# Syllable-Pattern-Based Unknown-Morpheme Segmentation and Estimation for Hybrid Part-of-Speech Tagging of Korean

Gary Geunbae Lee*
Pohang University of Science and Technology

Jeongwon Cha†
Pohang University of Science and Technology

Jong-Hyeok Lee‡
Pohang University of Science and Technology

*Most errors in Korean morphological analysis and part-of-speech (POS) tagging are caused by unknown morphemes. This paper presents a syllable-pattern-based generalized unknown-morpheme-estimation method with POSTAG (POStech TAGger),*[1] *which is a statistical and rule-based hybrid POS tagging system. This method of guessing unknown morphemes is based on a combination of a morpheme pattern dictionary that encodes general lexical patterns of Korean morphemes with a posteriori syllable trigram estimation. The syllable trigrams help to calculate lexical probabilities of the unknown morphemes and are utilized to search for the best tagging result. This method can guess the POS tags of unknown morphemes regardless of their numbers and/or positions in an* eojeol *(a Korean spacing unit similar to an English word), which is not possible with other systems for tagging Korean. In a series of experiments using three different domain corpora, the system achieved a 97% tagging accuracy even though 10% of the morphemes in the test corpora were unknown. It also achieved very high coverage and accuracy of estimation for all classes of unknown morphemes.*

## 1. Introduction

Part-of-speech (POS) tagging involves many difficult problems, such as insufficient amounts of training data, inherent POS ambiguities, and (most seriously) many types of unknown words. Unknown words are ubiquitous in any application and cause major tagging failures in many cases. Since Korean is an agglutinative language, it presents more serious problems with unknown morphemes than with unknown words because more than one morpheme can be unknown in a single word and morpheme segmentation is usually very difficult.

* NLP Laboratory, Electrical and Computer Engineering Division, Pohang University of Science and Technology (POSTECH), Pohang, 790-784, Korea. E-mail: gblee@postech.ac.kr.
† NLP Laboratory, Electrical and Computer Engineering Division, Pohang University of Science and Technology (POSTECH), Pohang, 790-784, Korea. E-mail: himen@postech.ac.kr.
‡ NLP Laboratory, Electrical and Computer Engineering Division, Pohang University of Science and Technology (POSTECH), Pohang, 790-784, Korea. E-mail: jhlee@postech.ac.kr.

[1] The binary code of POSTAG is open to the public for research and evaluation purposes at http://nlp.postech.ac.kr/. Follow the link OpenResources→DownLoad.

Previous techniques for guessing unknown words mostly utilize the guessing rules to analyze the word features by looking at leading and trailing characters. Most of them employ the analysis of trailing characters and other features such as capitalization and hyphenation (Kupiec 1992; Weischedel et al. 1993). Some of them use more morphologically oriented word features such as suffixes, prefixes, and character lengths (Brill 1995; Voutilainen 1995). The guessing rules are usually handcrafted using knowledge of morphology but sometimes are acquired automatically using lexicons and corpora (Brill 1995; Mikheev 1996; Oflazer and Tür 1996). Previously developed methods for guessing unknown morphemes in Korean are not much different from the methods used for English. Basically, they rely on the rules that reflect knowledge of Korean morphology and word formation. The usual way of handling unknown morphemes is to guess all the possible POS tags for an unknown morpheme by checking connectable functional morphemes in the same *eojeol* (Kang 1993).[2] However, in this way, it is only possible to guess probable POS tags for a single unknown morpheme when it occurs at the beginning of an *eojeol*. Unlike in English, in Korean, more than one unknown morpheme can appear in a single *eojeol* because an *eojeol* can include complex components such as Chinese characters, Japanese words, and other foreign words. If an *eojeol* contains more than one unknown morpheme or if the unknown morphemes appear in other than first position in the *eojeol*, all previous methods fail to efficiently estimate them. This is the reason why we try to avoid conventional guessing rules using word morphology features such as those proposed in Mikheev (1996) and Oflazer and Tür (1996).[3]

In this paper, we propose a syllable-pattern-based generalized unknown-morpheme estimation method using a morpheme pattern dictionary that enables us to treat unknown morphemes in the same way as registered known morphemes, and thereby to guess them regardless of their numbers or positions in an *eojeol*. The method for estimating unknown morphemes using the morpheme pattern dictionary in Korean needs to be tightly integrated into morphological analysis and POS disambiguation systems.

POS disambiguation has usually been performed by statistical approaches, mainly using the hidden Markov model (HMM) in English research communities (Cutting et al. 1992; Kupiec 1992; Weischedel et al. 1993). These approaches are also dominant for Korean, with slight improvements to accommodate the agglutinative nature of Korean. For Korean, early HMM tagging was based on *eojeols*. The *eojeol*-based tagging model calculates lexical and transition probabilities with *eojeols* as a unit; it suffers from severe data sparseness problems since a single *eojeol* consists of many different morphemes (Lee, Choi, and Kim 1993). Later, morpheme-based HMM tagging was tried; such models assign a single tag to a morpheme regardless of the space in a sentence. Morpheme-based tagging can reduce data sparseness problems but incurs multiple observation sequences in Viterbi decoding since an *eojeol* can be segmented in many different ways. Researchers then tried many ways of reducing computation due to multiple observation sequences, such as shared word sequences and virtual words (Kim, Lim, and Seo 1995) and two-ply HMM for morpheme unit computation but restricted within an *eojeol* (Kim, Im, and Im 1996). However, since statistical approaches take neighboring tags into account only within a limited win-

[2] An *eojeol* is a Korean spacing unit (similar to an English word), which usually consists of one or more stem morphemes and a series of functional morphemes.

