디지털 TV 환경에서 음성인식을 통한 동적 EPG 제어 시스템 설계 및 구현

Design and Implementation of the Speech Recognition-based Dynamic EPG Control System in Digital Broadcasting environment

김성원, Seong-Won Kim*, 나희주, Hee-Joo Na*, 시장현, Jang-Hyun Si*,
김정환, Jung-Hwan Kim*, 정문열, Moon-Ryul Jung*

*서강대학교 영상대학원 미디어공학과 디지털방송 연구실

요약 디지털 방송은 수많은 프로그램과 기존의 아날로그 방송에서 볼 수 없었던 다양한 서비스를 제공하며 발전하고 있다. 하지만 시청자들에게는 방송 서비스 채널과 기능이 많아질수록 원하는 채널을 검색하고 전환하는 과정이 어렵고 복잡한 일이 되어 버릴 수 밖에 없을 것이다. 이에 본 논문에서는 이러한 정보 획득과정의 축소를 위해 전통적인 리모콘으로 채널을 검색하고 이동하는 절차를 벗어나 음성인식을 통한 동적 EPG(Electronic Program Guide) 제어 시스템을 설계하고 구현하고자 한다. 이는 EPG정보와 시청자의 TV시청 성향 및 History를 기반으로 구동되는 시스템으로 음성대화의 구조적 정의가 가능한 VXML(VoiceXML) 인터프리터를 활용한다. 본 논문에서 제안하는 대화형 인터페이스는 다양한 디지털방송 서비스에 접목이 가능 할 것이며, 새로운 형태의 디지털 가전기기 파일럿 인터페이스 개발에 도움이 될 것이라 기대한다.

핵심어: 음성인식, EPG, VXML, 디지털방송

1. 서론

음성인식 기술의 발전으로 TTS(Text-to-Speech), STT(Speech-to-Text)가 가능해졌고 이를 바탕으로 인터넷 음성포탈, 음성을 이용한 사용자 가이드 등 다양한 콘텐츠의 등장으로 사람의 생활환경과 가까운 사용자 인터페이스가 개발되고 있다. 방송환경에서도 디지털 전환을 거치면서 현재 100개가 넘는 다채널 방송으로 프로그램 서비스를 지속적으로 확대하고 동영상의 전통적인 방송 콘텐츠뿐만 아니라 부가정보를 위한 서비스를 위해 발전해 나가고 있는 실정이다. 이렇듯 콘텐츠가 다양화되고 발전될수록 시청자는 보다 많아진 방송 콘텐츠에 적응하고 필요한 정보 및 데이터를 검색하고 찾아내기 위해 번거로운 리모콘 조작법에 익숙해져야만 하게 될 것이다. 이에 본 논문에서는 방송시청 편의성과 인간 친화적인 사용자 인터페이스 구현을 위해 음성인식에 대한 구조적인 정의가 가능한 VXML의 대화형 인터페이스와 음성인식을 통해 검출된 문장을 기반으로 EPG데이터를 질의하고, 그 결과를 통해 채널을 검색하고 반환하는 시스템을 제안한다. 본 연구를 위해서 필요한 부가 장비로는 음성인식을 위한 셋톱박스에 부착 가능한 하드웨어 모듈과 인식된 음성데이터의 텍스트 변환 모듈, 그리고 대화형 인터페이스를 위한 VXML 인터프리터와 구문분석을 위한 형태소 분석기가 필요하다. 하지만 셋톱박스의 하드웨어적인 구현과 Add-on 하드웨어 부착이 어려운 현실적인 제약으로 음성인식 디바이스 장치는 고려하지 않고 음성인식을 통한 정확한 문장의 도출은 가능하다는 가정을 전제로 한다. 도출된 문장은 구문 분석을 통해 키워드를 검출하게 되게 되는데, 키워드 검출을 위해 사용된 시스템 모듈은 스크립트 확장형태의 PHP HAM(Hangul Analysis Module) Extension 모듈을 사용하였다.

