

# 한글 TTS 시스템에서 문자열-발음열 변환기의 L-소리 첨가 전 처리기

정경석<sup>1)</sup> 박혁로

전남대학교 전산학과

{ksjung@dal.chonnam.ac.kr, hrpark@cs.chonnam.ac.kr}

## Ni-eun Addition Preprocessing of Grapheme to Phoneme Conversion for Korean TTS

Kyung Seok Jeong<sup>1)</sup> Hyuk-Ro Park

Dept. of Computer Science, Chon-Nam National University

### 요약

본 논문은 L-첨가 규칙을 전 처리를 통한 문자열-발음열 변환 기법을 소개한다. 한국어 TTS시스템에서의 고질적인 문제는 문자열-발음열 변환이라고 할 수 있는데, 그 이유는 한국어의 특징상 음운적 조건과 형태론적 조건등에 의해 다양한 방법과 예외처리를 요구하기 때문이다. 그 중 L-첨가 규칙은 위와 같은 대표적인 현상으로 많은 문제점과 그에 따른 연구를 필요로 하고 있다. 이 시스템은 형태소 분석을 선행한 후, 특수문자나 숫자를 정규화하고 L-첨가 규칙을 전 처리한 후, 음운변화 현상을 분석하여 선택적으로 규칙을 적용하여 발음열을 생성한다. 제안된 시스템은 기존의 시스템에 비해 더욱 효과적인 음운,형태소 변화를 가져옴과 함께, 특히 L-첨가가 적용되는 문장을 효과적으로 해결할 수 있어 TTS시스템에 좋은 결과를 가져오게 될 것이다.

### 1. 서론

최근 컴퓨터의 발전과 인터넷의 보급으로 숫자와 문자를 주로 처리하던 기존의 정보처리방식에서 벗어나 인간에게 친숙한 시각, 청각 형태의 정보를 처리하여 컴퓨터 사용자에게 편리함을 제공한다. 이 중 사람과 컴퓨터간의 인터페이스 수단으로 음성이 적합하다고 생각하여 음성합성 시스템과 음성인식 시스템이 상용화되고 있다. 특히 음성합성 시스템은 합성음의 명료성과 자연성 향상이 목표로 현재 상용화 단계에 이르렀지만 더욱 많은 연구를 필요로 하고 있다.[1]

한국어의 음절은 자음(초성), 모음(중성) 그리고 선택적인 자음(종성)으로 이루어진다. 한국어는 음절 기반의 표음 문자이므로 각각의 철자는 특정 문자열을 나타낸다. 그러나 실제 발음 시 철자에 나타난 문자열은 그대로 발음되지 않고, 음운 변화 과정을 통해서 여러 가지 다양한 발음열을 생성한다. 특히 한국어는 교착어이기 때문에 많은 어절들이 실질 형태소와 형식 형태소의 결합에 의해 생성된다. 더욱이 대화 음성에서는 두 개 이상의 어절이 하나의 언절로 묶여 발음되기도 한다. 이러한 한국어의 특징으로 인해 음운 변화는 주변 단어의 문자열과 형태소의 종류에 크게 영향을 받는다.[2]

규칙적인 음운 변환 과정 중 L-첨가는 형태론적으로나 음운론적으로 자동 발음 생성화에 비교적 많은 문제점과 예외상황을 가지고 있다. 예를 들어 '숨이'라는 명사와 조사가 결합한 어절의 경우와 '숨이불'이라는 복합어가 있는 경우, '숨이'는 /소미/로 발음되고 '숨이불'은 /숨니불/로 발음된다. 같은 형태의 음소 배열이지만 구성 형태소가 다르므로 첫 번째 경우는 연음 규칙이 적용된 경우이고, 두 번째는 L-첨가가 적용된 경우이다. 하지만 기존의 명사사전에는 '숨이불'이라는 단어가 한 복합어로 구성되어 있어 '숨이불'을 단일 명사로 구분하므로 '소미불'로 연음이 되 버리는 오류를 범하게 된다.