[3] Even though Turkish and Finnish are in the same class of agglutinative languages and German also has very complex morphological structures, in our view word formation is more diverse and complex in Korean than in these Western languages because of its mix of Oriental and Western culture.

dow (usually two or three), sometimes the decision fails to cover important linguistic contexts necessary for POS disambiguation. Also, approaches using only statistical methods are inappropriate for idiomatic expressions, for which lexical terms need to be directly referenced. And especially, statistical approaches alone do not suffice for agglutinative languages, which usually have complex morphological structures. In agglutinative languages, a word usually consists of one or more stem morphemes plus a series of functional morphemes; therefore, each morpheme should receive a POS tag appropriate to its functional role to cope with the complex morphological phenomena in such languages. Recently, rule-based approaches, which learn symbolic tagging rules automatically from a corpus, have been reconsidered, to overcome the limitations of statistical approaches (Brill 1995). Some systems even perform POS tagging as part of a syntactic analysis process (Voutilainen 1995). Following the success of transformation-based approaches, attempts have been made to use transformation rules in systems for tagging Korean (Im, Kim, and Im 1996). However, in general, rule-based approaches alone are not very robust and are not portable enough to be adjusted to new tagsets or new languages. Also, they usually perform no better than their statistical counterparts (Brill 1995). To gain portability and robustness and also to overcome the limited coverage of statistical approaches, we need to somehow combine the two approaches to gain the advantages of each. In this paper, we propose a hybrid method that combines statistical and rule-based approaches to POS disambiguation and can be tightly coupled with generalized unknown-morpheme-guessing techniques.

## 2. Linguistic Characteristics of Korean

Korean is classified as an agglutinative language. In Korean, an *eojeol* consists of several morphemes that have clear-cut morpheme boundaries. For example, *na-neun gam-gi-e geol-lyeoss-dda* 'I caught a cold' consists of 3 *eojeols* and 7 morphemes:[4] *na*('I')/T + *neun*('auxiliary particle')/jS, *gam-gi*('cold')/MC + *e*('adverb and conjunctive particle')/jO, *geol-li*('catch')/DR + *eoss*('past tense')/eGS + *dda*('final ending')/eGE. Below are the characteristics of Korean that must be considered for morphological-level natural language processing and POS tagging.

- POS tagging of Korean is usually performed on a morpheme basis rather than on an *eojeol* basis. Accordingly, morphological analysis is essential to POS tagging because morpheme segmentation is much more important and difficult than POS assignment. Moreover, morphological analysis should segment *eojeols* that contain unknown morphemes as well as known morphemes. Hence, unknown-morpheme handling should be integrated into the morphological analysis process. Because a single *eojeol* can have many possible analyses (e.g., *na-neun*: *na*('I')/T + *neun*('topic marker')/jS, *na*('sprout')/DR + *neun*('adnominal')/eCNMG, *nal*('fly')/DI + *neun*('adnominal')/eCNMG, morpheme segmentation is inherently ambiguous.

- Korean is a postpositional language with many kinds of noun endings (particles), verb endings, and prefinal verb endings. It is these functional morphemes, rather than the order of *eojeols*, that determine grammatical

[4] Here, "+" represents a morpheme boundary in an *eojeol* and "/" introduces the POS tag symbols (see Table 2).

**Table 1**
Sample distribution of unknown morphemes in Korean.

| Tag | # morphemes | Tag | # morphemes |
|---|---|---|---|
| MC | 2,888 (29.7%) | S | 1,358 (14.0%) |
| MPN | 650 (6.7%) | B | 603 (6.2%) |
| MPP | 235 (2.4%) | T | 50 (0.5%) |
| MPC | 56 (0.6%) | Symbol | 10 (0.1%) |
| MPO | 728 (7.5%) | Foreign word | 3,140 (32.3%) |

relations such as a noun's syntactic function, a verb's tense, aspect, modals, and even modifying relations between *eojeols*. For example, *ga*/jC is a case particle, so the *eojeol uri(we)-ga* has a subject role due to the particle *ga*/jC. Korean has a clear syllable structure within the morpheme; most nominal content morphemes keep their surface form when they are combined with functional morphemes.

- Korean is basically an SOV language but has relatively free word order compared with English. The weight $\alpha, \beta$ in Equation (1) (Section 4.1) reflects the fact that transition probability is less important in Korean than in English. However, Korean does have some word order constraints: verbs must appear in sentence-final position, and modifiers must be placed before the element they modify. So some order constraints must be selectively utilized as contextual information in the POS tagging process, which is taken well into account in the design of error correction rules (Section 4.3).
- Complex spelling changes (irregular conjugations) frequently occur between morphemes when two morphemes combine to form an *eojeol*. These spelling changes make it difficult to segment the original morphemes before the POS tag symbols are assigned.
- The unknown-morpheme problem in Korean differs in some ways from the unknown-word problem in English. In English, it is easy to identify unknown words because they occur between spaces. However, in Korean, since unknown morphemes are hidden in an *eojeol*, we only know that morphological analysis failed in that *eojeol*; pinpointing the exact unknown morphemes is usually difficult. This is why, unlike in English, it is not possible to fully guess an unknown morpheme using only affixes. The distribution of POS tags for unknown morphemes extracted from a 130,000-morpheme training corpus (9,718 unknown morphemes) is shown in Table 1. The distribution from even a small corpus shows that we need to estimate various parts of speech for unknown morphemes rather than simply guess them as nouns.

Table 2 shows the tagset that was used in the experiments reported in Section 5. The tagset was selected from hierarchically organized POS tags for Korean. We defined about 100 different POS tags, which can be used in morphological analysis as well as in POS tagging. We also designed over 300 morphotactic adjacency symbols to be used in morpheme connectivity checks for correct morpheme segmentation (to be explained in the next section). The POS tags are hierarchically organized symbols

**Table 2**
A tagset with 41 tags.

| Major category | Tag | Description |
|---|---|---|
| Nominal | MC | common noun |
| | MPN | person name |
| | MPC | country name |
| | MPP | place name |
| | MPO | other proper noun |
| | MD | bound noun |
| | T | pronoun |
| | S | numeral |
| Predicate | DR | regular verb |
| | DI | irregular verb |
| | HR | regular adjective |
| | HI | irregular adjective |
| | I | i-predicative particle |
| | E | existential predicate |
| | b | auxiliary verb |
| Modifier | G | adnoun |
| | B | adverb |
| Particle | y | predicative particle |
| | jC | case particle |
| | jS | auxiliary particle |
| | jO | adverb and conjunctive particle |
| Ending | eGE | final ending |
| | eGS | prefinal ending |
| | eCNDI | aux conj ending |
| | eCNDC | quote conj ending |
| | eCNMM | nominal ending |
| | eCNMG | adnominal ending |
| | eCNB | adverbial ending |
| | eCC | conjunctive ending |
| Affix | + | prefix |
| | − | suffix |
| Special symbol | su | unit symbol |
| | s‘ | left parenthesis |
| | s’ | right parenthesis |
| | s. | sentence closer |
| | s- | sentence connection |
| | s, | sentence comma |
| | sf | foreign word |
| | sh | Chinese character |
| | so | other symbol |
| Interjection | K | interjection |

that were iteratively refined from the eight major grammatical categories of Korean: nominal, predicate, modifier, particle, ending, affix, special symbol, and interjection. For a given morpheme, the acronym of a path name in the symbol hierarchy up to a certain level is assigned as a POS tag.[5] The rest of the detailed hierarchies, which are related only to morpheme connectivity, are independently assigned as morphotactic adjacency symbols. Therefore, we can use either full or partial path names as POS tags in order to adjust the total number of tags. The size of the tagset can thus be adapted by refining grammatical categories that are more pertinent to a given application. For example, for text-indexing applications, we refine nominals more than predicates since index terms are usually nominals in these applications.

