



## 저작자표시 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.
- 이차적 저작물을 작성할 수 있습니다.
- 이 저작물을 영리 목적으로 이용할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#) 

석사학위논문

스마트폰에서 음성인식 시스템 설계 및  
활용

2012년



한성대학교 대학원

멀티미디어공학과

멀티미디어공학전공

장 자 순

석 사 학 위 논 문

지도교수 이재문

# 스마트폰에서 음성인식 시스템 설계 및 활용

Design and Application of Speech Recognition System on  
Smart Phone

2011년 12월 일

한성대학교 대학원

멀티미디어공학과

멀티미디어공학전공

장 자 순

석사학위논문  
지도교수 이재문

# 스마트폰에서 음성인식 시스템 설계 및 활용

Design and Application of Speech Recognition System on  
Smart Phone

위 논문을 공학 석사학위 논문으로 제출함

2011년 12월 일

한성대학교 대학원

멀티미디어공학과

멀티미디어공학전공

장 자 순

장자순의 공학 석사학위논문을 인준함

2011년 12월 일

심사위원장 \_\_\_\_\_인

심 사 위 원 \_\_\_\_\_인

심 사 위 원 \_\_\_\_\_인

# 국 문 초 록

## 스마트폰에서 음성인식 시스템 설계 및 활용

한성대학교 대학원

멀티미디어공학과

멀티미디어공학전공

장자순

오늘날 21 세기에, 정보 기술의 급속한 발전의 배경중에 스마트 폰의 출현은 연구 및 업무에 전례 없는 편리를 주었고 다른 측면은 사람 사이의 다양성을 보다 효율적으로 커뮤니케이션이 될 수 있도록 하였다. 스마트폰 애플리케이션은 지루한 과정을 보다 쉽게 해 줄 수 있다.

본 논문은 음성 인식 기술로써 음성 인식을 제공하는 구글 클라우드 서비스를 사용하고, 스마트폰 애플리케이션으로 Google에 의존하며, 원래 응용 프로그램 입력보다 더욱 반영되고 개선되어져 음성인식은 스마트폰 활성화에 더욱 기여할 것이다.

**【주요어】** 스마트폰, 구글, 안드로이드, 애플리케이션, 음성인식

# 목 차

|                                   |           |
|-----------------------------------|-----------|
| <b>제 1 장 서론</b>                   | <b>1</b>  |
| 제 1 절 연구 배경                       | 1         |
| 제 2 절 연구 목적 및 구성                  | 4         |
| <b>제 2 장 관련연구</b>                 | <b>5</b>  |
| 제 1 절 안드로이드 개요                    | 5         |
| 제 2 절 음성인식 기술                     | 10        |
| 제 3 절 클라우드 컴퓨팅 시스템                | 15        |
| <b>제 3 장 안드로이드 기반 음성인식 시스템 설계</b> | <b>21</b> |
| 제 1 절 안드로이드 음성인식                  | 21        |
| 제 2 절 안드로이드 노트북 기반의 음성인식 시스템 설계   | 33        |
| <b>제 4 장 확장된 음성인식 시스템의 활용</b>     | <b>45</b> |
| 제 1 절 음성인식 시스템 응용 활용 사례           | 45        |
| 제 2 절 애플리케이션에 확장된 음성인식 시스템 활용     | 47        |
| <b>제 5 장 결론 및 향후연구</b>            | <b>58</b> |
| 제 1 절 평가 및 고찰                     | 58        |
| 제 2 절 기대효과 및 전망                   | 59        |
| <b>【참고문헌】</b>                     | <b>61</b> |
| <b>ABSTRACT</b>                   | <b>64</b> |
| <b>中文摘要</b>                       | <b>65</b> |

## 【표 목 차】

|                                   |    |
|-----------------------------------|----|
| [표 1] 스마트폰 OS점유율 현황.....          | 3  |
| [표 2] 버전별 안드로이드 OS의 별칭 및 사용률..... | 10 |
| [표 3] 적용기기별 안드로이드 OS 갱신현황.....    | 10 |
| [표 4] 클라우드 컴퓨팅의 정의.....           | 16 |
| [표 5] 음성인식 클래스.....               | 23 |
| [표 6] RecognizerIntent 상수표.....   | 23 |
| [표 7] 개발 환경 표.....                | 33 |



## 【그 립 목 차】

|                                                          |    |
|----------------------------------------------------------|----|
| <그림 1> 안드로이드 구성도.....                                    | 7  |
| <그림 2> Voice XML Interpreter 구조.....                     | 18 |
| <그림 3> 3G or 4G 스마트폰 구조.....                             | 19 |
| <그림 4> 음성인식 과정도.....                                     | 22 |
| <그림 5> Android 음성인식 화면.....                              | 22 |
| <그림 6> EXTRA_LANGUAGE_MODEL의 속성 값 상수 .....               | 25 |
| <그림 7> EXTRA_RESULTS의 속성 상수 값.....                       | 26 |
| <그림 8> 음성인식 실행의 예제 .....                                 | 27 |
| <그림 9> 음성인식 결과 출력 예제.....                                | 27 |
| <그림 10> 음성인식 예제 실행화면 .....                               | 28 |
| <그림 11> 블링고 실행화면.....                                    | 30 |
| <그림 12> 시리 실행화면.....                                     | 30 |
| <그림 13> 드라이브세이프리 프로 실행화면.....                            | 31 |
| <그림 14> 차차 앤서즈 실행화면.....                                 | 32 |
| <그림 15> 지비고 보이스 트랜슬레이션 실행화면.....                         | 32 |
| <그림 16> 음성인식 시스템의 구성도.....                               | 34 |
| <그림 17> Handler를 이용한 StartActivity구현 .....               | 34 |
| <그림 18> 시작 액티비티 실행화면 .....                               | 35 |
| <그림 19> Mynotepadactivity의 onCreateOptionsMenu 메소드 ..... | 35 |
| <그림 20> 메인 액티비티 실행화면.....                                | 36 |
| <그림 21> EditActivity의 Save버튼 리스너 생성 .....                | 37 |
| <그림 22> Bundle을 이용한 데이터베이스 저장.....                       | 37 |
| <그림 23> Bundle을 이용한 데이터베이스 저장 확인 .....                   | 38 |
| <그림 24> 편집 액티비티 실행화면.....                                | 38 |
| <그림 25> Xml을 이용한 버튼 생성.....                              | 38 |
| <그림 26> 음성인식을 실행.....                                    | 39 |
| <그림 27> 음성인식 결과를 출력.....                                 | 40 |
| <그림 28> 음성인식 실행화면.....                                   | 40 |
| <그림 29> 구글 클라우드 서버.....                                  | 41 |

|                               |    |
|-------------------------------|----|
| <그림 30> 노트북드 애플리케이션의 저장과정     | 42 |
| <그림 31> 데이터베이스 테이블 및 필드 정보    | 43 |
| <그림 32> 데이터베이스 생성             | 43 |
| <그림 33> 데이터 저장 화면             | 44 |
| <그림 34> 텔레매틱스 네비게이션           | 46 |
| <그림 35> 음성인식 기술이 적용된 셋톱박스과 TV | 47 |
| <그림 36> 구글 번역 실행              | 48 |
| <그림 37> 데이터 저장 및 구글 번역 발송 화면  | 49 |
| <그림 38> 구글 번역 실행결과 출력         | 50 |
| <그림 39> 안드로이드 매니페스트 권한 추가     | 51 |
| <그림 40> 구글 번역 결과 화면           | 51 |
| <그림 41> Gmail 실행              | 52 |
| <그림 42> Gmail 발송 화면           | 53 |
| <그림 43> Read it Later 실행화면    | 55 |
| <그림 44> 애플리케이션 공유 실행          | 56 |
| <그림 45> 애플리케이션 공유 화면          | 56 |
| <그림 46> 메시지 전송 화면             | 57 |
| <그림 47> 카카오톡 발송 화면            | 57 |



HANSUNG  
UNIVERSITY

# 제 1 장 서 론

## 제 1 절 연구 배경

스마트폰은 개인 휴대용 정보 단말기인 PDA 제품과 휴대폰간의 융합의 결과로 탄생되었다. 최초의 스마트폰은 IBM사가 1992년 COMDEX에 소개 했던 Simon이라 불리는 제품이었는데, 휴대폰의 기능에 캘린더, 주소록, 세계 시간, 계산기, 노트패드, 이메일, 팩스, 게임 등의 멀티미디어 기능을 제공하였다. 1996년 휴대폰 제작사인 노키아(Nokia)의는 Nokia 9000을 발표하게 되는데 이것은 HP의 고가의 PDA 모델과 노키아 베스트셀러 휴대폰이 융합한 결과였다. 이후 2001년 단말기 제조업체인 캐나다의 Rim(Research in motion)이 블랙베리(blackberry)를 출시하면서 대중화되기 시작하였다. 2007년 6월에는 미국 애플(Apple)사에서 첫 아이폰(iPhone)을 출시하게 된다. 휴대폰, 화이트스크린, 아이팟(iPod), 인터넷 단말기가 융합된 형태로 아이폰은 그해 미국의 시사주간지 타임이 선정한 '2007년 올해의 발명'으로 선정된다. 이후 2008년 7월 11일 아이폰 3G의 출시와 구글(Google)사에서 개발한 스마트폰 운영체제인 안드로이드(Android)가 출시되면서 현재에 이르고 있다.

스마트폰이란 PC와 같이 운영체제를 탑재하여 다양한 애플리케이션(응용프로그램, Application)을 설치 및 동작시킬 수 있는 휴대폰을 말한다. 스마트폰은 고성능 CPU와 상용 OS를 갖추고 있어 개인정보관리는 물론이고 음악과 동영상 등의 고용량 멀티미디어 콘텐츠의 저장과 재생이 가능하다. 또한, 일반사용자들이 자신의 취향에 따라 다양한 애플리케이션을 개발하거나 설치할 수도 있다. 따라서 개방된 플랫폼을 기반으로 애플리케이션을 지속적으로 활용 하는 것이 가능하다는 측면에서 스마트폰은 기존 휴대폰과 차별화된다.

인간과 기계가 음성으로 대화한다는 것은 인류의 꿈으로, 오래 전부터 연구되어 온 주제이다. 그 오랜 숙원인 기계와의 음성대화가 2010년 중반부터 붙어 닥친 스마트폰의 등장을 기점으로 음성을 통한 기계의 제어가

가능해졌고, 사람들은 이 인류의 오래된 꿈을 스마트폰 보급과 함께 이룰 수 있게 되었다.

모바일 음성인식이 스마트폰의 새로운 킬러 애플리케이션(앱)으로 급부상하고 있다. 최근 다음과 구글이 관련 서비스를 선보인 이후 이를 기반으로 한 앱이 등장했으며 한국어 지원이 안 되는 아이패드에서도 검색어 입력 방식으로 활용되는 등 다양한 형태로 확대되는 추세다. 스마트폰 개발자들에 따르면 구글이 한국어 음성인식을 내놓은 이후 이 기술을 활용한 멀티 음성인식과 문자음성변환(TTS) 입력 앱 등이 등장했다. 현재 등장한 앱들은 애플의 승인을 받아야 가능한 아이폰용 앱에 비해 대부분 응용기술 적용이 용이한 안드로이드 운영체제(OS) 기반으로 개발됐다.

구글은 2007년 9월 'GOOG-114'라는 이름으로 음성 인식 서비스를 처음 선보였다. 그리고 2010년 6월 영어, 중국어, 일본어 등에 이어 여덟 번째로 한국어 서비스를 제공하며 국내 시장에 발을 내디뎠다. 국내의 대표적인 포털 업체인 네이버와 다음이 음성 인식을 활용한 서비스를 제공했지만 구글의 만족도에는 미치지 못했다. PC 검색 점유율에서는 네이버와 다음에 밀려 5% 미만의 저조한 성적을 보이고 있었지만, 모바일에서는 유독 강세를 보였다.

음성 검색의 핵심은 음성 인식의 정확도에 있다. 음성 검색의 정확도를 높이려면 음성 검색 데이터를 되도록 많이 확보하고 있어야 한다. 마이크 슈스터 구글 음성 인식 연구원은 “음성 인식이 성공하려면 정확하고 신속해야 하는데 음성 검색 데이터가 늘어나면서 정확도가 개선되고 그에 따라 사용자가 늘어나는 선순환 구조가 이루어지고 있다”라고 말했다. 축적된 데이터가 많아야 음성의 각기 다른 억양이나 발음을 파악해 제대로 된 검색 결과를 제공할 수 있다는 뜻이다.

지금 95%의 인식률을 자랑하는 구글의 음성 인식 기술의 비결은 클라우드(cloud) 서버에 있다. 슈스터 연구원이 언급한 음성 검색 데이터가 저장되는 곳이 바로 클라우드 서버이다. 클라우드 서버는 인터넷상의 서버 여러 대가 마치 하나인 것처럼 동작하는 일종의 온라인 데이터 창고이다. 구글은 이 클라우드 서버에 성별,연령별, 사투리 등으로 구분된 총 2천3백

억 개의 영어 단어를 음성 데이터로 저장하고 있다. 음성 검색은 스마트폰을 통해 입력되는 음성 데이터를 클라우드 서버로 전송해 음성을 인식하고 그 결과를 단말기로 재전송해주는 것뿐이다. 구글코리아 관계자는 “스마트폰의 폭발적인 성장세에 힘입어 구글 모바일 트래픽은 단 6개월 동안 10배가량 증가했다. 특히 한국어 음성 검색은 그동안 출시되었던 여러 언어의 음성 검색 서비스 경험을 통해 축적된 기술을 바탕으로 한 것으로, 그 정확도나 로딩 속도에 자신이 있다”라고 말했다.

2010년 국내 IT 시장은 스마트폰의 돌풍에 의해 움직이는 시기였다. 2010년 1월, 100만명에 불과하던 국내 스마트폰 시장은 iPhone, 갤럭시, 옵티머스 시리즈등이 대중화에 성공하면서 빠르게 성장을 했다. 2010년 12월 말, 국내 스마트폰 사용자는 710만 명으로 11개월 만에 약 700% 성장을 한 것으로 알려졌다.

Gartner의 2011년 2분기 글로벌 스마트폰 판매량을 보면 안드로이드 OS는 4,678만 대로 43.4%의 시장점유율을 기록하고 있는데, 이는 작년 2분기에 1,065만 대 판매로 17.2%의 시장점유율을 기록한 것과 비교하면 아주 빠른 성장을 이룬 것이다. 또한 경쟁업체인 노키아의 심비안 OS가 22.1%로, 애플의 iOS가 18.2%, RIM 블랙베리 OS가 11.7%, 마이크로소프트의 윈도우즈 모바일 OS가 1.6%를 차지한 것과 비교해도 안드로이드 OS의 성장세를 실감할 수 있으며, 머지않아 스마트폰 OS 시장점유율 50%돌파도가능할것으로보인다.

표 1. 스마트폰 OS점유율 현황

| OS        | 2011년 2분기(M/S)  | 2010년 2분기(M/S)  |
|-----------|-----------------|-----------------|
| Android   | 4,678만 대(43.4%) | 1,065만 대(17.2%) |
| Symbian   | 2,385만 대(22.1%) | 2,539만 대(40.9%) |
| iOS       | 1,963만 대(18.2%) | 874만 대(14.1%)   |
| RIM       | 1,265만 대(11.7%) | 1,163만 대(18.7%) |
| Bada      | 206만 대(1.9%)    | 58만 대(0.9%)     |
| Microsoft | 172만 대(1.6%)    | 306만 대(4.9%)    |
| Others    | 105만 대(1.0%)    | 201만 대(3.2%)    |
| 계         | 10,774만 대(100%) | 6,206만 대(100%)  |

출처: Gartner의 자료를 <<Totaltelecom>>(2011.8.11)에서 재인용

국내 안드로이드 기반의 스마트폰 사용자는 2010년 8월 기준으로 아이폰을 앞선 300만 명에 이르고 있다. 시장조사기관 SA가 발표한 “아시아 태평양 지역과 이머징 마켓에 대한 스마트폰 판매 전망 보고서”에 따르면 2015년 국내에서 애플 아이폰 사용자는 18.9%를, 안드로이드 OS 기반의 스마트폰 사용자는 60.1%를 전망할 만큼 향후에도 안드로이드 기반 스마트폰이 국내 스마트폰 시장의 큰 비중을 차지할 것으로 생각된다. 따라서 다양한 애플리케이션을 통한 효율적인 정보 전달과 사용자 및 개발자의 편의를 위해 애플의 안드로이드 기반 음성인식 개발이 반드시 필요하다.

## 제 2 절 연구 목적 및 구성

본 논문의 연구목적은 공개운영체제인 Linux를 이용하고 오픈소스로 제공되는 모바일 운영체제인 안드로이드를 기반으로 하였다. 애플리케이션의 설계 및 활용은 음성인식 기술을 이용하였고 이를 통해 기능성 및 효율성을 개선하고 응용 영역을 넓혔다. 또한 안드로이드 음성인식 기술을 조사 및 분석하고 기존 애플리케이션에 음성인식 기능을 추가하여 이를 통해 다중 애플리케이션을 공유하는 과정을 연구했다.

