Rule-based Korean Grapheme to Phoneme Conversion Using Sound Patterns

Yu-Chun Wang[a] and Richard Tzong-Han Tsai[b]

[a] Department of Computer Science and Information Engineering, National Taiwan University, Taiwan
[b] Department of Computer Science and Engineering, Yuan Ze University, Taiwan

Abstract. Grapheme-to-phoneme conversion plays an important role in text-to-speech applications and other fields of computational linguistics. Although Korean uses a phonemic writing system, it must have a grapheme-to-phoneme conversion for speech synthesis because Korean writing system does not always reflect its actual pronunciations. This paper describes a grapheme-to-phoneme conversion method based on sound patterns to convert Korean text strings into phonemic representations. In the experiment with Korean news broadcasting evaluation set of 20 sentences, the accuracy of our system achieve as high as 98.70% on conversion. The performance of our rule-based system shows that the rule-based sound patterns are effective on Korean grapheme-to-phoneme conversion.

Keywords: sound pattern, grapheme-to-phoneme conversion, Korean

1 Introduction

In many fields of computational linguistics and natural language processing, grapheme-to-phoneme conversion is an important task. For example, in text-to-speech applications (van Santen *et al.*, 1997), the input text must be converted into phonemic representations that the speech synthesizer can use to generate correct and natural speech.

Our research in this area focus on Korean, which uses a phonemic alphabetical writing system. Although Korean writing system is phonemic, it is still limited by morphology. For instance, the sentence "못해" ([mo tʰɛ], I can't do it) is composed of two morphemes: "못" ([mos], can't) and "해" ([hɛ], do), not written in its actual pronunciation form "모태" ([mo] + [tʰɛ]). Korean written forms do not reflect their actual pronunciation. Therefore, we have to construct a system to convert the written forms into actual phonemic representation forms. Through our system, the phonemic representations of actual Korean pronunciation can be predicted correctly and then can be used in Korean text-to-speech applications to generation correct speech sounds.

Following the model of generative sound patterns proposed by (Chomsky and Halle, 1972), we analyze Korean phonology to reconstruct the sound patterns of Korean. (Wang, 1995) and (Kang, 2003) proposed the actual pronunciation and assimilation rules of Korean consonants and vowels. We follow the phonological rules he found and construct the Korean sound patterns. Next, we construct a rule-based system that applies these sound patterns to Hangul text strings to predict phonemic representations of pronunciation. We accept Korean text strings as input and generate the pronunciation by the sound patterns orderly. Then, the final conversion result is shown with International Phonetic Alphabet (IPA) symbols to perform evaluation or further application such as text-to-speech.

2 Related Work

In the previous work of grapheme-to-phoneme conversion, there are several approaches such as dictionary-based, rule-based, and statistical-rule-learning-based (Davel and Barnard, 2004). The

dictionary-based strategy (Zhang *et al.*, 2001) acquires a lot of phonological knowledge from a phonetic dictionary. It looks up each word in a dictionary and retrieve the corresponding pronunciation. The dictionary-based strategy suffers from out-of-vocabulary (OOV) problems severely because it is impossible to account for all novel words.

On the contrary, the rule-based strategy (Divay and Vitale, 1997) only requires a smaller set of rules that describe how to convert a grapheme into one or several phonemes. The rules can be applied to all of words without OOV problems. However, for some languages with complex writing systems such as Chinese and Japanese, building the rules is labor-intensive and very difficult to cover most possible situations.

The statistical-rule-learning-based approaches (Zhang and Chu, 2002) try to generate a rule set that can determine the phonemes automatically by statistical machine learning models, such as decision tree (DT) models or maximum entropy (ME) models (Guiasu and Shenitzer, 1985). Statistical-rule-learning-based methods can achieve satisfiable performance and also apply to many similar natural language processing tasks such as part-of-speech tagging (Ratnaparkhi, 1996) and named entity recognition (Tsai *et al.*, 2004). However, statistical-rule-learning-based strategy requires a sufficient large human-tagged corpus as a training set to have an accurate rule set.

Since Korean uses a phonemic writing system, the rule-based grapheme-to-phoneme conversion is not so difficult as Chinese ones. Therefore, we adopt rule-based approach. The advantages of rule-based grapheme-to-phoneme conversion are simple to implement, low efforts of human tagging, low memory consumption and efficient in computation. It is suitable to work on small embedded devices such as mobile phones, GPS navigation devices, and portable media players. It makes the interaction between humans and devices much more naturally.

