

국제 음소의 자동 생성을 활용한 연속음성인식에 관한 연구

김석동⁰, 홍성수*, 신좌철**, 우인성***, 강흥순****

호서대학교 컴퓨터공학과

{sdkim⁰, sshong*, ccshin**, wiscom***, hskang****}@hoseo.edu

A Study on the Continuous Speech Recognition for the Automatic Creation of International Phonetics

Sukdong Kim⁰, Seongsoo Hong*, Chwacheul Shin**, Insung Woo***, Heungsoon Kang****

Dept. of Computer Engineering of Hoseo University

요약

자동 음성 인식(Automatic Speech Recognition)기술은 세계적인 의사소통과 협력을 원활히 할 수 있는 가능성을 제시한다. 현재까지 대부분의 연구들은 주로 사용되는 단일 언어의 말하기에만 집중되어 있다. 따라서 다른 언어들과 함께 사용되는 특정 ASR 시스템을 도입하는 데에는 비싼 비용이 뒤따른다. 본 논문은 다국어 음성 인식에 대한 일반적 접근으로 각 나라 언어를 대표한 발음사전(어휘모델)을 만들기 위하여 음성 인식에 이용하는 어휘 모델을 만들기 위하여 음소 언어 인식(PLI, Phonetic Language Identity) 형식의 입력된 파일을 해석하는 국제 음소 엔진(IPE, International Phoneticizing Engine)를 제안한다. IPE는 독립적이며 규칙을 기본으로 한다. 어휘모델 생성 과정은 Java 언어로 구현된 프로그램에 의해 이루어지고, 이 과정들은 규칙 상충을 줄여주며, 언어학적 훈련을 받지 않은 사람의 규칙 생성도 가능하게 한다. IPE에 의해 생성된 어휘모델을 연속 음성 인식기에 적용한 결과 우리말 인식이 92.55%, 영어에 대하여 89.93%를 얻었다.

핵심용어 : 다국어 연속 음성 인식, 국제 음소 기호(IPA), 음소, 어휘 모델

Abstract

One result of the trend towards globalization is an increased number of projects that focus on natural language processing. Automatic speech recognition (ASR) technologies, for example, hold great promise in facilitating global communications and collaborations. Unfortunately, to date, most research projects focus on single widely spoken languages. Therefore, the cost to adapt a particular ASR tool for use with other languages is often prohibitive. This work takes a more general approach. We propose an International Phoneticizing Engine (IPE) that interprets input files supplied in our Phonetic Language Identity (PLI) format to build a dictionary. IPE is language independent and rule based. It operates by decomposing the dictionary creation process into a set of well-defined steps. These steps reduce rule conflicts, allow for rule creation by people without linguistics training, and optimize run-time efficiency. Dictionaries created by the IPE can be used with the speech recognition system. IPE defines an easy-to-use systematic approach that can obtain 92.55% for the recognition rate of Korean speech and 89.93% for English.

Key Word : Multilingual speech recognition, International phonetic alphabet, Phoneme, Lexical model

1. 서론

자기나라 언어 이외의 언어에 대해 다른 나라의 언어 인식 시스템을 도입할 때 많은 문제점들이 나타난다[1]. 이러한 시스템의 핵심 문제는 다른 나라 언어들에게 적용되지 않는 많은 전제들을 포함하고 있다. 예를 들면 언어에 따라 고유음가, 음성적 배열, 발음법이 다르기 때문에 종종 전문가에 의한 심도 깊은 접근방법이 필요하다. 전 세계는 인터넷으로 연결되어 서로 교류되는 정보량이 증가되는 추세이다. 현재 전 세계에서 통용되는 언어를 살펴보면 사용자 수가 중국어가 제일 많지만 공식적으로 사용하는 경우는 영어가 제일 많다. 특히 하나의 음성에 우리말과 영어가 공존하는 경우 두 나라의 음성을 동시에 인식하는 것이 필요하다. 본 연구는 그동안 나라별로 개별적으로 개발되는 음성 인식 시스템을 확장해서 영어와 우리말이 동시에 인식될 수 있는 혼합 음성 인식기에 대한 연구를 수행하기 위해 첫 단계로 국제적으로 통합된 음소를 정의한 IPA로 우리말에 적용시켜 가능성을 확인하고자 한다.