본 논문의 구성은 다음과 같다. 2장은 이론적 배경으로 안드로이드 OS 대해서 조사하고 음성인식 및 클라우드 컴퓨팅 기술을 서술한다. 3장은 안드로이드를 기반으로 음성인식 설명하고 애플리케이션의 구성 설명 또는 기존의 안드로이드에서 음성인식 관련 애플리케이션을 서술한다. 4장은 3장에서 작성된 음성인식 애플리케이션을 이용해서 구글 번역, Gmail, 애플리케이션 공유 기능을 추가한다. 5장은 결론 및 향후 연구를 제시한다.

## 제 2 장 관련연구

본 장에서는 안드로이드의 관련 지식, 음성인식 기술 및 클라우드 컴퓨팅 기술을 서술한다.

### 제 1 절 안드로이드 개요

#### 1. 안드로이드 정의

구글(google)의 안드로이드는 오픈소스 운영체제이며 스마트 폰이 추가되는 모바일 통신 장치를 타깃으로 하는 소프트웨어 프레임워크로 구성되어 있다. 안드로이드의 사전적 의미는 "인간의 모습을 한 로봇"이다. 즉, 스마트 폰 이외에 여러 가지로 확장이 될 수 있다는 구글의 미래 전략을 살펴 볼 수 있다. 정확히 설명하면 안드로이드 플랫폼은 운영체제, 미들웨어, 키(key)애플리케이션을 만들 수 있다.

안드로이드 OS는 폐쇄적인 타 OS에 비해 많은 부분을 휴대폰 제조사에 맞게 수정할 수 있도록 설계가 되어 있다. 삼성전자, LG, 모토로라 등 전 세계의 주요 휴대폰 제조사가 안드로이드 폰 개발에 참여하여 2010년 가장 주목받는 스마트폰 OS가 되었다. 안드로이드 SDK(Software Development Kit)는 개발언어가 자바이기 때문에 예전 휴대폰 애플리케이션을 만들 때 통신사 휴대폰 기종에 맞게 따로 개발해야 하는 번거로움을 없애 개인 개발자나 스마트폰 관련 응용 프로그램 개발자들에게 새로운 장을 열어 주는 계기가 되었다. 안드로이드 SDK는 Java 프로그래밍 언어를 사용하여 안드로이드 플랫폼 상에서 애플리케이션 개발을 시작하기 위해 필요한 도구들과 API 들을 제공한다.

#### 2. 안드로이드 특징

안드로이드는 운영체제의 핵심이라고 할 수 있는 커널을 공개 운영체제

인 리눅스에 기반한다. 리눅스는 상당히 오랜 기간 개발된 커널이므로 정교한 메모리 관리, 안정적인 멀티 스레드, 보안 등의 고급기능을 무료로 사용할 수 있다. 단, 모바일 환경에 어울리지 않는 크고 무거운 기능은 제거되었으며 알람, 디버거 등의 기능은 추가되었다. 리눅스를 모바일 환경에 맞게 개조한 것이다.

공식적으로 자바 언어를 사용한다. 고수준의 언어이므로 생산성이 높으며 하드웨어 추상 층을 제공하므로 전문 지식이 없어도 개발이 가능하다. 다만 고급 언어이다 보니 성능이나 섬세함에서는 다소 불이익도 존재한다.

검증된 많은 라이브러리들을 대거 포함하고 있어 웬만한 기능은 별도의 외부 라이브러리를 사용할 필요가 없다. 자바가 언어 차원에서 제공하는 라이브러리들 외에도 OpenGL, SQLite, FreeType 등의 써드파티 라이브러리까지 내장되어 있다. 오픈 소스 중에 쓸 만한 것을 모두 집대성해 놓은 셈이다.

플랫폼에 내장된 빌트인 프로그램과 사용자가 만든 프로그램이 동일한 API를 사용하므로 모든 프로그램은 동등하다. 원한다면 기본으로 제공되는 프로그램을 사용자가 원하는 것으로 언제든지 교체할 수 있다. 플랫폼을 구성하는 요소들을 자유롭게 선택할 수 있다는 면에서 유연성이 뛰어나다.

개방된 환경인만큼 개발 툴과 관련 문서들이 모두 무료로 제공된다. 심지어 운영체제의 핵심 소스까지도 대부분 공개되어 있다. 별도의 라이선스 비용이 들지 않으므로 단말기 가격이 저렴해지는 효과가 있으며 대중화에 유리해서 소프트웨어 수요도 많다.

### 3. 안드로이드 구조

안드로이드는 다섯 개의 계층은 <그림1>처럼 구분되어 있다.

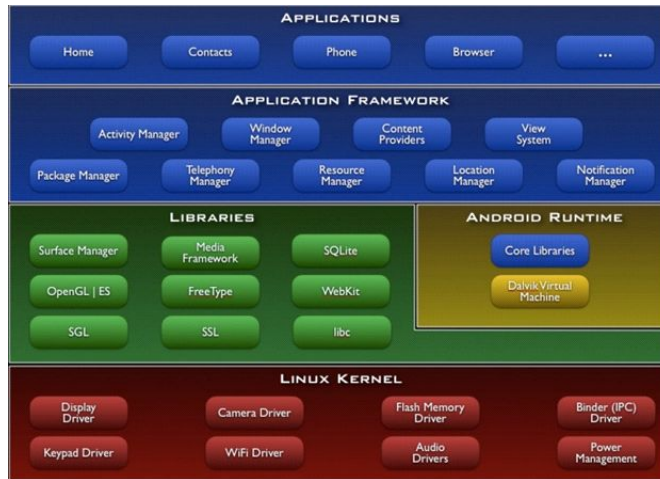


그림 1. 안드로이드 구성도

첫 번째는 H/W와 내부 메모리 관리, 프로세스 관리, 네트워킹 관리 등을 담당하는 리눅스 커널이다. 안드로이드 개발 중 포팅을 하는 개발자라면 이 커널 부분을 H/W에서 작동하도록 하는 일을 할 것이다. 두 번째는 C, C++로 구성된 Library 계층으로 핸드폰에 사용되는 H/W를 사용하기 위한 기능을 제공한다. 대표적인 Library는 그래픽을 구현하기 위한 Graphic 라이브러리와 동영상 재생을 위한 Media Code가 있다. 또한 안드로이드에서 Java Application을 동작시키기 위한 Runtime인 Dalvik VM이 위치해 있다. 세 번째는 안드로이드 Application을 작성하기 위해 필요한 기능들을 제공하는 Application Framework 계층이 있다. 안드로이드 Application 개발자는 필요한 기능 구현을 위해 Framework 계층에서 제공하는 API를 사용한다. 네 번째는 최상위 Application 계층으로 사용자가 사용하는 Application 들이 배포되는 계층이다. 개발자가 애플리케이션을 배포하면 여기에 포함되며 기본적으로 설치된 구글의 애플리케이션과 동등한 위치를 갖는다. 4개의 계층으로 구분된 안드로이드의 모든 소스는 오픈되어 있으며 각각 소스들을 빌드하여 특화된 안드로이드를 만들 수 있다.

#### 4. 안드로이드 애플리케이션

안드로이드 응용 프로그램(애플리케이션)은 적절한 권한만 있으면 누구든지 인스턴스화 할 수 있는 4개의 컴포넌트 구성된다.

#### 1) 액티비티(Activity)

액티비티는 안드로이드의 유저인터페이스를 말하며 실제 장치에서 화면을 구성하고, 사용자는 액티비티에서 키 입력이나 화면을 터치하는 등의 입력을 한다. 그리고 단지 하나의 액티비티 만이 포커스와 키보드 입력을 가질 수 있다. 액티비티는 생명주기를 가지고 있으며, 이러한 생명주기를 가지고 빠른 화면 전환을 하기도 한다.

#### 2) 서비스(Service)

서비스는 안드로이드에서 백그라운드로 실행되는 데몬(Daemon)과 같은 것을 말하며, 파일을 다운로드하거나 음악을 재생하는 것과 같이 유저 인터페이스가 사라진 후 어떤 오퍼레이션을 계속적으로 수행하는 경우를 말한다. 서비스는 시스템이 부팅될 때 시작할 수 있으며, RPC(Remote Procedure Call)를 사용하여 원격지의 시스템에 명령을 실행하거나 데이터를 주고받을 수 있다.

#### 3) 방송 수신자(Broadcast Receiver)

방송 수신자는 다른 애플리케이션으로부터 전달되는 메시지를 받는 메일박스과 같다. 일반적으로 애플리케이션은 내부적인 목적지로 메시지를 전달하기 위해 방송하며, 목적지 방송 수신자는 그 메시지를 받을 수 있게 된다. 방송 인텐트(Intent)는 브로드 캐스트 리시버에게만 전달되며, 결코 액티비티(Activity)나 서비스(Service)에는 전달되지 않는다.

#### 4) 콘텐츠 제공자(Content Provider)

콘텐츠 수신자는 관계형 데이터베이스를 사용하여 데이터를 저장하고 공유한다. 방송 수신자는 URI(Uniform Resource Identifier)와함께 구성된

다. 수신자를 위한 모든 URI는 “content://”라는 문자열로 시작된다. content: 스키마(Scheme)는 데이터가 콘텐츠 수신자에 의해 제어되는 것으로 간주한다. DB의 경우 콘텐츠 수신자를 통해 쿼리를 전송함으로써 SQL 인젝션(Injection)을 방지하는 효과도 있다. 이외에 파일을 공유할 수도 있다.

## 5. 안드로이드 현황 및 발전

구글은 현재 모바일 기기인 스마트폰과 태블릿 PC를 중심으로 자사의 안드로이드OS가 탑재되도록 제조업체들을 지속적으로 지원하고 있다. 이를 위해 구글은 주요 모바일 기기를 중심으로 안드로이드 OS를 지속적으로 업데이트하고 있다.

안드로이드 1.0 버전이 발표된 뒤로 여러 번의 업데이트가 있었으며, 업데이트의 내용은 주로 오류 수정이나 새로운 기능의 추가이다. 그리고 과거 안드로이드 2.2 버전(Froyo)까지 스마트폰과 태블릿 PC로 구분하지 않고 같은 안드로이드버전을 제공하였다. 따라서 스마트폰 및 태블릿 PC 제조업체들은 스마트폰과 태블릿PC의 기기를 구분하지 않고 구글이 제공하는 동일한 안드로이드 운영체제의 버전을 적용하였다. 하지만 구글이 안드로이드 운영체제를 2010년 12월 스마트폰용 안드로이드 2.3 버전(Gingerbread)과 2011년 1월 태블릿 PC에 최적화된 안드로이드 3.0 버전(Honeycomb)으로 나누어 공개, 제공하면서 안드로이드 OS가 스마트폰과 태블릿 PC용으로 구분되어 적용되기 시작했다. 그리고 태블릿 PC용으로 수정된 안드로이드 3.0 버전의 업그레이드 버전(3.1)도 더불어 제공하였다. 따라서 모토로라, 삼성전자, LG전자 등 주요 태블릿 PC 제조사 외에도 중견 태블릿 PC 제조사들은 안드로이드 3.0 · 3.1(Honeycomb)을 탑재한 태블릿 PC를 출시하고 있다.

표 2. 버전별 안드로이드 OS의 별칭 및 사용률

| Platform version              | Code-name   | Distribution |
|-------------------------------|-------------|--------------|
| Android 1.5                   | Cupcake     | 1.1%         |
| Android 1.6                   | Donut       | 1.4%         |
| Android 2.1                   | Eclair      | 11.7%        |
| Android 2.2                   | Froyo       | 45.3%        |
| Android 2.3 - Android 2.3.2   | Gingerbread | 0.5%         |
| Android 2.3.3 - Android 2.3.7 |             | 38.2%        |
| Android 3.0                   | Honeycomb   | 0.2%         |
| Android 3.1                   |             | 0.9%         |
| Android 3.2                   |             | 0.7%         |

출처: developer.android.com(2011. 3)

구글은 2011년 5월 구글의 I/O 개발자 컨퍼런스에서 향후 안드로이드 운영체제의 계획을 발표하였다. 안드로이드 차세대 버전(Android 4.0)인 아이스크림 샌드위치(Ice-cream sandwich)는 스마트폰용 버전과 태블릿 PC용 버전으로 나누어진 안드로이드 운영체제를 하나로 합친 것으로, 기기에 따라 안드로이드 버전을 구분하지 않아 스마트폰 및 태블릿 PC 모두에서 사용할 수 있는 운영체제이다.

표 3. 적용기기별 안드로이드 OS 갱신현황

| 적용기기   | 안드로이드 OS 갱신 현황                 |                                       |                                            |
|--------|--------------------------------|---------------------------------------|--------------------------------------------|
| 스마트폰   | 2.2 Froyo<br>(2010. 5월 SDK 공개) | 2.3 Gingerbread<br>(2010. 12월 SDK 공개) | 4.0 Ice-cream sandwich<br>(2011년 5월 계획 발표) |
| 태블릿 PC |                                | 3.0(3.1) Honeycomb<br>(2011. 2월 출시)   | 태블릿 PC (2011년 10월 공개 및 출시)                 |

## 제 2 절 음성인식 기술

본 절에서는 음성인식 기술 개요와 동향을 살펴보고 음성 인식 기술을

이용한 스마트폰 애플리케이션들에 대해 기술한다.

## 1. 음성인식 기술 개요

음성인식은 자동적 수단에 의하여 음성으로부터 언어적 의미 내용을 식별하는 것이다. 음성은 사용의 편리성 및 자연성 면에서 다른 인터페이스에 비해 우수하다는 장점으로 인해 여러 선진국에서 수십 년간 음성인식에 관한 다양한 연구를 진행하고 있다. 연구 초기의 소규모 단위의 화자 종속적인 발성음을 대상으로 한 연구에서 점차로 대어휘 단어의 화자 독립적인 단어 음성을 인식하는 단계를 거쳐서, 현재는 대어휘를 사용하는 연속 음성인식 단계에 이르고 있다. 이러한 연속 음성인식과 아울러 인식된 문장이나 단어 열로부터 의미를 추출하여 컴퓨터가 보다 지능적인 일을 수행할 수 있도록 하는 음성이해에 관한 연구와 낭독체가 아닌 구어체를 대상으로 한 자유 발화 음성인식 및 이해에 관한 연구도 병행되어 활발히 연구 되어 있다.

음성인식은 다른 입력방식과 차별화된 강점으로 다양한 분야에서 4가지 유형으로 활용되고 있다. 첫 번째 기능이 다양한 기기를 조작할 때 사용된다. 음성인식은 인간에게 친숙한 정보 전달 방법이기 때문에 별도의 학습이나 훈련 없이도 기기를 손쉽게 사용할 수 있어 사용자에게 편리함을 제공한다. 둘째 이동 및 작업 중에 정보를 입력할 때 활용된다. 손과 발이 자유롭지 못한 상황에서도 정보를 입력할 수 있기 때문에 안전성과 생산성을 제공 할 수 있다. 셋째 개인별 서비스 제공에 적합하다. 화자의 음성을 통해 신원(identity), 심리건강상태, 언어능력 등을 파악할 수 있으므로 보안, 의료, 교육 분야 등에서 개인별 서비스 제공이 가능하다. 넷째 실시간 정보 처리에 유용하다. 입력 속도가 타자에 비해 빠르기 때문에 고속 또는 실시간으로 정보를 처리해야 하는 콜센터, 병원, 방송 상황에서 신속하게 정보를 입력할 수 있다.

음성인식은 친숙한 의사소통 수단인 음성으로 기기와 소통하는 기술이다. 따라서 기업은 IT 기기에 익숙하지 않은 유아, 노인, 장애인 등을 고

려하여 음성으로 쉽게 조작할 수 있는 제품과 서비스를 개발하고, 때로는 음성만으로 정확한 정보 전달이 어려울 수 있으므로 터치, 동작인식 등 다른 입력방식과 음성인식을 결합하여 활용하는 방안도 모색해야 한다.

음성인식과 관련이 있는 이야기는 바벨탑, 아라비안 나이트 등 먼 옛날로 거슬러 올라가지만 실제적인 음성인식의 역사를 대략적으로 살펴보면 다음과 같다. 1950년대에는 Bell Lab에서 고립 숫자음에 대한 단독 화자에 대한 인식기를 개발하였고, RCA Lab에서는 차별적인 음절을 인식하는 시스템을 개발하였다. 그리고 영국의 College 대학에서 4개의 모음과 9개의 자음을 인식하는 음소 인식기를 개발하였고, MIT Lincoln Lab에서는 화자독립 모음 인식기를 개발하였다. 1960년대에는 안정적인 음성 검출기의 개발, 소련 학자들에 의한 동적 프로그래밍(Dynamic programming) 즉 DTW(Dynamic Time Warping)의 개념 확립, CMU(Carnegie Mellon University)에서 음소의 동적인 탐색 기법을 이용하여 연속어 음성인식에 대한 가능성을 확인하여 주었다. 1970년대에는 고립단어 인식 기술이 실제로 사용 가능한 수준이 되었으며, IBM에서 대어휘 음성인식을 위한 시도가 이루어 졌다. 또한 AT&T Bell Lab에서 화자독립 음성인식 시스템을 위한 노력이 본격적으로 시작되었다.