3 Method

In this section, we describe the Korean sound patterns and the construction of the rule-based system.

3.1 Korean Sound Patterns

In the following, we observe the phonological phenomenon in Korean. Furthermore, we will construct the specific sound patterns to describe the phonological rules of Korean.

3.1.1 Word Final Neutralization All the Korean consonants are distinctive in onset position, except ㅇ [ŋ]. However, there are only seven consonants can appear in the syllable coda position, such as [p, t, k, l, m, n, ŋ]. The stops are neutralized as homorganic stops, and fricatives and affricates as a stop [t]. The nasal and lateral consonants remain the same. The following is an example.

$$
\begin{array}{llll}
\text{앞 [ap}^\text{h}\text{]} & \rightarrow & \text{[ap}^\text{˺}\text{]} & \text{'front'} \\
\text{낮 [natɕ]} & \rightarrow & \text{[nat}^\text{˺}\text{]} & \text{'daytime'} \\
\text{벗 [pəs]} & \rightarrow & \text{[pət}^\text{˺}\text{]} & \text{'friend'}
\end{array}
$$

Therefore, we can construct a sound pattern for word final neutralization in Korean.

$$
\begin{bmatrix} +\text{consonantal} \\ +\text{obstruent} \end{bmatrix} \rightarrow \begin{bmatrix} -\text{aspirated} \\ -\text{voiced} \\ -\text{glottal constriction} \\ -\text{strident} \end{bmatrix} / \_\#\#
$$

In Korean writing system, Hangul, it might have double consonants in coda position. For example, the character 값 (kaps) has double consonants ㅄ (ps) in the coda position. However,

Korean restricts syllables to one final consonant in pronunciation. Therefore, one of the double consonants in coda position is deleted in pronunciation by the word final neutralization rule. For the above example, the actual pronunciation of the character 값 is "kap", not "kaps". We follow the Hangul Orthography Rules to determine which one of double consonants should be deleted.

3.1.2 Liaison In a Korean word, when the last consonant of a syllable is followed by a vowel, the consonant is moved to the first sound of the next syllable. The liaison rule is also applied to loanwords from Chinese, as known as Sino-Korean, although it does not happen in Chinese. The following is an example of liaison.

$$
\begin{array}{llll}
삼일 \text{ [sam il]} & \rightarrow & \text{[sa mil]} & \text{'three days'} \\
먹어 \text{ [mək}^\urcorner\text{ə]} & \rightarrow & \text{[mə gə]} & \text{'eat (impolite)'}
\end{array}
$$

For double final consonants, the two consonants all remain and the last consonant move to the next syllable as onset consonant.

$$
\begin{array}{llll}
앉아 \text{ [anʨ a]} & \rightarrow & \text{[an ʨa]} & \text{'sit'} \\
읽어 \text{ [ilk ə]} & \rightarrow & \text{[il kə]} & \text{'read'}
\end{array}
$$

The pattern of liaison rule is as follow.

$$
\begin{bmatrix} +\text{consonantal} \\ +\text{coda position} \end{bmatrix} \rightarrow \begin{bmatrix} -\text{coda position} \\ +\text{onset position} \end{bmatrix} / \_\_ \#V
$$

3.1.3 Nasalization An obstruent preceding a nasal become its homorganic nasal sound, such as [p] becomes [m], [t] becomes [n], and [k] becomes [ŋ]. Korean nasalization is a complete assimilation, not a secondary articulation. The following shows some examples.

$$
\begin{array}{llll}
앞만 \text{ [ap}^h\text{ man]} & \rightarrow & \text{[am man]} & \text{'front only'} \\
밖만 \text{ [pak̚ man]} & \rightarrow & \text{[paŋ man]} & \text{'outside only'} \\
낮만 \text{ [naʨ man]} & \rightarrow & \text{[nan man]} & \text{'daytime only'}
\end{array}
$$

The sound pattern of nasalization is shown as follow.

$$
\begin{bmatrix} +\text{consonantal} \\ +\text{obstruent} \end{bmatrix} \rightarrow \begin{bmatrix} +\text{nasal} \end{bmatrix} / \_\_ \begin{bmatrix} +\text{consonantal} \\ +\text{nasal} \end{bmatrix}
$$