## 3. Unknown-Morpheme Segmentation during Morphological Analysis

The agglutinative nature of Korean inevitably requires doing morphological analysis before POS tagging. Morphological analysis, which segments input texts into morphotactically connectable morphemes and assigns all possible POS tags to each morpheme by looking them up in a morpheme dictionary, is a basic step in natural language processing.

Our morphological analysis follows three general steps (Sproat 1992): morpheme segmentation, recovering original morphemes from spelling changes, and morphotactic modeling. Input texts are scanned from left to right, character by character,[6] to be matched with morphemes in a morpheme dictionary. The morpheme dictionary has a trie structured index for fast matching. It also has an independent entry for each variant surface form (called allomorph) of the original morpheme so the original morphemes can easily be reconstructed from spelling changes (see Table 3). For morphotactic modeling, we used the POS tags and the morphotactic adjacency symbols in the dictionary. The POS tags provide information about morpheme class, while the morphotactic adjacency symbols provide information about grammatical connectivity between morphemes needed to form an *eojeol*. The full hierarchy of POS tags and morphotactic adjacency symbols is encoded in the morpheme dictionary for each morpheme. Besides the morpheme dictionary, to model morphemes' connectability to one another the system uses an independent morpheme connectivity table that encodes all the connectable pairs of morpheme groups using the morphemes' tags and morphotactic adjacency symbol patterns. After an input *eojeol* is segmented by trie indexed dictionary searches, the morphological analysis checks whether each segmentation is grammatically connectable by looking in the morpheme connectivity table.

For unknown-morpheme segmentation, we developed a generalized method for estimating unknown morphemes regardless of their position and number. Using a morpheme pattern dictionary, our system can look up unknown morphemes exactly the same way it looks up known registered morphemes. The morpheme pattern dictionary covers all the necessary syllable patterns for unknown morphemes, including common nouns, proper nouns, adverbs, regular and irregular verbs, regular and irregular adjectives, and special symbols for foreign words. The lexical patterns for morphemes are collected from previous studies (Kang 1993) where the constraints on Korean syllable patterns regarding morpheme connectivity are well described. Table 4 shows some sample entries in the morpheme pattern dictionary, where $Z$, $V$, "*" are

[5] For example, nominal(M):proper-noun(P):person-name(N) is a three-level path name.

[6] The character sequence in *na-neun* is *n*, *a*, *n*, *eu*, *n*.

**Table 3**
Examples of morpheme dictionary entries. *MCC* is a full POS tag that identifies a common noun consisting of Chinese characters. *MCK* identifies a common noun consisting only of Korean characters. *DIgeo-la* represents a *geo-la* irregular verb, and *HIl* represents an *l* irregular adjective. *Yu* represents that *ga-gong* has a final consonant (*ng*). *D-ha*, *H-ha*, and *D-doe* are morphotactic adjacency symbols for predicate particles. Nominals that have a *D-ha* as a morphotactic adjacency symbol can be connected with predicate particles, and they play the role of a verb or adjective. In verb or adjective, *gyu* represents a regular form of an irregular conjugation, *bul* represents an irregular form of an irregular conjugation. *Eo* is a morphotactic adjacency symbol for vowel harmony when connecting with endings. *Chug-yag* represents that a particular verb (or adjective) contains the special contracted ending. ">" is a special symbol for adjacent direction (">"= right connection; "<"= left connection).

| POS-tag<original form> | (Allomorph) | [Morphotactic adjacency symbols] |
|---|---|---|
| MCC<ga-gong> | (ga-gong) | [yu>D-ha>H-ha>D-doe>] |
| MCK<geo-leum> | (geo-leum) | [yu>D-ha>] |
| DI*geo-la*<geon-neo-ga> | (geon-neo-ga) | [gyu>chug-yag>] |
| DI*d*<al-a-deud> | (al-a-deud) | [gyu>] |
| DI*d*<al-a-deud> | (al-a-deul) | [bul>eo>] |
| DI*s*<heu-li-jeos> | (heu-li-jeo) | [bul>eo>] |
| DI*s*<heu-li-jeos> | (heu-li-jeos) | [gyu>] |
| HI*l*<ga-neul> | (ga-neu) | [bul>] |
| HI*l*<ga-neul> | (ga-neul) | [gyu>eo>] |

**Table 4**
Sample entries in the morpheme pattern dictionary. Symbol meanings are explained in Table 3.

| POS-tag<original form> | (Allomorph) | [Morphotactic adjacency symbols] |
|---|---|---|
| HIl<ZV*gal> | (ZV*gal) | [gyu>eo>] |
| HIl<ZV*gal> | (ZV*ga) | [bul>] |
| HIb<ZV*ZVb> | (ZV*u) | [bul>] |
| HIb<ZV*ZVb> | (ZV*weo) | [chug-yag>] |
| HIb<ZV*ZVb> | (ZV*wa) | [chug-yag>] |
| DIs<ZV*jeos> | (ZV*jeos) | [gyu>] |
| DIs<ZV*jeos> | (ZV*jeo) | [bul>eo>] |
| DId<ZV*deud> | (ZV*deud) | [gyu>] |
| DId<ZV*deud> | (ZV*deul) | [bul>eo>] |

metacharacters that indicate a consonant, a vowel, and any number of Korean characters, respectively. For example, *go-ma-weo* 'thanks', which is a morpheme and an *eojeol* at the same time, is matched to (ZV*weo) (shown in Table 4, where it is *b*, irregular adjective (HI*b*)) in the morpheme pattern dictionary, which allows the system to recover its original morpheme form *go-ma**b***.