1980년대의 중요한 사건은 DARPA(Defense Advanced Research Projects Agency)에서 수행한 대어휘 연속어 음성인식 프로젝트이다. 이 프로젝트는 많은 연구기관에서 수행되었다. 이 프로젝트의 수행 기관은 다음과 같다. 카네기 멜론 대학(Carnegie Mellon University), BBN, Lincoln Labs, SRI, MIT, AT&T Bell Labs등이다. HMM(hidden Markov Model)은 DTW(Dynamic Time Warping) 이후로 음성인식에 사용되는 가장 중요한 알고리즘이다. 그런데 1980년대 중반까지는 HMM(hidden Markov Model)에 대해서 특정 기관(IBM, Institute for Defense Analyses, Dragon Systems)을 제외하고는 아무런 관심을 끌지 못하였다. 1980년대 말에는 NN(Neural Networks)이 음성인식에 본격적으로 응용이 되었다. NN(Neural Networks)은 1950년대에 등장하였지만 여러 가지 구현상의 문제를 해결하지 못하여 주목을 받지 못하였으나 1980년대에 들어서서 기술

의 많은 한계를 극복하고 주목을 받게 되었다. 현재 세계 대부분의 연속음성 인식 시스템은 HMM을 사용하고 있다. 1990년대에 음성인식 연구자들의 중요한 관심사는 대화체 음성 인식(Spontaneous Speech Recognition)을 어떻게 하느냐 이다. 음성인식의 역사가 40년 이상이 되었지만 아직까지도 자연스러운 발화 다시 말해서 친구사이의 대화에서 발성 되는 음성을 인식하는 것은 해결되지 않았으며, 명료하면서도 자연스러운 음성을 합성해 내기도 어려운 상태이다. 또한 무제한 영역에서 유능한 번역가의 세련된 번역문을 생성하기는 더욱 힘든 일로 남아 있다. 이러한 기술이 모두 개발되어야 외국어를 전혀 모르고도 외국인과 자연스럽게 대화를 나눌 수 있도록 하는 자동통역 시스템을 개발할 수 있다.

## 2. 음성인식 기술 현황 및 발전

현재 약 13억 달러 시장을 형성하고 있는 음성 소프트웨어에는 음성다이얼 휴대폰이나, 음성입력(Voice Dication) 소프트웨어 등이 있으며, 최근에는 지능형 음성인식 및 음성언어번역 소프트웨어 개발이 적극적으로 진행되면서 PC를 비롯한 휴대폰, 양방향 TV, 자동차 등에 내장될 것으로 예상된다. 또한 다국어 음성언어 번역 소프트웨어 공동연구를 위한 국제 컨소시엄(C-Star)이 형성되어 활발한 연구가 진행되고 있다.

- 음성처리 기술의 실용화는 고립단어 인식기술의 경우 1990년대 초반에 실용화가 시작되었으며, 그 예로 90년대 초반에 미국 AT&T에서 수신자 부담 전화의 처리에 고립단어 음성인식 기술을 사용했다.

- 1990년대 후반에 이르러 화자종속 연속음성 인식기술을 이용한 음성타프라이터 제품이 출시했다. 그 예로 1997년에 미국 Dragon사의 Naturally Speaking 이라는 제품이 출시했다.

- 미국 DEC사의 음성합성기 DecTalk, 영국의 천재물리학자 스티븐 호킹이 사용했다.

- 벨기에의 Lernout & Houspie사, 음성인식기 및 음성합성기 출시했고, L&H 음성인식기술의 성능은 소규모 어휘를 대상으로 하는 제품에 대

해서는 98% 이상, 중규모 어휘 음성인식시스템에 대해서도 95% 정도의 인식 성능을 보이고 있을 뿐 아니라, 1만 단어 이상의 대규모 어휘인식 시스템에 대해서도 90% 이상의 성능을 나타내고 있다.

- 드래곤 포인트&스피크 : 가정용 PC 시장을 겨냥해 출시된 제품으로, 연속음 인식기술을 채택해 자연스러운 디테이션을 지원한다. IBM의 비아보이스와 함께 90% 이상의 높은 인식률을 보이고 있다.

- IBM ViaVoice : 다양한 애플리케이션에서 디테이션을 구현한다. 문법을 고려해 문장을 받아 적고, 아울러 사용자가 디테이션한 발음을 녹음해 음성 데이터베이스를 구축한다.

- 커즈웨일 보이스패드 : 커즈웨일(Kurzweil)사는 약 16년간 음성인식 기술에 몸담은 회사로, 지난 해 6월 전세계 3대 음성인식업체로 꼽히는 L&H사에 합병된 곳이다. 연속음 인식은 불가능하다. 단어 하나씩 구별해 발음하는 고립단어 인식기술에서 진보된 연결단어 인식기술로, 단어 사이에 시간적 간격을 두지 않고 말해도 인식이 가능하다.

국내에서도 1980년도에 들면서부터 본격적인 음성인식에 관한 연구가 이루어져 오고 있다. 개발된 시스템으로는 한국전자통신 연구소의 자동통역시스템, 한국통신의 증권정보안내시스템, 삼성전자의 음성구동 퍼스널 컴퓨터, 음성구동 셀룰러폰(삼성, LG), 음성메모장치(공성통신) 등이 있으며 현재 성능개선 또는 상용화 중에 있다. 또 음성에 의한 로봇제어에 관한 연구, 음성에 의한 자동항법 장치 등에 관한 연구도 활발히 진행되고 있다.

최근 음성인식기술은 미국을 중심으로 구체적인 응용분야가 개척되어오고 있고 멀티모드/멀티미디어 환경 속에서의 다른 미디어와 통합에 관한 연구가 진행되고 있다. 향후 이러한 멀티미디어와 결합되는 연구가 더욱 활발하게 진행될 것으로 기대된다. 이러한 멀티모드/멀티미디어 기술의 활용분야로는 각종 멀티미디어 정보기기의 입출력 인터페이스, 자동차 네비게이션 시스템 개발, 시각 장애자를 위한 서비스 시스템, 대화형 자판기, 대화형 Robot, 3차 컴퓨터시스템 개발, 제품의 검사, 멀티모드 의료서비스, 각종 멀티모델 데이터 베이스 검색, 홈쇼핑, 자동 예약/문의 시스템, 음성

입출력 PC, 전자메일 시스템 개발, 멀티모드형 자동항법 장치 개발 등 그 분야는 이루 헤아릴 수 없다. 이와 같은 응용연구와 더불어 자연어 처리 기술을 적극적으로 이용하는 자연발화 대화체 연속음성인식에 관한 연구가 더욱 활발하게 진행될 것으로 생각된다. 이와 더불어 각국 간의 자동통역 전화에 관한 연구도 가속화될 것으로 보인다.

음성인식 전반적으로는 현재의 통계적 방법을 기반으로 실제의 대량의 음성데이터에 기초를 둔 일상 언어의 언어모델을 구축하는 것, 다수화자의 음성데이터에 기재하여 개인차의 모델을 구축하여 이에 의한 다수화자의 음성데이터의 적응화 알고리즘을 개발하는 것, 여러 종류의 잡음, 왜곡에 자동적으로 적응되는 방법을 확립하는 것 등이 중요한 기술적 과제로 될 것이다. 국내적으로는 하루빨리 대규모 한국어 음성데이터베이스가 구축되어 많은 음성연구자들이 공동으로 이용하여 서로의 연구결과를 평가하고 그 결과를 공유할 수 있는 기반이 조성되어야 할 것으로 생각된다.

### 제 3 절 클라우드 컴퓨팅 시스템

#### 1. 클라우드 컴퓨팅 시스템 정의

클라우드 컴퓨팅이란 인터넷 기술을 기반으로 소프트웨어나 스토리지 등과 같은 컴퓨팅 자원들을 가상화하여 하나로 통합하는 기술이며, 통합된 각각의 컴퓨팅 자원들을 하나의 서비스로서 사용자에게 제공하는 IT 환경이다. 사용자들이 클라우드 환경 내에서 제공하는 여러 서비스 중에서 필요로 하는 서비스만을 선택하여 언제 어디서나 제공 받을 수 있도록 해준다. 서비스 제공자는 사용자들의 수요의 변화에 따라 필요한 컴퓨팅 자원들을 가변적으로 구성할 수 있으며 사용자는 초기 구축에 대한 비용 부담 없이 서비스를 이용한 만큼 비용을 지불하면 된다는 특징을 가진다. 아래의 표 4는 클라우드 컴퓨팅의 정의에 대한 기술 표이다.

표 4. 클라우드 컴퓨팅의 정의

| 기관                 | 정의                                                                                                                   |
|--------------------|----------------------------------------------------------------------------------------------------------------------|
| Gartner            | 인터넷 기술을 활용하여 다수의 고객들에게 높은 수준의 확장성을 가진 자원들을 서비스로 제공하는 컴퓨팅의 한 형태                                                       |
| Forrester Research | 표준화된 IP 기반 기능들이 IP를 통해 제공되며 언제나 접근이 허용되고 수요의 변화에 따라 가변적이며 사용량이나 광고에 기반한 과금 모형을 제공하며 웹 혹은 프로그램적인 인터페이스를 제공하는 컴퓨팅      |
| Wikipedia          | 인터넷에 기반한 개발과 컴퓨터 기술의 활용을 말하는 것으로 인터넷을 통해서 동적으로 규모화 가능한 가상적 자원들이 제공 되어지는 컴퓨팅                                          |
| IBM                | 웹 기반 애플리케이션을 활용하여 대용량 데이터베이스를 인터넷 가상공간에서 분산 처리하고 이 데이터를 데스크톱 PC, 휴대 전화, 노트북 PC, PDA 등 다양한 단말기에서 불러오거나 가공할 수 있게 하는 환경 |

클라우드 컴퓨팅의 대표적인 서비스에는 SaaS, PaaS, IaaS가 있으며 그 외에는

XaaS라는 틀 안에서 다양한 서비스들이 존재한다.

#### 1)SaaS(Software as a Service)

SaaS란 서버에 설치된 소프트웨어를 인터넷을 통해 사용자에게 서비스로서 제공해주는 것이다. 최근에는 오피스 소프트웨어를 인터넷을 통하여 제공해주는 업체들이 늘어나고 있다. 대표적으로 구글 오피스와 네이버 위드를 이야기할 수 있는데 편리함과 무료라는 특성 때문에 많은 사용자가 이용하고 있다.

#### 2)PaaS(Platform as a Service)

PaaS란 소프트웨어를 개발하기 위한 플랫폼을 인터넷을 통하여 개발자들에게 서비스로서 제공해주는 것이다. 대표적으로 구글에서 서비스하는 Google App Engine 등이 있다.

### 3)IaaS(Infrastructure as a Service)

IaaS란 IT 인프라 환경을 서비스로서 제공해주는 것이다. 저장장치나 CPU 등의 하드웨어를 사용하여 그것들이 가지고 있는 ‘컴퓨팅 능력’을 서비스 한다. 대표적으로 아마존에서 서비스하는 Amazon S3(Simple Storage Service) 등이 있다.

## 2. VoiceXML 정의

VoiceXML(Voice eXtensible Markup Language)은 AT&T, IBM, Lucen Technologies, Motorola 등이 설립한 VoiceXML Forum에서 만들어졌다. VoiceXML Forum에서 제안한 대화형 음성서비스 시나리오를 제공하기 위해 고안된 XML 기반의 마크업 언어로 W3G(World Wide Web Consortium)에서 표준으로 공인하였다. VoiceXML의 구조 모델은 아래의 그림 1과 같이 크게 4가지 구성 요소로 이루어져 있다.

### 1) Document Server

Document Server는 VoiceXML Interpreter가 URI(Uniform Resource Identifier)형태로 요청하는 자원들을 VoiceXML Interpreter에게 전송하는 역할을 한다. 주로 기존의 Web Server를 이용하여 구축하게 되는데 이를 통해 Web Database에 접근하여 실시간 정보들을 반영할 수 있으며 Web Server가 지원하는 다양한 기술들을 활용할 수 있어 음성을 통해 Web Browser를 접근 하는 것과 같은 효과를 줄 수 있다.

### 2)Interpreter

Interpreter는 VoiceXML 문서를 적재하고 그 문서의 내용을 해석하는 VoiceXML 구조의 가장 핵심적인 요소이다. Interpreter의 내장된 XML Parser를 통해 VoiceXML 문서가 VoiceXML 문서의 형식을 정의하고 있는 DTD(Document Type Definition)에 맞게 작성되었는지 검사하고 태그

들을 통하여 표현된 음성 서비스 시나리오를 해석한다. 그리고 해석된 내용을 통해 Implementation Platform에 필요한 음성 입출력 명령을 내리며 Document Server를 통하여 필요한 문서를 다운로드하거나 다른 문서로 전환하면서 음성 서비스 시나리오의 수행을 제어한다.

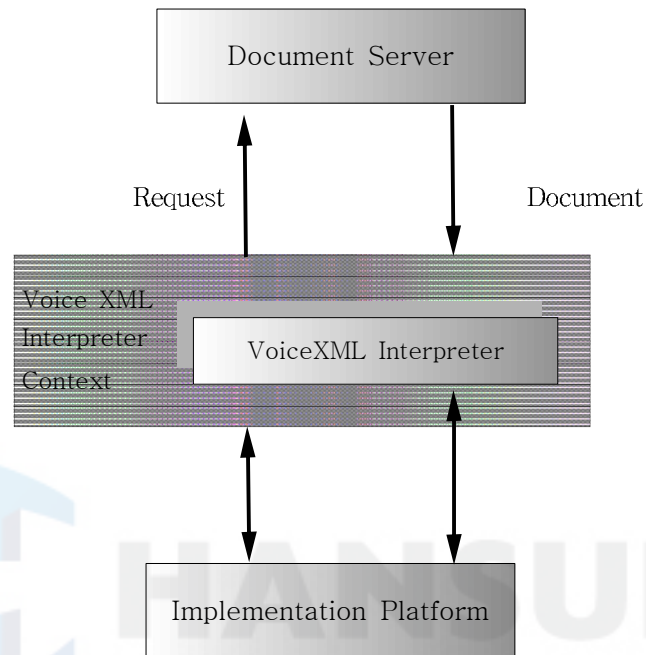


그림 2. Voice XML Interpreter 구조

### 3)Interpreter Context

Interpreter Context는 Interpreter를 통제하고 관리하는 역할을 수행한다. Interpreter에 필요한 자원을 할당 및 해제하며 시작 문서를 지정하고 환경 변수 등을 설정한다. 아래의 그림은 모바일 스마트 기기에 구축한 VoiceXML 플랫폼 구조이다.

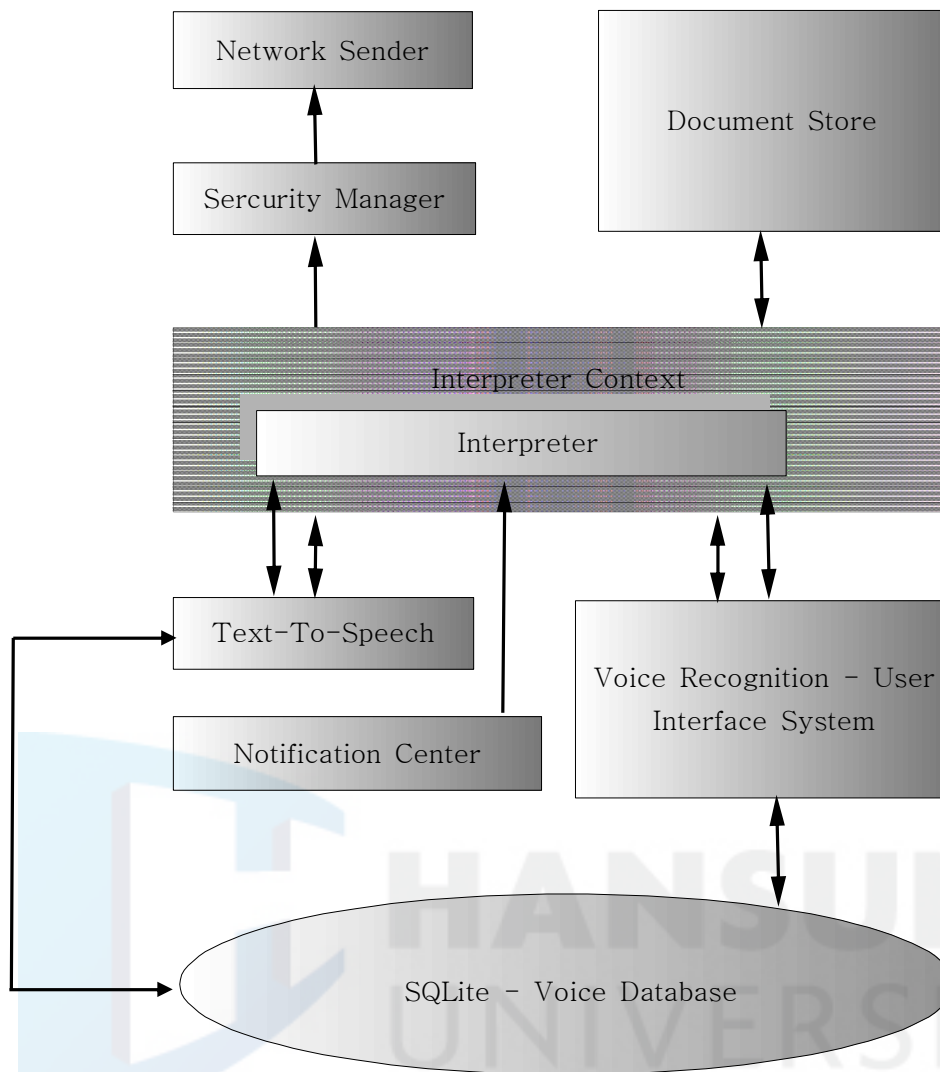


그림 3. 3G or 4G 스마트폰 구조

### 1) Voice Database

Voice Database에는 음성 인식과 합성 기술을 사용하기 위하여 음성 정보를 저장하고 있는데 이에 Voice Recognition - User Interface System TTS(Text-To-Speech)와 연동되어진다. Voice Database는 SQLite로 구축된다. SQLite는 서버가 없이 Stand-alone으로 동작하며 빠르고 크기가 작아 모바일용으로 특화된 관계형 데이터베이스 엔진이다.

