2001년 한국멀티미디어학회 춘계학술발표논문집

한국어 음성 합성을 위한 음가 변환 테이블 생성

이계영*, 임재걸*, 이태경**
동국대학교 컴퓨터학과*

Formation of A Phonetic-Value Look-up Table for Korean Voice Synthesis

Gyeyoung Lee*, Jaegeol Yim *, Taekyoung Lee**
Dept. of Computer Science, Dongguk University*

요 약

문법적으로 정확한 한국어 음성을 합성하려면 표준어 규정의 '표준 발음법'을 준용해야 한다. 따라서 한국어 음가 합성 시스템에 사용되는 한글을 음성으로 변환하여 주는 규칙은 '표준 발음법'을 완전하게 반영하며 또한 무결해야 한다. 기존의 연구에서는 표준 발음법을 검증없이 적용하여 왔고, 표준 발음법 자체에 모순이 있는가의 여부에 대해서도 체계적인 분석을 위한 시도가 전무하였다. 본 논문에서는 한국어 음가 생성의 기본 규칙으로 준용할 표준 발음법을 페트리넷으로 모델링하여 표준 발음법의 일관성을 검증하였다. 그리고, 음운 변동 현상을 여러 단계로 나누어 차례로 적용한다든지, 변동된 단어에 대하여 처음부터 다시 변환 작업을 재수행하는 기존의 음가 생성 방법의 문제점을 해결하기 위하여 한번의 테이블 탐색으로 모든 음운 변동이 완료되는 한국어 음성 합성을 위한 음가 변환 테이블을 구현하였다.

1. 서론

표기 언어인 한글을 한국어 음성으로 변환하려면 "표준어 규정(1988년, 문교부 고시 제 88-2호)" 중, 제 2부의 "표준 발음법[1]을 준용해야 한다. 따라서, 한국어 음가를 생성하는 컴퓨터 시스템에 사용되는 발음법은 '표준 발음법'을 완전히 표현해야 하며 또한 무결해야 한다. 본 논문에서는 표준 발음법의 모든 항을 페트리넷[2-3]으로 모델링한 다음 이들을 통합함으로써 표준 발음법을 완전히 표현하는 한 개의 커다란 페트리넷을 생성하고, 이 페트리넷의 일관성을 분석한다.

그리고, 일관성 분석의 결과로 나타나는 중복항에 대해서는 두 개의 항 중 하나를 삭제하고, 선택항은 원칙을 취하고 허용항을 삭제하며, 상충항은 적용의 우선 순위를 부여하거나 잘못된 정의를 삭제하여 한국어 음가 생성을 위한 표준 발음법을 검증한다.

이와 같이 표준 발음법의 일관성이 검증되어 무결한 표준 발음법이 정의된 후에는 이를 바탕으로 음가 변환 테이블을 생성하게 되며, 이 테이블은 음가 생성을 위하여 입력된 한글 단어나 문장의 자모와 형태소 정보만으로 한국어 표준 발음을 쉽게 추출할 수 있게 된다. 이러한 방법은 표기 언어인 한글을 음성 언어인 한국어 음가로 변환할 때 발생할 수 있는 개인간의 해석차를 없애고, 형태소 정보와 자모만으로 정확한 음가 변환이 이루어질 수 있도록 도와 준다.

한국어 음성 합성 시스템의 음가 변환 과정에서 검증되지 않은 임의의 순서로 한 단계씩 음가 변환을 시켜나가는 기존의 연구로는 음가 생성의 무결성을 입증할 수 없으며, 임의의 다단계로 처리하는 과정상에서도 비효율성이 발생할 수 있다. 따라서, 검증된

표준 발음법을 적용하여 구현되는 본 논문의 한국어 음가 변환 테이블은 1회의 테이블 검색으로 모든 한국어 음가 변환이 완료되므로, 검증되지 않은 임의의 순서로 한 단계씩 차례로 처리해 나갈때 발생할 수 있는 음가 변환의 오류와 변환 시간의 문제 등을 극복할 수 있다.