3.1.4 Palatalization The [t] and [t^h] are palatalized when they appear in the syllable coda position preceding a postposition or verb stem which begin with the vowel [i] or the syllable [hi]. The following is some examples.

$$
\begin{array}{llll}
맏이 & \text{[mat i]} & \rightarrow & \text{[ma ʨi]} & \text{'eldest child'} \\
같이 & \text{[kat}^h\text{ i]} & \rightarrow & \text{[ka ʨ}^h\text{i]} & \text{'together'} \\
굳히다 & \text{[kut hi ta]} & \rightarrow & \text{[ku ʨ}^h\text{i da]} & \text{'strengthen'}
\end{array}
$$

The pattern of palatalization is as follow.

$$
\begin{bmatrix} +\text{consonantal} \\ +\text{obstruent} \\ +\text{dental} \end{bmatrix} \rightarrow \begin{bmatrix} +\text{palatal} \end{bmatrix} / \_\_ \# \begin{Bmatrix} \text{[i]} \\ \text{[hi]} \end{Bmatrix}
$$

3.1.5 **Lateralization** When a nasal sound [n] is followed by a lateral [l] or a lateral [l] is followed by a nasal [n], the nasal [n] becomes the lateral [l]. The following shows some examples of lateralization.

반란 [pan lan]　　→　[pal lan]　　'rebellion'
칠년 [tɕʰil njən]　→　[tɕʰil ljən]　'seven years'
신라 [sin la]　　　→　[sil la]　　　'Silla Kingdom'

The sound patterns of lateralization are as follow.

$$
\begin{bmatrix} +\text{consonantal} \\ +\text{nasal} \\ +\text{continuant} \\ +\text{dental} \end{bmatrix} \rightarrow \begin{bmatrix} -\text{nasal} \\ +\text{lateral} \end{bmatrix} / \left\{ \begin{matrix} \underline{\quad} \begin{bmatrix} +\text{consonantal} \\ +\text{lateral} \end{bmatrix} \\ \begin{bmatrix} +\text{consonantal} \\ +\text{lateral} \end{bmatrix} \underline{\quad} \end{matrix} \right\}
$$

3.1.6 **/n/-Insertion** In a compound word, if the last consonant of the previous word is followed by a [i] or [j] of the next word, /n/ is inserted between the two words. The following shows some examples.

한여름　[han jə rɯm]　→　[han njə rɯm]　'midsummer'
꽃잎　　[k̚otɕʰ ipʰ]　　→　[k̚on nip̚]　　'petal'
댓잎　　[tɛs ipʰ]　　　→　[tɛn nip̚]　　'bamboo's leaf'

The sound pattern of /n/-insert is shown as follow.

$$
\emptyset \rightarrow [\text{n}]/\#\underline{\quad}
$$

3.1.7 **Aspiration** Korean lenis stops and affricates such as [p, t, k, tɕ] are aspirated when they are in syllable coda position and followed by the voiceless glottal fricative [h] or they follow the voiceless glottal fricative [h] which is in syllable coda position. Besides, the fricative [h] is deleted and the lenis stops or affricates move to the onset position of the next syllable. The following is a example.

급하다　[kɯp ha ta]　→　[kɯ pʰa da]　'hurry'
악하다　[ak ha ta]　　→　[a kʰa da]　　'evil'
어떻게　[ə t̚əh ke]　　→　[ə t̚ə kʰe]　　'how'

The sound pattern of aspiration is as follow.

$$
\begin{bmatrix} +\text{consonantal} \\ +\text{obstruent} \end{bmatrix} \rightarrow [+\text{aspirated}] / \left\{ \begin{matrix} \underline{\quad}[\text{h}] \\ [\text{h}]\underline{\quad} \end{matrix} \right\}
$$

$$
[\text{h}] \rightarrow \emptyset / \left\{ \begin{matrix} \underline{\quad} \begin{bmatrix} +\text{consonantal} \\ +\text{obstruent} \end{bmatrix} \\ \begin{bmatrix} +\text{consonantal} \\ +\text{obstruent} \end{bmatrix} \underline{\quad} \end{matrix} \right\}
$$

3.1.8 **Fortis** When the lenis obstruent consonants [p, t, k, s, tɕ] follow other obstruent consonants, they will become fortis consonants [p̚, t̚, k̚, s̚, tɕ̚]. Some examples are shown in the following.