이를 바탕으로 본 논문에서는 음성데이터의 대화형 인터페이스와 키워드 산출을 위한 구조를 정의하고, 검출된 키워드 기반의 EPG 정보 검출을 위한 시스템 설계와 구현방법에 대해서 설명한다. 본 논문의 2절에서는 인터페이스 정의를 위해 사용된 VXML에 대해서 설명하고, 3절에서는 구문분석의 방법에 대해서 다룬다. 그리고 4절에서는 동적 EPG 시스템에 대한 구성 및 각각의 프로세스에 대해서 설명하고, 마지막으로 5절에서는 본 논문의 결론을 맺고 향후 보완해야할 연구와 방향에 대해 서술한다.

2. VXML

VXML(Voice eXtensible Markup Language)은 XML에 기반을 둔 Markup language이다. HTML이 graphical web page를 만드는데 사용되는 것처럼 VXML은 spoken dialog를 정의하는데 사용되는 언어이다. 적용 가능 분야는 UMS(Unified Messaging System), ASR(automatic Speech Response) 같은 음성 관련 서비스 분야에 사용될 수 있다. 보이스포탈 서비스는 현재 그 필요성이 증대함으로써 여러 업체에서 수많은 해결 방법을 개발하였지만 이러한 시스템의 문제는 특정한 분야나 특정 서비스에 적합하도록 구현이 되어져 있으므로 이 기종간의 호환성이나 새로운 서비스의 추가 시 많은 시간과 비용이 필요하였다. 이러한 Closed system에만 적용 가능한 현재까지의 시스템에서의 문제점을 해결할 수 있는 것이 VXML이다.

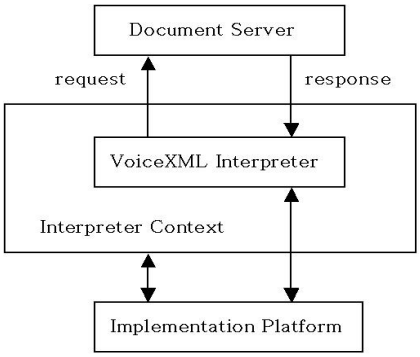

그림 1 VoiceXML Architectural Model

2.1. VXML 문서 구조

VXML Tag는 계층적인 구조를 가진다. 하나의 VXML 파일 안에는 Form Item이라고 정의되어진 여러 개의 <form>과 <menu>로 구성될 수 있는 여러 개의 Dialog가 존재할 수 있으며 각 dialog는 Field Item으로 정의된 5개의 <field>, <record>, <ransfer> <object>, <subdialog>와 같은 tag들에 따라서 지정된 작업을 수행한다. 본 논문에서 사용된 문서 구조는 다음과 같다.

main.vxml

ch.vxml desc.vxml

그림 2 VXML의 문서 구조

main.vxml은 최초 검색을 위한 종류, 즉 채널 검색 또는 내용 검색을 선택할 수 있는 type을 분리하기 위한 문서이다. 차후 보다 정확한 한글 태깅 모듈을 적재할 경우 이런 type 분리를 위한 문서는 생략 가능하다. ch.vxml에서는 입력된 숫자를 기반으로 EPG상의 채널 번호와 같은 번호를 호출하면 해당 서비스 ID를 STB로 전달한다. desc.vxml 문서는 입력된 문장에서 키워드를 도출하고 도출된 키워드를 우선 순위를 두고 STB로 전달하게 된다. 아래에는 설명한 3개 문서상의 syntax를 각각 나타낸다.

```
<?xml version="1.0" encoding="EUC-KR"?>
<vxml version="1.0">
  <form id="main">
    <field name="guide">
      <prompt>
        음성 프로그램 안내를 시작합니다.
      </prompt>
      <grammar>
        채널검색 | 내용검색
      </grammar>
      <filled>
        <if cond="type=='채널검색'">
          채널검색을 시작합니다.
          <var name="type" expr="'채널검색'"/>
          <goto next="ch.vxml"/>
        <elseif cond="type=='내용검색'"/>
          내용검색을 시작합니다.
          <var name="type" expr="'내용검색"/>
          <goto next="desc.vxml"/>
        <else/>
          잘못 선택하셨습니다.
          <reprompt/>
        </if>
      </filled>
      -> noinput, nomatch
    </field>
  </form>
</vxml>
```