## 2)Notification Center

Notification Center는 홈 네트워크 시스템으로부터 메시지를 수신 받고 Interpreter를 호출하는 역할을 한다. Application이 실행 중이 아닐 때에도 실시간으로 홈 네트워크 시스템으로부터 메시지를 수신 받을 수 있다는 특징을 가지고 있다.

## 3)Security Manager

Interpreter로부터 홈 네트워크 시스템에게 전달할 메시지를 전달 받아 사용자의 비밀 키로 전자서명 한 후 Network Sender에게 전달한다.

## 4)Network Sender

Security Manager로부터 전달 받은 메시지를 홈 네트워크 시스템에 전달하는 역할을 한다.



## 제 3 장 안드로이드 기반 음성인식 시스템 설계

### 제 1 절 안드로이드 음성인식

#### 1. 안드로이드 음성인식 개요

음성인식 기술은 휴대폰에 널리 응용 되었다. 음성은 인간에게 있어서 가장 자연스러운 정보전달 수단의 하나이다. 휴대폰을 사용하는 과정에서 대부분 손으로 입력하는 방식을 취해서 휴대폰과 상호작용을 한다. 하지만 휴대폰과 같은 비교적 작은 본체안의 가상키보드로 입력하는 작업이 사람에게 불편함을 줄 수 있기 때문에, 구글에서는 음성검색 서비스를 지원하게 된다. 2008년 11월부터 아이폰은 음성검색 기능을 사용하였다. 또한, 안드로이드1.5 SDK에서는 음성인식 기능을 강화하였으며, 음성검색영역까지 사용하게 되었다. 우리가 검색 창 옆에 있는 마이크처럼 생긴 버튼만 누르면(그림) 안드로이드는 음성인식기능을 가동해 요구하는 내용을 검색할 수 있다. 만약 음성이 정확하게 인식이 안 될 경우 안드로이드는 기계가 대략적으로 인식된 내용을 선택할 수 있게끔 나열해 줄 수도 있다. 이렇게 한 이유는 최대한 인간과 기계의 커뮤니케이션을 실현하게 한 방법이다. 머지않아 사람들은 더 편하게 사용할 수 있는 기능을 연구하여, 게임분야에서도 음성으로 조정하는 기능을 생길 거라고 기대한다. 안드로이드 음성인식 기술의 핵심은 클라우드 기술이다. 구글은 인식률을 높이기 위해 수백만 문장을 데이터베이스에 저장해줬다고 밝혔다. 데이터베이스에 저장된 각 문장들은 사용자가 말한 문장과 비교하여 말한 사람의 의도와 가장 비슷한 단어와 문장을 찍어 휴대폰에 표시한다.

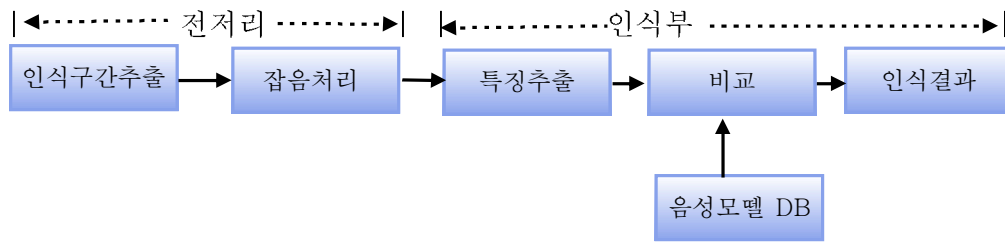


그림 4. 음성인식 과정도

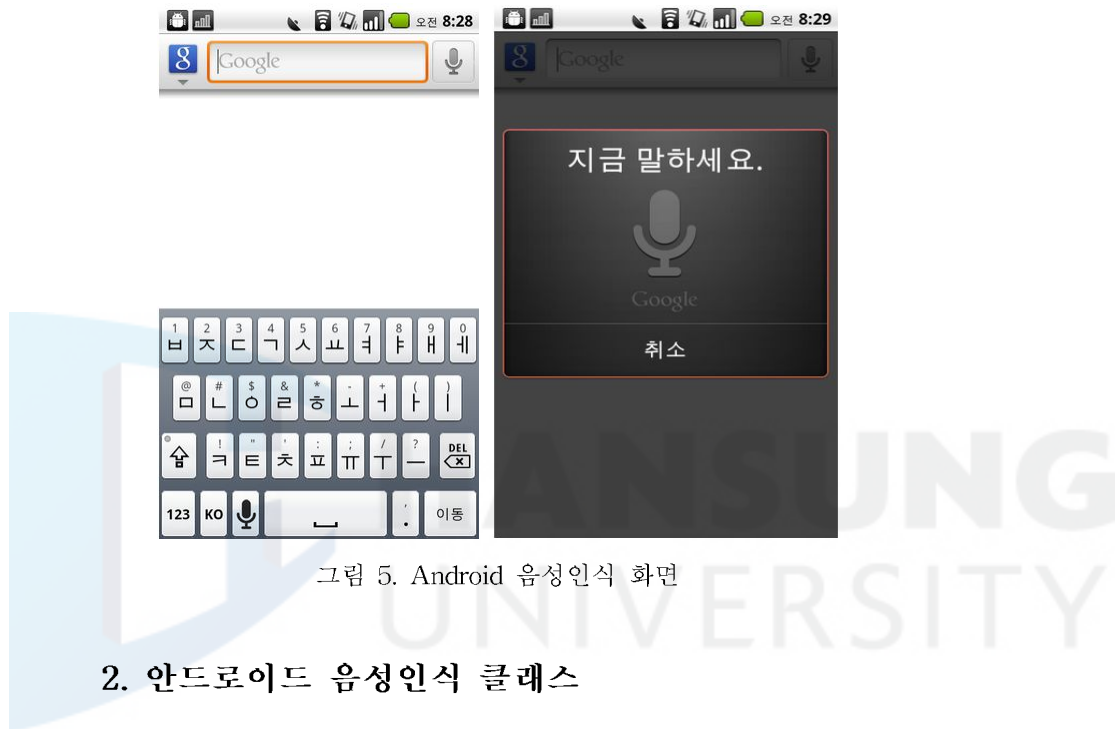


그림 5. Android 음성인식 화면

## 2. 안드로이드 음성인식 클래스

안드로이드OS는 음성인식기능을 이용하면 구글에서 제공하는 API에서 android.speech(안드로이드 스피치 api)를 사용한다.

표 5. 음성인식 클래스

| 클래스 명                       | 설명                                           |
|-----------------------------|----------------------------------------------|
| RecognitionService          | 이 클래스는 인식 서비스 구현을 위한 기본 클래스                  |
| RecognitionService.Callback | 이 클래스는 음성 인식 서비스로부터 받은 콜백을 사용자에게 전달하기 위한 클래스 |
| RecognizerIntent            | 음성인식을 사용하기 위한 Intent                         |
| RecognizerResultsIntent     | 음성 인식 결과를 표시하여 그에 관한 관점을 위한 Intent           |
| SpeechRecognizer            | 이 클래스는 음성 인식 서비스에 대한 액세스를 제공                 |

안드로이드 시스템이 권장하는 음성 인식 액티비티를 실행하기 위해서는 음성 인식 행위를 의미하는 시스템 상수 값을 이용하여 intent 객체를 생성한다. 이처럼 안드로이드에서 음성 인식을 사용하기 위해 필요한 상수 값이 RecognizerIntent 클래스에 정의되어 있다. 다음은 RecognizerIntent 클래스에서 자주 사용되는 상수 값에 대한 설명이다.

표 6. RecognizerIntent 상수표

| 상수     |                         | 설명                                                                             |
|--------|-------------------------|--------------------------------------------------------------------------------|
| String | ACTION_RECOGNIZE_SPEECH | 발생 시작을 위해 사용자에게 메시지를 표시하고, 인식기로 전달된다.                                          |
| String | ACTION_WEB_SEARCH       | 발생 시작을 위해 사용자에게 메시지를 표시하고, 인식기로 전달된다. 그리고 웹 검색 결과를 표시한다.                       |
| String | EXTRA_LANGUAGE          | 확장 IETF 언어 태그이다.(as defined by BCP47). for example "en-US".                    |
| String | EXTRA_LANGUAGE_MODEL    | 인식기가 ACTION_RECOGNIZE_SPEECH를 언제 실행하는지에 대한 정보이다. 인식기는 세부적으로 그 결과의 해당 정보를 사용한다. |
| String | EXTRA_MAX_RESULTS       | 결과 값이 반환되는 최대 개수이다.                                                            |
| String | EXTRA_PROMPT            | 사용자에게 말을 하도록 요청하는 텍스트이다.                                                       |
| String | EXTRA_RESULTS           | ACTION_RECOGNIZE_SPEECH가 수행될 때 인식 결과는 ArrayList<String>으로 보내진다.                |

|        |                           |                                 |
|--------|---------------------------|---------------------------------|
| String | LANGUAGE_MODEL_FREE_FORM  | free-form 기반으로 음성 인식하기 위해 사용된다. |
| String | LANGUAGE_MODEL_WEB_SEARCH | 웹 검색 기반 언어 모델을 사용한다.            |
| String | RESULT_AUDIO_ERROR        | 오디오 에러 시 반환되는 결과 값이다.           |
| String | RESULT_CLIENT_ERROR       | 일반적인 클라이언트 에러의 결과 값이다.          |
| String | RESULT_NETWORK_ERROR      | 네트워크 에러 시 반환되는 결과 값이다.          |
| String | RESULT_NO_MATCH           | 일치하는 결과가 없을 시 반환된 결과 값이다.       |
| String | RESULT_SERVER_ERROR       | 인식 서버에서 보내는 오류의 결과 값이다.         |

#### 1) ACTION\_RECOGNIZE\_SPEECH

사용자의 음성을 인식하기 위해 음성 인식 액티비티를 시작하도록 전달하는 상수 값으로, 이 상수 값을 인자로 가지는 인텐트 객체를 이용하면 음성 인식 액티비티를 실행할 수 있다. 음성 인식 액티비티의 경우 음성 인식의 결과를 가져와야 하기 때문에 startActivityForResult 메소드를 이용하여 액티비티를 실행해야 하며, 그 결과는 onActivityResult 메소드에서 EXTRA\_RESULTS를 통해 반환받거나 PendingIntent 의 객체로 얻을 수 있다. 이 상수 값으로 음성 인식 액티비티를 실행하는 경우 putExtra 메소드를 이용하면 음성 인식기에 다양한 정보를 넘겨줄 수 있는데, 그중 EXTRA\_LANGUAGE\_MODE 정보는 반드시 전달되어야 한다. 이외에 선택적으로 EXTRA\_MAX\_RESULTS, EXTRA\_LANGUAGE, EXTRA\_PROMPT, EXTRA\_RESULTS\_PENDINGINTENT, EXTRA\_RESULTS\_PENDINGINTENT\_BUNDLE 을 통해 정보를 전달할 수 있다.

#### 2) ACTION\_WEB\_SEARCH

이 상수 값을 이용하여 음성 인식 액티비티를 실행시키면 음성 인식 결과로 웹 검색을 수행하여 검색 결과를 표시해준다. 이 상수 값으로 음성 인식 액티비티를 실행하는 경우 마찬가지로 EXTRA\_LANGUAGE\_MODE

정보는 반드시 전달되어야 한다. 이외에 EXTRA\_PARTIAL\_RESULTS, EXTRA\_MAX\_RESULTS, EXTRA\_LANGUAGE, EXTRA\_PROMPT 정보를 추가적으로 전달할 수 있다.

### 3) EXTRA\_LANGUAGE\_MODEL

음성 인식 액티비티를 실행하는 경우 음성 인식기에 사용되는 음성 모델(Speech model) 정보를 명시하기 위해 사용되는 상수 값이다. 이 상수 값은 반드시 액티비티에 전달되어야 하며 전달할 수 있는 두 가지 음성 모델은 다음과 같다.

|                            |
|----------------------------|
| LANGUAGE_MODEL_FREE_FORM   |
| free-form 기반의 음성 인식 모델을 사용 |
| LANGUAGE_MODEL_WEB_SEARCH  |
| 웹 검색 조건을 기반으로 하는 언어 모델을 사용 |

그림 6. EXTRA\_LANGUAGE\_MODEL의 속성 값 상수

### 4) EXTRA\_PROMPT

사용자에게 음성을 요청할 때 보여주기 위한 문장을 설정할 수 있는 상수 값이다.

### 5) EXTRA\_LANGUAGE

음성 인식기에 인식되는 언어를 설정할 때 사용하는 상수값으로 IETE 언어 태그를 이용하여 설정 할 수 있다. 명시하지 않을 경우에는 기기의 환경 설정에 설정된 언어로 음성이 해석되지만, 이와 다른 언어로 음성 인식을 수행하기 위해서는 그에 맞는 값을 지정해주어야 한다. 음성 인식에 사용되는 주요 국가들의 IETE 언어 태그는 다음과 같다.

한국어(대한민국):ko-KR 일본어(일본):ja-JP 중국어(중국,북경):cmn-Hans-CN  
영어(미국):en-US 독일어(독일):de-DE 프랑스어(프랑스):fr-FR

### 6) EXTRA\_MAX\_RESULTS

음성 인식을 통해 반환될 결과의 수를 제한하는 상수 값으로 그 값은 정수여야 한다.

## 7) EXTRA\_RESULTS

ACTION\_RECOGNIZE\_SPEECH를 이용하여 액티비티를 호출한 경우 사용자의 음성을 인식하고 해석된 결과를 onActivity 메소드내에서 ArrayList<String> 객체로 반환받기 위한 상수 값이다. 음성 인식 액티비티 종료 후 호출되는 onActivity 콜백 메소드에서 액티비티의 결과 코드가 RESULT\_OK가 아닌 에러 값인 경우 해당 값들은 아래의 오류 내용을 의미한다.

|                                             |
|---------------------------------------------|
| RESULT_AUDIO_ERROR : 오디오 에러 시 반환되는 결과 값.    |
| RESULT_CLIENT_ERROR : 일반적인 클라이언트 에러의 결과 값.  |
| RESULT_NETWORK_ERROR : 네트워크 에러 시 반환되는 결과 값. |
| RESULT_NO_MATCH : 일치하는 결과가 없을 시 반환된 결과 값.   |
| RESULT_SERVER_ERROR : 인식 서버에서 보내는 오류의 결과 값. |

그림 7. EXTRA\_RESULTS의 속성 상수 값

## 3. 안드로이드 기본 음성인식 예제

RecognizerIntent 클래스를 이용하여 구성한 이 예제는 voice recognize 버튼을 클릭하여 실행된 액티비티를 통해 사용자의 음성을 입력받으면, 인식된 그 결과를 리스트뷰에 출력한다. 예제 RecognizerIntent 클래스 사용 부분 다음과 같다.

```

Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
// 음성 인식 행위에 대한 액티비티를 실행시킬 수 있는 인텐트 객체를 생성한다.
intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
// 음성 인식기에 대한 언어 모델 정보를 실행될 액티비티에 전달한다.
intent.putExtra(RecognizerIntent.EXTRA_PROMPT, "voice start");
// 사용자에게 음성 인식을 요청할 시 보여줄 텍스트 정보를 실행될 액티비티에 전달한다.
startActivityForResult(intent, VOICE_RECOGNITION_REQUEST_CODE);
// 인텐트 정보에 맞는 액티비티를 실행한다.