그리고, 본 연구에서 생성하는 한국어 음가 변환 테이블은 음가 변환기에서 표기 언어를 음성 언어로 바꾸어 주는 규칙(음운변동 규칙)에 해당하는 것으로 입력된 한글의 형태소 분석 결과와 구문 분석 결과에 따라서 동일한 문자 표기일지라도 다른 음가로 변환시킬 수 있게 된다.

2. 표준 발음법의 페트리넷 모델링

2.1 모델링 표기의 정의

(1) 입력 플레이스에 나타나는 표기

IC : 입력되는 초성　　IJ : 입력되는 종성
ICV : 입력되는 초성과 중성
IM : 입력되는 문법 형태소
IMJ : 입력되는 문법 형태소의 종성
IMCV : 입력되는 문법 형태소의 초성과 중성
-1 : 입력이 enable됨을 나타내는 표기

(2) 출력 플레이스에 나타나는 표기

OC : 출력되는 초성　　OJ : 출력되는 종성
OJN : 출력되는 종성을 발음 안함
+1 : 출력이 enable됨을 나타내는 표기

2.2 표준 발음법의 페트리넷 모델링

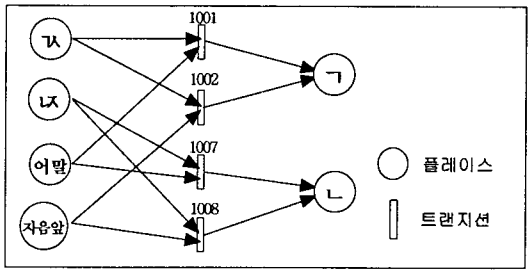
<표 1>의 표준 발음법 제10항은 겹받침에 대한 일반항과 '밟-'과 '넓-'에 대한 특수항으로 구분된다. 일반항은 'ㄻ'이 어말에 사용되면 'ㄱ'으로, 'ㄻ'이 자음 앞에서 'ㄱ'으로, 'ㄵ'이 어말에서 'ㄴ'으로 'ㄵ'이 자음 앞에서 'ㄴ'이 됨을 의미한다. 이것을 표현하는 페트리넷은 25개(겹받침 6, 어말 1, 자음 18)의 입력 플레이스와 4개의 출력 플레이스, 120개의 트랜지션으로 구성된다.

특수항의 예로 「'밟-'은 자음 앞에서 [밥-]으로, '넓-'은 '넓죽하다'와 '넓둥글다'의 경우에 각각 [넙쭈카다]와 [넙뚱글다]로 발음한다」라는 조항이 있다. 따라서, 일반항의 규정에 따르면 '밟-'은 [발-]로 발음될 듯 하지만 특수항의 규정에 따라서 [밥-]으로 발음되어야 하며, 마찬가지로 '넓-'도 특수항의 조건에 맞으면 [넙]으로 발음하여야 한다. 이 같은 예외 사항은 예외 사전으로 구축하여 별도로 처리하며, 나머지 항들도 정의된 용어와 용어의 내용에 따라 각각의 근접 행렬 표현으로 구성한다.

<표 1> 표준 발음법 제10항

제10항 겹받침 'ㄳ', 'ㄵ', 'ㄻ, ㄺ, ㄼ', 'ㅄ'은 어말 또는 자음 앞에서 각각 [ㄱ, ㄴ, ㄹ, ㅂ]으로 발음한다. 넋[넉]　　넋과[넉꽈]　　앉다[안따]　　여덟[여덜] 넓다[널따]　외곬[외골]　　훑다[훌따]　　값[갑] 없다[업:따] 다만, '밟-'은 자음 앞에서 [밥]으로 발음하고, '넓-'은 다음과 같은 경우에 [넙]으로 발음한다. (1) 밟다[밥:따]　　밟소[밥:쏘]　밟지[밥:찌] 밟는[밥:는 → 밤:는]　밟게[밥:께]　밟고[밥:꼬] (2) 넓-죽하다[넙쭈카다]　넓-둥글다[넙뚱글다]

(그림 1)은 페트리넷 정의와 격발의 정의[4-6]에 따라 10항의 규칙을 각각의 소항목으로 나누어 항목별로 모델링한 것이다. 10항 내의 각 규칙(소항목)을 페트리넷의 트랜지션과 사상시킬때, 조건부에 사용된 요인은 해당 트랜지션의 플레이스로 사용하였고, 출력부에 사용된 요인은 해당 트랜지션의 출력 플레이스로 모델링하였다. (그림 1)은 10항의 트랜지션 중에서 1001, 1002와 1007, 1008만을 통합된 페트리넷으로 나타낸 것이다.