백골　[pɛk kol]　→　[pɛk k̚ol]　'bone'
익사　[ik sa]　　→　[ik s̚a]　　'drown'
먹다　[mək ta]　→　[mək t̚a]　'eat'

The sound pattern of fortis is as follow.

$$
\begin{bmatrix} +\text{consonantal} \\ +\text{obstruent} \end{bmatrix} \rightarrow [+\text{fortis}] / \begin{bmatrix} +\text{consonantal} \\ +\text{obstruent} \end{bmatrix} \underline{\quad}
$$

3.2 Sound Pattern Order

In the above section, we described the sound patterns used to generate actual Korean pronunciation. In practice, these sound patterns are not applied simultaneously. Since different application order generates different pronunciation, we have to apply the patterns in the correct order. Table 1 shows the order of the sound patterns described in Section 3.1.

Table 1: Order of Sound Patterns

Precedence	Rule
1	/n/-Insertion
2	Palatalization
3	Liaison
4	Lateralization
5	Word Final Neutralization
6	Nasalization
7	Aspiration
8	Fortis

3.3 Implementation of Rule-based System

After analyzing the Korean sound patterns, we then construct a rule-based system to predict the actual Korean pronunciation. The overview of our system architecture is shown in Figure 1. The system comprises four stages: word segmentation, phoneme extraction, sound pattern processing, and IPA presentation. We have used the Java programming language and Java Virtual Machine to implement the system and the test environment.

3.3.1 Korean Word Segmentation In Hangul, there are no explicit word boundaries. However, there are spaces in the sentence to separate "eojeols," which are composed of a noun and postposition or a verb stem and ending. Sound assimilation seldom occurs across several eojeols in Korean. Therefore, we separate Korean sentences into several eojeols using space characters.

For the /n/-insertion pattern described in Section 3.1.6, we must further segment compound words to check if they match the /n/-insertion pattern or not. We use the maximal matching algorithm (MMA) with the MinJungSeoRim Korean dictionary to perform further segmentation as follows: First, we check to see if the string is in the dictionary. If it is, then it is a word and the algorithm stops. Otherwise, we discard the last syllabic block to see if the string minus this block exists or not. The algorithm repeats this step until it finds a word or reduced the string to one block.

3.3.2 Romanization and Phoneme Extraction Hangul combines two, or more often three, letters into syllabic blocks. When processing strings of these blocks, we first convert them into Roman characters using an online romanization tool[1]. For example, the character 넓, which is composed of four Hangul letters ㄴ (n), ㅓ(eo), ㄹ (r), and ㅂ (b), will be transcribed into "neorb". This romanization is based on the Revised Korean Romanization standard promoted by the South Korean Ministry of Culture.

Next, we extract each phoneme from the romanized string. Some monosyllabic vowels are transcribed into digraphs with two Roman letters, like "eo" (ə), "ae" (ɛ), and "eu" (ɯ). We have to identify them as single phonemes not diphthongs. In the design of Revised Korean Romanization, most diphthongs begin with "y" and "w" except "ㅢ" (ui). Therefore, we can distinguish the digraphs and diphthongs easily. We also record whether each phoneme is in the initial, medial, or final position in the phoneme data structure because some sound patterns are based on the position of the phoneme in the syllable.
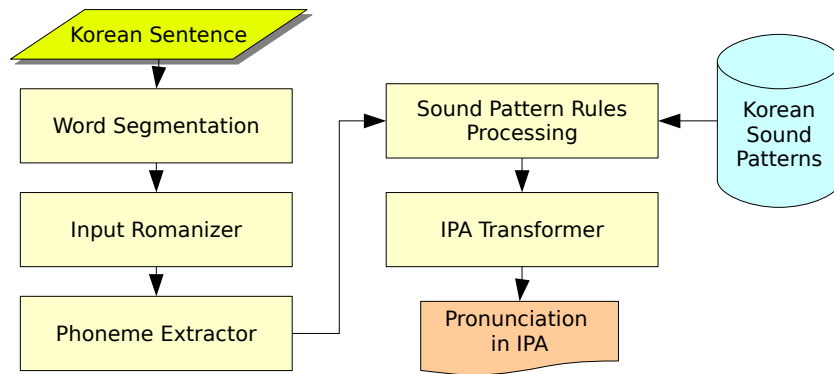
---

[1] http://www.kawa.net/works/ajax/romanize/hangul-e.html

Figure 1: System Architecture

3.3.3 Sound Pattern Rules Processing In this stage, we apply the Korean sound-pattern rules to the extracted phoneme sequences to predict the pronunciation following the application order defined in Section 3.2. We have also designed data structures for phonemes and sound patterns to make them computable in these sequential procedures.