본 논문은 일반화된 다국어 ASR에 대한 현실적인 시도에 있어서 필수적인 두 가지 중요한 요소에 초점을 맞추었다. 그 중 첫 번째는 이용자 편의에 대한 것이다. 언어학적 훈련을 받지 않은 이용자가 ASR 툴을 쉽게 이용할 수 있도록 단순한 디자인으로 되어있다. 개발자의 이용도에 좌우되고 언어적 지식을 이용하지만, 이용자에게 언어학적 또는 음성 인식에 대한 특수한 훈련은 필요 없다. 이것이 언어 독립 시스템[2,3]을 만들었던 이전의 시도들과 다른 점이다. 이전의 방식들은 대부분의 개발자에게 익숙한 단어군을 활용하거나 언어학적으로 훈련되지 않은 전문가에게 쉬운 문법을 제공하지 않았었다[4]. 두 번째 요소는 단계적인 언어 독립 음성화 과정에 대한 필요이다. 음성기술 면에서 볼 때 각 언어는 발음과 표기가 독특하며 두 가지 다 렉시콘에서 찾을 수 있다. 일단 음성 데이터가 일반 문자열로 전환하면 시스템의 다른 요소들은 더 이상 국제화를 필요로 하지 않으므로 렉시콘은 모든 음성 시스템 중 가장 기본적인데[5].

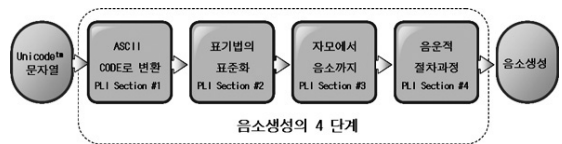
본 논문은 음성인식에 사용 가능한 렉시콘을 만들어내는 우리의 언어 독립 음성화 과정에 대하여 다루고 있다. 이 과정은 다음의 네 단계로 구성된다. (a)유니코드에서 코드 포인트 문자화, (b) 음성학적 표준화 규정, (c) 문자 최소단위에서 음소 규칙으로 실행 (d) 음운체계 실행 등이다. 이러한

단계들은 유니코드 문자열을 입력으로 받아들이고 상응하는 음성적 문자열을 산출하며, 동시에 일반적 문자열의 문제를 해결하여 컴퓨터상으로 더욱 실용적인 방법인 음성화 과정의 분리로 국제화 규칙 어플리케이션과 비교해서 충돌 규칙과 그 영향을 감소시켰다.

본 논문은 다음과 같이 구성되어있다. 2장에서는 음성 인식 시스템에서 이용할 수 있는 어휘모델 생성을 위한 과정을 설명하고, 3장에서 본 논문에서 사용한 음소 언어 인식 (PLI, Phonetic Language Identity) 문법구문론을 설명하고, 4장에서는 PLI 문법을 해석하는 국제 음성화 엔진에 대하여 다룬다. 마지막으로 6장에서는 본 논문에서 제시한 방법에 의해 생성된 어휘모델을 적용한 실험 결과를 보이고 있다.

2. 음소 모델 생성

음소 모델을 구성하기 위한 규칙들은 서로 상반된 규칙에 의해 언어 음성화(음소생성)[6,7]에 문제점들이 발생한다. 이러한 이유에서 우리는 적절한 4단계 절차로 구분하였다. 이 단계별 절차는 유니코드 음역, 표기법의 일반화, 일반화된 스트링의 음성화, 그리고 최종적으로 음소로 나타내기로 진행된다. 음소열 결과물은 ASR 시스템으로 통합된다. 그림(figure)1은 이 기본적인 설계를 보여준다.