(그림 1) 10항의 소항목 1001, 1002, 1007, 1008이 통합된 페트리넷

(그림 1)의 플레이스는 초기 마킹 Mo가 놓여지지 않은 상태이며, 플레이스에 놓여지는 초기 마킹은 입력되는 문자열의 형태소 분석 정보와 자모 분리 결과에 따라 동적으로 놓여지게 된다. 예를 들어, 입력 문자열로 "넋"이라는 글자가 입력되면 형태소 분석과 자모 분리 결과로 '어말'과 'ㄻ'이 생성되고 이들에 해당되는 플레이스에 토큰이 놓여진다. 그러면 트랜지션 1001이 활성화(enable)되며, 트랜지션 1001이 격발하게 되면 출력 플레이스 'ㄱ'에 토큰이 놓여지게 된다.

2.3 페트리넷 모델의 근접 행렬 변환

(그림 1)과 같이 작성된 페트리넷 모델을 통합과 분석을 용이하게 하기 위하여 근접 행렬(incidence matrix)로 표현하였다. 근접 행렬 C는 |P| x |T| 행렬이며, C의 일반항은 다음과 같이 정의한다[2,3].

c_{ij} = 1 if $(t_j, p_i) \in F$, -1 if $(p_i, t_j) \in F$, 0 otherwise

<표 2>는 (그림 1)의 페트리넷을 근접 행렬로 변환한 것으로 이해를 돕기 위하여 10항을 간소화하여 나타내었다. 즉, 실제로는 <표 4>의 입력 플레이스 중 '자음앞'은 18개의 자음으로 확장되어 모델링된다.

<표 2> 표준 발음법 10항의 페트리넷 근접 행렬

R10 ①	1001 ②	1002	1003	1004	1005	1006	1007	1008	1009	1010	1011	1012
ㄱ ③	-1⑦						-1 ⑦					
ㄴ ③		-1						-1				
ㄹ ③			-1						-1			
ㅅ ③				-1						-1		
ㅊ ③					-1						-1	
ㅂ ③						-1						-1
⑤												
어말 ④	-1⑦	-1	-1	-1	-1	-1						
자음앞④							-1 ⑦	-1	-1	-1	-1	-1
⑥												
ㄱ ⑤	+1⑧						+1 ⑧					
ㄴ ⑤		+1						+1				
ㄹ ⑤			+1	+1	+1				+1	+1	+1	
ㅂ ⑤						+1						+1

<표 2>의 근접 행렬에서 ①~⑧은 테이블을 설명하기 위한 주석 표시이며 각 주석이 표시하는 내용은 다음과 같다.

① 표준 발음법의 항 번호
② 항의 각 소항목을 나타내는 번호(트랜지션)
③ 장전 조건 1(입력 플레이스1)
④ 장전 조건 2(입력 플레이스2)
⑤ 격발 후 출력(출력 플레이스)
⑥ 각 조건을 구분하기 위한 공백
⑦ 해당 소항목의 조건부
⑧ 해당 소항목의 결론부

표준 발음법의 모든 항에 대하여 <표 2>와 같이 완성된 각각의 근접 행렬은 페트리넷의 통합과 일관성 검증 프로그램의 입력 자료로 사용하기 위하여 완성형 텍스트 형태로 저장된다.

2.4 표준 발음법의 일관성 검증

표준 발음법 일관성 검사 모듈은 작성된 각각의 근접 행렬 테이블 파일을 읽어들여 하나의 테이블로 통합하고, 통합된 결과를 트랜지션 단위로 비교하면서 일관성을 검사하는 기능을 수행한다. 이 시스템은 기능상으로 테이블 통합 모듈과 일관성 검사 모듈로 구성되어 있다.