3.3.4 IPA Transformer The system's output is converted into IPA symbols using a table that maps all romanized phoneme representations into the IPA symbols to make it readable.

4 Evaluation

4.1 Evaluation Set

In order to evaluate our system, we use a test set made up of recordings of actual Korean speech, specifically news broadcast video files from the Korean Broadcasting System's (KBS) website[2] dating from December 12, 2007 to January 3, 2008. We use the broadcast transcripts as input data. In order to reduce computational complexity, we randomly choose one or two sentences from each transcript to evaluate. And to avoid inconsistencies caused by regional Korean accents, we only use broadcasts by anchorpersons from Seoul. We then have a Korean linguist transcribe the broadcaster's speech into Korean phoneme representation forms, and we then convert them into IPA symbol strings as broad transcriptions. Table 2 shows the statistics of our evaluation set.

4.2 Evaluation Method and Result

We adopt string edit distance method to evaluate the results of our system. We compare the result strings of our system and the human transcriptions based on the IPA symbol characters standing for Korean phonemes. String edit distance has three kinds of errors: *insertion*, *deletion*, and *substitution*. Insertion indicates the result string inserts one more character than the transcription string; deletion stands for the result string lacks one character, and substitution means the result string changes one character of the transcription string. String edit distance can be implemented by the dynamic programming technique to perform the comparison efficiently. Moreover, we also utilize *I-score* (Kim *et al.*, 2002) to calculate the accuracy. *I-score* is defined as

$$I\text{-}score = \frac{c}{c+i+d+s},$$

where c, i, d, and s denote the number of correct phonemes, insertion errors, deletion errors, and substitution errors respectively. The evaluation results are shown in Table 3.

---

[2] http://news.kbs.co.kr

Table 2: Statistics of Evaluation Set

	Count
Sentences	20
Words	346
Phonemes	615

5 Discussion

The evaluation shows that our system achieves as high as 98.70% accuracy to convert Korean graphemes into phonemes. In the following, we discuss the effectiveness of our system and analyze the error cases.

5.1 Effectiveness of Sound Pattern Rules

Through the evaluation, we find that some sound patterns are robust. These patterns are applied all the times in our evaluation set such as nasalization, palatalization, and lateralization. The results shows that these assimilations always happen in Korean speaking.

Table 3: Convertion Results

	Count
Correct sentences	15
Correct phonemes	606
Insertion errors	5
Deletion errors	1
Substitution errors	2
I-score	**0.9870**

5.2 Error Cases

5.2.1 Sound Losses Sound losses cause the most insertion errors. The vowel [ɯ] and the semivowel [j] are sometimes omitted when they do not appear in first syllables. In our evaluation set, the vowel [ɯ] in the words 편의점 ([pʰjən **ɯ**i tɕəm], "convenient store") and 용의자 ([joŋ **ɯ**i tɕa], "suspect") are deleted and pronounced as [pʰjə ni tɕəm] and [joŋ i tɕa] by the anchorpersons. The semivowel [j] is also deleted in the word 중계차 ([tɕuŋ k**j**e tɕʰa], "SNG mobile car") as [tɕuŋ ke tɕʰa].

The lost of the whole syllable also happens in our evaluation. In the eojeol 기능은 ([ki nɯŋ ɯn], "function" + nominative postpostion), the last syllable [ɯn] is totally lost in the speaking.

The consonant [h] may be omitted when it follows a nasal [n]. For example, the word 문화관 광부 ([mu**n h**wa kwan kwaŋ pu], "Ministry of Culture and Tourism") is pronounced as [mu **n**wa kwan kwaŋ pu]. However, there is a counterexample. The consonant [h] which follows a nasal [n] in the sentence 전해주시죠 ([tɕən hɛ tɕu si tɕjo], "tell me") is not deleted.