그림 2 main.vxml 문서

```
<?xml version="1.0" encoding="EUC-KR"?>
<vxml version="1.0">
  <form id="channel">
    <field name="guide">
      <prompt>
        프로그램 채널 검색을 시작합니다. 채널 번호를 말씀하세요
      </prompt>
      <grammar>
        <?
          // Database Connection
          // select distinct 채널번호 from 프로그램 테이블
        ?>
      </grammar>
      <filled>
        <?
          // <if...  채널 Matching 조건
          ......
        ?>
        <else/>
          방송중인 채널이 아닙니다. 다시 말씀해 주세요.
          <reprompt/>
        </if>
      </filled>
      -> noinput, nomatch    </field>
  </form>
</vxml>
```

그림 3 ch.vxml 문서

```
<?xml version="1.0" encoding="EUC-KR"?>
<vxml version="1.0">
  <form id="desc">
    <field name="guide">
      <prompt>
        검색하고자 하는 정보를 말씀하세요.
      </prompt>
      <filled>
        <submit next="키워드 전송 스크립트(Socket)"/>
      </filled>
    </field>
  </form>
</vxml>
```

그림 4 desc.vxml 문서

3. 구문 분석

VXML을 통해 대화형 인터페이스를 구조화하고 음성으로부터 산출된 문장을 분석하여 색인어를 찾을 수 있다. 한국어 문장에서 같은 문자열이나 단어, 혹은 형태소가 문맥에 따라서 서로 다른 품사를 가질 수 있다. 따라서 그 문장에 가장 적당한 품사를 선택하는 과정이 필요하다. 품사 태깅은 그림 6과 같이 크게 각 단어에 대하여 가능한 모든 품사를 생성하는 '모호성을 생성하는 부분' 과 여러 품사 중에서 가장 적절한 품사를 선택하는 '모호성을 해소하는 부분' 으로 구성된다.[2]

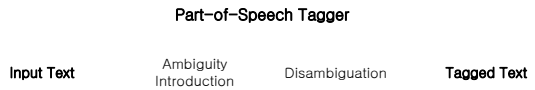

그림 5 Tagging 구조

한국어는 교착어이므로 형태소 단위의 품사 태깅을 위하여 형태소 분석을 거치게 되며 이 과정이 모호성을 생성하는 부분이 된다. 모호성을 해소하는 방법은 통계적 방법이 현재 많이 사용되고 있다. 지금까지 정확한 형태소 분석을 위한 많은 연구들이 선행되었고 현재는 90% 이상의 정확도에 다다른 상태이며 매우 안정적인 알고리즘에 근거한 연구를 바탕으로 한 구문 분석 연구가 많이 있다. 본 논문에서 사용한 한국어 구문 분석은 국민대학교 언어공학, 정보검색 연구실에서 공개한 HAM 소스를 기반으로 작성되었다. HAM은 99% 이상의 정확도와 안정성을 가지고 있는 것으로 평가되고 있으며 관련 연구 자료도 많이 나와 있는 상태이기 때문에 본 논문에서는 구문 분석 및 색인어 산출 알고리즘에 대해서는 언급을 하지 않도록 하겠다. HAM을 응용한 PHP Extension 모듈을 사용했기 때문에 PHP 모듈이 구축된 웹시스템에 이식성이 높고, http 프로토콜을 통한 파라미터 전송방식의 수용이 용의하여 본 논문에서 제시하는 Return Path를 통한 셋톱박스에서의 접근이 가능하다.