```

그림 8. 음성인식 실행의 예제

음성인식 결과는 `ArrayList<String>` 형태로 변환되며, `onActivityResult()`에서 `getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS)` 사용하여 받을 수 있다. 인식결과가 여러 개일 경우 가장 정확도가 높은 순서대로 결과가 배치된다. 예제에서는 가장 정확도가 높은 결과 하나를 `TextView()`에 표시하도록 하였다.

```

protected void onActivityResult(int requestCode, int resultCode, Intent data){
    if (requestCode == VOICE_RECOGNITION_REQUEST_CODE
        && resultCode == RESULT_OK) {
        // 결과 값이 있고, 액티비티를 시작할 때
        VOICE_RECOGNITION_REQUEST_CODE로 설정해둔 결과 값인 경우
        ArrayList<String> matches =
            data.getStringArrayListExtra(RecognizerIntent.EXTRA_RESULTS);
        // 사용자의 음성을 인식하여 해석한 결과를 String 객체로 반환한다.
        mList.setAdapter(new ArrayAdapter<String>
            (this, android.R.layout.simple_list_item_1, matches));
        // ArrayList 객체와 출력 리소스를 설정한 어댑터 객체를 생성하여
        simple_list_item_1에 설정한다.
    }
}

```

그림 9. 음성인식 결과 출력 예제

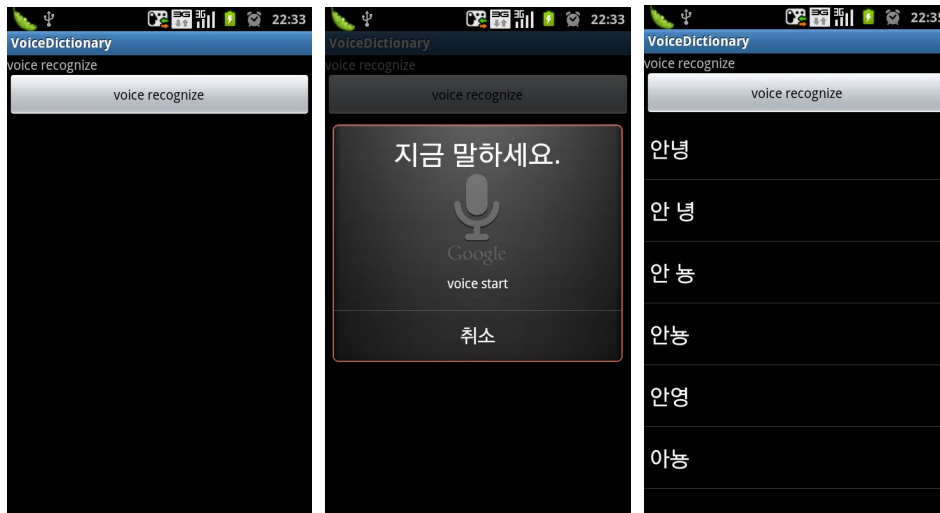


그림 10. 음성인식 예제 실행화면

#### 4. Text to Speech 음성합성

TTS(Text to Speech)란 글자, 문장, 숫자, 기호 등을 사람이 일반적으로 발성하는 형태의 음성으로 변환하는 것을 말한다. TTS 기술은 시각이 아닌 청각으로 정보를 전달하므로 자동차 운전이나 다른 곳에 집중하기 어려운 상황 등과 같이 시각에 의존하는 작업과 병행이 가능하다는 장점이 있다. 음성 합성 기술은 수시로 변할 수 있는 정보를 사람에게 알려주기에 가장 쉽고 간편한 방법이며, 시각 장애를 가진 사람을 위한 매우 효율적인 정보 제공수단이기도 하다.

TTS는 실제 응용 방식에 따라 크게 두 가지로 구분될 수 있다. 제한된 어휘개수와 구문구조의 문장만을 합성하는 제한적 TTS 또는 자동 음성응답 시스템(Automatic Response System)과 임의의 문장을 입력받아 음성 합성하는 무제한 TTS 시스템이 있다. 전자는 텔레뱅킹, 114안내, 증권조회 등과 같이 고유명사나 숫자를 발성하는 간단한 분야에서 널리 이용되고 있으며, 후자는 이메일 및 인터넷 신문 리더, 교통정보, 일기예보 등과 같이 단어나 문장 수에 구애받지 않는 시스템에서 사용된다.

제한적 TTS 시스템은 문장 전체 또는 간단한 단어의 조합으로 생성된

음성을 재생하므로 사용하는 단어나 합성 문장의 개수, 기억 용량의 제한, 음절 간 음운현상, 문장에 따른 억양 등의 문제로 자연스러운 문장단위의 합성음이 필요한 분야에서 사용되지 못하고 있다. 그러나 무제한 TTS 시스템은 단어보다 작은 음절, 반 음절, 음소, 반음소 등의 단위로 운율을 처리하므로 단어나 문장의 개수에 제한 없이 문장을 합성할 수 있고 제한적 TTS보다 자연스러운 합성음을 생성할 수 있다.

## 5. 기존의 안드로이드에서 음성인식 관련 애플리케이션

구글의 안드로이드와 애플의 iOS 등 오늘날의 휴대폰 플랫폼은 개선된 음성 인식 기능을 제공하기 때문에 음성 인식이 점차 인기를 얻고 있다. 그러나 가트너의 수석 애널리스트 투옹 누옌은 모바일 음성 인식의 부흥은 그러한 요인만이 아니고 개선된 사용자 인터페이스에 대한 수요, 특히 자신의 휴대폰과 상호작용하기 위해 터치스크린에만 의존하고 싶지 않은 사용자들로부터의 수요도 한 몫을 차지한다고 설명했다.

### 1) 블링고(Vlingo)

개인 비서는 단지 부자와 유명인들만이 이용할 수 있는 것이 아니다. 가상 세계에서는 스마트폰 사용자들도 개인 비서를 이용할 수 있다. 안드로이드 그리고 iOS 등의 플랫폼을 지원하는 빙고 앱은 페이스북 상태 업데이트, 문자 메시지 전송, 혹은 웹 검색 등 사용자의 명령을 수행한다. 블링고의 iOS 앱 스토어에 인용된 문구에서 MC 해머는 "블링고는 마약과 같다"라고 말했다.

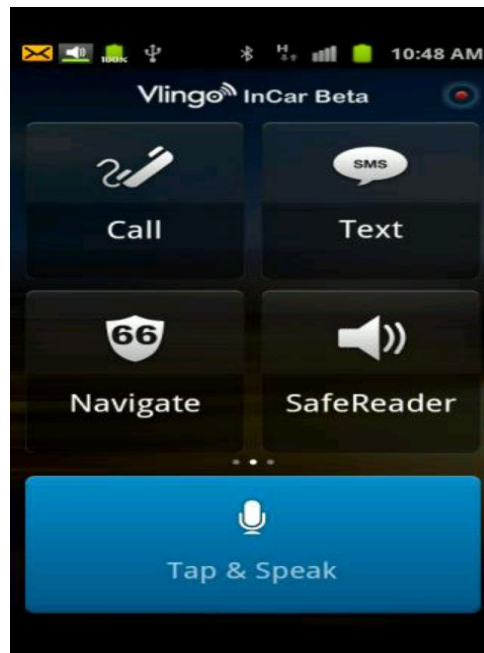


그림 11. 블링고 실행화면

## 2) 시리 어시스턴트(Siri Assistant)

블링고와 유사하지만 식당 찾기, 예약하기, 극장 티켓 예매하기, 그리고 택시 예약하기 등에 좀 더 초점이 맞춰진 시리 어시스턴트를 이용한다. 그리고 마치 사람처럼, 시리 어시스턴트는 시간이 지남에 따라 사용자의 개인 선호도를 학습한다.



그림 12. 시리 실행화면

### 3) 드라이브세이프리 프로(DriveSafe.ly Pro)

운전 중에 문자 메시지를 작성하는 것은 위험할 뿐만 아니라 많은 여러 장소에서 그렇게 하는 것이 불법이다. 드라이브세이프리 프로(DriveSafe.ly Pro)를 사용하면 운전 중 문자 메시지 작성을 위해 화면을 터치하지 않아도 된다. 이 어플은 사용자에게 수신된 문자 메시지를 읽어주고 음성으로 답장을 보낼 수 있는 기능을 제공한다.(안드로이드용 무료 버전은 수신 메시지를 읽어주는 기능만 제공한다)



그림 13. 드라이브세이프리 프로 실행화면

### 4) 차차 앤서즈(ChaCha Answers)

개략적인 정보를 통해 검색을 수행하고자 한다면 검색 엔진이 유용하다. 그러나 구체적인 특정 질문이 있는 경우, 단순한 질문에 대한 여러 페이지의 검색 결과를 살펴보는 것은 비생산적이다. 그때 차차 앤서즈를 사용하면 된다. 차차 앤서즈는 사용자가 입력한 질문에 대한 답을 제공하는 서비스이다. 차차 앤서즈는 토성에서 질문자의 몸무게가 얼마가 될 것인지에 대한 대답도 제공한다.

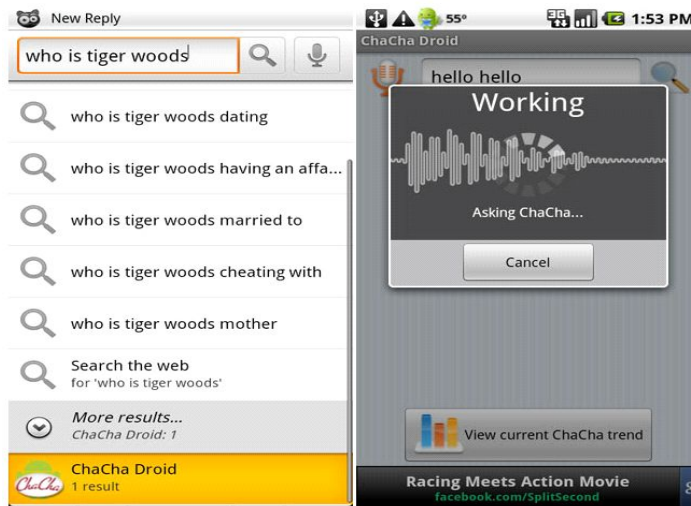


그림 14. 차차 앤서즈 실행화면

#### 5) 지비고 보이스 트랜슬레이션(Jibbigo Voice Translation)

상대방이 자신의 말을 알아 들을 수 없다면 대화하는 것이 무슨 소용이 있겠는가? 그럴 때는 지빙고를 이용한다. 안드로이드를 지원하는 이 양방향 통역 어플은 사용자의 말을 듣고 다른 언어로 사용자의 말을 변환해준다. 지빙고는 영어, 필리핀어, 프랑스어, 독일어, 일본어, 한국어, 표준 중국어, 그리고 스페인어 등 8개 국어를 지원한다. 외국어 발음에 자신이 없는 경우, 변환된 말을 직접 읽지 말고 지빙고가 읽어주는 것을 상대방에게 들려주는 편이 낫다.

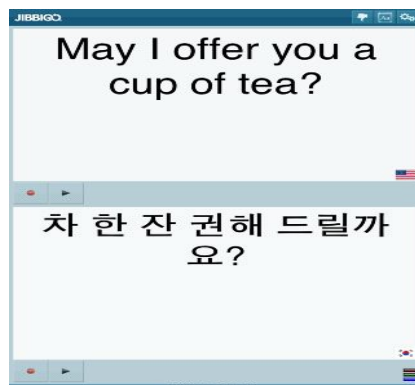


그림 15. 지비고 보이스 트랜슬레이션 실행화면

## 제 2 절 안드로이드 노트북 기반의 음성인식 시스템 설계

본 장에서는 안드로이드 노트북 기반의 음성인식 시스템을 설계하고 이를 구현하는 과정을 기술한다.

### 1 애플리케이션 개발 환경

안드로이드 노트북의 기반의 음성인식 시스템을 개발하기 위해 사용된 하드웨어 및 소프트웨어 개발 환경은 아래 표 7과 같다.

표 7. 개발 환경 표

| 구분  | 기능별항목  | 사양                    |
|-----|--------|-----------------------|
| H/W | CPU    | Intel Core i5 CPU 650 |
|     | RAM    | 3G                    |
|     | HDD    | 250G                  |
| S/W | 운영체제   | Windows 7 Premium K   |
|     | 개발 언어  | Android 2.3, XML      |
|     | 데이터베이스 | SQLite                |
|     | 프로스랩   | Eclipse               |

### 2 애플리케이션 구현 및 설계

본 논문에 안드로이드 노트북 기반의 음성인식 시스템을 다음 구성도로 구현한다. 이 애플리케이션이 시작 액티비티, 메인 액티비티, 편집 액티비티, 음성인식, 구글 클라우드 서버에 분석, 데이터 저장, 데이터 공유 8개의 부분으로 구성되었다.

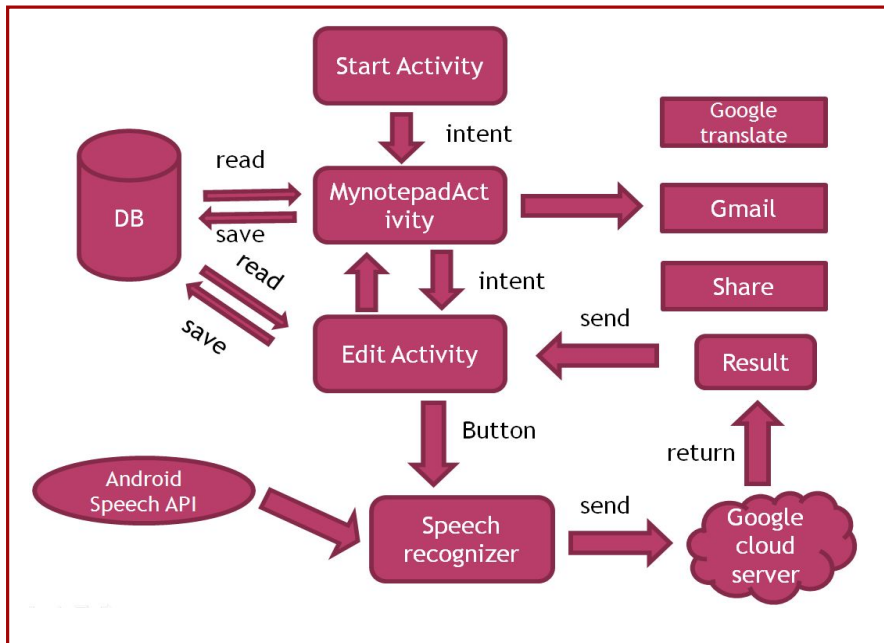


그림 16. 음성인식 시스템의 구성도

### 1) 시작 액티비티

보통 애플리케이션의 첫 화면에 로고를 보여주고 일정 시간 후 자동으로 다음으로 넘어가거나 사용자가 터치할 경우 넘어가도록 구성이 되어 있는 것을 보았을 것이다. 위의 두 가지 방법 중에서 일정 시간이 지난 후에 이동하는 방법에 대해서 알아본다.

다음은 Android의 Handler를 이용하여 구현한 모습이다.

```
new Handler().postDelayed(new Runnable(){
//시간이 지연하기 위한 익명의 Handler를 사용한다
    public void run(){
        Intent i = new Intent(startactivity.this , MynotepadActivity.class);
        // Intent를 이용해서 메인 액티비티로 이동하는 Intent를 생성한다.
        startactivity.this.startActivity(i);
        // Intent를 이용하여 메인 액티비티로 이동한다.
        startactivity.this.finish();
        // 시작 화면 종료
    }
}, 2000);
// 2000ms 후에 실행되는 Handler를 생성한다.
```

그림 17. Handler를 이용한 StartActivity구현



그림 18. 시작 액티비티 실행화면

## 2) 메인 액티비티

메인 액티비티(Mynotepadactivity)는 private필드가 있고, onCreate(), onCreateOptionsMenu(), onOptionsItemSelected()등 세 개의 오버라이드된 메소드가 있다. onCreate() 메소드는 액티비티가 시작될 때 호출되고, 이 때 정보를 초기화한다. 그리고 onCreateOptionsMenu()메소드는 액티비티에서 메뉴를 볼 수 있게 해준다. 사용자가 메뉴 버튼을 클릭했을 때 선택할 수 있는 메뉴가 나타난다. 다음 그림은 메뉴 버튼을 눌렀을 때 나타나는 메뉴화면이다.

```
public boolean onCreateOptionsMenu(Menu menu)
{
    menu.add(0, INSERT_ID, 0, "add");
    menu.add(0, EXIT_ID, 0, "exit");

    return super.onCreateOptionsMenu(menu);
}
```

그림 19. Mynotepadactivity의 onCreateOptionsMenu 메소드

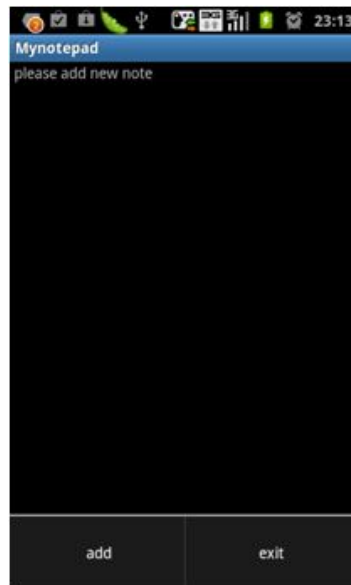


그림 20. 메인 액티비티 실행화면

OptionsItemSelected()메소드는 메뉴에서 발생하는 이벤트를 처리하기 위해 사용하는데, 이 프로그램에서는 사용자가 메뉴를 선택하면 호출된다.

### 3) 편집 액티비티(Editactivity)

편집 액티비티는 먼저 레이아웃 파일을 사용하여 뷰를 설정한다. 이 애플리케이션에서 타이틀과 바디 텍스트, save 버튼, voice recognize(음성인식 입력)버튼은 사용자가 입력하거나 클릭할 수 있는 요소이다. 이 4가지 요소는 프로그램을 할 때 프로그래머가 제어할 필요가 있다. 사용자가 타이틀과 바디에 어떤 내용을 입력하면 입력한 내용을 얻어서 데이터베이스에 저장해야 한다. 데이터베이스에 저장된 내용으로 갱신할 때는 타이틀과 바디는 데이터베이스로부터 내용을 얻어 와야 한다. 이 화면에 버튼을 클릭했을 때 먼저 save 버튼의 경우 setOnClickListener()메소드를 호출한다. 이때 파라미터로는 OnClickListener 인터페이스가 들어간다.