[그림 1] 언어 독립적인 음소생성의 4단계

언어의 음성화(음소생성) 작업은 네 가지 일련의 규칙을 생성하는 것으로 구성된다. 기계적 학습을 바탕으로 하는 방법과[8]는 다르게 우리의 최종목표는 사전적인 결과물을 얻는 과정을 완전히 자동화하는 것이 아니고 사용자(원어민)가 다국어를 말할 수 있도록 하는 것이다.

1단계 - 유니코드의 ASCII로 변환

:오늘날 유니코드는 ASCII 코드와 함께 일반적인 표준 문자열로 받아들여지고 있다. 유니코드는 정해진 크기의 문자열로, 각 텍스트 구성요소가 16 비트(UCS-2)로 암호화되며, 언어들의 일관성

을 이루어낸다. 유니코드[9]는 ASCII 코드가 수용할 수 없는 확장된 코드를 가지고 있다. 변환의 목적은 각 대상 언어 유니코드 값을 ASCII 문자열로 대응시키는 것이다. 변환은 스트링을 기본으로 하는 내부문자 생성이 가능하게 하고, 음성화 과정에 적절하여 현재의 ASCII 형식 ASR 시스템의 필수조건에 상응한다. 또한 변환은 텍스트 구성요소에 포함되어있는 추가의 정보를 추출해 낼 수 있도록 해준다. ASCII의 원래 디자인은 미국 영어에 적합하며, 라틴 알파벳의 각 문자에 대한 단일 코드 값을 가지고 있다. 이는 하나의 텍스트 구성요소가 한 개 이상의 언어학적 현상이 보여주는 일부 언어에서는 적당하지 못하다. 예로는 프랑스어 모음은 판독하기 힘든 표시들을 포함하고 있다. 한글에서는 각각의 텍스트가 음절이고 텍스트 구성요소는 최대 4개의 자모(단음)를 포함하고 있다. 언어에 따라 변환 단계는 복잡성에서 차이가 난다. 적은 개수의 문자를 가진 음성적 언어에 있어서는 문제가 되지 않지만 많은 개수의 표의문자 스크립트의 언어에는 매우 복잡하다.

2단계 - 표기의 표준화

: 언어는 시간이 지나면서 진화하기 때문에 표기상 복잡하다. 발음과 표기의 관계는 직관에서 벗어나갈 때가 많다. (예 영어: “knight”는 “nite” 처럼 발음된다.) 어떤 언어에서는 같은 단어를 여러 가지로 표기하기도 한다. (예, 아이티 프랑스어에서 “pwezidan”와 “presidan”) 동음이의어 (예, 영어의 “know,” “no”)는 의미적 차이를 나타내는 표기를 강화한다. 이러한 예들은 음성적 표준화의 필요성을 보여준다. 따라서 음성 기술 시스템은 주요 스크립트에서 음성 관계를 만들어 낼 수 있다. 1단계처럼 이 단계의 목표는 직관적이고 자명한 것이라서 언어학적으로 교육 받지 않은 원어민도 사용할 수 있다.

3단계 - 자모에서 음소까지.

이 단계에서는 표준화된 표기법으로 기본적인 자모에서 음소까지를 만들어내고 발음 조합에 의한 잔여 문맥에 초점을 맞춘다. 비음화와 같은 음소

상호작용이 이 단계에서 규칙대로 생성될 필요는 없다.

4단계 - 음운적 절차

; 발음 규칙은 주로 성도를 사용하여 만들어내는 소리에 의한 복합적 상호작용에 따라 결정된다. 4단계는 발음상의 문맥에 따라 이음간의 차이를 구분한다. 4단계는 또한 불필요한 부분을 삭제한다. 예를 들어 프랑스어에서 동일한 음소가 중복되면 한번만 발음된다. (예, “tourette”: T UW R EH T T T UW REH T)

3.음소언어인식(PLI) 문법

이 부분은 앞서 설명한 4단계에서 사용한 문법을 다룬다. 이 문법은 지역적으로 단순 변형을 이용한 음성화 과정의 모델로서 간편함이라는 우리의 목표를 담고 있다. 사용자는 PLI 문법 체계를 이용해서 텍스트 파일을 만든다.