테이블 통합 모듈은 각각의 파일로 저장되어 있는 각 항들의 파일을 읽어 들여 각 플레이스를 비교하고, 새로운 플레이스의 추가와 기존의 플레이스에 새로운 '-1' 값 또는 '+1' 값을 추가하는 역할을 하게 된다. 일관성 검사 모듈은 통합된 테이블에서 트랜지션 단위로 비교하면서 중복항과 상충항을 검색하는 기능을 수행한다.

3. 구현 및 실험

3.1 일관성 검사 모듈

(그림 2)는 통합된 테이블을 대상으로 중복 또는 상충이 발생하는 트랜지션을 검사하는 알고리즘이다. Step 1에서는 통합된 근접 테이블을 구조체 배열로 읽어 들인다.

```
Input : 표준 발음법 근접 행렬 파일
Output : 통합된 표준 발음법의 근접 행렬 파일

FILE x : 각 테이블의 트랜지션을 통합한 후 저장할 배열
FILE n : 통합 대상 테이블의 파일명
m : 통합 대상 테이블을 읽어 저장할 배열명
int i=0, j=0, k=0

BEGIN
Step 1 : While 처리되지 않은 테이블이 남아 있는 동안
Step 2 : 테이블 로딩(화일명n, 저장될 배열명m)
Step 3 : FOR i=0 TO n의 트랜지션 수
           n의 i번째 트랜지션과 트랜지션에 연결된
           플레이스를 읽어 옴
           FOR j=0 TO x의 트랜지션 수
            x의 j번째 트랜지션과 트랜지션에 연결된
            플레이스 load
            IF i번째 트랜지션의 플레이스와
             j번째 트랜지션의 플레이스가 같으면
                  THEN GOTO Step 3
                  ELSE : x에 새로운 플레이스 추가
                         GOTO Step 3
Step 4 : 통합 근접 테이블 기록(트랜지션, 플레이스)
           k=플레이스의 마지막 트랜지션 위치값
           IF 입력 플레이스 THEN k+1 = -1
           IF 출력 플레이스 THEN k+1 = +1
Step 5 : 통합 근접 테이블 출력(화일명x)
END
```

(그림 2) 테이블 통합 알고리즘

Step 2에서는 Table 배열의 첫 번째 트랜지션부터

하나의 트랜지션을 읽어 온 후, 연결된 입력 플레이스와 출력 플레이스를 읽어 들인다.

Step 3에서는 두 번째 트랜지션부터 하나의 트랜지션과 연결된 입력 플레이스와 출력 플레이스를 선택하며, Step 4에서는 Step 2와 3에서 읽어 들인 트랜지션을 비교하여 입력 플레이스 1과 입력 플레이스 2가 같고 출력 플레이스가 틀리는 경우(상충)에 대한 정의를 하며, 입력 플레이스 1과 입력 플레이스 2, 출력 플레이스가 모두 같으면 중복된 정의로 판별한다. Step 5에서는 검증 결과를 출력한다.