본 논문에서는 이러한 문제를 해결하기 위하여 L-첨가가 일어날 수 있는 복합어를 제외한 사전을 이용한 L-첨가 전 처리기를 구현하여 L-첨가 규칙을 해결하고자 하고 또한, L-첨가의 음성학적 규칙과 이용 사례를 연구하여 더 나은 L-첨가 발음열을 얻고자 한다. 제안된 시스템은 단순 형태소 분석된 기존의 시스템의 단점을 보완한 시스템으로 TTS의 성능 개선에 도움을 줄 수 있을 것이다. 2장에서는 관련연구를 논하고 3장에서는 한국어의 음운 변화와 L-첨가 규칙에 대해서 더욱 알아보고, 4장에서는 제안하는 모델에 대해 설명

하고, 제 5장에서는 실험에 대해 기술하고 마지막으로 제 6장에서는 결론 및 향후 연구에 대해 논하겠다.

**2. 관련연구**

기존의 문자열-발음열 변환기 연구들은 크게 형태론적 지식, 음성학적 지식과 의미기반 지식을 요구한다. 하지만 일반적으로 의미기반을 제외하고 형태소 분석과 음운, 형태론적 규칙에 의거하여 발음열을 자동 생성하였다. 하지만 여러 예외적 상황과 특히 ㄴ-첨가 규칙에는 좋은 결과를 제시하지 못 함을 지적할 수 있었다.[1][3]

본 논문은 위와 같은 문제점을 해결하고자, ㄴ-첨가 전 처리기와 예외적 상황을 수집함으로써 문제를 해결하고자 한다.

**3. 한국어의 음운 변화와 ㄴ-첨가 규칙**

한국어의 음운변이는 하나의 형태소 안에서, 형태소와 형태소 사이에서, 그리고 어절과 어절 사이에서 일어날 수 있고, 음운적 조건에 의한 것 이외에 형태론적 조건에 의해서 일어나는 음운변이도 있기 때문에 음운론적, 형태론적 분석을 통한 처리를 해야한다.

예를 들어 “학교”라는 단어는 형태소 안에서 음운 변이가 일어나서 /하교/가 되고, “책과”에서는 “책”과 “과”사이에서 음운변이가 일어나 /채과/가 된다. “여섯 개”는 띄어쓰기 단위인 어절사이에서 음운변이가 일어나 /여섯개/로 된다.

표준 발음법[3] 27항 ‘ㄴ’소리 첨가규칙을 보면 합성어 및 파생어에서, 앞 단어나 접두사의 끝이 자음이고 뒤 단어와 접미사의 첫 음절이 “이, 야, 여, 요, 유”일 경우에는 ㄴ소리 첨가를 첨가하여 “니, 나, 너, 뇨, 뉴”로 발음한다고 나온다. 예를 들어 “홀이불”은 /혼니불/로, “막일”은 /막닐/로, “맨입”은 /맨닙/으로 변환된다. 그리고 ㄹ받침 뒤에 첨가되는 ㄴ소리는 /르/로 발음한다. 예를 들면, “술잎”은 /술립/으로 “실익다”는 /실릭다/로 발음된다. 마지막으로, 두 단어를 이어서 한 마디로 발음할 경우에도 이에 준한다. 예를 들면, “한 일, 서른 여섯, 짓이기가다”는 /한 닐, 서른 녀섯, 짓 니기다/로 된다. 하지만 예외적으로 일어나지 않는 경우도 있는데, “6.25”는 “유기오”, “송별연”은 /송벼련/으로 바뀐다.

그리고 통계학적으로 [표 1]처럼 뒷 낱말이 /ㄱ/로 시작할 경우나 ㄴ-첨가가 훨씬 자연스럽고 한자어에서 첫 음절 기저형이 /ㄴ, ㄹ/로 시작하는 경우에 주로 일어난다고 한다. 또한, 한자들 중에서 역(驛), 염(炎,鹽), 욕(慾,浴), 용(用), 유(油), 육(肉)은 셋째 음절 이하에서 ㄴ-첨가되고 약(藥), 열(熱)은 둘째 음절 이하에서 ㄴ-첨가된다.[4] 하지만 한자와 고유어의 판별은 형태소분석기로 분별 할 수 없기 때문에 마지막 조건을 고려하기엔 어렵다 하겠다.

