

한국어 발음 변환기(G2P)의 현황과 성능 향상에 대한 언어학적 제안*

문성민^{**}, 김수한^{***} 고언숙[†]
아주대학교^{**}, 조선대학교^{***}, 조선대학교[†]

Seongmin Mun, Su-Han Kim, Eon-Suk Ko. 2022. A proposal to improve on existing Grapheme-to-Phoneme conversion models informed by linguistics. *Language and Information* 26.2, 27-46. Grapheme-To-Phoneme (G2P) converts an input letter sequence into its corresponding pronunciation. Several rule-based G2P models are available for Korean, but their performance in view of individual phonological rules in Korean is not well-investigated. We establish phonological rules to be reflected in a G2P model for Korean, and test the model performance of two most popular G2P models of Korean, i.e., g2pk and KoG2P, regarding their performance on each of these rules. We created a golden corpus to evaluate the performance of the current G2P models based on manual phonetic transcription. We then measured the performance of two G2P models, and identified the phonological rules in which the models show relative success or failure in deriving the correct output. We implemented additional phonological rules such as h-deletion, ui-variation, consonant place assimilation, and restructured the ordering of rules such as h/th-neutralization in the model. We show that our revised model makes a substantial improvement on model performance. Further, we argue that the major limitation of the current rule-based approaches to G2P is in its binary approach to phonological rules and lack of information about prosodic boundaries. We propose that a rule-based G2P system should reflect the stochastic nature of phonological processes informed by existing research on the gradient nature of phonological rule application as a function of factors such as lexical frequency and speech register.

Key words: Grapheme-to-Phoneme, rule-based, NLP, phonological variation

* 이 논문은 2022년 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(No. 2019-0-01371, 뇌·인지 발달과정의 기초-영아단계 모사형 실세계 상호작용 경험 기반 객체 관련 개념의 기계학습 기술 개발)과 대한민국 교육부와 한국연구재단의 인문사회분야 중견연구자지원사업의 지원(NRF-2021S1A5A2A01073515)을 받아 수행한 연구임.

** 제1저자, (16499) 경기도 수원시 영통구 월드컵로 206, 아주대학교 인문과학연구소, 연구조교수, E-mail: stat34@ajou.ac.kr

*** 공동저자, (61452) 광주광역시 동구 필문대로 309, 조선대학교 영어영문학과, 석사과정, E-mail: suhan3804@gmail.com

† 교신저자, (61452) 광주광역시 동구 필문대로 309, 조선대학교 영어영문학과, 부교수, E-mail: eonsukko@chosui.ac.kr

1. 서론

한국어 문자로 작성된 문자열에 해당하는 기저형을 발음열에 해당하는 표면형으로 변환하는 기술인 G2P(Grapheme-To-Phoneme)는 기계가 인간의 발화와 유사한 발음열을 생성하는데 있어 중요한 기술이기 때문에 전산 언어학 혹은 자연어 처리 연구 분야에서 활발히 연구되어 오고 있다(Divay and Vitale, 1997). 일반적으로 영어를 포함한 외국어의 경우 문자열은 발음열과 일대일로 대응하지 않지만 한국어의 경우 표음 문자를 사용하기 때문에 대부분의 문자열과 발음열을 일대일로 대응하여 표준 발음 산출을 할 수 있다. 예를 들어, 한국어 단어 ‘가방’의 경우 문자열과 발음열에 대한 일대일 연결이 가능하기 때문에 문자열과 일치하는 ‘가방’으로 발음열이 산출된다. 하지만 문자열과 발음열의 산출이 항상 일대일로 대응하는 것은 아니다. 예를 들어, 2009년 NH농협에서 기업의 이미지 광고로 이용한 광고 문구 ‘같이의 가치’의 경우를 생각해 보자. 광고 문구에서 사용된 단어 ‘같이’와 ‘가치’라는 두 단어의 경우 서로 전혀 다른 의미를 가지고 있지만 실제 문자열을 발음으로 산출하면 두 단어 모두 ‘가치’로 발음이 산출 되는 것을 알 수 있다. 두번째 단어 ‘가치’의 경우 문자열이 발음열과 일치하는 데에 반해, 첫번째 단어 ‘같이’의 경우 첫음절의 종성 ‘ㄷ’이 조사 ‘이’ 앞에서 초성으로 재음절화 되는 동시에 ‘ㅈ’으로 변환되는 구개음화(palatalization) 규칙이 적용되어 [가치]로 발음되기 때문이다.¹⁾ 따라서 한국어의 문자열을 발음열로 변환하는 기술은 단순히 문자열의 음소 정보를 발음열의 발음 정보로 연결하는 것으로 구현 될 수 없으며, 형태론, 음운론, 그리고 통사론에 대한 언어학적 지식과 이를 전산화 할 수 있는 기술의 융합으로 구현 될 수 있다.

한국어 발음 변환기를 개발하는 방법은 크게 문자열의 단어에 대한 발음열 정보를 기록하고 있는 사전을 사용하여 일대일로 연결하는 방법과 음운론적 규칙을 전산화하여 문자열을 발음열로 변환하는 방법으로 나누어 볼 수 있다(Dutoit, 1997; Kim et al., 2002). 우선, 사전을 사용하는 개발 방법은 최대한 많은 문자열 단어들의 발음열 정보를 저장하고 있어야 하기 때문에 크기가 매우 큰 사전을 사용하는 것이 특징이며 신조어처럼 최근에 생성되는 문자열 단어에 대한 발음열 정보와 같이 모든 문자열의 발음열 정보를 저장 할 수 없기 때문에 별도의 사전까지 병합하여 사용된다. 뿐만 아니라 사전의 크기가 너무 커져 변환기가 문자열을 발음열로 산출 할 때 소모되는 시간이 크기 때문에 이를 감소시키기 위해 문자열의 단어가 아닌 형태소만을 사전에 저장하고 형태학적 규칙을 적용하여 굴절어, 파생어, 복합어에 대한 발음을 예측하는 방식으로 개발이 이루어진다. 하지만 이러한 노력에도 불구하고 사전을 사용하는 방법은 변환해야 하는 단어의 수가 늘어 날 때 변환 작업에 소요되는 시간이 기하급수적으로 증가한다는 단점을 가지고 있다. 다음으로, 음운론적 규칙을 사용하는 개발 방법은 문자열의 단어가 발음열로 변화하는 음운론 규칙과 더불어 수의적인 발음으로 인해 음운론 규칙 적용의 결과를 예측하기 어려운 ‘ㄴ첨가’와 같은 규칙의 적용을 받는 변환 단어들에 대한 작은 크기의 사전을 사용하는 방법이다. 이 방법은 언어학 분야의 사전 연구 결과를 기반으로 음운론적인 규칙을 컴퓨터가 인식할 수 있도록 전산화하여 적용하고 규칙 적용이 불가능한 일부의 단어에 대한 사전만을 사용하기 때

1) 국립국어원에서 규정하고 있는 한국어 표준 발음법 제 17항에서는 구개음화(palatalization)에 대해 “받침 ‘ㄷ, ㅌ(ㅌ)’이 조사나 접미사의 모음 ‘ㅣ’와 결합되는 경우에는, [ㅈ, ㅊ]으로 바꾸어서 뒤 음절 첫소리로 옮겨 발음한다.”라고 규정하고 있다. 구개음화와 재음절화 사이의 적용은 규칙 적용의 순서와 무관하다.

문에 사전을 사용하여 발음을 변환하는 방법에 비해 발음 변환 과정에 소모되는 시간이 적다는 것이 특징이다.

최근에는 인공지능 기술이 발전함과 동시에 발음 변환(G2P) 기술에 대한 관심이 더 높아지고 있으며, 국내 기업에서는 네이버의 ‘클로바(CLOVA)’, KT의 ‘기가지니(GiGA Genie)’, SKT의 ‘누구(NUGU)’ 등의 인공지능 스피커에서 발음 변환 기술이 사용되고 있다. 인공지능 기술에 발음 변환기를 적용하는 방법은 다양하지만 최근에는 구글에서 개발된 ‘텐서플로우(TensorFlow)’, 페이스북에서 개발된 ‘파이토치(PyTorch)’, 오픈AI에서 개발된 ‘트랜스포머(Transformer)’와 같이 Python 환경에서 작동하는 AI 프레임워크들이 많이 배포됨에 따라 Python 환경에서 작동하는 발음 변환기가 서비스 및 연구에 많이 사용되고 있다(예: Cho et al., 2020; Hong et al., 2018; Lee and Kim, 2018). Python 환경에서 문자열을 발음열로 변환해주는 발음 변환기는 2021년 카카오에서 개발된 ‘PORORO’, 2021년 서울대학교에서 과학기술정보통신부의 사업으로 개발된 ‘SMART-G2P’, Park (2019)에서 공개한 ‘g2pk’, 그리고 Cho (2017)에서 공개한 ‘KoG2P’가 대표적이다. 이 중에서도 카카오의 ‘PORORO’는 Park (2019)의 ‘g2pk’를 기반으로 작동하고 서울대학교의 ‘SMART-G2P’는 Park (2019)의 ‘g2pk’와 Cho (2017)의 ‘KoG2P’를 기반으로 작동하기 때문에 사실상 Park (2019)의 ‘g2pk’와 Cho (2017)의 ‘KoG2P’가 현재 Python 환경에서 작동하고 많이 사용되고 있는 발음 변환기라고 할 수 있다.