Sound loss seems not so regular. It depends on the preference or the speaking speed of the speakers and varies irregularly. In order to overcome sound loss problem, we may have to adopt a probabilistic model to learn the variations from the corpora to determine sound loss should happen or not.

5.2.2 Sound Patterns across Eojeols In our system, sentences are split into several eojeols and then the sound patterns are applied to these eojeols separately. However, we find that the assimilations may across eojeols in some cases. For example, in the sentence "선택, 하셨는지 요?" ([sən tʰɛk, **h**a sjən nɯn tɕi jo], "Did you make a choice?"), the aspiration rule is applied among the second and the third syllable to make the actual pronunciation as [sən tʰɛ, **kʰ**a sjən nɯn tɕi jo] even though there is a comma to separate them. To deal with this problem, we might have to merge a Korean shallow parser to analyze the structure of the sentences more detailedly.

6 Conclusion

In this paper, we have constructed a rule-based Korean grapheme-to-phoneme conversion system based on sound patterns. Korean writing system is phonemic, but still limited by morphology. It causes the graphemes can not reflect their actual pronunciation. Therefore, it is necessary to build a grapheme-to-phoneme conversion system to convert the texts into their phonemic transcription of actual pronunciation for some research fields like text-to-speech. We follow the analysis of Korean phonology to make eight sound patterns to describe the process of the conversion from grpheme to phoneme. Moreover, we define the order of the sound patterns to generate the correct phonemes. In order to evaluate the system, we build an evaluation dataset by collecting the news broadcasting videos with their Korean text transcription, and the phonemic transcriptions from the speaking of news videos by a Korean linguist.

The evaluation results show that the conversion accuracy of our system is up to 98.70%. It proves that a rule-based system is effective for Korean grapheme-to-phoneme conversion problem. The error cases can be divided into two categories. One is the loss of the sounds in the speaking. The other is the application of sound patterns across several eojeols.

In the future, we will try to compare our system with other kinds of conversion systems such as the conversion systems based on machine learning models or phonetic pattern dictionaries. Besides, we will adopt more natural language processing tools, like morphological analyzers, part-of-speech taggers, and parsers into the processing stages of our system. It will make the conversion system have more abilities to deal with the cases involving morphology and semantics.

References

Chomsky, N. and M. Halle. 1968. The sound pattern of English. Harper and Row.

Davel, M. and E. Barnard. 2004. A default-and-refinement approach to pronunciation prediction. *The 15th Annual Symposium of the Pattern Recognition Association of South Africa.*

Divay, M., and A. J. Vitale. 1997. Algorithms for grapheme phoneme translation for English and French: Applications for database searches and speech synthesis. *Computational Linguistics*, 23(4), 495-523.

Guiasu, S., and A. Shenitzer. 1985. The principle of maximum entropy. *The Mathematical Intelligencer*, 7.

Kang, O.-M. 2003. *Korean Phonology*. Tae Hak Sa.

Kim, B., G.G. Lee and J.-H. Lee. 2002. Morpheme-Based Grapheme to Phoneme Conversion Using Phonetic Patterns and Morphophonemic Connectivity Information. *ACM Transactions on Asian Lnguage Information Processing*, 1(1), 65-82.

Ratnaparkhi, A. 1996. A Maximum Entropy Model for Part-Of-Speech Tagging. *EMNLP-96.*

Tsai, R. T.-H., S.-H. Wu and W.-L. Hsu. 2004. Mencius: A Chinese Named Entity Recognizer Based on a Maximum Entropy Framework. *Computational Linguistics and Chinese Language Processing*, 9(1), 65-82.

van Santen, J.P., R.W. Sproat, J.P. Olive and J. Hirschberg. 1997. *Progress in Speech Synthesis.* Springer-Verlag.

Wang, C. 1995. *Korean Phonetics.* Buffalo Publishing.

Zhang, Z. and M. Chu. 2002. A Statistical Approach for Grapheme-to-Phoneme Conversion in Chinese. *Journal of Chinese Information Processing*, 16, 39-45.

Zhang, H., J. Yu, W. Zhan and S. Yu. 2001. Disambiguation of Chinese Polyphonic Characters. *The First International Workshop on MultiMedia Annotation (MMA2001).*