OnClickListener는 버튼과 같은 뷰가 클릭되었을 때 이벤트를 처리하기 위해 콜백을 등록하는 인터페이스이다. 다음과 같이 프로그래밍하면

saveButton버튼을 클릭했을 때 onClick()메소드가 호출된다.

```
saveButton.setOnClickListener(new View.OnClickListener()
{
    public void onClick(View view)
    {

    }
})
```

그림 21. EditActivity의 Save버튼 리스너 생성

사용자가 새 노트에 타이틀과 바디를 입력하고 save 버튼을 누를 것이다. 이때 프로그램에서는 사용자가 입력한 타이틀과 바디의 내용을 데이터베이스에 저장하도록 onClick() 메소드를 구현하면 된다. Editactivity를 호출한 Mynotepadactivity에 전달할 내용이 있다면 번들을 사용하면 된다. putExtras() 메소드를 이용하면 Intent에 번들의 내용을 삽입할 수 있다. 이렇게 해서 Editactivity를 인텐트한 Mynotepadactivity에 번들을 넘길 수 있다. 번들을 만들고 타이틀과 바디를 데이터베이스에 저장하는 소스는 다음과 같다.

```
Bundle bundle = new Bundle();
bundle.putString(DbHelper.KEY_TITLE, mTitleText.getText().toString());
bundle.putString(DbHelper.KEY_BODY, mBodyText.getText().toString());

if (mRowId != null) {
    bundle.putLong(DbHelper.KEY_ROWID, mRowId);
}
```

그림 22. Bundle을 이용한 데이터베이스 저장

Intent를 새로 생성하고 생성된 Intent에 번들을 설정한 후 setResult()를 호출한다. 이런 측면에서 Intent는 일종의 번들(title, body, mRowId)을 운반하는 역할도 한다. setResult()메소드는 결과 코드와 Intent를 반환하는 메소드로 오류가 없다면 결과 코드에 RESULT\_OK를 설정하면 된다.

```
Intent mIntent = new Intent();
mIntent.putExtras(bundle);
setResult(RESULT_OK, mIntent);
finish();
```

그림 23. Bundle을 이용한 데이터베이스 저장 확인

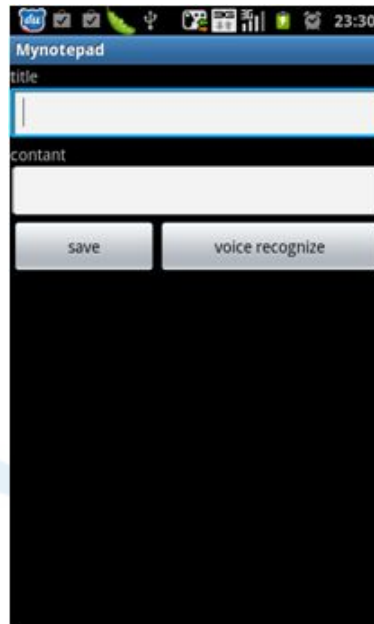


그림 24. 편집 액티비티 실행화면

#### 4) 음성인식

편집 액티비티에 사용자 데이터를 입력하는 방법에는 소프트 키보드 방식과 음성인식 기능을 이용할 수 있다. 먼저 음성인식 버튼을 추가한다.

```
Button saveButton = (Button) findViewById(R.id.confirm);
Button voiceButton = (Button) findViewById(R.id.Button01);
```

그림 25. Xml을 이용한 버튼 생성

음성인식기를 실행하는 메소드를 작성한다. 음성인식기를 실행하는 인텐

트에는 putExtra()를 사용하여 부가 설정을 할 수 있다. 음성인식 언어로 현재 단말기에서 사용하고 있는 언어를 사용한다.

```
voiceButton.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        try {  
            Intent intent = new Intent  
                (RecognizerIntent.ACTION_RECOGNIZE_SPEECH);  
            intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,  
                RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);  
            intent.putExtra(RecognizerIntent.EXTRA_PROMPT, "voice start");  
            startActivityForResult(intent,  
                VOICE_RECOGNITION_REQUEST_CODE);  
        } catch (ActivityNotFoundException e) { }  
    }  
});
```

그림 26. 음성인식을 실행

음성 인식 결과 중 정확도가 높은 순서대로 ListView에 표시하도록 하고, 결과를 선택하면 타이틀이랑 바디 창에 표시된다.

```

protected void onActivityResult(int requestCode, int resultCode, Intent data) {

    if (requestCode == VOICE_RECOGNITION_REQUEST_CODE
        && resultCode == RESULT_OK) {

        results = data.getStringArrayListExtra
            (RecognizerIntent.EXTRA_RESULTS);

        mList.setAdapter(new ArrayAdapter<String>(this,
            android.R.layout.simple_list_item_checked, results));

        mList.setOnItemClickListener(new AdapterView.OnItemClickListener() {

            public void onItemClick(AdapterView<?> arg0, View arg1,
                int arg2, long arg3) {

                String    value= results.get(arg2);
                mTitleText.setText(mTitleText.getText()+ value);
                mBodyText.setText(mBodyText.getText()+ value);

            }
        });
    }
    super.onActivityResult(requestCode, resultCode, data);
}

```

그림 27. 음성인식 결과를 출력

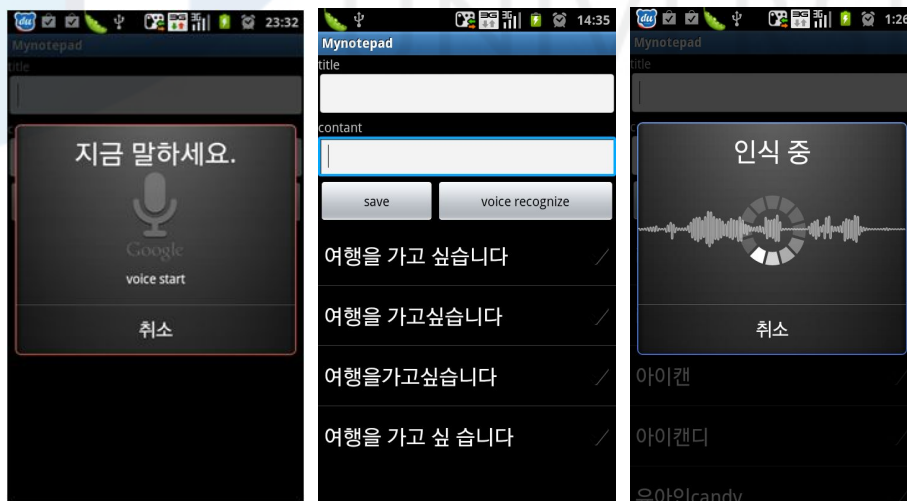


그림 28. 음성인식 실행화면

#### 5) 구글 클라우드 서버

구글은 클라우드 서버에 성별, 연령별, 사투리로 구분한 총 2,300억 개의 영어단어를 음성데이터로 저장하고 있고 이를 활용하여 음성인식을 실시함으로써 처리한다.

구글 서버 클라우드 방식의 분산형 음성인식을 사용해 대용량의 음성인식을 수행한다. 분산형 음성인식에서 스마트폰이나 지능형 단말기는 입력되는 신호로부터 음성 구간을 검출하고 검출된 음성 신호를 통신망을 통해 서버로 전달하는 기능을 수행한다. 실제 음성인식과 관련된 작업은 대규모 음성인식 서버에서 수행하며 그 결과를 클라이언트에 전송하게 된다.

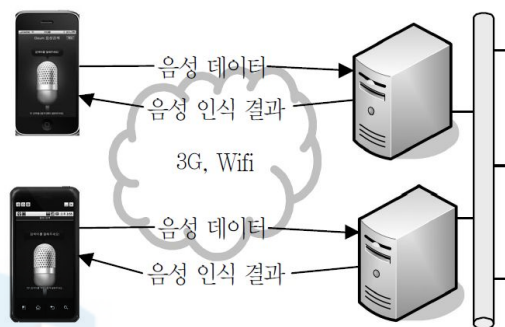


그림 29. 구글 클라우드 서버

#### 6) 데이터 저장

구글은 메모할 수 있는 노트패드 애플리케이션을 제공하고 있다. 이를 이용하여 음성인식 기술을 가지고 있는 애플리케이션을 작성할 수 있다. 본 논문에서 개발하는 애플리케이션이 구글에서 제공하는 노트패드 애플리케이션을 참조하여 개선 후 작성되는 것이다. 기본 노트패드 애플리케이션 저장과정은 <그림 30>과 같다.

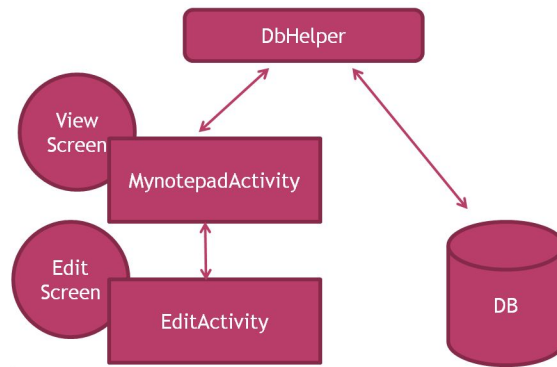


그림 30. 노트패드 애플리케이션의 저장과정

이 애플리케이션에서 생성하는 노트 데이터는 안드로이드에서 제공하는 SQLite데이터베이스를 이용하여 저장되거나 갱신된다. 이때 프로그램은 SQLite 데이터베이스에 직접 접근하는 것이 아니라 NotesDbAdapter클래스를 통해 접근한다. NotesDbAdapter클래스를 이용하면 필드 이름으로 데이터를 조회할 수 있다. 만약 테이블이 존재하지 않을 경우 SQLite 기본 메소드를 통해 테이블을 쉽게 만들 수 있다.

이번 프로젝트에서 사용할 데이터베이스는 'notes'라고 부르는 테이블을 하나 가지고 있다. 그리고 이 테이블에는 \_id, title, body와 같이 3개의 필드가 정의되어 있다.(그림) 이 중에서\_id는 데이터베이스를 조회하고 갱신할 때 키로 사용되며, 기본키의 역할을 한다. 안드로이드에서는 기본키에 해당하는 필드 이름에 언더라인(\_)을 붙이는 명명 규칙이 있다. 이 규칙에 의해 id도 \_id로 사용한다. 다른 2개는 데이터를 저장할 간단한 텍스트 필드이다.

```

private static final String DATABASE_NAME = "data";
private static final String DATABASE_TABLE = "notes";
private static final int DATABASE_VERSION = 2;

public static final String KEY_ROWID = "_id";
public static final String KEY_TITLE = "title";
public static final String KEY_BODY = "body";

```

그림 31. 데이터베이스 테이블 및 필드 정보

프로그램은 NotesDbAdapter클래스를 이용하여 안드로이드 시스템과 커뮤니케이션을 하는데, 이렇게 중간자적인 역할을 하는 NotesDbAdapter 같은 클래스를 만들어 사용하는 것이 일반적인 방법이다. DatabaseHelper 클래스가 생성되며 getWritableDatabase()메소드가 호출되면, 실제 데이터베이스가 생성되고 데이터베이스는 오픈 처리된다.

```

private DbHelper dbHelper;
private SQLiteDatabase db;

@Override
public void onCreate(Bundle savedInstanceState) {
    ...
    dbHelper = new DbHelper(this);
    db = dbHelper.getWritableDatabase();
    ...
}

```

그림 32. 데이터베이스 생성

NotesDbAdapter클래스의 메소드들을 간략히 설명한다.

#### 1) createNote(String title, String body)

createNote()에는 2개의 파라미터가 있는데, 이 중 하나는 새 노트의 타이틀을 의미하고, 다른 하나는 바디를 저장하는 스트링이다. 이 2개의 파라미터로 데이터베이스에 노트를 생성한다. 만약 새 노트가 오류 없이 생성된다면 createNote()는 새로 생성된 노트의 \_id를 반환한다.

2) deleteNote(int index)

deleteNote()는 rowId를 파라미터로 전달받아 특정 노트를 데이터베이스로부터 삭제한다.

3) fetchAllNotes()

fetchAllNotes()메소드는 데이터베이스에 저장되어 있는 모든 노트를 가리키는 커서를 반환하는데, 이때 커서는 테이블 행의 집합을 반환한다.

4) fetchNote()

fetchNote()는 fetchAllNotes()와 비슷하지만 rowId로 명시된 한 개의 노트만 반환한다.

5) updateNote()

updateNote()는 rowId, title, body를 파라미터로 해당 rowId의 행을 title, body로 갱신한다.

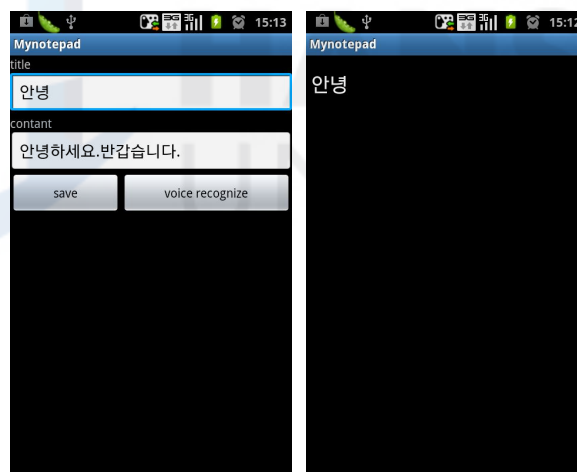


그림 33. 데이터 저장 화면

## 제 4 장 확장된 음성인식 시스템의 활용

### 제 1 절 음성인식 시스템 응용 활용 사례

음성인식 기술을 적용한 일반적 응용 사례는 유/무선 통신망 환경 기반 서비스, 단말기 기반 서비스, PC 기반 응용 서비스로 분류하고 있다. 음성인식 기술의 응용은 어린이 장난감에서부터 자동차에 이르기까지 적용이 안 된 분야가 없을 만큼 다양한 분야에 적용되어 있음을 알 수 있다.

#### 1. 텔레매틱스에서 목적지 입력 응용 서비스

텔레매틱스를 위한 음성인식 응용은 텔레매틱스 내비게이션(일명 “ESTk-Laser”라 함)에 수십만 어휘의 목적지를 음성 인식으로 처리함을 말한다. 기존의 텔레매틱스 내비게이션은 터치 스크린 방식으로 되어 있어서 사용자가 펜이나 손으로 목적지를 입력하는 방식이었고 음성인식 기술이 포함된 내비게이션의 경우 수십에서 수백 단어 급의 등록된 목적지를 인식하거나 장치를 제어하기 위한 명령어 인식이 일반적이다. 여기에서 소개하는 수십만 단어의 어휘를 처리할 수 있는 내장형 텔레매틱스 내비게이션 단말기의 하드웨어 사양은 ARM11 600MHz CPU이고, 운영체제는 Windows CE 5.0에 기반한다. 사용자의 사용 시나리오는 자동차의 핸들에 부착된 리모컨 장치에 음성인식 버튼을 할당하여 조작함으로써 음성인식을 할 수 있도록 하였다. 위에서 설명한 바와 같이, 대어휘를 처리해야 하는 텔레매틱스 내비게이션이 제한된 메모리와 저성능의 CPU를 갖는 제약 사항을 극복하기 위하여 고속 디코딩 기술에서 설명한 바와 같이 2단계 탐색 알고리즘을 구조적으로 채택하였다. 또한, 자동차 잡음 환경에 대해서도 강인한 인식을 할 수 있도록 화자 적응 및 환경적응 음향 모델링 기법과 잡음 처리 기술이 적용되었다. 또한, 잡음 환경에서 배경 잡음을 거절하기 위한 발화검증 기술이 사용되었다.



그림 34. 텔레매틱스 네비게이션

<그림 34>은 음성인식 기술이 적용된 텔레매틱스네비게이션 단말기와 리모컨을 보여주고 있다. <그림 34>은 26만 단어로 구성되는 목적지 입력 음성 응용 서비스에 대해 실제 텔레매틱스 전용 단말기에서 10-best 인식 후보 제시 인터페이스의 예이다.

## 2. 음성인식 TV 가이드 응용 서비스

디지털 TV의 빠른 보급과 더불어 IPTV 시범서비스 실시, DMB, 와이브로, WCDMA, 케이블 방송이 활성화 됨에 따라 우리가 선택할 수 있는 채널 수는 180여 개, 제공되는 프로그램 수는 13,800여 개이다. 이와 같이 사용자가 선택할 수 있는 채널 수와 프로그램은 다양하지만, 일일이 리모컨을 사용하여 원하는 채널과 프로그램을 선택하는 것은 매우 불편한 일이다.

