객체를 주고 받으며 구현된 화자 확인 기술을 사용할 수 있다. (그림 3-1. SVAPI Framework Overview)

현재 SVAPI는 소스와 Microsoft Windows 95/NT용 버전이 공개되어 있으며 곧 Unix용 버전도 공개될 예정이라고 한다.

SVAPI를 따르는 화자 확인 어플리케이션 또는 엔진을 구현하기 위해서는 SVAPI를 구성하고 있는 객체, 즉 구체적인 클래스들을 구현해야 한다. 이들은 구현하는 계층에 따라 어플리케이션에 의해 구현되는 객체와 엔진에 의해 구현되는 객체로 나누어지며 어떤 것은 순수 추상 클래스(pure abstract class)의 형태로 존재한다.(그림 3-2. SVAPI의 클래스 계층)

따라서 이들 클래스들을 인스턴스화된 객체로

사용할 수 있도록 구현하기 위해서는 미리 정의된 객체들의 의미와, 객체들 간의 관계를 나타내는 메쏘드를 이해해야 한다.

4. 시스템 구성

구현된 데모 시스템의 하드웨어는 Windows 95/NT를 운영체제로 삼는 인텔의 펜티엄 메인 프로세서를 장착한 IBM PC호환 기종이다. 또한 Windows 95/NT에서 사용 가능하며 8KHz, 8bit μ-law 또는 선형 PCM 디지털 파형을 입출력 할 수 있는 사운드 카드가 있어야 한다.

테스트된 최소 사양의 시스템은 펜티엄 133, 16Mbyte RAM을 기본 사양으로 하는 국산 노트북 컴퓨터였다. 내장된 사운드 블래스터 호환 장치는 데스크탑 PC에서 사용되는 사운드 블래스터보다 상대적으로 잡음이 상당히 심한 편이었지만 화자 확인 결과에 큰 영향은 없었다. 또한 마이크의 성능도 음질에 어느 정도 영향을 미치는데 일반적인 마이크보다 Electret 마이크가 좀 더 좋은 성능을 보이는 것으로 나타났다.

소프트웨어는 Visual C++ 5.0을 이용하여 개발된 어플리케이션과 엔진으로 구성된다. 어플리케이션은 여러명의 화자를 등록하거나 확인할 수 있도록 구성되어 있으며 요구된 화자의 성문 모델이 없을 경우 등록 절차를 거치게 된다.(그림 4-2.)

등록 절차는 동일한 어구를 3번 반복하여 발성함으로써 끝난다. 만약 성문 모델이 존재할 경우 자신이 등록한 어구를 발성함으로써 화자 확인 테스트를 거칠 수 있다. 결과로서 본인임을 주장한 화자에 대한 수락/거부를 알 수 있다.(그림 4-2)

성문 모델 생성과 확인에 소요되는 시간은 모두 1초 이내로 화자가 시간 지연을 느끼지 못할 정도이다.

5. 엔진 설계

화자 확인 기술은 크게 음성 특징 추출, 성문 모델링 및 유사도 계산 부분 그리고 계산된 유사도에 대한 가설 테스트(hypothesis testing) 부분으로 나눌 수 있다.[5] 구현된 화자 확인 엔진은 끝점 검출을 포함한 LPCC/MFCC 추출 루틴으로 음성 특징 추출 부분을 구현하였으며 성문 모델링 및 유사도 계산 부분은 HMM 알고리즘을 사용하였다. 마지막으로 hypotesis testing부분은 정규화된 스코어와 분덕값을 비교하는 간단한 Null Hypothesis Testing 방법을 사용하였다.

끝점 검출 루틴은 공중전화망을 거쳐 8KHz, 8bit μ-law 방식으로 샘플링되는 음성을 기준으로 역시 같은 환경을 기준으로 제안된 한국 통신의 끝점 검출 알고리즘을 참고하여 설계하였다.[6] 음성 파형은 16bit 선형 PCM 파형으로 변환되어 처리되고 끝점 검출 루틴은 약 100msec의 묵음 구간을 가정하여 에너지와 영교차율의 분덕값을 설정하고 이를 기준으로 음성 구간을 검출한다. 끝점 검출을 위한 분석 구간은 작을수록 좀 더 정확한 끝점을 검출 할 수 있는데 데모 시스템은 5msec의 분석 주기와 10msec의 길이의 분석 구간을 가진다.