PLI 텍스트 파일은 4부분을 담고있다. 각 부분은 스스로 키워드 “#1”, “#2”, “#3” 와 “#4” 에 의해 일렬로 구분된다. 모든 부분은 파일에 순서대로 나타난다. 각 단계는 Unicode TM로부터 음성체계로 변환하는 구분된 단계를 따르는 일련의 규칙을 포함하고 있다. PLI 체계는 모든 단계에서 동일하다. “소스 스트링 [탭문자] 대상 스트링”. PLI 과정은 입력된 소스 스트링을 출력된 대상 스트링으로 만들어낸다. 한 단계의 대상 스트링은 다음 단계의 입력 데이터이다. 문자 // 에 뒤따르는 모든 문자들은 주석으로 간주된다. 쉽게 읽 수 있도록 주석을 사용한다.

섹션 #1: 각각의 PLI 라인은 탭 문자와 ASCII변환 코드에 따르는 16진법 대문자의 유니코드 코드값을 가지고 있다. 유니코드 코드 값이 사용될 수 있기 때문에 아니라 PLI 포맷이 ASCII 표준이 될 수 있도록 하기 때문에 유니코드 값을 적용한다. 사용자가 각 PLI 문장 뒤에 실제 유니코드 텍스트 구성요소를 포함한 주석을 첨부한다.

문장의 예제 : D4AD {pVt} //팔

섹션 #2: 이 부분은 변환된 문자 스트링, 탭 문자, 표준화된 음역 스트링을 가지고 있다.

음역된 문자 스트링들은 섹션 1 과정에서 만들어진 것들이다. Knight 의 'kn' 에서 'n' 까지 맵핑하기 위해 우리는 PLI 문장 "kn[TAB]n" 을 사용한다.

섹션 #3: 이 부분의 규칙에는 섹션 #2에서 발생한 표준화 된 변환 문자열, 탭 문자, 음소 문자열이 있다. 두 가지 규칙은 다음과 같다. "th[TAB][TH]" 와 "i[TAB][AY]"

섹션 #4: 이 부분은 섹션 #3에서 발생된 연속적 음소 스트링, 탭 문자, 최종 연속 음소 스트링을 담고 있다. 또한 종종 불필요한 부분을 제거한다.

특수한 문자들을 사용하기도 된다. 이러한 부분의 사용자의 결정에 달린 것이지만, 괄호(())와 같은 경우 단일 발음 단위를 둘러싸기 위해 사용한다. 예를들어, 음소 AY는 [AY]로 표기하고, 한글 음절 '한' 은 [han]으로 표기한다. PLI 문법은 또한 "+" 와 "#" 의 두 가지 제한된 문자를 사용한다. "+" 문자는(스페이스 마커) 단어내의 스페이스를 나타낸다. "+" 는 PLI가 예외사전과 규칙을 기본으로 하는 문법 시스템의 두 가지의 기능을 할 수 있도록 해준다. 예: :s+i [Z]+ [ay].

"#" 문자를 담고 있는 규칙 (무음)은 소리내지 않음을 나타낸다. 예: [HH]+ #+

```
<consonants> = {bcdfghijklmnpqrstvwxyz}
//1개의 가변에 있는 자음을 분류하는
<consonant-sound> = {BKDFGHHJKLNMNPKRSTVWXYZ}
//자음 그룹 1개의 가변에서 소리가 난다
i<consonant> [IY] <consonant-sound> //사용 규칙에 있는 분류 가변
```