Once the unknown morphemes are identified and recovered using the pattern dictionary, when checking the unknown morphemes to see if they are connectable, the system can use the same information about adjacent morphemes in the unknown morphemes' *eojeol* that it would use if they were known morphemes. This is the reason why our method can be called "generalized" and can identify unknown morphemes regardless of their position and number in an *eojeol*. The actual POS estimation is integrated into the POS tagging process that will be described in Section 4.2. The essential

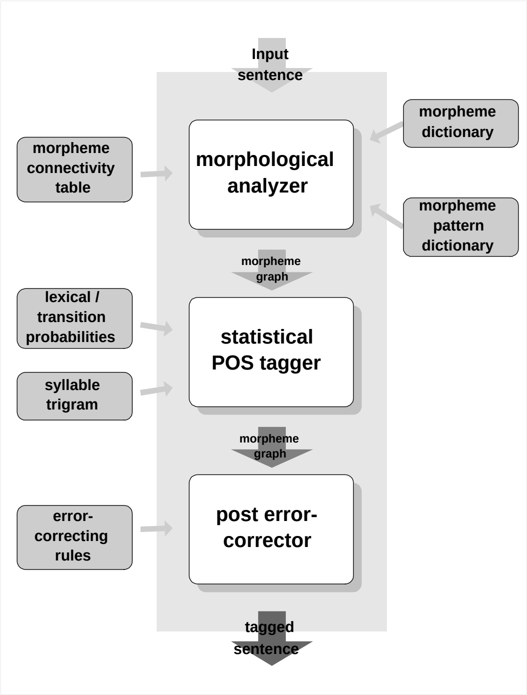

**Figure 1**
Statistical and rule-based hybrid architecture for POS tagging of Korean.

idea of the morpheme pattern dictionary is to pre-collect all the possible general lexical patterns of Korean morphemes and encode each lexical syllable pattern with all the candidate POS tags. Therefore, the system can assign initial POS tags to each unknown morpheme simply by matching the syllable patterns in the pattern dictionary. In this way, unlike previous approaches, ours does not need to incorporate a special rule-based unknown-morpheme-handling module into its morphological analyzer, and all the possible POS tags can be assigned to unknown morphemes just as they are to known morphemes.

## 4. A Statistical and Rule-Based Hybrid Tagging Model

Figure 1 shows a proposed hybrid architecture for POS tagging of Korean with syllable-pattern-based generalized unknown-morpheme guessing. It has three major components: the morphological analyzer with unknown-morpheme handler, the statistical POS tagger, and the rule-based error corrector. The morphological analyzer segments the morphemes from input *eojeols* and reconstructs the original morphemes from spelling changes by recovering the irregular conjugations. It also assigns all possible POS tags to each morpheme by consulting a morpheme dictionary. The unknown-morpheme handler, which is tightly integrated into the morphological analyzer, assigns initial POS tags to morphemes that are not registered in the dictionary, as explained in the previous section. The statistical POS tagger runs the Viterbi algorithm (Forney 1973) on the morpheme graph to search for the optimal tag sequence for POS disambiguation. To remedy the defects of a statistical POS tagger, we developed an a posteriori error correction mechanism. The error corrector is a rule-based transformer

(Brill 1995), and it corrects mistagged morphemes by consulting lexical patterns and necessary contextual information.

### 4.1 The Statistical POS Tagger

The statistical POS tagging model takes the morpheme graph (output of the morphological analyzer) and selects the best morpheme and POS tag sequence[7] for sentences represented in the morpheme graph. The morpheme graph is a compact way of representing multiple morpheme sequences for a sentence. Each morpheme's tag is a node in the graph and its morpheme connectivity is a link. Our statistical tagging model is modified from the standard bigrams (Cutting et al. 1992) using Viterbi search plus on-the-fly extra computing of lexical probabilities for unknown morphemes. The equation used for the statistical tagging model is a modified bigram model with left-to-right search,

$$T^* = argmax_T \prod_{i=1}^{n} Pr(t_i \mid t_{i-1})^{\alpha} \left( \frac{Pr(t_i \mid m_i)}{Pr(t_i)} \right)^{\beta} \tag{1}$$

where $T^*$ is an optimal tag sequence that maximizes the forward Viterbi scores. $Pr(t_i \mid t_{i-1})$ is a bigram tag transition probability and $\frac{Pr(t_i|m_i)}{Pr(t_i)}$ is a modified morpheme lexical probability. $\alpha$ and $\beta$ are weights and are set at 0.4 and 0.6, respectively, which means that lexical probability is more important than transition probability given the relatively free word order of Korean. This equation was finally selected after extensive experiments using the following six equations:

$$T^* = argmax_T \prod_{i=1}^{n} Pr(t_i \mid t_{i-1}) Pr(m_i \mid t_i) \tag{2}$$

$$T^* = argmax_T \prod_{i=1}^{n} Pr(t_i \mid t_{i-1})^{\alpha} Pr(m_i \mid t_i)^{\beta} \tag{3}$$

$$T^* = argmax_T \prod_{i=1}^{n} Pr(t_i \mid t_{i-1}) Pr(t_i \mid m_i) \tag{4}$$

$$T^* = argmax_T \prod_{i=1}^{n} Pr(t_i \mid t_{i-1})^{\alpha} Pr(t_i \mid m_i)^{\beta} \tag{5}$$

$$T^* = argmax_T \prod_{i=1}^{n} Pr(t_i \mid t_{i-1}) \frac{Pr(t_i \mid m_i)}{Pr(t_i)} \tag{6}$$

$$T^* = argmax_T \prod_{i=1}^{n} Pr(t_i \mid t_{i-1})^{\alpha} \left( \frac{Pr(t_i \mid m_i)}{Pr(t_i)} \right)^{\beta} \tag{7}$$

In the experiments, we used the 10,204-morpheme training corpus from the *Kemong Encyclopedia*.[8] Table 5 shows the tagging performance of each equation.

Training of the statistical tagging model requires a parameter estimation process for two different parameters, that is, morpheme lexical probabilities and bigram tag transition probabilities. Several studies show that using as much tagged material as

[7] Because a Korean *eojeol* can be segmented in many different ways, selecting the best morpheme segmentation sequence is as important as selecting the best POS sequence in POS tagging.