ㄴ-첨가가 되는 현상	예 제
뒷 낱말이 /ㄱ/인 경우	스물 여덟[스물 려덟], 옷깃을 여민다.[을 기를 려민다], 지난 여름[지난 녀름]
첫 음절이 /ㄴ, ㄹ/인 경우	신여성(新女性)[신녀성], 열역학(熱力學)[열력학], 신혼여행(新婚旅行)[신혼녀행]

표 1. 대표적인 ㄴ-첨가 규칙

본 논문에서는 위와 같은 음운적 조건과 형태론적 조건을 따지면서, 형태소 사이, 어절 사이의 음운 현상을 처리하고, ㄴ-소리의 효과적인 변환을 위해 통계학적인 접근 시도로 ㄴ-소리 첨가 전 처리기를 구현하고 실험 평가하겠다.

**4. 문자열-발음열 변환 기법**

**4.1 문자열-발음열 변환기의 구조**

실제 이 시스템의 문자열-발음열 변환기의 구조는 [그림 1]과 같다. 한국어의 음운 변이는 주로 품사에 의해 많은 영향을 받으므로 형태소 분석기를 이용하여 형태소 분석을 선행하고 형태소 분석기를 통해 품사 정보를 얻은 실제 문자들 중 특수 문자나 숫자 등은 정규화 과정을 통해 실제 발음열로 변환이 된다.

정규화까지 된 문자열들은 ㄴ-첨가 전 처리기를 거친 후 발음열 변환을 선택적으로 처리하게 되는데, 먼저 예외적인 단어와 구들을 포함한 사전들을 통하여 그 에 해당하는 문자열들은 예외처리를 시키고, 실제 발음열로 변환시키기 위해 자소분해사전을 이용하여 자소들로 분해한다. 그 다음 분해된 자소들은 발음자소변환사전을 참고하여 음운, 형태론적 변화를 통하여 발음자소들로 변환된다. 마지막으로 변환된 발음자소들을 다시 음절로 합성하여 최종 발음열을 얻는다.

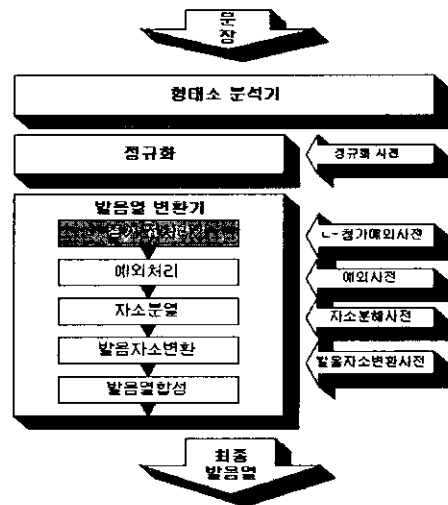


그림 1. 문자열-발음열 전체 구조도

4.2 L-첨가 전 처리기

[그림 2]는 L-첨가 전 처리기를 나타내고자 한다.

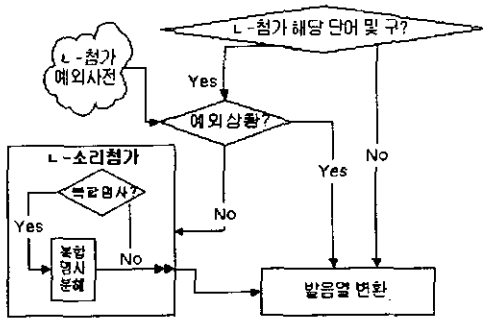


그림 2. L-첨가 전 처리기 Flowchart

합성어 및 파생어에서, 앞 단어나 접두사의 끝이 자음이고 뒤 단어와 접미사의 첫 음절이 “이, 야, 여, 요, 유” 일 경우에는 L-첨가를 하는데 L-첨가가 일어나지 않는 예외적인 단어나 구들은 L-첨가 예외사전을 통해 제외하고, 해당 단어와 구들은 또 다시 복합명사인지의 유무를 판단한다. 복합명사인 경우는 복합명사 분해 알고리즘을 통해 해결하고 복합명사가 아닌 동사나 구일 경우도 또한 L-첨가를 적용한다.