〈표 1〉 PLI 문법에서의 변수그룹 체계

변수 그룹은 나중에 클래스로 표현될 수 있는 문장 구성 요소의 벡터로 정의한다. 변수 그룹을 이용하면 PLI 규칙에서 각각의 요소를 명확히하게 한다. 또한 '〈' 와 '〉' 기호사이 선택한 변수 이름을 넣어 변수 그룹을 설정한다. 표 1에 보인바와 같이 PLI 파일은 두 가지 변수 그룹, <consonants> 과 <consonant_sound>으로 표현한다. 변수 그룹은 사용하기 전에 선언 한다. 일반적으로 두 가지 계수 벡터의 각 요소들은 반드시 서로간에 1대1 맵핑을 해야 하는데, 이는 계수내의 물리적 위치가 PLI 변환 맵핑을 좌우하기 때문이다. 표 1의 <consonants> 와 <consonant-sound> 변수는 같은 수의 나열된 요소를 가지고 있다. 자음 c는 K

로 표시된다. 대상이 아무런 변수를 가지고 있지 않을 때에는 이러한 1대1 규칙에서 예외 된다. 이 경우 확장된 소스 전체가 변화되지 않은 같은 대상으로 맵핑된다. : o<variable> o# and o<variable> o

변수 그룹은 사용자가 지정하는 것이므로 PLI 체계는 모든 수준의 정교함을 허용한다. 언어학자는 많은 음성적 특성을 가지고 변수 그룹을 생성할 수 있는 반면 비언어학자라는 자음과 모음을 분리할 수 있다. 이 유연성으로 인해 언어학자나 비언어학자를 가리지 않고 사용자들 모두가 PLI 문법을 사용할 수 있었다.

4. 국제 음성화 엔진 (IPE)

IPE는 자바로 구현한 PLI 포맷의 크로스 플랫폼 번역기이다. 이것은 유니코드 텍스트 파일과 유니코드 파일에서 쓰여진 언어에 관련된 PLI 파일을 필요로 하는 명령어 라인 어플리케이션이다. 음성 인식 시스템에 의해 사용할 수 있는 어휘 모델이 출력된다. IPE는 잘 구성된 PLI를 받아드려 이를 번역하여 임의의 언어에 대한 어휘 모델을 생성하는 프로그램이다. 필요에 따라 4단계의 일부 또는 전부를 설정할 수 있고 트레이스 모드에서 PLI를 진행할 수 있다. IPE는 소스 문자열의 길이에 따라 내림차순으로 각 섹션의 모든 규칙을 분류한다. 같은 소스 문자열 길이를 가진 규칙들은 원래 순서를 유지한다. 분류된 규칙은 앞에서 먼저 정의한 규칙을 먼저 따른다. 규칙들은 왼쪽에서 오른쪽으로 순서로 적용된다. IPE 방식을 테스트하기 위해 다양한 어휘 모델을 만들어 음성 인식기에 그것들을 사용하였다. PLI파일을 생성하기 위해 텍스트 자료 내의 가장 자주 쓰이는 단어를 선택하였는데, 이는 대부분의 발음 현상을 만날 수 있는 집합이기 때문이다.

한글.(Korean).

한글은 ASCII 로 다루기 매우 힘들고 추가적인 규칙을 필요로 하는 음성을 이용하기 때문에 매우 어려운 과제였다. PLI를 생성하는데 900단어가 이용되었다. 섹션 #1에서 IPE는 각 유니코드 한글 음성 음절을 3자모의 상응하는 스트링으로 맵핑하였다. PLI 섹션 #1 파일은 IPE가 10분에 걸쳐 자동적으로 맵핑한 11,179개

의 코드 포인트를 담고 있었다. 섹션 #2는 240개의 복잡한 규칙을 담고 있다. 한글은 규칙적인 언어로 각 자모의 발음은 3자모 전체 음절에서의 위치에 따라 결정되기 때문이다. 발음에 영향을 주는 음절 내의 상호작용이 있고, 상응하는 PLI규칙이 여기에서 나타난다. 특히 만들어진 자모(한 자모가 다른 둘의 합체를 나타내는)는 완전히 문맥에 따라 발음된다. 섹션 #3은 자모의 위치에 따라 가능한 3가지 발음을 만들어내는 한 가지 규칙만을 포함한다. 섹션 #4는 10개의 비성 발음을 만들어냈고 규칙들을 삭제하였다.