음성 특징은 일반적으로 음성 인식에서 효과적인 것으로 알려진 LPCC와 MFCC를 각각 사용한다. 초기에 개발된 엔진에서는 F-ratio의 측정을 통해 LPCC보다 MFCC가 화자 확인을 위한 변별력이 뛰어난 것으로 나타났다.[7] 추출된 음성 특징은 일반적으로 프레임간의 변화를 수용하기 위해 미분값을 추가하는 것이 성능 향상에 도움이 되는 것으로 알려져 있는데 데모 시스템에서는 1차 미분값을 추가한 음성 특징 벡터를 사용하였다. 음성 특징 추출을 위한 분석 주기는 8msec, 분석 구간은 16msec 길이로 정했다.

성문 모델은 발성의 길이에 따라 가변적으로 결정되어지며 다중 가우시안 분포를 갖는 5개 이상의 출력 상태를 가지는 모델을 사용하였다. 한편 문턱값 설정은 엔진의 성능을 좌우하는 요소로서 음성 특징과 성문 모델의 형태에 따라 다르다. 현재 구현된 엔진에서는 사칭자의 경우 -20 이상의 유사도를 갖는 것으로 나타났다.

이러한 화자 확인 기술을 SVAPI 기반의 엔진으로 구현하기 위해 SV_Utterance, SV_Model, SV_Engine등의 클래스를 설계하였다. SV_Utterance는 끝점 검출 및 특징 추출 알고리즘을, SV_Model은 HMM 알고리즘을, SV_Engine은 hypothesis testing 알고리즘을 각각 포함한다.

6. 엔진 평가

엔진의 성능은 각각 온라인과 오프라인으로 평가되었다.

온라인 테스트는 구현된 시스템에서 화자가 직접 마이크를 통해 등록하고 확인하는 방식이며 오프라인 테스트는 미리 전화선을 통해 수집된 데이터베이스를 통해 평가하는 방식이다. 각각의 평가에서 성문 등록을 위한 발성은 3번으로 가정하였다.

온라인 테스트는 테스트 조건이 일정하지 않으므로 객관적인 결과를 제시할 수 없지만 자신의 이름을 성문 모델로 등록하고 확인했을 경우 만족할 만한 성능을 나타내었다.

오프라인 테스트는 좀 더 객관적인 성능을 평가하기 위해 수집된 데이터 베이스를 사용하였다. 평가에 사용된 음성 데이터 베이스는 나래 이동 통신의 협조로 구축되었으며 전화를 직접 사용하여 6명의 남자 화자와 3명의 여자 화자가 자신을 이름을 각각 40회를 발성하여 구축하였다. 발성된 음성은 Dialogic의 전화선 입력 보드를 통해 8KHz, 8bit μ-law 의 디지털 음성 파형으로 변환되어 저장하였다.[6]

테스트는 각각 남, 여를 구분하여 이루어졌는데 이는 남녀의 사이가 매우 크므로 남녀간의 확인 에러를 포함시킬 경우 전체적으로 False Accept 오류만을 낮추는 효과를 나타내고 상대적으로 False Reject 오류를 계산하기 위한 데이터의 개수가 작기 때문이다. 또한 EER(Equal Error Rate)을 계산하는 과정에서는 두 가지의 에러가 정규화되므로 한 가지의 에러가 낮거나 높은 것이 영향을 미치지 못하는 이유도 고려되었다.

화자의 성문 모델을 등록하기 전에 오프라인으로 끝점 검출을 수행하여 등록과 확인에 사용되는 최종 음성 파형을 준비했으며 이때 숨소리와 히소리 등의 발성 잡음으로 인해 끝점이 정확하게 검출되지 않는 음성 데이터도 등록과 확인에서 제외시키지 않았다. 결과적으로 평가에 사용된 음성의 길이에 대한 정보를

표 1.에 나타내었다.