3.1. 색인어를 통한 EPG 정보 질의

고유명사와 숫자가 중심이 되는 색인어(키워드) 중심으로 추출하며 색인어 기반으로 EPG 정보의 질의가 충분한 이유는 방송에서 검색을 위한 기본적인 요소는 일반적으로 다음과 같이 정리 할 수 있다. 첫째, 연기자 또는 프로그램 관련 제작자 이름, 특정 연예인 이름, 채널명 등을 지시하는 말. 둘째, 특정 채널 번호 및 시간 등이 포함 될 수 있는 말. 그리고 일반적인 카테고리 명칭의 고유명사(NNP)를 고려 할 수 있다. 이렇듯 3가지의검색 요소를 크게 분류해보면 고유명사(NNP), 숫자(SCD)로 단편화하여 정의 할 수 있다. 그리고 예외 사항으로 고유명사 또는 숫자의 품사 태깅이 없을 경우 명사(NN)에 우선순위를 두고 핵심어를 추출한다. 추출된 핵심어는 String 배열 객체로 구성하여 Return Path를 통해 클라이언트 즉 셋톱박스로 전송되게 된다. 전송된 객체를 셋톱박스의 EPG애플리케이션은 MPEG-2 TS(Transport Stream)으로부터 Section Filtering을 통해 하루 분량의 채널 키 기반의 프로그램 제목 및 프로그램 Description 정보를 Hashtable구조의 Data-Pool를 구성하고 EPG 데이터를 저장한다. Pool에 대해서 해당 핵심어를 기준으로 키 기반 색인 시스템으로 색인어의 검색을 실시한다.

4. 동적 EPG 제어 시스템

EPG 제어 시스템은 사용자가 EPG 정보를 검색하고 선택한 후에 리모콘을 통한 채널변환이 이루어지는 단계를 생략하고 음성인식을 통하여 시청자의 요구조건을 채널번호 뿐만 아니라 EPG의 Element 및 TV 시청 기록, 선호 채널 등을 고려하여 다소 애매모호한 시청자의 음성 요청에 반응 및 키워드를 검출하고 EPG 데이터 질의를 통한 채널을 반환 또는 채널 전환을 허용하는 시스템을 위해 설계되었다.

실험을 위해서 EPG 애플리케이션은 OCAP환경에서 채널에 종속되지 않게 하기 위해 OOB(Out-of-Band)로 전송된다. 이는 케이블 환경의 특성이 잘 반영된 표준으로 트랜스포트 스트림으로 전송되는 PSIP[1]정보를 OOB로 전송하고 이에 대해서 SCTE[2]에서는 OOB-SI(Service Information)표준을 정하였다. OOB-SI는 PODExtendedChannel로 접근이 가능하다. 때문에 OOB-SI 분석을 위해 PODExtendedChannel을 통한 Section Filtering으로 필요한 프로그램의 제목, 간략 줄거리 등의 정보를 얻을 수 있다. 아래는 OOB-SI 구조를 나타내며 프로그램 정보를 필터링하기 위한 과정은 아래의 구조를 준수하여 필터링을 하게 된다.

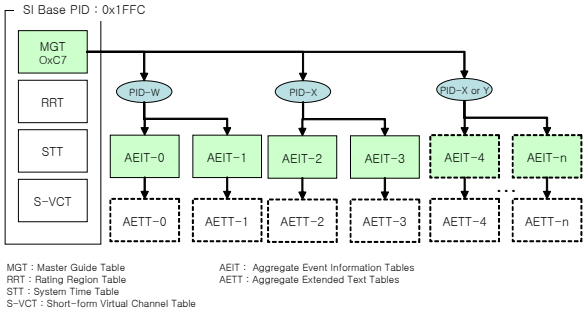

그림 6 OOB-SI 구조도

1) PSIP(ATSC A/65 Program and System Information Protocol) : ATSC 지상파 디지털 방송 표준

2) SCTE(Society of Cable Telecommunications Engineers) : 유선전기통신산업 관련 미국표준단체

OOB를 통한 섹션 필터링에서 MGT는 기존의 PSIP 섹션 필터링에 사용된 API를 사용하면 되지만, AEIT[1], AETT[2]는 다중 섹션 전송으로 인해 RingSectionFilter의 API를 제공한다.

```
SectionFilterGroup filterGroup = new SectionFilterGroup(1);
ts = PODExtendedChannel.getInstance();

try {
  filterGroup.attach(ts, this, null);
} catch (FilterResourceException e) {
  e.printStackTrace();
} catch (InvalidSourceException e) {
  e.printStackTrace();
} catch (TuningException e) {
  e.printStackTrace();
} catch (NotAuthorizedException e) {
  e.printStackTrace();
}

// For MGT
TableSectionFilter tableFilter = filterGroup.newTableSectionFilter();
tableFilter.addSectionFilterListener(this);

// For AEIT, AETT
RingSectionFilter ringFilter = filterGroup.newRingSectionFilter(size);
ringFilter.addSectionFilterListener(this);
```