문자열의 단어를 발음열로 변환하는 작업은 복잡한 음운론적 규칙과 규칙으로 해소되지 않는 불규칙 변환에 대한 사전 정보를 활용하는 작업으로서 언어학적 지식, 특히 형태론과 음운론에 대한 지식을 필요로 한다. 그런데 앞서 언급한 두 개의 발음 변환기가 현재 인공지능 서비스 및 연구에 사용되고 있는 대표적인 변환기라고 할 수 있음에도 불구하고 두 발음 변환기가 실제 음운론적인 발음 변환 과정을 얼마나 잘 반영하고 있는지에 대한 보고는 미흡한 실정이다(예: Yoon and Brew, 2006). 따라서 본 연구는 언어학 분야에서 발표된 사전 연구들을 기반으로 발음 변환 과정에서 반영해야 하는 음운론적인 규칙들을 대해 정리하고 정리된 음운론적 규칙들이 두 발음 변환기에 얼마나 잘 적용되어 있는지에 대한 현황을 탐색하고자 한다. 또한 기존의 현황을 보완하고자 새로운 규칙 반영을 제안하고자 한다.

연구의 목적을 위해 본 연구는 다음의 과정으로 연구를 진행하였다. (1) 사전 연구들을 기반으로 발음 변환 과정에서 반영해야 하는 음운론적인 규칙들에 대한 정리, (2) 두 명의 언어학 전문가(본고의 제1, 제2저자)들로 검수자 집단을 만들고 정리된 음운론적 규칙에 대한 평가 말뭉치 구축, (3) 구축된 말뭉치를 사용하여 두 발음 변환기의 성능을 측정 및 보고, (4) 두 발음 변환기에 적용되지 않은 음운론적 규칙에 대한 반영 기준을 추가로 제시. 이와 같은 과정을 통해 기존의 두 발음 변환기의 성능을 향상시키기 위한 언어학 기반의 방안을 제시하였으며 본 연구에 활용된 발음 변환기를 다음 사이트에 제공하였다(<https://github.com/seongmin-mun/KoG2Padvanced>).

2. 한국어의 음운론적 발음 변환 규칙

한국어의 발음 변환 과정에서 적용되는 음운론적 규칙들을 정리하기 위해 본 연구는 이전에 언어학 분야에서 발표된 사전 연구들과 국립국어원의 ‘표준 발음법’을 참고하였다. 한국어의 발음 변환 과정에 적용되어야 하는 음운론적 규칙들은 다음과 같이 요약할 수 있다.

2.1. 자음 중화(Coda neutralization)

한국어 음절 구조는 표면형에서 음절말에는 평 파열음과 비음, 유음(예: ㄱ, ㄷ, ㅂ, ㄴ, ㄹ, ㅁ, ㅇ)만 실현된다는 제약이 있다.²⁾ 자음 중화는 음절말에서 실현될 수 없는 자음이 기저형에 나타나고 후행하는 분절음이 자음일 때, 해당 자음의 조음 위치에 따라 불파음으로 교체되는 현상이다.

- (1) /ㄱ, ㄴ/ → [ㄱ] / ____ {C, #}: 동넉[동넉], 박[박]
- (2) /ㅈ, ㅊ, ㅍ, ㅌ/ → [ㄷ] / ____ {C, #}: 넷넉[넷넉], 있대을때, 넷넉
- (3) /ㅍ/ → [ㅂ] / ____ {C, #}: 쫌넉[쫌넉], 쫌넉[쫌넉]

자음 중화는 대다수의 음운 규칙들을 급여(feeding)하므로(예: 옷만[옴만→온만], 깎대[깎다→깎따] 등) 발음 변환기를 개발 할 때는 음절말의 자음들을 7개로 줄여주는 작업이 우선되어야 한다. 단, 이 과정에서 ‘ㅎ’, ‘ㅌ’에 대한 중화는 연관된 다른 음운 규칙보다 나중에 진행된다. 그 이유는 우선 ‘ㅎ’에 대한 규칙 중 뒤에서 언급할 유기음화의 경우, 규칙이 실현되는 일부 조건이 자음 중화를 출혈(bleeding)하고(예: 낱고[나코], 놓대[노타] 등) ‘ㅌ’에 대한 규칙 중 구개음화의 경우, ‘ㅌ’이 기저형에 실현되어 있는 상태에서 구개음화가 일어나기 때문에 두 자음의 중화는 해당 규칙을 적용시킨 후 중화가 적용된다.

2.2. 자음군 단순화(Coda cluster simplification)

앞서 언급했던 자음 중화와 같은 이유로 음절말에 자음군이 오는 경우, 표면형으로 변환할 때 두 자음 중 하나의 자음만 실현되는 단순화 과정을 진행한다.³⁾ 자음군 단순화는 크게 앞 자음이 실현되

2) 국립국어원에서 규정하고 있는 한국어 표준 발음법 제8항과 제9항에서는 자음 중화에 대해 아래와 같이 정의하고 있다.

제8항: 받침소리로는 ‘ㄱ, ㄴ, ㄷ, ㄹ, ㅁ, ㅂ, ㅇ’의 7개 자음만 발음한다.

제9항: 받침 ‘ㄱ, ㄴ, ㄷ, ㄹ, ㅁ, ㅂ, ㅇ’은 어말 또는 자음 앞에서 각각 대표음 [ㄱ, ㄷ, ㅂ]으로 발음한다.

3) 국립국어원에서 규정하고 있는 한국어 표준 발음법 제10항, 제11항, 제14항에서는 자음군 단순화에 대해 아래와 같이 정의하고 있다.

제10항: 겹받침 ‘ㄴㄷ, ㄴㄹ, ㄹㄷ, ㄹㄹ, ㄹㅁ, ㅁㄷ, ㅁㄹ, ㅁㅁ’은 어말 또는 자음 앞에서 각각 [ㄱ, ㄴ, ㄷ, ㅂ]으로 발음한다.

제11항: 겹받침 ‘ㄹㄹ, ㄹㅁ, ㅁㅁ’은 어말 또는 자음 앞에서 각각 [ㄱ, ㄷ, ㅂ]으로 발음한다.

제14항: 겹받침이 모음으로 시작된 조사나 어미, 접미사와 결합되는 경우에는, 뒤엣것만을 뒤 음절 첫소리로 옮겨

는 경우, 뒷 자음이 실현되는 경우, 재음절화로 인해 음절말 자음군의 뒷 자음이 다음 음절의 초성으로 재음절화가 이루어지는 경우로 분류할 수 있다. 단, 표준 발음법 제14항에서 ‘ㅅ’이 들어간 자음군 뒤에 모음이 오며 연음이 실현되는 경우에 ‘ㅅ’이 뒤 음절의 어두가 될 적에는 ‘ㅅ’이 경음으로 실현된다. 자음군 단순화의 예시는 아래와 같다.

(1) 앞 자음이 실현되는 경우

/ㄱ/ → [기] / ____ {C, #}: 몯[목], 녀[녀], 녀과[녀과]

/ㄴ/ → [니] / ____ {C, #}: 앓는[안는]

/ㄹ, ㄴ, ㄹ/ → [리] / ____ {C, #}: 늙다[널따], 외곬[외골], 핏대[할따]

/ㅁ/ → [비] / ____ {C, #}: 껌[갑], 없대[업따]

(2) 뒷 자음이 실현되는 경우

/ㄹ/ → [기] / ____ {C, #}: 침[칙], 침습[칙습]

/ㅍ/ → [비] / ____ {C, #}: 읊대[읍따]

/ㄹ/ → [미] / ____ {C, #}: 삶[삼], 삶과[삼과]

(3) 재음절화 되는 경우

/ㄱ/ → [기씨] / ____ V: 몯이[목씨]

/ㄴ/ → [니씨] / ____ V: 곱이[골씨]

/ㅁ/ → [비씨] / ____ V: 없이[업씨]

/ㅍ/ → [미포] / ____ V: 읊어[읍퍼]

/ㄹ/ → [리기] / ____ V: 읽어[일거]

/ㄴ/ → [니즈] / ____ V: 앓아[안자]

/ㄹ/ → [리비] / ____ V: 밟아[발바]

/ㄹ/ → [리티] / ____ V: 핏대[할타]

/ㄹ/ → [리미] / ____ V: 삶아[살마]

이때 (1)의 ‘ㄹ’, ‘ㄹ’의 단순화는 경음화가 같이 일어난다. 표준 발음법 제25항(예: 밟고[발꼬] 핏대[할따])은 자음군 뒤에서 실현되는 경음화로, 이러한 실현 조건의 경우, 자음군 단순화 뒤 경음화를 적용시킨 후 자음군 단순화를 진행한다.

2.3. ㅇ탈락(/h/-deletion)

한국어에서 기저형에 모음 또는 공명자음(예: ‘ㄴ’, ‘ㄹ’, ‘ㄹ’, ‘ㅇ’)앞과 모음 사이에 ‘ㅎ’이 실현될
 발음한다.(이 경우, ‘ㅅ’은 된소리로 발음함.)

경우 표면형에서 ‘ㅎ’이 탈락한다. 다만 ‘ㅎ’탈락은 실현되는 환경과 단어의 품사, 빈도에 따라 실현율이 변하는 변이 현상이라는 많은 선행 연구가 있다. 우선 ‘ㅎ’탈락은 용언에서는 필수적으로 탈락하지만 명사에서는 수의적인 현상이며, 고빈도 단어(예: ‘조용하다’, ‘사람한테’ 등)가 저빈도 단어(예: ‘오진하다’, ‘오만한데’ 등)보다 실현율이 높고 형태소 경계가 선행할 때보다 선행하지 않을 때, 모음보다는 공명자음 뒤에서, 비음(예: ‘ㄴ’, ‘ㄹ’, ‘ㅇ’)보다는 유음(예: ‘ㄹ’)뒤에서, 그리고 ‘ㅇ’보다는 ‘ㄹ’ 또는 ‘ㄴ’ 뒤에서 활발하게 일어남을 확인할 수 있었다(예: 결혼[결혼~거론] > 신혼[신혼~시논] = 심화[심화~시파] > 황혼[황혼~황온] > 이혼[이혼~이온])(강옥미, 2003; 노석은, 2003; 엄태수, 2014). ‘ㅎ’탈락도 자음 중화, 자음군 단순화와 마찬가지로 다른 음운 규칙들을 급여하므로(e.g. 똥네[똥네→똥레], 똥는[똥는→똥른] 등) G2P모델에 우선적으로 적용되어야 할 규칙 중 하나이다. (1)과 (2) ‘ㅎ’탈락의 예시이다. 추가적으로 (3)와 같이 ‘ㅎ(ㄸ, ㄹㅎ)’뒤에 ‘ㅅ’이 결합하는 경우 표면형에서 ‘ㅆ’으로 실현되는 축약 규칙도 함께 적용한다.