	화자	평균 (samples)	표준편차 (samples)
남	10200	7675	757.9
	10204	4565	1297.4
	10246	7230	1144.3
	10291	5718	923.2
	10292	6016	694.2
	21234	5765	956.8
여	21235	8893	869.5
	21236	4358	512.8
	21237	3948	719.1

표 1. 각 화자의 발성 길이

화자의 성문 모델은 40회의 발성 중 3회를 선택하여 이루어졌으며 나머지 37회의 발성을 False Reject 오류를 평가하기 위해 사용하였으며 남자의 경우 나머지 5명에 대한 발성 모두를 그리고 여자의 경우 나머지 2명에 대한 발성 모두를 False Accept 오류를 평가하기 위해 사용하였다.

또한 등록에 사용된 발성 상태에 따른 오류를 최소화하기 위해 총 40회의 발성 중 순차적으로 3회씩을 선택하여 이상과 같은 실험을 화자 당 약 12~13회 반복 수행하였다.

실험 결과 분석된 EER을 계산하여 가장 좋은 것과 가장 나쁜 EER을 제외한 나머지 실험 결과를 이용하여 평균 EER을 계산하였다. 보통 가장 나쁜 EER을 보인 경우는 등록에 사용된 음성에 발성 잡음이 포함된 경우가 대부분이었다.

실험 결과는 표 2.와 같다.

	화자	EER(%)	평균(%)
남	10200	4.18	2.85
	10204	6.52	
	10246	0.86	
	10291	1.42	
	10292	1.43	
	21234	1.03	
여	21235	0.95	2.44
	21236	2.66	
	21237	1.42	

표 2. 각 화자의 EER

실험 결과 남자의 경우 10200, 10204 화자의 EER이 상대적으로 높게 나타났다. 특히 10204화자의 EER이 높은 것은 상대적으로 발성의 길이가 매우 짧고 변화가 심한 것에서 그 이유를 찾을 수 있다.

7. 결론

본 논문은 공중 전화망 어플리케이션 적용을 목표로 개발된 어구 종속 화자 확인 엔진 및 데모 시스템에 대하여 기술하였다.

특히 구현된 시스템을 실용적인 시스템에 적용될 수 있도록 SV 컨소시엄에서 제안한 SVAPI를 개발 및 운용 환경으로 사용하였다.

구현된 어구 종속 엔진은 데모 시스템을 통한 온라인 테스트와 미리 수집된 데이터 베이스를 통한 오프라인 테스트로 평가되었다. 성문은 0.5~1초 사이의 어휘를 3번 발성하여 등록하였다. 오프라인 테스트에서 엔진의 평가 결과 남, 여 각각 2.85%, 2.44%의 EER을 나타내었다.

현재 데모 시스템에서 구현된 엔진은 추후 전화선 환경을 고려한 최적화로 보다 나은 성능을 기대할 수 있을 것으로 사료된다.

앞으로 채널 잡음 및 주변 잡음에 강인한 끝점 검출 알고리즘, 화자에 따른 문턱값의 정규화 방법 그리고 성문 모델의 적응 기술들을 보완할 예정이다.

8. 참고문헌

[1] SV committee, "SVAPI Users' Guide for C++", http://www.srapi.com/svapi, Oct 3, 1997

[2] AARON E. Rosenberg, "Automatic Speaker Verification: A Review", Proceedings of the IEEE, vol. 64, No. 4, pp. 475-487, April 1976.