그림 7 Section Filtering API

4.1. 시스템 구성

시스템 구성의 사전제한사항으로 셋톱박스에 마이크와 같은 음성인식을 위한 디바이스가 준비되어 있으며 해당 디바이스로부터 음성데이터 접근 및 변환에 대한 권한을 가지는 하드웨어 제어 장치가 적재되어 있어야 한다.

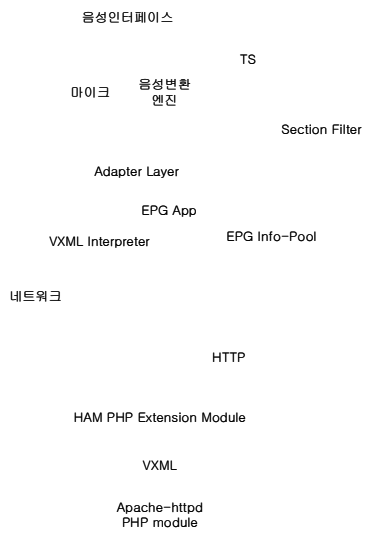

그림 8 EPG 시스템 구조도

1) AEIT(Aggregate Event Information Tables) : event의 제목, 스케줄 정보 제공

2) AETT(Aggregate Extended Text Tables) : event의 부가정보 제공

그리고 셋톱박스에는 음성데이터 추출을 위한 음성 변환 모듈이 탑제되며, EPG 애플리케이션은 변환된 음성 데이터를 Return Path를 통한 HTTP를 이용하여 VXML파일을 URL로 접근하여 내부 파서를 통해 대화형 인터페이스를 가진다. URL로 접근을 하는 과정에서 음성에서 변환된 문장을 해당 VXML 파일 대상으로 GET Type으로 전송하고 서버 내부 스크립트 HAM Interpreter를 통해 색인어를 검출하게 되고 EPG 애플리케이션의 내부 파서는 색인어를 EPG 정보저장소에서 해당 프로그램 Descriptor를 찾고 채널 정보를 반환한다. 찾는 방법은 AEIT에서 검출된 각 channel 별 Subject와 AETT의 description 정보를 추출하고 String을 whitespace 기준으로 Tokenized Subject 또는 Description을 Key Base로 채널(SRC_ID)Body 타입으로 저장소에 저장한다.

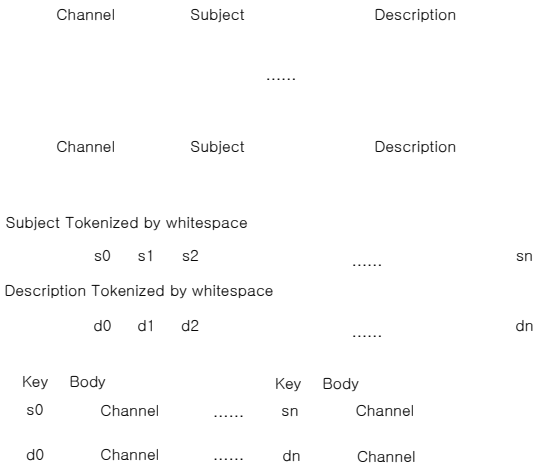

그림 9 EPG Pool 구성 프로세스

그리고 EPG 애플리케이션은 반환된 채널로 튜닝한다. 세부적인 시스템의 프로세스는 그림 8 과 같다.