[8] Provided by the Electronics and Telecommunications Research Institute (ETRI).

**Table 5**
Tagging performance (all in %) of each equation. The "*eojeol*" row shows *eojeol*-unit tagging performance, and the "morpheme" row shows morpheme-unit performance.

| | Eq. (2) | Eq. (3) | Eq. (4) | Eq. (5) | Eq. (6) | Eq. (7) (Eq. (1)) |
|---|---|---|---|---|---|---|
| *Eojeol* | 86.80 | 90.48 | 89.40 | 89.62 | 91.73 | 92.48 |
| Morpheme | 91.32 | 94.93 | 94.40 | 94.48 | 95.77 | 96.12 |

possible for training gives much better performance than unsupervised training using the Baum-Welch reestimation algorithm (Merialdo 1994). We therefore decided to use supervised training using tagged corpora with relative frequency counts. The three necessary probabilities can be estimated as in Equations (8)–(10),

$$Pr(t_i \mid m_i) \approx f(t_i \mid m_i) = \frac{N(m_i, t_i)}{N(m_i)} \tag{8}$$

$$Pr(t_i) \approx f(t_i) = \frac{N(t_i)}{\sum_{n=1}^{N_{ts}} N(t_n)} \tag{9}$$

$$Pr(t_i \mid t_{i-1}) \approx f(t_i \mid t_{i-1}) = \frac{N(t_{i-1}, t_i)}{N(t_{i-1})} \tag{10}$$

where $N(m_i, t_i)$ indicates the total number of occurrences of the morpheme $m_i$ together with the specific tag $t_i$, while $N(m_i)$ indicates the total number of occurrences of the morpheme $m_i$ in the tagged training corpus. $N_{ts}$ indicates the total number of POS tags in the tagset. $N(t_{i-1}, t_i)$ and $N(t_{i-1})$ can be interpreted similarly for two consecutive tags $t_{i-1}$ and $t_i$.

A beam search strategy is utilized for high-speed tagging. For each morpheme in the sentence, the highest probability, $P_h$, of the tag is recorded. All other tags associated with the same morpheme must have probabilities greater than $\frac{P_h}{\gamma}$ for some constant beam size $\gamma$; otherwise, they are discarded. The beam may introduce search errors, but, in practice, search efficiency can be greatly improved with virtually no loss of accuracy.

### 4.2 Lexical Probability Estimation for Unknown-Morpheme Guessing

The lexical probabilities for unknown morphemes cannot be precalculated using Equation (8) since we assume the unknown morphemes do not appear in the training corpus, so a special on-the-fly estimation method must be applied. We suggest using syllable trigrams since Korean syllables can play an important role in restricting units for guessing the POS of a morpheme. The lexical probability $\frac{Pr(t_i|m_i)}{Pr(t_i)}$ for unknown morphemes can be estimated using the frequency of syllable trigram products according to the formula in (11)–(13) (Nagata 1994),

$$m = e_1 e_2 \ldots e_n \tag{11}$$

$$\begin{aligned} \frac{Pr(t \mid m)}{Pr(t)} &\approx Pr_t(e_1 \mid \#, \#) Pr_t(e_2 \mid \#, e_1) \\ &\quad \times \prod_{i=3}^{n} Pr_t(e_i \mid e_{i-2}, e_{i-1}) \end{aligned}$$

$$\times\ Pr(\# \mid e_{n-1}, e_n) \tag{12}$$

$$\begin{aligned} Pr_t(e_i \mid e_{i-2}, e_{i-1}) \quad \approx \quad & f_t(e_i \mid e_{i-2}, e_{i-1}) \\ & + f_t(e_i \mid e_{i-1}) \\ & + f_t(e_i) \end{aligned} \tag{13}$$

where $m$ is a morpheme, $e$ is a syllable, $t$ is a POS tag, "#" is a morpheme boundary symbol, and $f_t(e_i \mid e_{i-2}, e_{i-1})$ is a frequency datum for tag $t$ with co-occurrence syllables $e_{i-2}, e_{i-1}$, and $e_i$. Trigram probabilities are smoothed by Equation (13) to cope with the data sparseness problem. For example, *Park-jong-man* is the name of a person, so it is an unknown morpheme. The lexical probability that *Park-jong-man* should be assigned the tag MPN (person name) is estimated using the following formula:

$$\begin{aligned} \frac{Pr(MPN \mid Park - jong - man)}{Pr(MPN)} \quad \approx \quad & Pr_{MPN}(Park \mid \#, \#) \\ & \times Pr_{MPN}(jong \mid \#,\ Park) \\ & \times Pr_{MPN}(man \mid Park, jong) \\ & \times Pr_{MPN}(\# \mid jong, man) \end{aligned} \tag{14}$$

In *Park-jong-man*, *Park* is usually a family name. If the first position of this morpheme is a family name, the probability that MPN is the correct tag becomes higher than the probability that the other tags are correct. Table 6 shows the distribution of $Pr(Park \mid \#, \#)$ for each possible tag. In Equation (14), $Pr_{MPN}(Park \mid \#, \#)$ represents the popularity of the tag MPN for the morpheme *Park-jong-man*.

All the trigrams for Korean syllables were precalculated and stored in the database and are applied with the candidate tags during the unknown-morpheme POS guessing and smoothing portion of the statistical tagging process.