- (1) / ㅎ / → Ø / {ㄴ, ㄹ} ____ ㄴ: 앞네[안네], 앞는[안는], 똥네[똥네→똥레], 똥는[똥는→똥른]
 (2) / ㅎ / → Ø / {ㄴ, ㄹ, V} ____ V: 옹아[오래], 앞은[아는], 아흠[아흠→아음], 신혼[신혼~시논]
 (3) / ㅅ / → [ㅆ] / {ㅎ, ㄸ, ㄹㅎ} ____: 당소[다:쏘], 많소[만:쏘], 싫소[실쏘]

2.4. 유기음화(Aspiration)

표준 발음법 제12항에서 규정하는 ‘ㅎ’ 축약은 크게 두 가지로, (1)~(4)와 같이 ‘ㅎ’이 다른 장애음과 결합하여 해당 장애음이 유기음으로 축약되는 유기음화와 유기음화 중 ‘ㅅ’이 ‘ㅎ’과 결합하여 실현되는 경우, 품사에 따라 표면형이 다르게 나타나는데 용언 어간과 접미사의 경계에서 실현될 때는 ‘ㅅ’으로 축약되고 나머지는 ‘ㅅ’이 ‘ㄷ’으로 중화된 후 ‘ㅌ’으로 축약된다. ‘ㅎ’축약의 예시는 아래와 같다.

- (1) ㄱ, ㄷ, ㅈ → ㅋ, ㅌ, ㅊ / {ㅎ, ㄸ, ㄹㅎ} ____: 농고[노코], 앞던[안텐], 옹지[올치]
 (2) ㄱ, ㄷ, ㅈ → ㅋ, ㅌ, ㅊ / ____ ㅎ: 각하[가카], 망형[마텃], 젓혀[저쳐]
 (3) ㄷ → ㅌ / ____ ㅎ: 옷 하나[온하나 → 오타나], 낮 한 때[난한때 → 나탄때]
 (4) ㄹ, ㄹㅎ, ㅌ → ㅋ, ㅍ, ㅊ / ____ ㅎ: 밝히대[발키대], 밝히대[발피대], 앉히대[안치대]

2.5. 구개음화(Palatalization)

한국어에서 음절말 ‘ㄷ’, ‘ㅌ’이 모음 ‘ㅣ’또는 이중모음 ‘ㅑ’, ‘ㅓ’, ‘ㅕ’, ‘ㅗ’, ‘ㅛ’와 결합할 때 ‘ㄷ’, ‘ㅌ’이 ‘ㅈ’, ‘ㅊ’으로 교체되어 뒤 음절 첫소리로 재음절화 되는 현상이다.⁴⁾ (1)의 예시는 일반적인 구개음화

4) 국립국어원에서 규정하고 있는 한국어 표준 발음법 제17항에서는 구개음화에 대해 아래와 같이 정의하고 있다. 제17항: 받침 ‘ㄷ, ㅌ(ㅌ)’이 조사나 접미사의 모음 ‘ㅣ’와 결합되는 경우에는, [ㅈ, ㅊ]으로 바꾸어서 뒤 음절 첫소리로 옮겨 발음한다.

현상이며 (2)의 경우, 앞서 언급했던 유기음화 현상이 구개음화를 거쳐하며 일어나는 음운현상이다.

- (1) /ㄷ, ㅌ/ → [ㅈ, ㅊ] / ____ { ㄹ, ㅌ, ㅋ, ㅍ, ㅍ } : 굳이[구지], 같이[가치], 덧붙여[덧부쳐]
 (2) ㄷ → ㅈ / ____ { ㄹ, ㅌ, ㅋ, ㅍ, ㅍ } : 달히대[다티다→다치대], 묻혀[무터→무쳐]

2.6. ㅎ, ㅌ 중화(/h/, /t^h/-neutralization)

2.1.에서 언급했듯이 구개음화까지 모두 진행된 상태에서 남은 자음인 ‘ㅎ’과 ‘ㅌ’이 음절말에 왔을 때 ‘ㄷ’으로 중화하는 규칙을 적용한다.

- (1) /ㅌ, ㅎ/ → [ㄷ] / ____ {C, #} : 갈다[갈때], 히읏[히읃]

2.7. ㄴ 첨가(/n/-Insertion)

표준 발음법 제29항에서 ‘ㄴ’첨가는 합성어 및 파생어에서, 앞 어근이나 접두사의 끝이 자음이고 뒤 단어나 접미사의 첫 음절이 ‘이, 야, 여, 요, 유’인 경우에는, ‘ㄴ’ 음을 첨가하여 [니, 냐, 녀, 뇨, 뉴]로 발음한다고 규정하고 있다. 예시는 아래와 같다. ‘ㄴ’첨가는 비음화를 급여하므로(예: 막일[막닐→망닐], 깻잎[깻입→깻닙→깻닙] 등) G2P모델에서 비음화보다 먼저 적용되어야 할 규칙이다(강옥미, 2003). 그러나 ‘ㄴ’첨가의 경우 기저형에서 실현되는 조건을 만족함에도 예외적으로 적용이 되지 않는 단어가 많아(예: 독약, 그림일기 등) 일반적으로 발음 변환기에는 ‘ㄴ’첨가가 실현되는 어휘 사전이 사용된다(국경아, 김주원, 이호영, 2005).

- (1) Ø → ㄴ / [명사 어근] ____ { ㄹ, ㅌ, ㅋ, ㅍ, ㅍ } 로 시작하는 명사 어근: 맨입[맨닙], 담요[담뇨]

2.8. 비음화(Nasalization)

비음화는 크게 두 가지로, 표준 발음법 제18항에서 규정하고 있는 기저형에서 음절말에 오는 장애음 뒤에 비음이 올 경우 비음의 영향을 받아 표면형에서 기저형의 장애음과 같은 조음 위치인 비음으로 동화되는 현상과 제19항에서 규정하고 있는 기저형에서 ‘ㄹ’ 앞에 ‘ㄱ’, ‘ㅍ’, ‘ㄷ’, ‘ㅇ’이 올 때 ‘ㄹ’이 같은 조음위치의 비음인 ‘ㄴ’으로 교체되는 현상이 있다. 이때 (2)의 예시와 같이 ‘ㄹ’의 비음화가 장애음의 비음동화를 급여하는 규칙이 되는 경우도 있다. 비음화의 예시는 아래와 같다.

- (1) /ㄱ, ㄷ, ㅍ/ → [ㅇ, ㄴ, ㅁ] / ____ { ㄴ, ㅁ } : 먹는[멍는], 있는[인는→인는], 밥 먹어[밤머거]
 (2) /ㄹ/ → [ㄴ] / ____ { ㄱ, ㅍ } : 막론[막논→망논], 협력[협녁→협녁]
 (3) /ㄹ/ → [ㄴ] / ____ { ㅁ, ㅇ } : 담력[담녁] 침략[침녁], 강릉[강녕]

2.9. 유음화(Lateralization)

표준 발음법 제20항에서 유음화는 기저형에서 ‘ㄴ’이 ‘ㄹ’의 앞 또는 뒤에 올 경우 ‘ㄹ’의 영향을 받아 표면형에서 ‘ㄴ’이 같은 조음위치의 유음인 ‘ㄹ’로 동화되는 현상이라고 규정하고 있다. 유음화의 예시는 아래와 같다.

- (1) ㄴ → ㄹ / ____ ㄹ: 난로[날로]
- (2) ㄴ → ㄹ / ㄹ____: 줄넘기[줄럼끼]

2.10. 경음화(Tensing)

한국어에서 평 장애음(예: ‘ㄱ’, ‘ㄷ’, ‘ㅂ’, ‘ㅅ’, ‘ㅈ’)이 경음(예: ‘ㄲ’, ‘ㄸ’, ‘ㅃ’, ‘ㅆ’, ‘ㅉ’)으로 교체되는 현상이다. 앞서 언급한 우선적으로 적용된 규칙인 표준 발음법 제25항의 겹받침 뒤에서의 경음화 외에 실현되는 조건은 첫째, (1)과 같이 평 장애음끼리 결합했을 때, 뒤에 오는 장애음의 경음화, 둘째, (2)와 같이 표준 발음법 제24항의 어간 받침 ‘ㄴ(ㄴ), ㄹ(ㄹ)’ 뒤에 결합되는 ‘ㄱ, ㄷ, ㅂ, ㅅ’의 경음화, 셋째, (3)과 같이 표준 발음법 제26항의 한자어에서 ‘ㄹ’받침 뒤에 연결되는 ‘ㄷ’, ‘ㅅ’, ‘ㅈ’의 경음화, 넷째, (4)와 같이 표준 발음법 제27항의 관형사형 ‘-(으)ㄹ’ 뒤에 연결되는 평 장애음의 경음화, 마지막으로 (5)와 같이 표준 발음법 제28항의 표기상으로는 사이시옷이 없더라도, 관형격 기능을 지니는 사이시옷이 있어야 할(휴지가 성립되는) 합성어의 경우, 뒤 어근의 첫 소리가 평 장애음일 경우의 경음화가 있다. 하지만 이 중에 발음 변환기에 적용될 수 있는 규칙은 (1)과 같은 평 장애음끼리 결합했을 때의 경음화 정도인데 이유는 (2), (4)처럼 품사에 따라, (3)과 같이 단어의 종류에 따라 경음화 규칙이 다르게 적용되는 경우가 있고 (5)와 같은 규칙은 예외가 많아 모델에 적용시키기에는 어려운 규칙들이 있기 때문이다. 특히 (2)의 어간 받침 ‘ㄴ(ㄴ), ㄹ(ㄹ)’ 뒤에서 실현되는 경음화의 경우, 현재까지의 기술로는 대부분의 형태소 분석기 자체에서 오류를 일으켜(예: ‘안다[안따][抱], ‘안다[안다][知]를 형태소 분석기 상에서 구분하지 못함) 모델에 적용시키는데에 어려움이 있다.