영어(English).

영어(ASCII에서 표현된)는 유니코드로부터 들어온 가장 쉬운 언어이다. 섹션 #1에서 단지 54개의 규칙이 필요했다. 섹션 #2에서 영어는 많은 발음상 예외를 가지고 있기 때문에 PLI파일은 어려웠다. 220개의 예외와 286개의 표준화된 규칙은 오랜 실험 시간을 필요로 한다. 이러한 수고는 더 나은 규칙 상충 감지 매커니즘이 필요하다는 것을 보여준다. 섹션 #2에서 방대한 양의 표준화가 이루어졌으므로 섹션 #3에는 47개의 규칙만을 가지고 있다. 섹션 #4에서 우리는 “-ed”로 끝나는 것에 대한 규칙과 소리 나지 않는 문자들을 지우는 많은 규칙들을 추가하였다.

일반적으로 음성적으로 더 정확한 철자법에서 단어를 다시 쓰기 위해서 PLI 섹션 #2를 이용하기를 권장한다. 영어를 위해 PLI를 개발하면서 우리는 같은 섹션에서 음소도 생성하도록 이 기능을 개조하였다.(286개의 표준화된 규칙, 47.5%가 음소를 포함) 이 섹션에서 우리가 만들어낸 음소들은 서기소를 이용한 음성을 표현한다. 이 방법으로 섹션 #2를 이용하는 것이 좋을 때도 있지만 원하지 않는 효과를 만들어내기도 한다. 다음의 4가지 규칙을 고려하였다:

Rule #1:	tion	shan
Rule #2:	ns+	nz+
Rule #3:	ty+	tee+
Rule #4:	rite	rayt

[표 2] PLI 규칙의 예

IPE는 규칙 #1을 이용해서 “portions”에서 “porshans”으로 맵핑되고, 규칙 #2에 의해 “porshanz”으로 맵핑된다. 그러나 “porshanz”은 사실상 음성학적 결과물인 [P][AO][R][SH][AH][N][Z]의 보기 좋은 시각적 표현이다. 이제 다음의 “majority”과 “write.”를 살펴보자. 규칙 #3을 적용한 뒤 두 단어는 “majoritee”와 “write.”가 된다. 그러나 규칙 #4를 적용한 뒤 이 두 단어는 “majorayte”와 “wrayt”가 된다. 하지만 이것을 옳지 않다. 이 문제는 규칙 #3과 규칙 #4의 수정과 특정 음성 기호를 통해 해결할 수 있다. (ty+ [IY]+ // 수정된 규칙 #3, ite ayt // 수정된 규칙 #4). 이러한 규칙을 이용하면 단어는 majorit[IY]와 wrayt가 되고 올바른 음성 표현으로 유도된다.

5. 실험결과

영어 음성에 대하여 2,997개의 단어 사전과 68명의 화자가 총 136개의 문장에 대하여 실험을 하였다. 본 논문에서 제안한 방법에 의해 만든 단어사전에 의해 초기에는 20.72%의 오류율을 보였다. 비용을 들이지 않았다는 점(시간과 비용)을 고려해볼 때 이것을 상당히 낙관적인 시작점이었다. 최소한도의 전문 미세 조정으로 10개의 PLI 규칙을 수정하여 오류율을 10.07%로 낮추었다. 이것이 수동으로 만들어진 결과에 많이 못 미치긴 하지만 PLI가 일단 생성되면 거의 노력 없이 극적인 개선을 이루어 낼 수 있음을 보여주었다.