### 4.3 A Posteriori Error Correction Rules

Statistical morpheme tagging is widely known to cover only a limited range of contextual information. Moreover, it cannot refer to lexical patterns as a context for POS disambiguation. As mentioned earlier, because Korean *eojeols* have very complex morphological structures, it is necessary to look at the functional morphemes selectively to determine the grammatical relations between *eojeols*. For these reasons, we designed error correction rules for *eojeols* to compensate for the estimation and modeling errors

**Table 6**
The distribution of $Pr(Park \mid \#, \#)$ for each tag.

|  | MC | MPN | MPC | MPP | MPO | T |
|---|---|---|---|---|---|---|
| No. of "##Park" | 125 | 2081 | 0 | 0 | 8 | 0 |
| No. of "##" | 115,841 | 25,915 | 589 | 1,209 | 50,671 | 4,255 |
| $Pr(Park \mid \#, \#)$ | 0.001 | 0.080 | 0.000 | 0.000 | 0.000 | 0.000 |
|  | B | DR | DI | HR | HI |  |
| No. of "##Park" | 5 | 17 | 2 | 0 | 9 |  |
| No. of "##" | 9,169 | 21,119 | 13,555 | 2,243 | 5,217 |  |
| $Pr(Park \mid \#, \#)$ | 0.000 | 0.000 | 0.000 | 0.000 | 0.001 |  |

**Table 7**
Examples of rule schemata used to extract the error correction rules automatically from the tagged corpus. The POSTAG system has about 24 rule schemata of this form.

| Rule schema | Acronym description |
|---|---|
| N1FT | the tag of the first morpheme (FT) of the next *eojeol* (N1) |
| P1LT | the tag of the last morpheme (LT) of the previous *eojeol* (P1) |
| N2FT | the tag of the first morpheme (FT) of the *eojeol* after the next one (N2) |
| N3FT | the tag of the first morpheme (FT) of the second *eojeol* after the next one (N3) |
| P1LM | the lexical form of the last morpheme (LM) of the previous *eojeol* (P1) |
| P1FM | the lexical form of the first morpheme (FM) of the previous *eojeol* (P1) |
| N1FM | the lexical form of the first morpheme (FM) of the next *eojeol* (N1) |

[current *eojeol* or morpheme] [rule schemata, referenced morpheme or tag]
→ [corrected *eojeol* or morpheme]

**Figure 2**
Error correction rule format.

of the statistical morpheme tagger. However, designing the error correction rules with knowledge engineering is tedious and error prone. Instead, we adopted Brill's approach (Brill 1995) whereby the error correction rules are learned automatically from a small amount of tagged corpus. Fortunately, Brill showed that one does not normally need a large amount of tagged corpus to extract the symbolic tagging rules compared with statistical tagging. Table 7 shows examples of carefully designed rule schemata used to extract the error correction rules for Korean, where a rule schema designates the context of rule applications (i.e., the morpheme position and the lexical/tag decision in a context *eojeol*).

The form of the rules that can be automatically learned using the schemata in Table 7 is shown in Figure 2, where $[current\ eojeol\ or\ morpheme]$ consists of the morpheme (with current tag) sequence in an *eojeol*, and $[corrected\ eojeol\ or\ morpheme]$ consists of the morpheme (with corrected tag) sequence in the same *eojeol*. For example, the rule $[meog(\text{'Chinese ink'})/MC + eun/jS][N1FT, MC] \rightarrow [meog(\text{'to eat'})/DR + eun/eCNMG]$ says that the current *eojeol* was statistically tagged as a common noun (MC) plus auxiliary particle (jS), but if the next *eojeol*'s (N1) first-position morpheme tag (FT) is also MC, the *eojeol* should be tagged as a regular verb (DR) plus adnominal ending (eCNMG). This statistical error is caused by the ambiguity of the morpheme *meog*, which has two meanings: 'Chinese ink' (noun) and 'to eat' (verb). Since morpheme segmentation is very difficult in Korean, many tagging errors also arise from the morpheme segmentation errors. Our error correction rules can also cope with these morpheme segmentation errors by correcting the errors in the whole *eojeol* at once. For example, the following rule can correct morpheme segmentation errors: $[jul/MC + i - go/jO][P1LM,] \rightarrow [jul - i/DR + go/eCC]$. This rule says that the *eojeol* *jul-i-go* is usually segmented as a common noun, *jul* 'string, rope', plus the adverb and conjunctive particle *i-go*, but when the morpheme *eul* appears before the *eojeol*, it should be segmented as a regular verb, *jul-i* 'shrink', plus the conjunctive ending *go*. This kind of segmentation error correction can greatly enhance the tagging performance. The rules are automatically learned by comparing the correctly tagged corpus with the output of the statistical tagger. The training is leveraged so the error correc-

**Table 8**
Performance of the statistical tagger (all in %) on three document sets, using three progressively degraded versions of the tagger.

| Document set | Version 1 | Version 2 | Version 3 |
|---|---|---|---|
| Set 1 | 96.4 | 89.5 | 87.1 |
| Set 2 | 96.0 | 92.8 | 89.0 |
| Set 3 | 96.7 | 88.7 | 84.8 |
| Total | 96.4 | 90.3 | 87.0 |

tion rules are gradually learned as the statistically tagged texts are corrected by the rules learned so far.

## 5. Experimental Results

### 5.1 Embedded Performance with Hybrid POS Tagging

For morphological analysis and POS tagging experiments, we used a 130,000-morpheme balanced training corpus for statistical parameter estimation and a 50,000-morpheme corpus for learning the a posteriori error correction rules. The training corpus was collected from various sources such as Internet documents, encyclopedias, newspapers, and school textbooks.

For test sets, we carefully selected three different document sets, aiming for broad coverage. The first document set (Set 1: 25,299 morphemes, 1,338 sentences), which was collected from the *Kemong Encyclopedia*,[9] a hotel reservation dialogue corpus,[10] and assorted Internet documents, contains about 10% unknown morphemes. The second document set (Set 2: 15,250 morphemes, 574 sentences), which consists solely of Internet documents from assorted domains, such as broadcasting scripts and newspapers, contains about 8.5% unknown morphemes. The third document set (Set 3: 20,919 morphemes, 555 sentences), which comes from a standard Korean document set called KTSET 2.0[11] including academic articles and electronic newspapers, contains about 14% unknown morphemes (mainly technical jargon). Table 8 shows our system's statistical tagging performance for these three document sets, using three progressively degraded versions of the tagging mechanism. Version 1 is a full version using the statistical method. Version 2 is a somewhat degraded version that does not use the system's unknown-morpheme guessing capability but treats all the segmented unknown morphemes as nouns (the typical method of estimation). Version 3 is an even more degraded version that rejects all unknown morphemes as tagging failures; this version does not even perform unknown-morpheme segmentation during morphological analysis. This experiment verifies the effectiveness of our unknown-morpheme segmentation and guessing techniques, as shown by the sharp performance drops between Versions 1, 2, and 3. As another experiment showed, the automatically acquired a posteriori error correction rules also proved to be useful. In this experiment, two versions of the hybrid tagger were tested on the three document sets. Version 1 was the full POSTAG system with unknown-morpheme segmentation, guessing, and

[9] From the Electronics and Telecommunications Research Institute (ETRI).
[10] From Sogang University, Seoul, Korea.
[11] From KT (Korea Telecom).