따라서, 음성인식 기술과 대화처리 기술을 결합한 TV 가이드 응용 서비스를 제공함으로써 리모컨을 통한 채널 선택과 프로그램 선택의 불편함을 해소하고, 사용자와의 자연스러운 대화를 통해 적절한 정보를 제공해 줄 수 있다. TV 가이드 응용 서비스는 셋톱 형태로 개발되고 있으며, FSN 기반 연결어 인식 및 핵심어 인식 기능이 제공된다. 뿐만 아니라 다양한 문형을 수용하기 위해 대화 말뭉치를 바탕으로 소규모 영역 온톨로지 기반의 문맥 자유 문법으로 음성인식 문법을 작성하는 방법이 적용되었다. TV 프로그램은 EPG 제공사이트를 통하여 실시간으로 최신 정보를 받을 수 있도록 하였다. <그림 21>은 음성인식 기술이 적용된 셋톱박스를 보여준다. 이 셋톱박스과 연결된 TV 모니터를 통해 <그림 21>과 같은 TV

프로그램 정보 검색을 음성으로 수행할 수 있다. (그림4)는 음성인식 TV 가이드 사용 시나리오를 간단히 보여준다. 사용자가 리모컨에 부착된 마이크와 음성인식 버튼을 사용하여 “오늘 MBC 드라마 검색해”라고 발성하면 하단에 인식결과가 나오고, TV 화면의 검색리스트에 해당하는 프로그램 정보를 보여준다. 사용자는 검색된 정보에 대해 자세한 정보를 원하면 추가로 볼 수도 있고, 녹화, 예약녹화 등 다양한 기능 제어도 음성으로 가능하다.



그림 35. 음성인식 기술이 적용된 셋톱박스과 TV

## 제 2 절 애플리케이션에 확장된 음성인식 시스템 활용

음성인식 기능 있는 노트북 애플리케이션은 저장되는 데이터를 다른 애플리케이션과 활용 할 수 있다.

### 1. 구글 번역

구글 번역(Google Translate)은 구글이 무료로 제공되는 통계 기반 기계 번역 서비스이며, 문자의 단락, 문서나 웹페이지를 다른 언어로 뜻을 옮기는 것을 도와 준다. 대한민국에서는 “구글 번역기”라는 용어도 이따금씩 사용되지만 구글이 밝히는 정식 용어는 “구글 번역”이다. 이 서비스는 2007년에 처음 선보였다. 그 전까지 구글은 Babel Fish, AOL, Yahoo! 등의 번역 서비스에도 쓰였던 SYSTRAN 기반 번역기를 사용하였다. Babel

Fish와 AOL, Yahoo!, MSN 등 내부적으로 SYSTRAN를 사용하는 번역 서비스와 달리 Google은 독자적인 번역 엔진을 사용하고 있다. Google 번역 엔진, 기계 번역 중에서도 통계적 기계 번역이라는 기술을 이용하고 있다. 이 방법은 the United Nations Documents를 데이터로 닦아왔다. 번역된 데이터는 200억 정도의 단어로 구성된다.

이 서비스는 음성인식을 통하여 문장을 번역할 수도 있으며 또한 웹페이지를 번역할 수도 있다. 번역할 수 있는 단락의 수(줄바꿈 태그 "<br>" 등으로 표시)는 페이지 단위로 제한되어 있지만, 웹페이지의 텍스트가 "<br>"없이 분할되어 있는 경우는 수천 어로 이루어진 장대한 웹페이지도 번역할 수 있다.

노트패드에서 translate to Korean버튼을 클릭하면 구글 번역 어플리케이션으로 노트패드의 데이터가 전송된다. 그 후 구글 번역에서 다시 번역 버튼을 누르면 데이터가 한국어로 번역되어 화면에 출력된다.

MynotepadActivity에 번역옵션 추가하는 코드는 다음 같이 작성한다.

```
switch (item.getItemId()) {  
    ...  
    case TRANSLATE_ID:  
        AdapterContextMenuInfo info2 = (AdapterContextMenuInfo)  
                                           item.getContextMenuInfo();  
        c.moveToPosition( (int)info2.id-1 );  
        String a= c.getString(2);  
        Intent intent1 = new Intent(MynotepadActivity.this, translate.class);  
        Bundle bundle = new Bundle();  
        intent1.putExtra("word",a);  
        startActivity(intent1);  
        return true;  
    ...  
}
```

그림 36. 구글 번역 실행

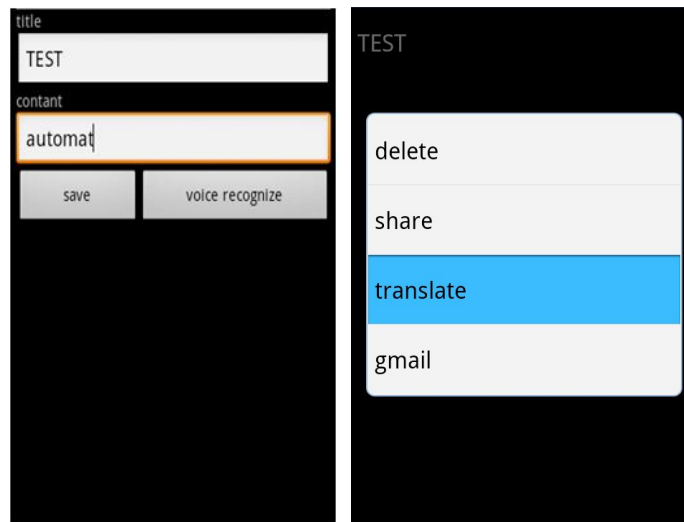


그림 37. 데이터 저장 및 구글 번역 발송 화면

번역하기 위해서 구글 번역 애플리케이션이 실행되어야 한다. 전체 프로젝트에 구글 번역 애플리케이션이 새로운 액티비티로 생성되며 구글 번역 어플리케이션을 사용하기 위해서는 Google Translate API for Java(매니페스트에 선언하지 않음)라는 외부 라이브러리를 추가해야 한다. 이 번역 애플리케이션은 노트패드 바디에 있는 데이터를 얻고 번역 버튼을 누르면 번역 결과를 아래의 TextView에 표시한다. 번역할 원본 텍스트의 언어를 자동으로 감지하여 한글로 번역해야 한다. 이를 위해 다음과 같이 액티비티 코드를 작성한다.

```

public void onCreate(Bundle savedInstanceState) {

    ...
    Bundle extras = getIntent().getExtras();
    if (extras != null) {
        inputText.setText(extras.getString("word"));
    }
    Button translateButton = (Button)findViewById(R.id.translateButton);

    translateButton.setOnClickListener(new OnClickListener(){
        @Override
        public void onClick(View v) {
            String after = new String();
            try{
                after = Translate.execute("word", Language.AUTO_DETECT,
                                         Language.KOREAN);

                translatedTextView.setText(after);
            }catch(Exception e){
                e.printStackTrace();
            }
        }
    });
}

```

그림 38. 구글 번역 실행결과 출력

노트패드에서 받은 문장을 번역하려면 execute()메소드를 사용한다.  
 public String execute(String text, Language from, Language to)  
 from인자의 언어로 작성된 text 문자열을 to 인자의 언어로 번역한다.  
 Translate.execute()메소드로 실행한 번역 결과는 반환 값으로 돌아온다.  
 번역할 문장의 언어를 자동으로 감지하려면 from의 인자로 자동 감지  
 (Language.AUTO\_DETECT)를 사용할 수도 있다. Translate.execute()메  
 서드는 구글 서버를 사용하여 문장을 번역하므로 manifest.xml 문서에 인  
 터넷 사용 권한을 추가해야 한다.

```
<uses-permission android:name="android.permission.INTERNET"/>
```

그림 39. 안드로이드 매니페스트 권한 추가

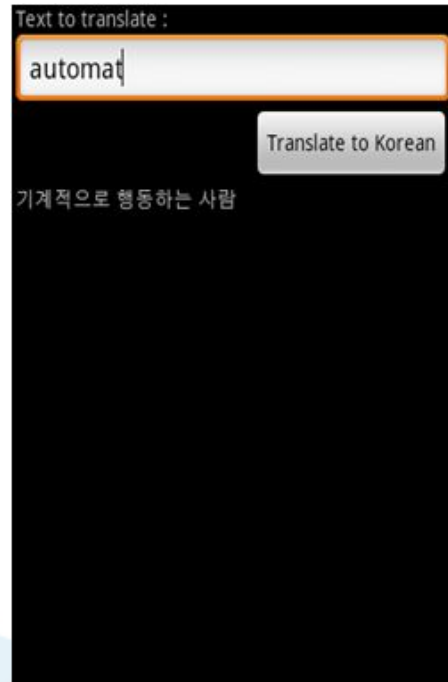


그림 40. 구글 번역 결과 화면

## 2. Gmail 발송

Gmail은 구글이 제공하는 대용량(계정당 7.1 기가바이트 이상) 무료 POP3와 웹메일 서비스(IMAP 지원)로서, 검색 지향적 인터페이스와 ‘대화 형식으로 보기’라는 독특한 기능을 가지고 있다. Gmail은 Ajax 프로그래밍을 사용한 대표적인 웹사이트이다.

현재 노트패드에서는 Gmail을 이용할 수 있다. Gmail 라이브러리를 이용할 때 canvas를 통해 로딩 중이라는 화면을 보여주게 되며 클라이언트 사이트 스토리지는 오프라인 시 데이터를 보관하기 위해 사용된다. canvas는 Mozilla의 Bepin에서 인터페이스의 렌더링에 사용되고 있어 응용프로그램 UI에 많이 사용될 것으로 보인다. 인터페이스 쪽을 살펴보면

border-image나 CSS Transitions등 새로운 기능을 사용하고 있으며, 이미지는 data: URI를 이용하고 HTTP 요청을 최소화하려는 것으로 파악되고 있다.

HTML 5에 대한 사용에 대해서는 “Google code blog의 HTML 5 and WebKit pave the way for mobile web applications”라는 포스팅글에서도 확인할 수 있다. 지금까지 Java ME에 의한 응용프로그램 위주였던 서버측 웹 응용프로그램을 생각해보면, 초기의 Android에 내장되어 있던 WebKit의 기능에 의해 HTML 5를 사용하게 되었다고 쓰여져 있다. HTML 5나 CSS 3를 사용하는 것이 아직 브라우저마다 대응하고 있는 현실과 다르기 때문에 일반적인 웹 환경에서는 크로스 브라우저에 대응하는 시간적인 수고를 들이는 것이 현실적이지 못하지만, 모바일용 Gmail처럼 같은 렌더링 엔진을 사용하는 기기라면 일반적인 웹환경의 제약이 없어져 상당히 자유롭게 응용프로그램을 만들 수 있다고 본다.

노트패드 애플리케이션 보면 저장된 데이터를 지메일으로 상대방에게 보낼 수 있다. 액티비티 소스 코드에서 Intent 클래스에 ACTION\_SEND 메소드를 사용한다.

```
switch (item.getItemId()) {
    ...
    case Gmail_ID:
        AdapterContextMenuInfo info1 =
            (AdapterContextMenuInfo)
            item.getContext().getContextMenuInfo();
        c.moveToPosition((int)info1.id-1);
        String x= c.getString(2); // 1: title, 2: body
        Intent sendIntent = new Intent(Intent.ACTION_SEND);
        sendIntent.putExtra(Intent.EXTRA_TEXT, x);
        sendIntent.setType("text/plain");
        startActivity(Intent.createChooser(sendIntent, null));
        return true;
    ...
}
```

그림 41. Gmail 실행

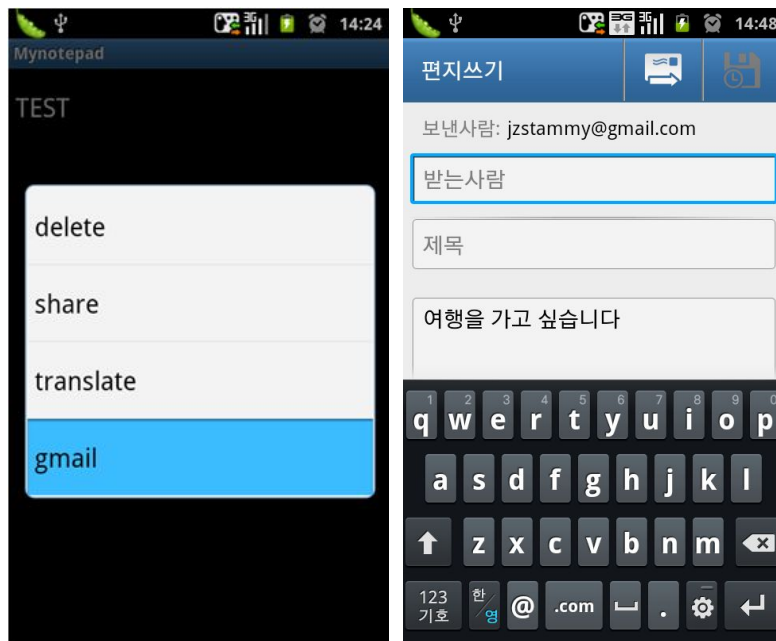


그림 42. Gmail 발송 화면

### 3. 애플리케이션 공유(share)

프로그램이 되도록이면 specific한 기능을 하나라도 더 가져야 한다고 생각하는 사람은 아마 아이폰 유저일 것이다. 웹브라우저에서 read it later 어플리케이션에 저장하는 기능도 가지고, 구글 리더로 피드백을 추가하는 기능도 있고, 또한 문서 뷰어는 이메일 첨부파일을 가져오는 기능과 가능하면 여러가지 문서의 포맷을 지원하며, 편집기능도 있으면 좋겠다 라고 생각할 것이다.

물론 프로그램이 올인원이면 좋을 것이다. 하나의 애플리케이션 안에서 모든 일을 다 처리할 수 있으나 단점은 프로그램이 덩치가 커지고, 커진 만큼 버그가 많아질 것이다. 따라서 하나의 프로그램이 여러 가지 기능을 모두 뛰어나게 갖추긴 어려운 것이다.

안드로이드 프로그램에서 메뉴를 눌러보면, 공유라는 항목을 가지고 있는 프로그램들이 흔히 있다. 한 프로그램에서 다른 프로그램으로 데이터를 넘겨주는 기능이다. 이를테면 웹브라우저는 웹주소를, 파일관리자는 파일

을, 그림관련 프로그램은 그림파일을, 드랍박스같은 웹하드 프로그램은 파일 자체나 링크를, 음악 프로그램은 음악파일을, 문서 프로그램은 문서 내용을 공유한다. 한편으로는 공유를 받을 수 있는 프로그램들이 있다. 예를 들면 Gmail은 모든 종류의 파일을 공유 받을 수 있고, 드랍박스로 모든 종류의 파일을 공유 받을 수 있고, flickr 업로드 프로그램은 사진을 받으며 read it later나 rss 리더는 웹주소를 받을 수 있다.

이 두개가 만나면 하나의 프로그램이 specific한 기능을 일일이 하나하나 지원하지 않더라도 자신이 지원하지 않는 기능을 다른 프로그램의 힘을 빌어 지원할 수 있다. 하나의 예를 들자면, 웹브라우저로 웹서핑을 하다가 read it later로 저장하고 싶을때, 그 페이지에서 share를 누른 후, 웹주소를 read it later로 보내면 끝난다. 웹브라우저 자체는 read it later로 스크랩하는 기능을 가지고 있지 않지만, share 기능을 지원하고, read it later 프로그램의 힘을 빌어 해당 기능을 지원하게 되는 셈이다. 다른 예로 파일관리자는 웹하드 업로드 기능이 없지만, 파일관리자에서 어떤 파일을 선택하여 share를 선택 한 후 드랍박스와 같은 웹하드 프로그램으로 보내면 그 파일은 드랍박스로 바로 업로드된다. 드랍박스는 자체적으로는 이메일 첨부 기능이 없지만, 드랍박스에 올라온 파일을 선택하여 share를 고른 다음 Gmail 프로그램으로 보내면, 바로 파일을 다운로드 받아서 Gmail로 첨부한다. 음악프로그램은 곡 전체를 벨소리로 지정하는 기능은 있어도, 음악의 일부만 편집해서 벨소리로 만드는 기능은 없지만 share로 ringtone 프로그램으로 보내면 벨소리를 만들 수 있다. 또는 음악프로그램에서 메일로 첨부한다든지, 웹하드로 올린다든지, 에버노트에 첨부하는 등의 모든 것들이 가능하다. 또는 페이스북 관련 프로그램에서 특정 글을 share를 통해 에버노트로 보내면, 그 글의 내용으로 에버노트에 업로드가 된다. 다른 측면에서, 만약 에버노트란 프로그램을 설치한다면, 그와 동시에 폰에 이미 설치된 share 기능을 가지는 모든 프로그램이 share 기능을 통해 에버노트 업로드 기능을 확장하게 되는 셈이다.