- (1) /ㄱ, ㄷ, ㅂ, ㅅ, ㅈ/ → [ㄲ, ㄸ, ㅃ, ㅆ, ㅉ] / {ㄱ, ㄷ, ㅂ}____: 잡기[잡끼], 각도[각또], 답변[답뵤]
- (2) /ㄱ, ㄷ, ㅂ, ㅅ, ㅈ/ → [ㄲ, ㄸ, ㅃ, ㅆ] / [어간 받침 ㄴ, ㄹ]____: 신고[신꼬], 꺾안대[껴안따], 젊지 [점찌]
- (3) /ㄷ, ㅅ, ㅈ/ → [ㄸ, ㅆ, ㅉ] / [어간 받침 ㄴ, ㄹ]____: 갈등[갈뵤], 말살[말쌀], 갈증[갈쵸]
- (4) /ㄱ, ㄷ, ㅂ, ㅅ, ㅈ/ → [ㄲ, ㄸ, ㅃ, ㅆ, ㅉ] / [관형사형 -(으)ㄹ]____: 할 것을(할 꺼슬), 할 바를 (할 빠를)
- (5) 문+고리[문꼬리], 눈+동자[눈뵤자], 신+바람[신빠람], 산+새[산쵸], 손+재주[손쵸주]

2.11. ‘의’ 변이(/ui/-variation)

한글 맞춤법 제9항, 표준 발음법 제5항에서 자음으로 시작하는 ‘의’ 또는 어두에 오지 않는 ‘의’는 표면형에서 ‘ㅣ’로도 실현되고, ‘의’가 관형격 조사로 사용될 때 ‘예’로도 실현된다고 규정하고 있다. 선행 연구에서는 비어두 위치에 오는 ‘의’는 약 68%가 ‘이’, 28%가 ‘예’로 실현되며, 어두에 오는 ‘의’는 94%가 ‘으’로 실현된다는 결과가 있다(박선우 2019). ‘의’ 변이 현상에 관한 예시는 아래와 같다.

- (1) 어두에서는 /으/로 처리: 의재[으재], 의사[으사]
- (2) 자음으로 시작하는 ‘의’는 /이/로 처리: 희망[히망], 떠대[띠대]
- (3) 비어두에서는 /이/로 처리: 거의[거이], 주의[주이]
- (4) 관형격 조사 /의/는 [예]로 실현: 민주주의의[민주주이에]

2.12. 재음절화(Resyllabification)

한국어에서 일어나는 재음절화는 표준 발음법 제13항의 어근 뒤에 오는 모음으로 시작하는 형태소의 종류에 따라 크게 두 가지 경우로 나눌 수 있는데 하나는 뒤에 오는 형태소가 형식 형태소(예: 조사, 어미, 접미사)인 경우 음절 말 자음이 제 음가대로 뒤 음절의 두음이 되는 현상과 표준 발음법 제15항의 뒤에 오는 형태소가 실질 형태소인 경우 음절말 자음이 중화된 후 뒤 음절의 두음이 되는 현상으로 나눌 수 있다. 그러나 이 규칙도 형태소 분석기 자체의 오류로 인해 재음절화를 형태소의 종류에 따라 완벽하게 발음 변환기에 규칙을 적용하기에는 어려움이 있다.

- (1) 어근 + 형식형태소: 옷+이[오시], 걸+에[거태]
- (2) 어근 + 실질형태소: 옷+안[온안→오단], 걸+옷[걸온→거돋]

2.13. 자음 위치동화(Coda place assimilation)

선행하는 음절의 말이 후행하는 음절의 두음의 조음위치에 영향을 받아 선행하는 음절 말 자음의 조음위치가 후행하는 음절 초 자음의 조음위치와 같아지는 수의적 음운현상이다(강옥미 2003; 박선우 2008). 예시는 아래와 같다.

- (1) /ㄴ, ㄷ/ → [ㅁ, ㅂ] / ____ {ㅁ, ㅂ, ㅍ, ㅃ} (치경음 → 양순음): 옷보대[온뽀대] ~ [웁뽀대]
- (2) /ㄴ, ㄷ/ → [ㅇ, ㄱ] / ____ {ㄱ, ㅋ, ㆁ} (치경음 → 연구개음): 옷고름[온꼬름] ~ [웁꼬름]
- (3) /ㅁ, ㅂ/ → [ㅇ, ㄱ] / ____ {ㄱ, ㅋ, ㆁ} (양순음 → 연구개음): 숯김[웁김] ~ [숙김], 짐꾼[짐꾼] ~ [징꾼]

3. 연구의 방법

3.1. 발음 변환기 성능 평가를 위한 말뭉치 구축

본 연구는 사전 연구에서 많이 사용되고 있는 두 한국어 발음 변환기(i.e., g2pk, KoG2P)의 성능 평가를 위해 두 명의 언어 전문가로 구성된 검수자 집단을 만들고 앞 장에서 정리한 각 한국어 발음 변환 규칙에 대한 평가 말뭉치를 구축하였다. 평가 말뭉치 구축 작업에는 언어적 자료로 국립국어원에서 10월 31일 배포한 ‘모두의 말뭉치’의 구어와 문어 말뭉치가 사용되었다(국립국어원, 2022). 평가 말뭉치 구축을 위한 단계는 다음과 같다. 우선, JSON자료 구조⁵⁾로 기록되어 있는 말뭉치를 Python으로 불러온 뒤 구어와 문어의 문장만을 추출하여 텍스트 형식의 파일(.txt)에 문장을 추출한 파일의 이름과 추출된 문장을 별도로 저장하여 하나의 파일로 생성하였다. 다음으로, 앞서 정리한 발음 변환 과정에서 중요한 음운론적 규칙에 해당하는 예시를 기반으로 각 음운론적 규칙 변환이 발생하는 문장을 따로 추출하였다. 문장을 추출한 이후에는 추출된 문장을 두 명의 검수자가 발음 변환 규칙에 근거하여 발음열로 변환하였으며, 변환된 문장에서 해당 음운론적 규칙이 적용된 대표 단어를 추출하여 별도의 변수로 생성하였다. 변환 작업을 위한 기준으로는 한국어 어문 규범에서 제공하는 표준 발음이 기준으로 사용되었으며, 그 외에도 한국어 어문 규범에는 자세한 기술이 없지만 음운론 연구에서 활발하게 기술되고 있는 현상인 공명음 사이에서의 ‘ㅎ’탈락(결혼[겨론], 신혼[시논], 이혼[이온]), 자음 위치동화(웃고름[으꼬름 → 옥꼬름]) 등의 수의적인 언어 변이 현상도 포함하여 작업을 진행하였다. 음운론적 발음 변환 규칙을 적용하여 최종 구축된 평가 말뭉치는 그림 1과 같다.