**Table 9**
Performance of the hybrid tagger (all in %) on three document sets, using two versions of the tagger.

| Document set | Version 1 | Version 2 |
|---|---|---|
| Set 1 | 97.2 | 96.4 |
| Set 2 | 96.9 | 96.0 |
| Set 3 | 97.4 | 96.7 |
| Total | 97.2 | 96.4 |

**Table 10**
Unknown-morpheme estimation performance (all in %). Experiments were performed on three different document sets as before. *#UKM* designates the number of unknown morphemes in each document set and their percentage. Recall (*Rec.*) measures the coverage of the estimation and precision (*Pre.*) demonstrates its accuracy.

| Document set | #UKM | Rec. | Pre. |
|---|---|---|---|
| Set 1 | 2,531 (10.0%) | 93.9 | 94.8 |
| Set 2 | 1,303 (8.5%) | 92.9 | 88.9 |
| Set 3 | 2,889 (13.8%) | 98.0 | 85.5 |
| Total | 6,723 (10.8%) | 94.9 | 89.7 |

rule-based error correction. Version 2 did not employ a posteriori error correction rules (the same system as Version 1 in the first experiment). Performance dropped between Version 1 and Version 2 (see Table 9); however, the drop rates were mild due to the performance saturation at Version 1, which means that our statistical tagger alone already achieves state-of-the-art performance for tagging of Korean morphemes.

### 5.2 Unknown-Morpheme Segmentation and Guessing Performance

To see the independent performance of unknown-morpheme handling more precisely (explained in Sections 3 and 4.2), we separated the unknown-morpheme performance from hybrid tagging experiments. Using the same test corpus, we measured the coverage and correctness of our unknown-morpheme estimation techniques. Table 10 shows the results, which were evaluated by the metrics defined as follows:

$$\begin{aligned} Recall &= \frac{\#\,unknown\ morphemes\ detected}{\#\,unknown\ morphemes} \quad (segmentation\ performance) \\ Precision &= \frac{\#\,unknown\ morphemes\ correctly\ estimated}{\#\,unknown\ morphemes\ detected} \quad (guessing\ performance) \end{aligned}$$

When the morphological analyzer meets an unknown morpheme, it is important to detect first whether it is unknown or not, because sometimes, due to incorrect segmentation, an unknown morpheme can be incorrectly processed as a known one. Our system reached an average recall level of 94.9%. Once the unknown morphemes are detected, the correct POS needs to be estimated. Our system tries to guess the POS

**Table 11**
Unknown-morpheme estimation performance (all in %) for each POS tag. *N/A* means the morpheme with the corresponding tag does not appear in the corpus. Recall (*Rec.*) measures the coverage of the estimation, and precision (*Pre.*) demonstrates its accuracy.

| | Set 1 | | Set 2 | | Set 3 | | Total | |
|---|---|---|---|---|---|---|---|---|
| POS tag | Rec. | Pre. | Rec. | Pre. | Rec. | Pre. | Rec. | Pre. |
| MC | 96.9 | 95.4 | 94.5 | 91.7 | 93.9 | 72.5 | 95.1 | 86.5 |
| MPN | 80.0 | 86.7 | 87.4 | 95.0 | 100.0 | 100.0 | 89.1 | 93.9 |
| MPC | 54.3 | 73.7 | 72.7 | 37.5 | N/A | N/A | 42.3 | 37.1 |
| MPP | 75.2 | 63.3 | 86.9 | 75.0 | 100.0 | 100.0 | 87.4 | 79.4 |
| MPO | 79.4 | 79.4 | 94.8 | 68.3 | 100.0 | 93.8 | 91.4 | 79.7 |
| B | 87.9 | 100.0 | 42.9 | 66.7 | 100.0 | 100.0 | 76.9 | 88.9 |
| T | N/A | N/A | 100.0 | 100.0 | N/A | N/A | 100.0 | 100.0 |
| S | 99.8 | 100.0 | 99.0 | 100.0 | 100.0 | 100.0 | 99.6 | 100.0 |
| Foreign word | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 | 100.0 |
| Special symbol | 100.0 | 100.0 | N/A | N/A | 100.0 | 100.0 | 100.0 | 100.0 |

of every open class item including common nouns, proper nouns, pronouns, numbers, adverbs, and others.[12] The average precision of 89.7% reflects very accurate guessing considering the range of POSs that need to be estimated. Table 11 shows the system's unknown-morpheme guessing performance for each POS tag.

To show the pattern dictionary's utility, we conducted another experiment in which we gradually reduced the morpheme dictionary size to see the smooth hybrid tagging performance (same as in Table 9) drops. As the morpheme dictionary gets smaller, POSTAG becomes more dependent on the morpheme pattern dictionary and also on the unknown-morpheme estimation process. From the full dictionary (with 65,000 nouns), we randomly deleted 5,000 nouns step by step for this series of experiments. (We deleted only nouns because noun estimation is the best arena for showing the system's unknown-morpheme estimation power.) Figure 3 shows the results. Even if the POSTAG system relies heavily on unknown-morpheme estimation instead of on more accurate dictionary lookups, the performance drop is very slow. This result explains why POSTAG can be used on open domain materials such as Internet documents even when only a small morpheme dictionary is available.

## 6. Conclusion

This paper presents a pattern-dictionary-based unknown-morpheme estimation method for generalized and powerful unknown-morpheme segmentation and guessing for a hybrid POS tagging system. Generalized unknown-morpheme handling is a new and powerful idea that adopts a morpheme pattern dictionary and syllable-based lexical probability estimation. The morpheme pattern dictionary enables the system to segment unknown morphemes in the same way as registered morphemes without any separate rules for Korean, and thereby to handle them regardless of their numbers or positions in an *eojeol*. The paper also presents an error-corrective statistical and

[12] Pronouns, numbers, and adverbs may be considered as closed classes. However, in real-world corpora, we frequently find unexpectedly coined terms in these classes since Korean word formation is affected by very diverse sources such as foreign words, old Chinese words, archaic pure-Korean words, and so on.