Phonetic Acoustic Models	Word Error Rates (%)
IPE English:	20.72
IPE English with minimal changes *	10.07

* : 10 단어의 수정 (3개의 단어를 추가 {kansas,saint,arriving} 또한 7개의 빈도수가 많은 단어의 수정 {the,to,what,a,for,from,are}.)

<표 3> 영어 음성의 단어인식 결과

위와 같은 방법을 한국어에 적용하였다. 언어 모델에 사용된 텍스트 자료와 음성 데이터 자료는 온라인 사이트에서 공개적으로 수집했고, KSC 완성형 코드에서 얻어진 텍스트는 유니코드(UCS-2)로 전환되었고 문장의 발음에 따라 구분되었으며, 학습 자료는 162명(70명의 여성과 92명의 남성)이 발음한 총 21시간 동안의 음성을 사용하였다. 발음 사전은 앞서 언급한 한국어 PLI로 만든 IPE에 의해 생성하여

이 발음 사전을 학습과 인식에 모두 사용하였다. 인식에 사용한 음성은 1명의 남성과 여성의 음성을 이용하였고 학습에 참여하지 않은 음성이다. 인식에 사용한 음성 자료는 총 100개의 문장으로 길이는 13.11분 짜리 음성이다.

LM text corpus size: 14358 Unique words in transcript: 310 Dictionary size: 8550 words	Trigram Language Model Perplexity: 6.25 Entropy: 2.64 bits	Bigram Language Model Perplexity: 38.48 Entropy: 8.40 bits	Unigram Language Model Perplexity: 1895.80 Entropy: 10.89 bits
Word Error Rate (%)	8.45	15.67	25.25
Syllable Error Rate (%)*	5.54	9.70	16.61

[표 4] 한국어 음성의 단어 및 문장인식 결과

[표 4]는 한국어 음성실험의 결과를 보여준다. 영어 음성 모델과 한국어 음성모델을 비교하였으나 여러 언어에 걸친 ASR 시스템 비교는 어려운 것이 사실이다. 그럼에도 불구하고 [표 3]과 [표 4]는 결과적으로 한국어 음성모델과 영어 음성모델이 유사하게 동작하는 것을 보여주었다. 한국어 음성모델이 PLI파일을 이용하여 자동적으로 생성되는 발음 사전을 이용하여 만들어졌고 영어 음성모델은 수동으로 만들어진 사전을 가지고 전통적으로 조절되었기 때문에 본 논문에서 제안한 방법이 수동으로 만들어진 어휘 모델의 실용적인 대안으로 생각된다.

6. 결론

본 논문에서 우리는 음성화 사전을 생성할 수 있는 언어 독립의 규칙을 기본으로 하는 기술을 제시하였다. 이 기술은 현재의 자동 연속 음성 인식시스템에 의해 이용될 수 있고 새로운 대상 언어에서 음성 인식의 신속한 전개를 용이하게 하는 가능성을 보여준다. 본 논문에서 제안한 PLI 문법과 IPE 해석기는 음성 기술이 쉽게 세계화 될 수 있는 가능성을 보인다. PLI는 독립된 4단계의 과정에 의해 관련된 언어 정보를 추출하기 때문에 컴퓨터 조작에 의한 복잡성을 줄이고 절차를 매우 단순화하였으며, 강력하고 이용하기 쉬우며 스크립트와 음성 사이의 관계를 효과적으로 구성할 수 있다.

언어학적으로 훈련되지 않은 사용자가 PLI 데이터를 생성하여 음성 기술을 다국적으로 처리할 수 있는 것이 본 논문의 최종 목표이다. 기본적으로 컴퓨터와 익숙한 원어민의 노력이 필요하지만 개발자가 언어학적으로 전문가일 필요

는 없다. 이는 PLI문법과 IPE해석기를 사용한 음성인식시스템의 활용을 통하여 음성인식에 필요한 비용과 노력을 상당히 감소시킬 수 있다.