| | A | B | C | D | E | F |
|----|-----------------------------|-------------------------|--|---------|-------------------|----------------|
| 1 | rule | filename | underlying | surface | target_underlying | target_surface |
| 2 | coda neutralization | Original_WARW1900000086 | 동넉 들머리로 갓 솟은 아츰동넉 들머리로 갓 소슨 아츰동넉 | 동넉 | 동넉 | 동넉 |
| 3 | coda neutralization | Original_WARW1800000048 | 선처림 가느다란 뽕뽕낙타 ¹ 선처림 가느다란 뽕뽕낙타 ¹ 창밖 | 창밖 | 창밖 | 창밖 |
| 4 | coda neutralization | Original_WARW1800000007 | 창안에 면대감 부인이 불공창안에 면대감 부이니 불공웃 | 웃 | 웃 | 웃 |
| 5 | coda neutralization | Original_WARW1900000043 | 세상 위에 밤낮 퍼져 있거라세상 위에 밤낮 퍼져 있거라밤낮 | 밤낮 | 밤낮 | 밤낮 |
| 6 | coda neutralization | Original_WARW1900000043 | 쑥쑥 자라고 있다 쑥쑥 자라고 있다 | 있다 | 있다 | 있다 |
| 7 | coda neutralization | Original_SBRW1900016598 | 에킨대 아~ 그러니까 맞출 에킨대 아~ 그러니까 맞출 | 맞출 | 맞출 | 맞출 |
| 8 | coda neutralization | Original_WARW1900002860 | 말 빠진 독에 물 붓기라는 말 빠진 독에 물 붓기라는 말 | 말 | 말 | 말 |
| 9 | coda neutralization | Original_SBRW1900016791 | 지하고 또 우리 앞 동네 살지하고 또 우리 앞 동네 사지 | 앞 | 앞 | 앞 |
| 10 | coda neutralization | Original_SBRW1900017741 | 허만 음~ 그냥 뭐 평양 시 ² 하면 음~ 그냥 뭐 평양 시 ² 하고 | 약꼬 | 약꼬 | 약꼬 |
| 11 | coda neutralization | Original_SBRW1900011324 | 망치와 낫 그리고 가운데 망치와 낫 그리고 부시 안는 | 낫 | 낫 | 낫 |
| 12 | coda cluster simplification | Original_SBRW1800000072 | 네 이런거부터 좀 고만해네 이런거부터 좀 고만해아웃 | 웃 | 웃 | 웃 |
| 13 | coda cluster simplification | Original_SARW1800000620 | 열부 학생만이 창문 옆에 열부 학생만이 창문 여덟 안는 | 안는 | 안는 | 안는 |
| 14 | coda cluster simplification | Original_SARW1800000456 | 한 번 주행을 시작하면 별5한 번 주행을 시작하면 별5여덟 | 여덟 | 여덟 | 여덟 |
| 15 | coda cluster simplification | Original_WARW1900000088 | 따라서 부모를 잃은 그들은따라서 부모를 이룬 그들은삼 | 삼 | 삼 | 삼 |
| 16 | coda cluster simplification | Original_WARW1900000094 | 그의 목소리는 무슨 노래의그 목소리는 무슨 노래에울조리고 | 음조리고 | 음조리고 | 음조리고 |
| 17 | coda cluster simplification | Original_WARW1900000095 | 그들은 영원한 안식을 위해 그드론 영원한 안식을 위해 값비싼 | 값비싼 | 값비싼 | 값비싼 |
| 18 | coda cluster simplification | Original_WARW1900000600 | 마음이 외골로 곧거나 아주마음이 외골로 곧거나 아주외골로 | 외골로 | 외골로 | 외골로 |
| 19 | coda cluster simplification | Original_WARW1800000007 | “달 잠들라구 그라.” 달 자들라구 그라 | 달 | 달 | 달 |
| 20 | coda cluster simplification | Original_SBRW1900000006 | 노래가 광장하 밝고 경쾌하노래가 광장이 밝고 경쾌하 밝고 | 밝고 | 밝고 | 밝고 |

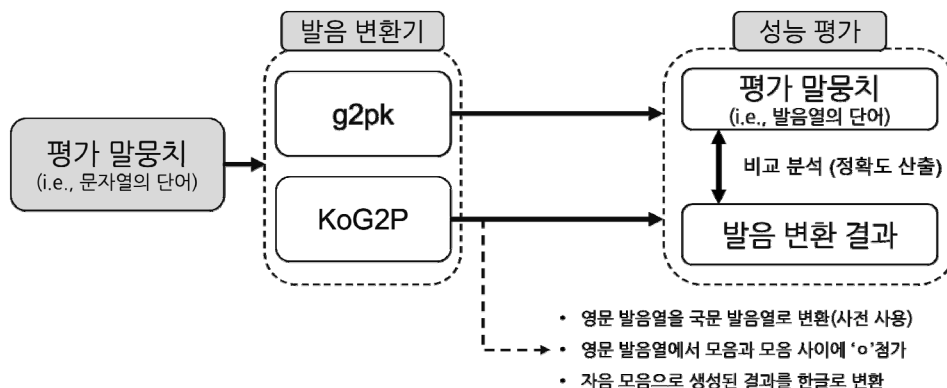
<그림 1> 본 연구에서 사용된 평가 말뭉치를 보여주는 스크린 샷

- 5) JSON은 키(key)와 값(value)로 이루어진 데이터로 Python에서 제공하는 사전(dictionary) 자료 구조와 그 구조가 일치해 Python에서 사용하기에 편리한 자료 구조이다.

평가 말뭉치는 그림 1에서 보여주는 모습과 같이 총 6개의 변수로 구성되어 있으며 변수에 대한 설명은 다음과 같다. 우선, 'rule' 변수는 앞 장에서 정리한 발음 변환 규칙 중 반드시 반영되어야 하는 13가지의 규칙에 대한 정보를 나타내고 있다. 다음으로 'filename' 변수는 rule 변수의 규칙이 포함되어 있는 문장을 추출한 모두의 말뭉치 파일의 이름을 나타내고 있으며, 'underlying'과 'surface'는 'rule'에 작성된 음운론적 규칙이 포함되어 있는 문장의 문자열과 발음열에 대한 정보가 기록되어 있다. 마지막으로, 'target_underlying'과 'target_surface'는 문장내에서 'rule'에 기록되어 있는 규칙이 직접적으로 반영되어 있는 단어에 대한 문자열과 발음열의 정보를 나타내고 있다. 본 연구에서는 그림 1에서 보여주는 말뭉치를 구축하는 모든 과정에서 두 검수자가 협업하여 작업을 진행하였으며, 최초 두 검수자가 변환한 발음열에 대한 일치 정도는 0.842였고 이후 논의를 거쳐 최종 평가 말뭉치를 구축하였다. 최종 평가 말뭉치는 13가지 음운론적 규칙에 따라 각 10개의 예문을 포함하고 있으며, 총 $130 * 6$ 의 데이터로 구성되어 있다.

3.2. KoG2P와 g2pk 평가를 위한 실험 설계

본 연구에서는 Python 기반으로 작동하는 두 개의 발음 변환기에 대해 언어학적인 관점에서 평가를 수행하였으며, 평가에는 앞서 언급한대로 사전 연구에서 많이 사용되고 실제 인공지능 서비스에서도 많이 사용되고 있는 'g2pk'와 'KoG2P'가 사용되었다. 실험에는 사전에 두 명의 검수자가 작성한 평가 말뭉치가 사용되었으며, 평가 말뭉치에 포함되어 있는 6개의 변수 정보 중 각 발음 변환 규칙에 대한 'target_underlying'과 'target_surface' 변수가 평가 과정에 사용되었다. 또한 두 발음 변환기에 대한 성능 평가를 위한 작업 환경으로 Google에서 제공하는 Colab이 사용되었으며, konlpy, jamo, pandas와 같은 Python 패키지들이 같이 사용되었다(Dong, 2015; McKinney, 2011; Park, 2014).



<그림 2> 발음 변환기 성능 평가를 위한 실험 설계

발음 변환기 성능 평가를 위한 실험의 과정(그림 2)은 다음과 같다. 우선, 평가 말뭉치에서 각 음운론적 규칙에 따라 작성된 문자열의 단어가 작성되어 있는 ‘target_underlying’ 변수가 입력 값으로 각 발음 변환기에 전송된다. 입력 값을 전송 받은 발음 변환기는 해당 문자열을 발음열로 변환하여 산출 결과를 생성한다. 이 과정에서 ‘g2pk’의 경우 입력 받은 한국어 문자열을 한국어 발음열로 산출하기 때문에 평가 말뭉치에서 제공하는 발음열 변수인 ‘target_surface’와의 비교가 용이하지만 ‘KoG2P’의 경우 한국어 문자열을 로마자 표기법의 발음열로 산출하기 때문에 산출 결과를 평가 말뭉치와 비교하기에 용이하지 않다. 따라서 본 연구는 ‘KoG2P’가 산출하는 로마자 발음열을 한국어 발음열로 변환하기 위해 다음의 과정을 추가하였다. 우선, 현재 ‘KoG2P’에서 오픈 소스를 통해 제공하는 한국어 자음 및 모음에 대한 로마자 표기법 사전을 사용하여 로마자 발음열을 한국어 자음과 모음으로 일대일 대응하여 변환하였다. 변환하는 과정에서 초성에 사용된 ‘ㅇ’의 경우 로마자 발음열에서는 그 정보를 기입하지 않기 때문에 한글 중성에 사용되는 모음 이후에 모음이 오는 경우 ‘ㅇ’을 삽입하여 한글의 형태에 맞게 표기하였다. 마지막으로 자음과 모음의 연속으로 작성되어 있는 발음열 결과를 Python의 jamo패키지에서 제공하는 join_jamos함수를 통해 한글로 표기하여 발음열로 산출하였다. 두 발음 변환기가 입력 받은 문자열 단어를 발음열 단어로 산출한 이후에는 각각의 산출 결과가 평가 말뭉치에서 제공하는 발음열 단어와 일치하는지를 비교하였고 총 130개의 문자열 단어에 대한 발음열 단어를 비교함으로써 각 발음 변환기의 성능을 측정하였다. 예를 들어, 10개의 문자열 단어를 발음열로 산출하였을 때, 발음 변환기의 산출 결과 중 9개의 결과가 평가 말뭉치의 발음열 정보와 일치하면 발음 변환기의 성능은 90%(0.9)로 측정된다.

4. 한국어 발음 변환기에 대한 성능 평가

4.1. 평가 결과 및 보고

본 연구에서는 G2P로 지칭되는 문자열의 입력 값을 발음열로 변환하는 기술로써 사용되는 ‘g2pk’와 ‘KoG2P’가 언어학적인 관점에서 실제 발음 변환 과정에서 반영되어야 하는 음운론적 규칙을 얼마나 잘 반영하는지에 대한 평가를 수행하는 것을 목적으로 진행되었다. 실험 결과는 다음과 같다.