3.2 재처리 모듈

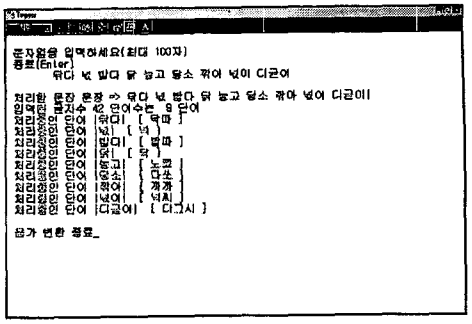
(그림 3)은 압축된 테이블을 대상으로 하여 출력된 음운 변동의 결과값이, 다른 음운 변동의 원인이되는 압축된 Tab[39][55]배열의 첫 번째 초성과 첫 번째 종성 위치의 내용을 읽어와서 c'에는 초성의 인덱스 번호를 저장하고, j'에는 종성의 인덱스 번호를 저장한다. Step 2에서는 Tab[c'][j']와 Tab[c][j]의 내용을 비교하여 내용이 틀릴때에 "변경 요청"이라고 출력한다. Step 3에서는 더이상의 "변경 요청"이 발생하지 않을 때까지 반복하여 검사하며, Step 4에서 검색된 재처리 결과를 출력한다.

```
Input : 압축된 표준 발음법 음가 변환 규칙
Output : 재처리된 음가 변환 규칙
Tab[39][50] : 압축된 테이블
int c,c',j, j'
BEGIN
Step 1 : FOR j=0 To 39(종성의 수)
         FOR c=0 To 55(초성의 수)
           IF Tab[c][j].Cho가  NULL이 아니면
             c' ← Tab[c][j].Cho에 기록된 초성의 인덱스 번호
           IF Tab[c][j].Jong이 NULL이 아니면
             j' ← Tab[c][j].Jong에 기록된 내용의 인덱스 번호
Step 2 :  IF Tab[c'][j].Jong과 Tab[c][j].Cho이 NULL이 아니면
          THEN IF Tab[c][j].Jong 과 Tab[c'][j'].Jong이 다르면
                  THEN print("변경 요청");
               IF Tab[c][j].Cho 와 Tab[c'][j'].Cho가 다르면
                  THEN print("변경 요청");
Step 3 : 변화된 초성과 종성이 있으면 Goto Step 1
Step 4 : 재처리 검색 결과 출력
END
```

(그림 3) 압축된 음가 변환 테이블의 재처리 알고리즘

4. 실험 및 결과

(그림 4)는 음가 변환기의 출력 화면을 보인 것이다. 음가 변환기는 입력되는 문자열을 각 단어 별로 분리하여 단어별 음가 변환 결과를 출력한다.



(그림 4) 음가 변환기의 출력 결과

5. 결론 및 향후 과제

본 논문에서는 한국어 음성 합성을 위한 정확한 한국어 음가 변환과 음성 인식 분야의 후처리 기본 규칙으로 사용할 수 있는 '표준 발음법'을 페트리넷으로 모델링할 수 있음을 보였다. 그 결과로서 표준 발음법 상의 각 규칙간에는 선택, 중복, 상충의 문제가 있음이 밝혀졌으며, 이를 배제한 한국어 음가 변환기를 구현하고 무결한 음가변환 테이블을 제정 하였다.

모델링된 페트리넷을 대상으로 일관성 검사를 한 결과, 모두 39개의 상충항과 1개의 중복항을 찾아내었으며, 21개의 재처리 항목이 검출되었다. 따라서, 본 논문에서는 검증 결과를 바탕으로 하여 중복되거나 상충되는 항이 없는 무결한 표준 발음법으로의 수정 방향을 제시하였고, 정형화된 음운 변동 규칙을 이용하여 최종 테이블을 작성하였으므로 본 연구에서 작성된 테이블은 무결성을 유지한다고 할 수 있다.

참 고 문 헌

[1] 문화교육부, "표준어 규정", 문교부 고시 제 88-2호, 1988.

[2] T.Murata, "Pertri nets: properties, analysis and applications," Proceeding of the IEEE, Vol77. no.4, pp.541-580, April 1989.

[3] 이계영, 임재걸, "한글 받침 발음법의 페트리 네트 표현", 동국논집 14집, pp. 155-167, 1995.

[4] 임재걸, 이계영, 김경징, "페트리넷을 이용한 표준 발음법 분석 시스템 디자인", 한국정보과학회 봄학술발표논문집, pp. 369-371, 1999.

[5] 이계영, 임재걸, 김경징, 김규식, "페트리넷을 이용한 표준 발음법 분석 시스템 구현", 한국정보처리학회 춘계학술발표논문집, pp. 609-612, 1999.

[6] 이계영, 임재걸, 김경징 "표준 발음법 페트리넷을 이용한 음가 변환기 설계",한국멀티미디어학회 춘계학술발표논문집 제2권1호, pp. 339-344, 1999.