본 연구의 효율성을 알아보기 위해 실험적으로 자동과 수동으로 만든 어휘모델에 대하여 영어음성을 실험한 결과 최소한의 전문가적 조율만으로도 인식률을 향상시킬 수 있음을 보여준다. 또한 한국어에 대하여 같은 방법으로 생성한 어휘모델을 적용하여 영어 시스템과 유사한 인식률을 얻었다. 추후에는 중국어를 비롯한 여러 언어에 대하여 확장한 어휘 모델 생성에 대하여 연구를 진행하겠다.

참고문헌

- [1] J.Allen, "From Lord Rayleigh to Shannon: How do we decode speech?" Proc. ICASSP-2002, Orlando, (2002)
- [2] J.-L. Gauvain and L. Lamel, "Large vocabulary continuous speech recognition:Advances and application," Proc. IEEE, col.88, no.8, pp.1181-1200, (2000)
- [3] Steeneken, H.J.M, and Lamel, L.F. "SQUALE : Speech Recognizer Quality Assessment for Linguistic Engineering," , Proceedings ARPA Workshop on Spoken Language Technology, Plainsboro, New Jersey, (1994)
- [4] Tatsuo Matsuoka, Katsutoshi Ohtsuki, Takeshi Mori, Sadaoki Furui and Katsuhiko Shirai, "Large-Vocabulary Continuous-Speech Recognition Using a Japanese Business Newspaper (NIKKEI)," Proc. Of the ARPA Workshop on Spoken Language Technology, Austin TX, Morgan Kaufmann, Cohen, Ed., (1996)
- [5] Lin-Shan Lee, et. al., "A Real-time Mandarin Dictation Machine for Chinese Language with Unlimited texts and very large Vocabulary," ICASSP '90, 65-68, (1990)
- [6] Li Deng, "Integrated-multilingual Speech Recognition using Universal Features in a functional Speech Production Model," ICASSP '97, 1007-1010, (1997)
- [7] Sejnowski, T.J. and Rosenberg, C.R., "Nettalk: a parallel network that learns to read aloud," The Johns Hopkins University Electrical Engineering and Computer Science

Technical Report, JHU/EECS-86/01, (1986)

[8] Federking, R., Rudnicky, A., Hogan, C., Eskenazi, "M. DIPLOMAT," ACL-EACL '97, (1997)

[9] The Unicode Consortium, The Unicode Standard, version 2.0, (Addison-Wesley Publishing Company, 1996)



김 석 동 (Sukdong Kim)

1982년 이주대학교 전자공학과(학사)
1993년 이주대학교 대학원 전자공학과(석.박사)
1984년 - 현재 호서대학교 컴퓨터공학과 교수

관심분야 : 음성 응용 프로그램, 음성인식



홍 성 수 (Seongsoo Hong)

1978년 광운대학교 전산학과(학사)
1990년 광운대학교 대학원 전산학과(석.박사)
1984년 - 현재 호서대학교 컴퓨터공학과 교수

관심분야 : 디지털 콘텐츠, 모바일 프로그램



우 인 성 (Insung Woo)

2000년 호서대학교 컴퓨터공학과(학사)
2003년 호서대학교 첨단정보 대학원 컴퓨터공학과(석사)
2007년 호서대학교 대학원 컴퓨터공학과(박사수료)

관심분야 : 게임프로그램, Voice XML



신 좌 철 (Chwacheul Shin)

1990년 호서대학교 전산과(학사)
1996년 호서대학교 대학원 전산과(석사)
2007년 호서대학교 대학원 컴퓨터공학과(박사)

관심분야 : 그래픽 프로그램, 음성 응용



강 흥 순 (Heungsoon Kang)

1990년 호서대학교 전산과(학사)
1996년 호서대학교 대학원 전산과(석사)
2007년 호서대학교 대학원 메카트로닉스전공(박사과정)

관심분야 : 그래픽 프로그램, Voice XML