<표 1> 언어학적 발음 변환 규칙에 대한 두 발음 변환기의 성능 결과

| 음운론적 규칙 | g2pk 산출 성능 | KoG2P 산출 성능 |
|---------|------------|-------------|
| 자음 중화 | 1.0 | 1.0 |
| 자음군 단순화 | 1.0 | 1.0 |
| ‘ㅎ’탈락 | 0.4 | 0.4 |
| 유기음화 | 0.9 | 1.0 |
| 구개음화 | 0.8 | 0.9 |

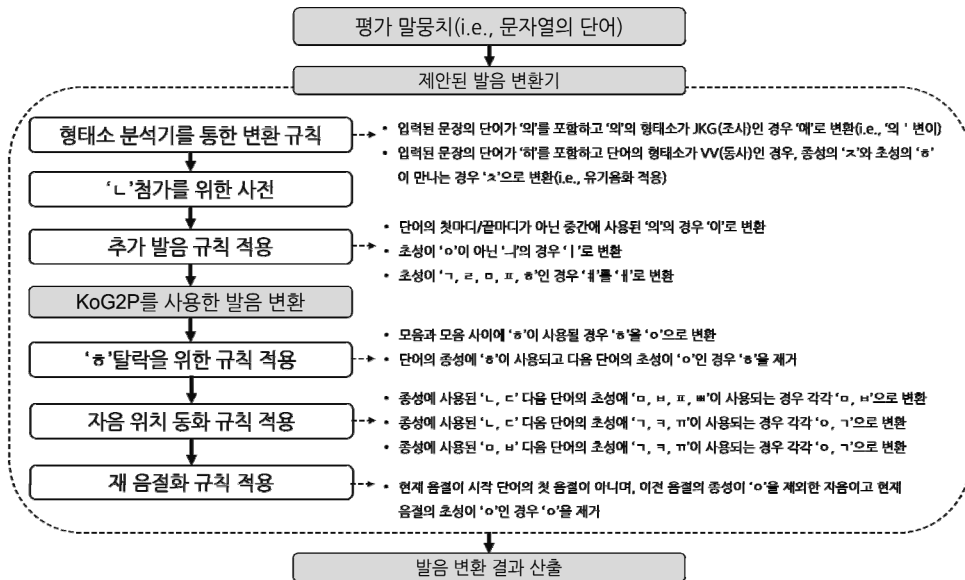
| | | |
|---------|-----|-------|
| ‘ㅎ,ㅌ’중화 | 0.7 | 0.7 |
| ‘ㄴ’첨가 | 0.3 | 0.5 |
| 비음화 | 1.0 | 1.0 |
| 유음화 | 1.0 | 1.0 |
| 경음화 | 1.0 | 1.0 |
| ‘의’ 변이 | 0 | 0 |
| 재음절화 | 1.0 | 1.0 |
| 자음 위치동화 | 0 | 0 |
| 평균 | 0.7 | 0.731 |

실험의 결과, 표 1 에서 보듯, ‘g2pk’와 ‘KoG2P’ 두 변환기 모두 70%를 상회하는 성능을 보였다. 세부 내역을 살펴보면 두 변환기 모두 자음 중화, 자음군 단순화, 유기음화, 구개음화, 비음화, 유음화, 경음화, 재음절화 등의 규칙에 대해서는 90% 이상의 비교적 높은 성능을 보였고 특히 자음 중화, 자음군 단순화, 비음화, 유음화, 경음화, 재음절화에서는 두 변환기 모두 100% 의 성능을 보였다. 반면, ‘ㅎ’탈락, ‘ㅎ,ㅌ’중화 에 대해서는 40~70%의 성능으로 결과에 미진한 부분이 있음을 보였다. 실제로 평가 말뭉치에서 사용된 ‘결혼, 이혼, 아홉’의 경우, ‘ㅎ’탈락이 적용되어 발음열에서는 ‘겨론, 이온, 아웁’이 되어야 하지만 ‘g2pk’와 ‘KoG2P’ 모두 ‘ㅎ’탈락을 반영하지 못하고 문자열 그대로인 ‘결혼, 이혼, 아홉’을 발음열로 산출하였다. 특히 ‘ㄴ’첨가 규칙에 대해서는 두 변환기 모두 30~50%이하의 낮은 성능을 보였다. 이는 ‘ㄴ’첨가 규칙이 상당히 복잡한 언어학적 요소들의 상호 작용에 의해 영향을 받으며, 화자들이 이 규칙을 수의적으로 적용하기 때문이다. 예를 들어, 선행 연구에 따르면 ‘ㄴ’보다는 ‘ㄷ, ㄱ, ㄴ, ㄷ’가 뒤에서 실현될 때(예: 막일[망닐] < 담요[담뇨]), 그 중에서도 ‘ㄷ’가 뒤에서 실현될 때(예: 담요[담뇨] < 식용유[시공뉴]), 앞에 공명자음이 실현될 때(예: 막일[망닐] < 큰일[큰닐]), 그 중에서도 ‘ㄷ, ㄴ’이 올 때(예: 탕약[탕낙] < 줌약[죤낙]) 그리고 짧은 단어보다 긴 단어에서(예: 막일[망닐] < 수학여행[수양녀영]) 더 자주 ‘ㄴ’첨가가 일어난다(Jun, 2015). 마지막으로 ‘의’ 변이와 자음 위치동화에 관한 규칙이 두 변환기 모두에 반영되어 있지 않음을 알 수 있다. 예를 들어, 평가 말뭉치에 사용된 자음 위치동화인 ‘준비, 잔고를’의 경우, 자음 위치동화가 적용되어 발음열에서는 ‘죤비, 잔고’로 발음열이 산출되어야 하는데 두 발음 변환기 모두 문자열 그대로인 ‘준비, 잔고’로 발음열을 산출하였다.

4.2. 성능 향상을 위한 언어학적 제안

두 발음 변환기에 대한 성능 평가 결과, 13가지의 음운론적 규칙 중 두 발음 변환기 모두 자음 중화, 자음군 단순화, 유기음화, 구개음화, 비음화, 유음화, 경음화, 재음절화 등의 규칙에 대해서는 높은 성능을 보여주는 반면에 ‘ㅎ’탈락, ‘ㅎ,ㅌ’중화, ‘ㄴ’첨가, ‘의’ 변이, 자음 위치동화 규칙에 대해서는 두 변환기 모두 낮은 성능을 보였다. 이에 본 연구에서는 기존의 변환기의 성능 향상을 위해 ‘ㅎ’탈락, ‘ㅎ,ㅌ’중화, ‘의’ 변이, 자음 위치동화 등 두 변환기에서 낮은 성능을 보여준 규칙에 관한 추가 규칙을

첨가하고 규칙으로 설명하기 복잡한 ‘ㄴ’첨가 현상에 대해 작은 규모의 사전을 생성하여 예외 처리하는 방식의 방법을 제안하였다. 본 연구에서 제안하는 방법에 대한 순서도는 아래의 그림 3과 같다.



<그림 3> 본 연구에서 제안하는 발음 변환기의 순서도

본 연구에서는 이전의 실험 결과에서 더 높은 성능을 보인 'KoG2P'를 기반으로 두 발음 변환기에 적용되지 않은 음운론적 규칙과 'ㄴ'첨가 사전을 추가함으로써 발음 변환기의 성능을 높였다. 본 연구에서 제안하는 발음 변환기의 작동 순서는 다음과 같다. 우선, 입력된 문장을 단어 단위로 읽어 들이고 형태소 분석기를 기반으로 조사로 사용된 '의'와 동사의 굴절형으로 사용된 '하'와 관련된 규칙을 추가한다. 이때, '의'의 품사를 분석하고 발음 규칙을 적용하는 작업은 다른 음운 규칙이 적용된 이후에도 적용이 가능하지만 'ㅈ'이 'ㅎ'과 만날 때, 'ㅊ'으로 실현되는 유기음화의 경우, 자음 중화를 출현하는 규칙이므로 자음 중화 규칙을 포함하는 KoG2P를 사용한 발음 변환 전에 선행하여 사용되어야 한다. 다른 음운 규칙을 적용하기에 앞서 형태소 분석기를 우선 적으로 다음으로 'ㄴ'첨가에 대한 사전을 불러와 입력된 단어가 사전에 있는 경우 사전에서 정의하는 발음으로 변환하고, '의' 변이와 'ㅋ' 변환에 대한 규칙을 적용한다. 이후에는 'KoG2P' 발음 변환기를 사용하여 문자열의 단어를 발음열로 변환한다. 마지막으로 'ㅎ'탈락, 자음 위치동화, 재음절화에 대한 규칙을 적용하여 최종 발음 변환 결과를 산출한다.

4.3. 개선된 발음 변환기에 대한 성능 평가 결과 및 보고

본 연구에서 앞서 진행한 발음 변환기의 성능 평가를 통해 ‘g2pk’와 ‘KoG2P’의 경우, 자음 중화 자음군 단순화, 유기음화, 구개음화, 비음화, 유음화, 경음화, 재음절화 등의 규칙은 잘 반영하고 있지만 ‘ㅎ’탈락, ‘ㅎ,ㄷ’중화, ‘ㄴ’첨가, ‘의’ 변이, 자음 위치동화에 대한 규칙은 잘 반영되어 있지 않음을 확인하였다. 이에 본 연구에서는 두 발음 변환기 중 더 나은 성능을 보여준 ‘KoG2P’를 기반으로 추가적인 음운론적 규칙을 그림 3과 같이 적용하였다. 본 연구에서 제안하는 발음 변환기와 기존 발음 변환기의 성능 평가 결과는 표 2와 같다.