4.3 L-첨가를 위한 복합명사 분해 알고리즘

기존 명사사전에서 “이, 야, 여, 요, 유”가 있는 단어 중에 실제 L-첨가가 일어날 단어를 추출한다. 이 추출된 단어를 제외한 명사사전이 실제 L-첨가를 위한 복합명사 사전이 되고, 이 사전을 통하여 최종적으로 복합명사의 역방향 분해 알고리즘을 적용하여 복합명사의 L-첨가의 적용 여부를 알 수 있다. 실제 복합명사의 적용 사례는 무한하다 할 수 있으므로, 이러한 복합명사 분해 알고리즘을 통한 분해는 꼭 필요할 수 있다 하겠다.[6]

5. 실험 및 고찰

위 논문의 문자열-발음열 변환기는 특수문자와 숫자를 발음열로 바꾸기 위해 정규화 사전(normalization.h)을 이용하고 한국어 표준 발음법을 기반으로 문자열-발음열 테이블(jaso.h)로 구성된다. 그리고 한 음절을 초성, 중성, 종성으로 바꾸기 위해 자소분해사전(ksc5601.h)을 포함하고 예외 상황을 처리하기 위해 L-첨가 예외사전(ni.h)과 예외사전(exception.h)을 구축하였다.

기존 명사사전(56210개)에서 L-첨가 해당 단어는 946개였다. L-첨가의 효과적인 해결을 위해 L-첨가용 명사를 제외한 사전을 구축하였는데, 이를 L-첨가용 명사사전이라 하고, 이 사전을 통하여 모든 “이, 야, 여, 요, 유”에 해당하는 복합명사들을 L-첨가 유무를 해결하였

다. 그리고, 복합명사가 아닌 동사 및 구일 경우도 L-첨가를 적용할 수 있어 더욱 자연스러운 발음열을 얻어 낼 수 있었다.

6. 결론 및 향후연구

본 논문에서는 음운, 형태론적 정보를 이용하고 L-첨가 전 처리기를 이용한 자동 문자열-발음열 변환 시스템을 제안했다. 그리고 뒷 낱말이 /ㄱ/로 시작할 경우에는 L-첨가가 훨씬 자연스럽게 한자어에서 첫 음절 기저형이 /ㄴ, ㄹ/로 시작하는 경우에 주로 일어난다는 원리로 통계적 기법을 시도하였다.

향후 연구로는 충분한 양의 예외 상황과 실험 데이터를 통한 성능평가를 통해 더 나은 통계적 접근과 예외처리를 해야 할 것이고 영문과 구문분석을 수행하여 문자열-발음열 변환기로 개선한다면 뛰어난 문자열-발음열 변환 시스템이 될 것이고 더 나아가 한국어 TTS시스템의 성능 향상에 기여할 것이다.

참고 문헌

- [1] 김병창, 이원일, 이근배, 이종혁 “한국어 TTS를 위한 무제한 단어 자소열-음소열 변환”, HCI'98 학술발표대회 논문집, pp319-323, 1998, 2.
- [2] 전재문, 차선화, 정민화 “형태음운론적 분석에 기반한 한국어 발음 생성”
- [3] ByengChangKim, GenuBaeLee, JongHyeokLee, “Hybrid Grapheme to Phoneme Conversion for Unlimited Vocabulary”, submitted to ACM transactions on asian language information processing, 1999, 7.
- [4] 교육부, 국어 어문 규정집, 대한교과서주식회사, 1994.
- [5] 이호영, 국어 음성학, 태학사, 1996
- [6] 이현민, 박혁로 “복합명사의 역방향 분해 알고리즘”, 한글 및 한국어 정보처리 학술대회 논문집, pp56-59, 2000, 10.