[3] "CAVE - Speaker Verification in Banking and Telecommunications", Computer Science Research at Ubilab, K.-U. Mazel and H.-P. Frei, pp.153-162, Nov. 1996

[4] "Speaker Verification in The Telephone Network: Reaserch Activities in The CAVE Project", ESCA, Eurospeech97, pp.971-974

[5] Chin-Hui Lee, "TUTORIAL: Fundamentals of Speaker and Utterance Verification", ICASSP 1997

[6] 안정모, 계영철, 구명완, "전화음성인식을 위한 효율적인 끝점검출", 대한 전자 공학회 추계 종합 학술 대회 논문집 제 17권 2호, pp.1475-1478, 1994

[7] 이계영, 유병민, 송영신, 이병두, "무선호출용 VMS에서 음성비밀번호 확인을 위한 화자확인 시스템", 제 8회 통신 정보 합동 학술대회, pp. 899-903, 1997년

[8] 정재호, "화자 확인을 위한 화자 확인 시스템" 디에어통통신 기술지 통권 5호, pp.4-10, 1996

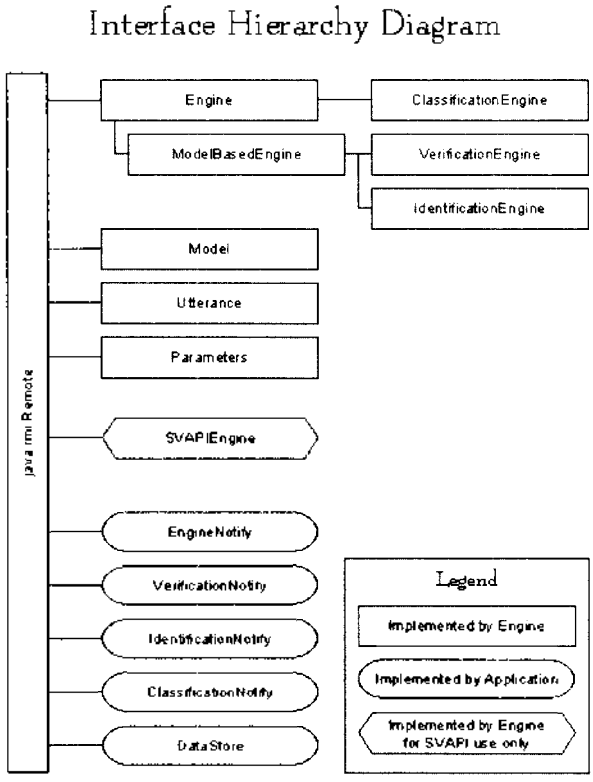

그림 3-1. SVAPI 클래스 계층
(from [1])

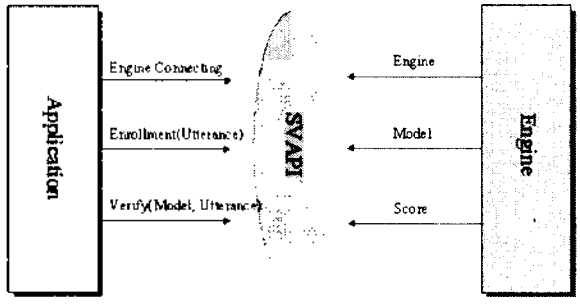

그림 4-1. SVAPI framework Overview

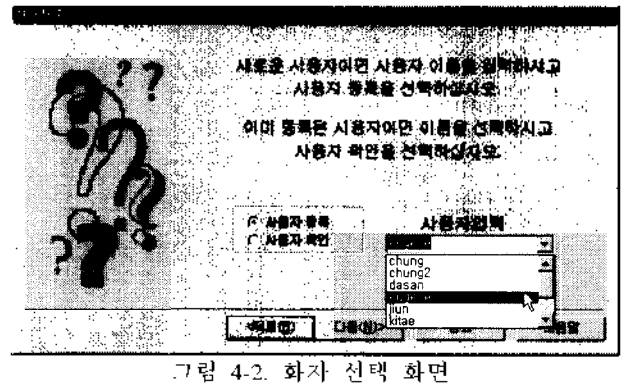

그림 4-2. 화자 선택 화면

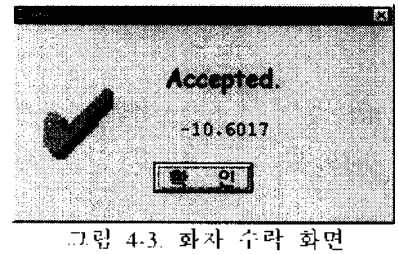

그림 4-3. 화자 수락 화면