<표 2> 두 발음 변환기와 본 연구에서 제안하는 발음 변환기의 성능 평가 결과

| 음운론적 규칙 | g2pk 산출 성능 | KoG2P 산출 성능 | 본 연구의 발음 변환기 성능 |
|---------|------------|-------------|-----------------|
| 자음 중화 | 1.0 | 1.0 | 1.0 |
| 자음군 단순화 | 1.0 | 1.0 | 1.0 |
| ‘ㅎ’탈락 | 0.4 | 0.4 | 1.0 |
| 유기음화 | 0.9 | 1.0 | 1.0 |
| 구개음화 | 0.8 | 0.9 | 0.9 |
| ‘ㅎ,ㄷ’중화 | 0.7 | 0.7 | 0.7 |
| ‘ㄴ’첨가 | 0.3 | 0.5 | 0.7 |
| 비음화 | 1.0 | 1.0 | 1.0 |
| 유음화 | 1.0 | 1.0 | 1.0 |
| 경음화 | 1.0 | 1.0 | 1.0 |
| ‘의’ 변이 | 0 | 0 | 1.0 |
| 재음절화 | 1.0 | 1.0 | 1.0 |
| 자음 위치동화 | 0 | 0 | 0.9 |
| 평균 | 0.7 | 0.731 | 0.938 |

실험 결과, 본 연구에서 제안하는 발음 변환기의 성능을 90%를 상회하는 결과를 보여주었으며, 13가지 음운론적 규칙에서 낮게는 70%, 높게는 100%의 변환 성능을 보여주었다. 특히 이전에 두 발음 변환기에 잘 적용되지 않았던 ‘ㅎ’탈락과 ‘ㄴ’첨가에서 평균적으로 30% 높은 성능을 보여주었으며, 전혀 적용되지 않았던 ‘의’ 변이와 자음 위치동화에서는 90%에서 100%로 높은 성능을 보여주었다.

5. 토론

음운 규칙에 바탕하여 문자열을 발음열로 변환할 때 가장 어려운 부분은 규칙의 수의적 적용에 관한 부분이다. 앞서 논의한 ‘ㄴ’첨가와 같은 예 외에도 한국어에서 수의적 음운 규칙 적용의 예는 다양하다. 예를

들어 명사의 종성이 ‘ㄷ, ㅈ, ㅌ’으로 끝날 경우 뒤에 모음으로 시작하는 조사 ‘이, 에, 으’ 등이 올 경우, 기존의 음운 규칙에 의하면 재 분절화에 따른 연음규칙에 따라 종성의 음가가 살아야 한다(예: 꽃이 → [꼬치]). 그러나 상당한 경우 화자들은 음운론적으로 정당화되지 않는 ‘ㅅ’ 발음을 적용하여 [꼬시]라는 발음을 산출한다. 뿐만 아니라 ‘ㄷ, ㅌ’ 이 조사 ‘이’ 앞에서 ‘ㅈ, ㅊ’으로 변하는 구개음화 규칙 또한, 규칙에 의해 예측이 되지 않는 조사 ‘에’ 혹은 ‘으’에도 적용하는 경우가 종종 관찰된다(예: 발에 → [바체], 발은 → [바춘]). 명사의 종성이 모음 앞에서 ‘ㅅ’으로 변화하는 것은 음운 규칙에 의한 것이라기 보다 한국어에 가장 빈번하게 관찰되는 자음의 변이가 ‘ㄷ’ ~ ‘ㅅ’ (예: 옷~옷이 → [웁]~[오시]) 이기 때문에 유추에 의해 ‘꽃’과 같은 단어에도 [꼇]~[꼬시]와 같은 변이를 적용시키기 때문으로 규칙으로 설명하기는 힘든 부분이다.

이러한 수의적 음운 규칙 적용은 음운 규칙에 바탕한 발음열 도출을 매우 어렵게 하기 때문에, 사전을 생성하여 처리하는 방법이 있으나 이는 본래의 규칙 기반 변환기 생성의 취지에도 어긋나고 사전에 포함되지 못한 단어에 대한 오류가 생성될 수 있는 여지가 있기 때문에 가능한 규칙 기반 변환을 추구하는 방향이 옳다. 또한, 사전을 활용한다고 하더라도 수의적인 발음 규칙을 완전히 설명할 수 있는 것은 아니다. 가령 어떤 화자는 ‘꽃이’라는 단어를 [꼬치]로 발음하기도 하지만 [꼬시]로 발음하기도 하기 때문에 일대일 대응만으로는 문자열을 발음열로 변환하는 것에 한계가 있다.

한 가지 생각해 볼 수 있는 방법은 언어학의 선행 연구들을 참조하여 이와 같은 변이가 각 환경에서 어느 정도의 빈도로 일어나는지에 대한 정보를 얻어 이를 활용하는 것이다. 예를 들어 앞서 기술한 명사의 종성에 위치한 발음 변이의 경우 Choi (2004)에 따르면 ‘꽃’의 종성 ‘ㄷ’이 모음 앞에서 연음규칙의 적용으로 원래의 ‘ㄷ’으로 발음되는 비율은 다음과 같다: ‘이’ 앞에서는 73%, ‘으’ 앞에서는 68%, ‘에’ 앞에서는 47%. 또 같은 논문에서 ‘발’의 종성 ‘ㅌ’이 원래의 ‘ㅌ’으로 발음되는 비율은 다음과 같다: ‘이’ 앞에서 77%, ‘으’ 앞에서 77%, ‘에’ 앞에서 88%. 이러한 비율에 대한 정보를 발음열 도출 시 적용한다면 사전을 활용하는 것보다 더 정확한 성능을 도출할 수 있다.

이러한 수의적인 음운 규칙의 적용은 또한 대화의 상대에 따라서도 달라진다. 사회 언어학에서 잘 알려진 바와 같이 사람들은 격식이 있는 자리 혹은 상대와 대화할 때는 발음을 규범 문법과 비슷하게 분명히 하려는 경향이 있는 반면 친근한 상대와 격식이 없는 대화를 할 때는 발음을 덜 분명하게 하려는 경향이 있다. 우리가 방송, 특히 뉴스를 진행하는 사회자가 ‘꽃이’를 [꼬시]가 아닌 [꼬치]라고 발음하는 것을 쉽게 상상할 수 있는 것이 예가 될 수 있다. 사회적 거리와 더불어 음운 규칙의 적용에 영향을 미치는 것으로 잘 알려진 또 하나의 요소는 어휘의 빈도수이다. 가령 영어에서 단어 끝의 ‘t’ 나 ‘d’가 수의적으로 탈락하는 현상은, 빈도수가 높은 ‘just’와 같은 단어에서 빈도수가 낮은 ‘bust’와 같은 단어에서보다 더 빈번하다는 것이 잘 알려져 있다(Bybee, 2002; Patrick, 1991). 앞서 말한 ‘꽃이’와 같은 명사 끝 자음의 변이에 관한 현상 또한 ‘꽃이’와 같이 빈번한 단어에서 ‘꽃이’와 같은 희귀한 단어에 비해 좀 더 지저형의 ‘ㄷ’을 살려 연음법칙을 적용한 [꼬치]와 같은 발음이 더욱 흔하다는 것이 보고되었다(Ko and Jun, 2022).

이와 같은 음운의 변이 현상은 일견 복잡해 보이나 음운론 연구의 자료만 있다면 기존의 일대일 대응 방식을 넘어 스토캐스틱(stochastic)한 방법의 음운 규칙을 적용할 수 있게 됨으로써 사전 활용을 최소화 하며 성능을 향상 시킬 수 있는 근본적인 방법이다. 이는 또한 언어 현상 자체가 스토캐스틱하기 때문에 언어 산출 모델에 가장 근접한 방법이 될 수 있다. 이러한 이유로 문자열로 작성한 한국어

를 발음열로 변환하는 작업은 기존의 일대일 방식의 음운규칙 적용만으로는 높은 정확도를 달성할 수 없으며, 많은 변이성에 대한 언어학적 지식, 특히 형태론과 음운론에 대한 지식이 반영되어야 높은 성능의 발음 변환기를 개발 할 수 있다.

본고에서 제시한 음운론적 규칙의 수의적인 적용 방식에 대한 해법과 더불어 또 하나 극복해야 할 주제는 음운 규칙의 적용 범위에 관한 문제이다. 가령 저해음 뒤에서 일어나는 평저해음의 경음화 현상 (post-obstruent tensing)은 해당 저해음이 억양구 (Accentual Phrase; Jun, 1993) 안에 있을 때 적용되며, 두 저해음 사이에 억양구의 경계가 존재 할 때에는 작용되지 않는다. 예를 들어 {가죽 장갑}_{AP}의 경우 [장]은 앞 음절의 종성 [ㄱ]으로 인해 [짱]으로 발음되지만 {가죽}_{AP}{종류별}_{AP}과 같은 경우에는 앞 음절에 종성 [ㄱ]이 있음에도 불구하고 [ㄱ]과 [중] 사이에 존재하는 억양구 경계로 인해 [중]에 경음화가 적용되지 않는다. 따라서 문장의 어떤 위치에 억양구의 경계가 형성되는가 하는 문제가 규칙으로 도출될 수 있는가 하는 주제와 연결되어 있고 이는 운율 음운론의 이론적 발전을 규칙화 해야 하기 때문에 상당한 정도의 언어학적 지식을 필요로 하는 부분이다.

추가적으로, 더 정확하게 발음 변환기의 성능을 분석하기 위해서는 ‘모두의 말뭉치’ 뿐만 아니라 소셜 미디어에서 사용하는 텍스트 등을 포함한 더 다양한 말뭉치가 평가 말뭉치에 포함 되는 것이 이상적이다. 본 연구에서 사용된 국립국어원의 ‘모두의 말뭉치’의 경우, 정형화된 한국어가 많이 포함되어 있고 특히 구어 말뭉치보다 문어 말뭉치의 비중이 많기 때문에 발음 변환기의 평가 대상으로 사용하기에 한계가 있다고 할 수 있다. 또한, 앞서 언급된 바와 같이 다양한 언어적 혹은 사회적 현상으로 인해 본 연구에서 제시한 13개의 음운론적이 규칙만으로는 실제 대화에서 발생하는 변환 규칙을 모두 설명하기에는 여전히 부족한 면이 있다. 이는 근본적으로 언어학, 특히 음운론 분야의 연구의 한계에 기인하는 부분이 있다. 따라서 향후 수의적 음운 규칙에 대한 자료가 축적될 경우 이에 대한 규칙화를 통해 G2P 알고리즘을 보완 및 확장하고, 또한 운율 음운론에 기반한 억양구 경계 생성에 대한 규칙을 추가함으로써 우리는 더욱 더 사전에 대한 의존도를 축소시키고 규칙 기반 산출 발음열의 정확도를 높일 수 있을 것이다.