현재는 EPG 애플리케이션에 음성인식을 위한 부가적인 모듈의 적재가능 여부를 판단 할 수는 없으며 차후 USB Add-On 방식의 마이크와 같은 디바이스 드라이버를 셋톱박스에 적재하고 미들웨어가 음성인식 즉, 마이크 디바이스에 접근 가능한 API를 제공하여야 한다. 클라이언트(STB)와 서버(Interpreter)간의 음성 데이터를 전송형태는 VXML로 전송하며 VXML 인터프리터 서버는 VXML을 클라이언트 VXML 인터프리터로 전송하고 음성에서 변환된 문장을 서버측으로 역전송하여 HAM Extension 모듈을 통해 색인어를 추출하고 EPG 애플리케이션에서 EPG정보를 대상으로 질의하게 된다. 그리고

최종적으로 질의 결과에 따른 채널 정보를 받은 EPG 애플리케이션은 채널변환 모듈을 통해 해당 채널로 Locator를 변환한다.

```
public void selectChannel(int source_id) {
    // Create a new service context.
    setServiceContext();
     Locator locator = null;

     // JavaTV Locators are created by a LocatorFactory
     LocatorFactory locatorFactory = LocatorFactory.getInstance();
     // Now create the Locator. This can throw an exception,
     // so we enclose it in a 'try' block.
     try {
          locator = locatorFactory.createLocator("ocap://0x"+ Integer.toHexString(source_id));
     } catch (MalformedLocatorException mle) {
          System.out.println("Can't create the locator object - malformed locator");
          mle.printStackTrace();
     }
     Locator[] locators = new Locator[1];
     locators[0] = locator;

     try {
          if (sc != null)
               sc.select(locators);
     } catch (javax.tv.locator.InvalidLocatorException ile) {
          System.out.println("Invalid Locator when selecting the new service");
          ile.printStackTrace();
     } catch (InvalidServiceComponentException isce) {
          System.out.println("Invalid service component when selecting the new service");
          isce.printStackTrace();
     }
}
```

그림 10 EPG 애플리케이션 내부 채널관리 모듈 소스

4.2. 서버 시스템 구성

서버 시스템은 BSD/UNIX 기반 시스템(FreeBSD)에 httpd processor를 적재하고 httpd 서버는 Apache에 mod_php를 DSO방식 적재하였으며, 추가적으로 HAM Parsing을 위한 모듈은 PHP Extension 모듈을 이용하였다.

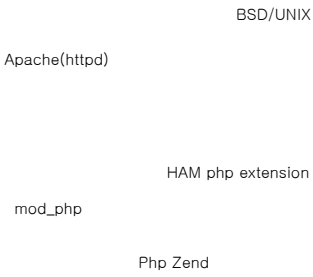

그림 11 서버 시스템 Stack

4.3. 클라이언트 시스템 구성

EPG 애플리케이션은 Xlet프로그램으로 작성되며, OOB로 전송된다. STB는 마이크로부터 생성되는 음성 데이터 리소스에 대해 접근하는 API를 가지고 URL 클래스를 통해 VXML파일을 Call 하고 Call Event를 통해 Httpd 서버로부터 VXML을 전송받고 해석과정을 거친다. 해석과정과 동시에 음성으로부터 추출된 문장은 Socket 통신을 통해 GET 타입으로 서버로 전송하고 Return Value를 기다린다. Return Value 코드에 따라 Exception 처리 및 EPG 정보를 통한 채널 변환 Event를 수행한다.