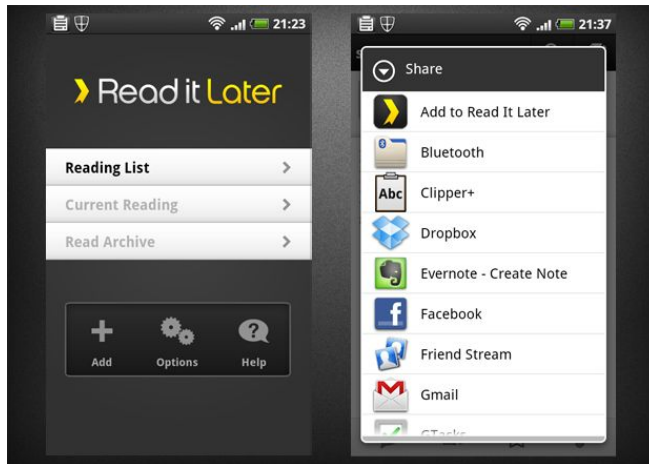


그림 43. Read it Later 실행화면

이런 서로 다른 프로그램들 기능의 조합을 통한 융통성 있는 기능의 확장은 안드로이드의 큰 장점이고 비교우위라고 생각한다. 따라서 안드로이드 프로그램은 자신이 가장 잘하는 기능에 집중하면 되지, 종합선물세트가 되어야 할 필요가 없다. 아마 앞으로도 종합선물세트 같은 프로그램은 안드로이드에서는 찾기 어려울 것이고, 누군가는 계속 안드로이드에는 쓸만한 프로그램이 없다고 할 것이다. 하지만 안드로이드에서 웹하드 프로그램은 다른 기능이 필요 없다. 파일 업로드, 다운로드만 잘하면 되는 것이다. 그런 면에서 다음클라우드 같은 프로그램에 불만이 있다. 안드로이드용 다음클라우드는 share를 지원하지 않지만, 다음클라우드에 있는 파일을 한메일에 첨부해주는 기능이 있다. 하지만 한메일이 아닌 다른 이메일을 사용하는 사람들에게겐 필요 없는 기능이다. 만약 Share를 지원한다면, 해당 파일을 share로 Gmail로 넘기면 그만인데, 만족스럽지 않은 이메일 첨부기능을 사족처럼 가지고 있는 프로그램인 것이다.

노트패드에서 저장된 데이터도 share 기능을 추가하여 여러 소셜 어플리케이션에 공유 할 수 있다. 추가 방법은 Intent 클래스에 ACTION\_SEND 메소드를 사용하며 코드 부분은 다음과 같다.

```

switch (item.getItemId()) {
    ...
    case SHARE_ID:
        AdapterContextMenuInfo info3 =
            (AdapterContextMenuInfo) item.getMenuInfo();
        c.moveToPosition((int)info3.id-1);
        String b= c.getString(2); // 1: title, 2: body
        Intent intent=new Intent(Intent.ACTION_SEND);
        intent.setType("text/plain");
        intent.putExtra(Intent.EXTRA_SUBJECT,"");
        intent.putExtra(Intent.EXTRA_TEXT,b);
        intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        startActivity(Intent.createChooser(intent, "share"));
        return true;
    ...
}

```

그림 44. 애플리케이션 공유 실행

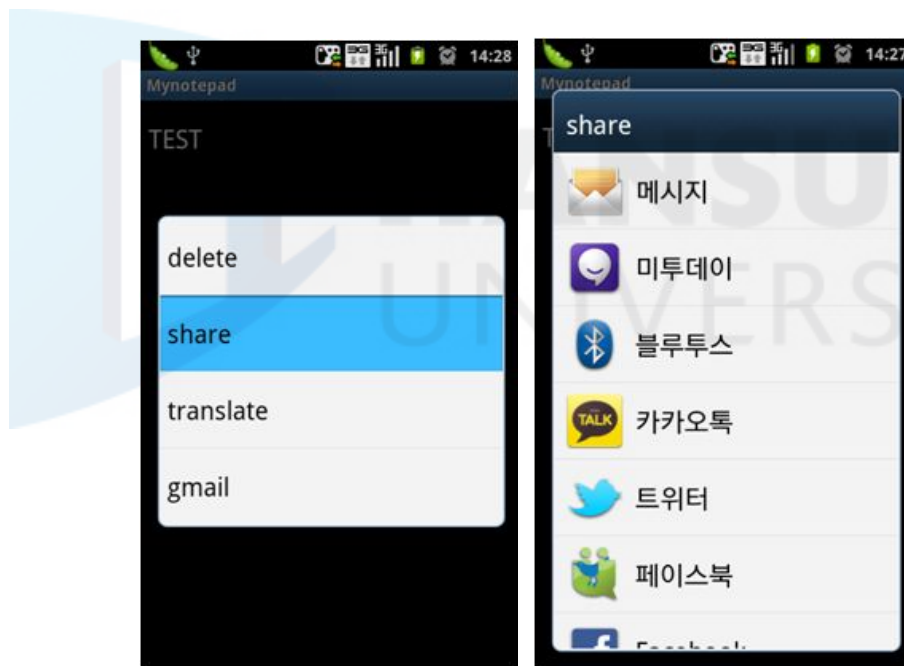


그림 45. 애플리케이션 공유 화면

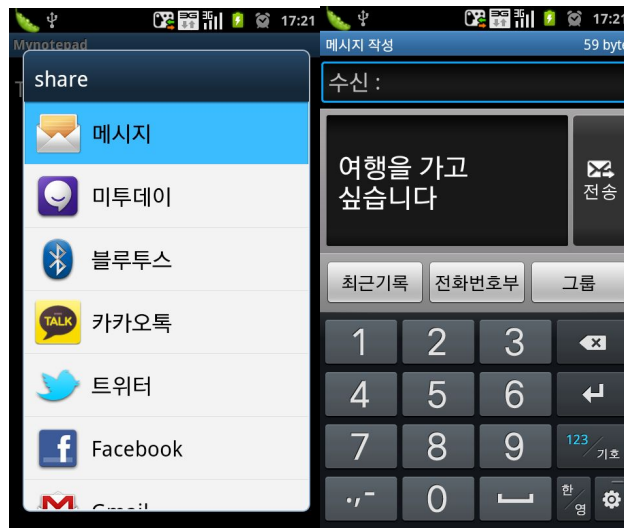


그림 46. 메시지 전송 화면

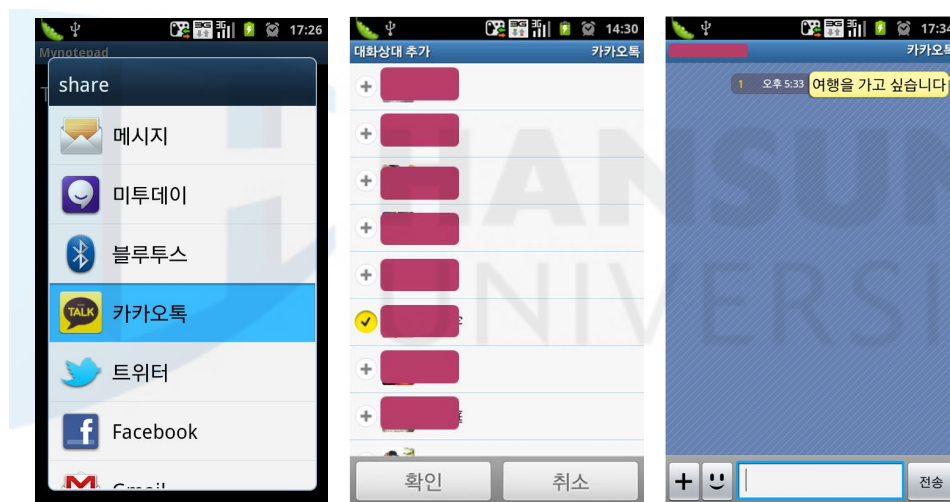


그림 47. 카카오톡 발송 화면

## 제5장 결론 및 향후연구

### 제 1 절 평가 및 고찰

본 연구에서는 안드로이드 기반으로 노트패드의 사용자가 Mynotepad 애플리케이션을 통하여 입력된 음성정보를 문자정보로 전환하여 이를 구글 번역, G메일, 공유등 기능에서 사용할 수 있게 하는 것은 테스트를 수행하였다.

기존의 안드로이드 기반 스마트폰에서 음성검색 애플리케이션이 구글 음성인식 서비스를 통해서 사용자가 음성정보를 문자정보로 전환하여, 구글 검색엔진을 연결해 검색 기능을 수행할 수 있는 반면에 Mynotepad에서는 이런 문자정보를 저장 및 재편집을 가능하게 만들었다. 그런데 음성검색을 수행하기 전에 안드로이드 기반 스마트폰에서 음성인식기 언어를 미리 설정해 주어야 한다. 다음으로 애플리케이션을 가동하고 설정된 언어와 같은 언어로 사용하게 되면 사용자가 입력된 음성정보를 구글 클라우드 서버에 전달하여 분석하게 된다. 이 과정을 거쳐서 나온 결과는 정확도 순서대로 listview에 나타난다. 동시에 음성정보 길이와 네트워크의 속도, 휴대폰 성능에 따라 식별 시간은 2초부터 10초까지 다양하게 나타났다.

이 과정을 거쳐 인식된 정확한 문자를 선택하고 휴대폰의 저장장치를 통해 저장하고 구글 번역, G메일, 소셜 애플리케이션 공유 등 추가기능을 수행할 수 있다. 이는 기존에 소프트 키보드 입력 방식과 달리 키보드가 없는 상태에서 문자 입력 및 데이터 전송 또는 공유를 가능하게 한다. 안드로이드 기반의 애플리케이션 개발자가 이런 기능을 통해 전통적인 소프트 키보드로 문자 입력방식에다 음성인식 입력방식을 추가하면 애플리케이션 사용자에게 입력 효율성과 편의성이 향상된다. 안드로이드 기반 스마트폰에서 다양한 애플리케이션(증권, 뉴스, 기상, 지식, 쇼핑 등)에 적용한다면 더욱 효율적이고 사용자 편의적인 정보 전달이 가능할 것으로 생각

된다. 동시에 구글이 제공하는 다양한 언어 지원을 이용하여 입력된 문자 정보를 다른 언어로 전환해서 언어적인 경계를 없애고 정보이용의 다양성을 실현할 수 있을 것이다.

## 제 2 절 기대효과 및 전망

기존의 Android에서 제공되는 구글 음성인식기에서는 입력된 음성정보를 직접 검색 창에 입력하고 검색하는 방식으로 수행하였으므로 사용자의 불편을 초래하였다.

본 연구에서는 이런 제한점을 개선해서 입력된 음성정보를 문자정보로 전환해서 저장, 편집 등 기능을 추가하였고 동시에 입력된 문자를 재사용(번역, 이메일, 공유)이 가능하게 만들었다. 또한 구글이 제공하는 다국어 언어 지원을 이용해 다양한 언어를 음성으로 입력해서 문자정보로 이용을 가능하게 만들었다.

Android 플랫폼의 노트 패드 전통 소프트 키보드 입력방식에 음성인식 기능을 추가하기위해 Android speech API 중에 RecognizerIntent 클래스 객체를 사용하여 사용자의 문자를 입력하는 과정에서 또 하나의 선택을 제공해주며, 사용자가 문자를 입력하는 효율성과 편의성을 향상시킬 수 있다.

언제 어디서든 음성으로 문자를 입력하는 서비스를 제공할 수 있기 때문에 기존에 입력방식을 의지하는 어플리케이션을 활용하고자 하는 개발자들은 본 연구에서 음성인식을 활용한 문자입력방법을 이용하여, 안드로이드 스마트폰 어플리케이션 사용자에게 더 쉽고 빠르고 편리하게 정보를 입력할 수 있게 되었다. 따라서 기존의 안드로이드에서 제공한 음성인식기의 여러 가지 단점이 보완 및 개선되었다고 생각한다.

향후 연구 과제로는 음성인식하는 과정에서 문장의 길이를 설정하는 것, 특히 사용자가 비교적으로 긴 정보를 입력할 경우에 어떤 방식으로 정보를 저장하고 이어지는 문제는 앞으로 이 어플리케이션을 지속적인 발전이 가능한 부분이며, 연구할 과제로 남았다.

음성으로 문자를 입력하는 서비스를 통해 스마트폰 어플리케이션을 사용하는 사람들은 더욱더 쉽고 편리한 입력방식으로 많이 사용이 될 것이라고 생각된다.



## 【참고문헌】

### 1. 국내문헌

강준구, 「음성 인식 시스템에서 사용자 인터페이스 연구」, 한국과학기술원, 1994

금태권, 「안드로이드 OS 기반 한국어 TTS 서비스의 설계 및 구현」, 원광대학교대학원, 2010.

김개원, 「클라우드 컴퓨팅 기반의 모바일 음성 인식 시스템」, 서울시립대학교 대학원, 2011.

김대근, 태지호, 「스마트폰 이용자의 매개 경험에 관한 연구」, 인문콘텐츠 제19 호, 2010.

김동희, 「안드로이드 애플리케이션 보안취약점 점검에 관한 연구」, 동국대학교국제정보대학원, 2011.

김정훈, 『구글의 안드로이드 프로그래밍』, BM 성안당, 2010년 pp. 218-313.

김태호, 『(기초부터 다지는) 커니의 안드로이드』, 인사이드, 2011. pp.320-324.

김한웅, 「안드로이드 기반의 스마트폰 소프트 키보드 개발 및 사용성 평가」, 아주대학교대학원, 2011.

동역 메카트로닉스 연구소, 『음성합성과 음성인식 시스템』, 영진출판사, 1990년.

유귀현, 「안드로이드 플랫폼 기반의 영상특징 추출 및 인식 알고리즘 연구」, 세종대학교 대학원, 2011.

이상복, 「음성 인식을 위한 효율적인 사전 구성에 관한 연구」, 전북대학교 대학원, 2003.2.

정보통신산업진흥원, 「모바일과 클라우드로 재조명받는 ‘음성인식」」, 『최신

IT동향』, 정보통신산업진흥원, 2011, pp 27-37.

하얀돌이, 「VoiceXML을 이용한 모바일 음성인식 시스템구현」, 동신 대학교 대학원, 2002.

한송희, 「안드로이드 기반 일본어문자 학습 어플리케이션개발」, 계명대학교 교육대학원, 2010.

홍인숙, 「Voice Browser용 음성인식 시스템 개발」, 한밭대학교정보통신전문대학원, 2011.



## 2. 국외문헌

- Bowen Zhou, Xiaodong Cui, Songfang Huang, Martin Cmejrek, Wei Zhang, Jian Xue, Jia Cui, Bing Xiang, Gregg Daggett, Upendra Chaudhari, Sameer Maskey, Etienne Marchere .2010. The IBM speech-to-speech translation system for smartphone: Improvements for resource-constrained tasks, *Computer Speech & Language*.
- Donn Felker, Joshua Dobbs. 2010. *Android Application Development for Dummies*, For Dummies.
- Jing Zheng; Mandal, A.; Xin Lei; Frandsen, M.; Ayan, N.F.; Vergyri, D.; Wen Wang; Akbacak, M.; Precoda, K. 2010. Implementing SRI's Pashto speech-to-speech translation system on a smart phone. *Speech Technol. & Res. Lab., SRI Int., Menlo Park, CA, USA*. 133 - 138
- Mednieks, Zigurd/ Dornin, Laird/ Meike, G. Blake. 2010. *Programm Android*, Oreilly & Associates Inc
- MILLER, CLAIRE CAIN. 2010. *Better Voice Commands For Android Phones*, COMPUTER software -- Reviews; MOBILE device applications; GOOGLE Inc.
- Quan Zhou; Ning Jiang; Englehart, Kevin; Hudgins, Bernard. 2011. Improved Phoneme-Based Myoelectric Speech Recognition. *IEEE Transactions on Biomedical Engineering*.2016-2023.
- Sayed Y.Hanshimi,Satya Komatineni,Dave MacLean.2010. *Pro Android 2*, Apress
- Scharenborg, Odette; Norris, Dennis; ten Bosch, Louis; McQueen, James .Nov/Dec2005. How Should a Speech Recognizer Work?. *Cognitive Science*. 867-918.
- Shanwn Van Every, Steve Bull , Wallace Jackson. 2010. *Developing Graphics,Music,Video and Rich Media Apps for Smartphones and Tablets*, Apress.
- 楊丰盛. 2010. Android 應用開發揭秘(Android Unleashed).机械工業出版社 (ChinaMachinePress). 280-282

# **ABSTRACT**

## **Design and Application of Speech Recognition System on Smart Phone**

**JIANG, ZI SHUN**

**Major in Multimedia Engineering**

**Dept. of Multimedia Engineering**

**Graduate School, Hansung University**

In the 21 st century communications technology high speed development period, smart phones become very popular means of communication. Smart phones of all kinds of application is our life to become more efficient. In the smart phone, all kinds of applications, the realization of speech recognition skills become very popular. This thesis, through to the Google android smartphone system principle and application analysis, through Google's speech recognition provide services, speech recognition programs get more extensive application, the original application convenience and efficiency improvement, make the program of the user groups more widely.

## 中文摘要

### 智能手機語音識別系統的設計與應用

蔣子舜

多媒體工學科

多媒體工學專業

漢城大學校，一般大學院

在當今21世紀信息科技高速發展的背景下，智能手機的出現給人們的生活、學習、工作等各方面帶來前所未有的便利，使人們之間的交流變得更有效率性和多樣性。智能手機中大量的應用程序使原本需要很繁瑣的過程變得簡單。本論文是依託谷歌安卓系統智能手機為背景，通過在手機應用程序中使用語音識別技術，利用谷歌提供的語音識別云服務，將原有的應用程序輸入方式得到改善，體現出更好的通用性，將智能手機和語音識別得到更好的結合。