6. 결론

본 연구에서는 한국어 발음 변환 과정에 적용되어야 하는 음운론적 규칙을 정리하고 현재 AI기술에서 사용되는 한국어 발음 변환기(i.e., g2pk, KoG2P)가 이를 잘 반영하고 있는지에 대해 현황을 탐색하였다. 또한 두 발음 변환기에 적용되지 않은 음운론적 규칙을 정리하고 이를 반영하는 방안을 제시하는 것을 목적으로 수행되었다. 본 연구의 결과를 요약하면 다음과 같다.

첫째, 한국어 발음 변환 과정에 적용되는 음운론적 규칙은 13가지로 요약된다. 본 연구는 현재의 발음 변환기에 대한 성능 평가를 수행하기에 앞서 언어학 분야의 선행 연구와 국립국어원의 한국어 어문 규범을 바탕으로 발음 변화 과정에서 적용되어야 하는 음운적으로 규칙을 정리하였다. 본 연구에서 정리하여 제시한 음운론적 규칙에 대한 세부 내역을 살펴보면 자음 중화, 자음군 단순화, ‘ㅎ’탈락, 유기음화, 구개음화, ‘ㅎ,ㅌ’중화, ‘ㄴ’첨가, 비음화, 유음화, 경음화, ‘의’ 변이, 재음절화, 자음 위치

동화로 나누어진다.

둘째, 본 연구에서 살펴본 두 발음 변환기는 ‘ㅎ’탈락, ‘ㅎ,ㅌ’중화, ‘의’ 변이, 자음 위치동화에 대한 음운론적 규칙을 반영하지 못하고 있다. 본 연구에서는 발음 변환기에 관련된 사전 연구들과 최근의 AI기술 동향을 기반으로 대표적인 한국어 발음 변환기로 ‘g2pk’와 ‘KoG2P’를 선정하였으며 두 발음 변환기가 앞서 정리한 음운론적 규칙을 잘 반영하는지에 대한 성능 평가를 수행하였다. 성능 평가 결과, 두 발음 변환기는 자음 중화, 자음군 단순화, 유기음화, 구개음화, 비음화, 유음화, 경음화, 재음절화 등의 규칙은 잘 반영하고 있지만 ‘ㅎ’탈락, ‘ㅎ,ㅌ’중화, ‘ㄴ’첨가, ‘의’ 변이, 자음 위치동화에 대한 규칙은 잘 반영되어 있지 않음을 확인하였다.

셋째, 음운론적 규칙을 추가로 정리하고 전산화하여 반영할 경우, 발음 변환기의 성능을 향상시킬 수 있다. 본 연구는 앞서 두 발음 변환기에 적용되지 않은 ‘ㅎ’탈락, ‘ㅎ,ㅌ’중화, ‘ㄴ’첨가, ‘의’ 변이, 자음 위치 동화에 대한 규칙을 전산화하여 반영하였으며, 두 발음 변환기보다 20%이상 향상된 성능을 보고하였다. 그 중에서도 ‘ㅎ’탈락, ‘의’ 변이, 자음 위치동화의 경우 기존의 발음 변화기에서 누락된 음운론적 규칙으로 본 연구에서 제시하는 발음 변환기가 평균적으로 두 발음 변환기보다 높은 성능을 보여주는데에 크게 기여하였다.

본 연구는 언어학 특히 형태론과 음운론에 관련된 사전 연구를 바탕으로 음운론적 규칙을 정리하고 발음 변환 과정에 반영하는 경우 기존의 발음 변환기의 성능을 향상시킬 수 있다는 것을 제시하였다. 더불어 발음 변환기의 추가적인 성능 향상을 위해서는 수리적인 발음 적용 규칙에 대한 이해와 운율 음운론에 기반한 억양구 도출에 관한 이해가 규칙에 반영되어야 함을 제시하였다. 이는 한국어 발음 변환기 개발에 있어 전산화 능력도 중요하지만 언어학적인 관점이 보다 중요하다는 것을 시사하며 언어학적 지식과 전산 기술이 잘 융합되면 사전에 대한 의존도를 최소화하면서도 보다 정확한 발음 변환기를 구현 할 수 있음을 제시한다.

<참고문헌>

- 강옥미. (2003). *한국어 음운론*. 태학사.
- 국경아, 김주원, & 이호영. (2005). 선호도 조사를 통한ㄴ첨가 현상의 실현 양상 연구. *말소리*, 53, 36-59.
- 국립국어원. (2017). *한국어 어문 규범*. 2022, 11, 08 검색. <https://kor norms.korean.go.kr/main/main.do>
- 국립국어원. (2022). *모두의 말뭉치*. 2022, 11, 08 검색. <https://corpus.korean.go.kr/#none>
- 노석은. (2003). *국어 연구개 비음 /ㅇ/의 음절 내 위치에 대한 실험음성학적 연구*, 고려대학교 석사학위논문.
- 박선우. (2008). 한국어 위치동화의 실험음성학적 분석. *언어연구*, 25(2), 45-65.
- 박선우. (2019). 한국어 이중모음 /의/의 음향적 특성과 유형에 대하여. *현대문법연구*, 102, 165-183.
- 서울대학교. (2021). *SMART-G2P*. 2022, 11, 08 검색. <https://github.com/SMART-TTS/SMART-G2P>

- 엄태수. (2014). 현대국어 음운현상의 상대적 강도에 대하여. *새국어교육*, 99, 345-374.
- 카카오 브레인. (2021). PORORO: Platform Of neuRal mOdelS for natuRal language prOcessing. 2022, 11, 08 검색. <https://kakaobrain.github.io/pororo/index.html>
- Bybee, J. (2002). Word frequency and context of use in the lexical diffusion of phonetically conditioned sound change. *Language Variation and Change* 14(3), 261-290. <https://doi.org/10.1017/S0954394502143018>.
- Cho, Won-Ik, Kim, Seok-Min, & Kim, Nam-Soo. (2020). Towards an efficient code-mixed grapheme-to-phoneme conversion in an agglutinative language: A case study on Korean transliteration. *Proceedings of the The 4th Workshop on Computational Approaches to Code Switching*, 65-70.
- Cho, Yejin. (2017). KoG2P. Available online: <https://github.com/scarletcho/KoG2P> (accessed on 08 November 2022).
- Choi, Hyewon. (2004). *A survey of standard pronunciation III* [phyocunpalum silthecosa III]. Seoul: The National Academy of Korean Language.
- Divay, M., & Vitale, A. J. (1997). Algorithms for grapheme-phoneme translation for English and French: Applications. *Computational Linguistics* 23(4), 495-523.
- Dong, Joshua. (2015). jamo. Available online: <https://python-jamo.readthedocs.io/en/latest/> (accessed on 08 November 2022).
- Dutoit, T. (1997). *An introduction to Text-to-Speech synthesis*. Kluwer Academic Publishers.
- Hong, Yoon-Seok, Ki, Kyung-Seo, & Gweon, Gahgene. (2018). Automatic miscue detection using RNN based models with data augmentation. In *Proc. Interspeech 2018*. 1646-1650.
- Jun, S.-A. (1993) The phonetics and phonology of Korean prosody, Ph. D. Dissertation, Ohio State University.
- Jun, J. (2015). Korean n-insertion: A mismatch between data and learning. *Phonology*, 32(3), 417-458. doi:10.1017/S0952675715000275
- Ko, E.-S. & Jun, J. (2022). Phonological variation in child-directed speech is modulated by lexical frequency. Retrieved from psyarxiv.com/wfqug
- Kim, B., Lee, G.G., & Lee, J.-H. (2002). Morpheme-based grapheme to phoneme conversion using phonetic patterns and morphophonemic connectivity information. *ACM Transactions on Asian Language Information Processing*, 1(1), 65-82.
- Lee, Younggun & Kim, Taesu. (2018). Learning pronunciation from a foreign language in speech synthesis network. arXiv preprint. arXiv:1811.09364.
- McKinney, Wes. (2011). Pandas: a foundational Python library for data analysis and statistics. In *SC Workshop on Python for High Performance and Scientific Computing*.
- Park, Kyubyong. (2019). G2pk. Available online: <https://github.com/Kyubyong/g2pK>

- (accessed on 08 November 2022).
- Park, Lucy. (2014). KoNLPy. Available online: <https://konlpy.org/ko/latest/index.html> (accessed on 08 November 2022).
- Patrick, P. L. (1991). Creoles at the intersection of variable processes: (TD) deletion and past marking in the Jamaican Mesolect. *Language Variation and Change* 3 (2): 171-189. <https://doi.org/10.1017/S095439450000051X>.
- Yoon, Kyuchul, & Brew, Chris. (2006). A linguistically motivated approach to graphemeto-phoneme conversion for Korean. *Computer Speech and Language*, 20 (4): 357-381.
- Wang, Yu-Chun, & Tsai, Richard Tzong-Han (2009). Rule-based Korean grapheme to phoneme conversion using sound patterns. 23rd Pacific Asia Conference on Language, Information and Computation, 843-850.

Received on: November, 15, 2022

Revised on: December, 14, 2022

Accepted on: December, 14, 2022