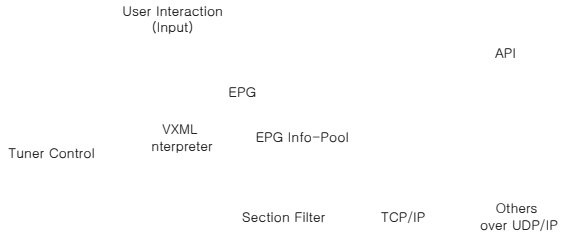

그림 12 클라이언트 시스템 Stack

5. 실험 및 향후 과제

현재는 마이크와 같은 User Interaction 부분 API 제공을 받지 못하기 때문에 장기적인 연구를 위해 STB를 위한 마이크 디바이스 적재를 위한 실험 환경을 구축하고 있다. 이번 연구 과정의 이러한 현재 환경적 제약으로 인해 실험 환경은 PC상에서 음성을 입력받을 수 있도록 구성하였으며, 음성을 택스트로 변환하는 모듈의 가정을 제외한 다른 프로세스는 본 논문에서 제안하고자 하는 음성인식 기반의 동적 EPG 제어 시스템과 동일하다. 클라이언트의 음성인식 파트를 에뮬레이션하기 위한 준비로 마이크를 부착한 PC에서 충북대학교에서 개발된 객체지향 음성인식기인 ezCSR를 이용해 음성인식 및 TEXT 변환 단계를 테스트하였으며 본 논문에서는 음성인식 부분에서 앞서 언급한것과 같이 음성인식의 안정성을 신뢰하고 구현된 간단한 음성대회 인터페이스를 통한 핵심어 검출을 중심으로 설계하였고 EPG 저장소를 생성하고 핵심어에 적합한 채널을 검출하는 시스템으로 구현하였다. 본 논문에서 구현된 시스템은 OCAP 1.0 표준을 지원하는 "HUMAX OC-2500" 셋톱박스를 이용하여 EGP 저장소를 생성하고 음성인식 결과에 따른 채널정보를 추출하는 실험을 하였다.

본 연구를 통해 EPG 인터페이스의 보다 넓은 범위로의 확장 가능성을 제시하였으며, 아직까지는 STB의 저장공간, I/O 및 프로세서 성능의 제약으로 인해 개선의 여지가 많지만 STB에 응용 가능한 확장된 네트워크 파일시스템의 도입과 USB와 같은 다양한 Add-On 디바이스의 지원이 원활하게 된다면, 현재 디지털 방송은 보다 많은 콘텐츠를 포용하기 위해 다양한 방법으로 시청자들에게 콘텐츠를 제공할 수 있을 것이다.

또한 네트워크 환경에 따라서 "음성 변환 - 데이터 추출 - 전송 - VXML 파싱 - Value Return - 채널변환 / Event" 에 이르는 프로세스를 분석하여 성능 개선을 위한 데이터의 Request / Response 프로토콜의 재정의가 가능할 것이다. 이 실험의 결과에 따라서 향후 개인화된 데이터 기반 및 음성의 패턴 정보와 음성을 통한 보다 정확한 음성 특징 벡터분석으로 감정의 변화까지도 분석하고 인공지능 TV 시스템을 구

축하는데 도움이 되었으면 한다.

※ 본 연구는 2005년도 과학기술부 기초과학연구사업 특정기초연구 "디지털 TV를 위한 개인맞춤형 EPG 개발 및 사용성 평가"에 의해 지원되었음.

참 고 문 헌

[1] 김한수, 황인준, “VoiceXML 기반 EPG검색 시스템”, 정보공학회논문지, 컴퓨팅의 실제 제 10권 제 4호, 2004

[2] 신상헌, 이근배, 이종혁, “통계와 규칙에 기반한 2단계 한국어 품사 태깅 시스템” 정보과학회논문지(B) 제 24권 제 2호, 1997

[3] 박정열, “2단계 방법을 사용한 빠른 한국어 TAG 구문 분석기 구현”, 2004년도 한국정보과학회 가을 학술발표논문집 Vol.31 No.2, 2004

[4] ANSI/SCTE 90-2. "SCTE Application Platform Standard OCAP 2.0 Profile", 2005

[5] ANSI/SCTE 65 2002(formery DVS 234) "Service Information Delivered Out-Of-Band for Digital Cable Television"

[6] ATSC Standard A/101, "Advanced Television System Platform", August, 2005

[7] Digital Video Broadcasting(DVB), "Multimedia Home Platform 1.0.3", ETSI TS 101 812 V1.3.1. June, 2003

[8] ATSC Standard A/100. "DTV Application Software Environment - Level 1 (DASE-1)", March, 2003

[9] DAVIC, "DAVIC 1.4.1 Specification Part 9 : Information Representation", 1999

[10] Edward M. Schwalb, "iTV Handbook", Prentice Hall, 2004

[11] 이재훈, 정문열, “복수의 프로그램 속성 값 지정을 통한 EPG User Interface”, 한국방송공학회 논문지 10권 1호, 2005

[12] Steven Morris,"Interactive TV Standards", Focal Press, 2005

[13] http://www.voicexml.org

[14] http://www.speechframe.com

[15] http://www.php.net