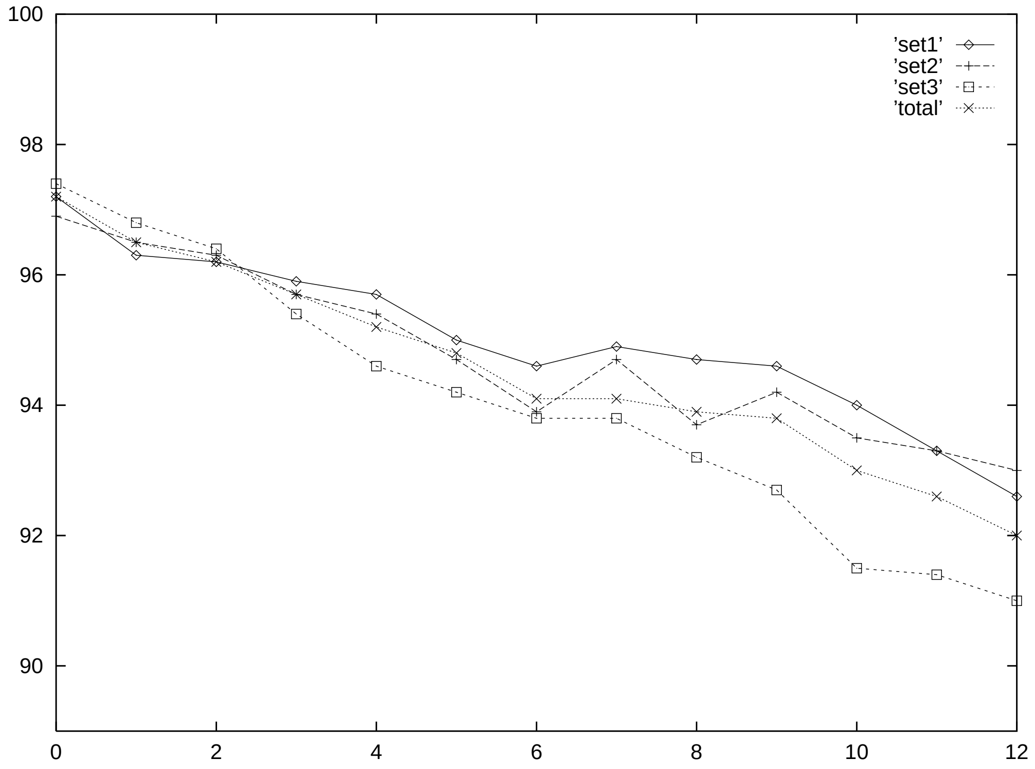

**Figure 3**
Hybrid tagging performance change (all in %), showing the utility of the pattern dictionary. Experiments were performed on three different document sets as before. The *x*-axis designates the number of deletion steps whereby the morpheme dictionary was decreased (by 5,000s) from its full size of 65,000 nouns (Step 0) to 5,000 nouns (Step 12).

rule-based hybrid POS tagging method that exhibits many novel features such as an experiment-based statistical model for Korean, rule-based error correction, and hierarchically expandable tagsets. The POSTAG system was developed to test these novel ideas, especially for agglutinative languages such as Korean. (Japanese, being similar to Korean in linguistic characteristics, will be a good target for testing these ideas.) Unlike previous systems, POSTAG is a hybrid tagging system; such a system has never been tried before, but it turns out to be most suitable for agglutinative languages such as Korean. POSTAG mainly applies a state-of-the-art HMM tagger for morphemes but considers multiple observations in the Viterbi score calculation. Because of the complexity of the morpheme sequence in a Korean *eojeol*, a morpheme-based HMM's tagging accuracy is relatively low for Korean, compared with its accuracy for English. POSTAG compensates extremely well for the limitations of HMMs by rule-based error correction. The error correction rules are automatically learned to selectively correct HMM tagging errors. Similar hybrid methods have been tried for English, but they integrate HMM tagging and rule-based tagging at the same level (Tapanainen and Voutilainen 1994). POSTAG integrates morphological analysis with the generalized unknown-morpheme segmentation so that unknown morphemes can be processed in the same manner as registered morphemes during tagging. POSTAG also employs hierarchical tagsets that are flexible enough to expand/shrink according to the given application. The hierarchical tagset is a novel idea. Most tagging systems for Korean have applied flat, fixed tagsets and have suffered from using varying tagsets in various applications. However, POSTAG's tagsets, based on the over 100 finely differentiated POS symbols for Korean are hierarchically organized and are flexibly reorganizable according to the application at hand. The hierarchical tagsets can be mapped to any other existing tagset as long as they are fairly well classified and therefore can encourage corpus sharing in the Korean-tagging community. POSTAG is constantly being improved through expansion of its morpheme dictionary, pattern dictionary, and tagged corpus for statistical and rule-based learning. Since the generalized unknown-morpheme handling is integrated into the system, POSTAG proves to be a good tagger for open domain applications such as Internet indexing, filtering, and summarization, and we are now developing a Web indexer using POSTAG technology.

## Acknowledgments

This project was partly supported by KOSEF (teukjeongkicho #970-1020-301-3, 1997.9-2000.8) and a Ministry of Education BK21 program awarded to the Electrical and Computer Engineering Division of POSTECH. We would like to thank JunHyeok Shim for coding the unknown-morpheme estimation experiments. An earlier version of this paper was presented at the 6th Workshop on Very Large Corpora in Montreal, 15–16 August 1998.

## References

Brill, E. 1995. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21:543–565.

Cutting, D., J. Kupiec, J. Pedersen, and P. Sibun. 1992. A practical part-of-speech tagger. In *Proceedings of the 3rd Conference on Applied Natural Language Processing*, pages 133–140.

Forney, G. 1973. The Viterbi algorithm. *Proceedings of the IEEE*, 61:268–278.

Im, H. S., J. D. Kim, and H. C. Im. 1996. Transformation rule-based tagging considering Korean characteristics. In *Proceedings of the Spring Conference of the AI SIG Meeting of the Korean Information Science Society*, pages 3–10. (Written in Korean.)

Kang, S. S. 1993. *Korean Morphological Analysis Using Syllable Information and Multiple-Word Units*. Ph.D. thesis, Department of Computer Engineering, Seoul National University. (Written in Korean.)

Kim, J. D., H. S. Im, and H. C. Im. 1996. Morpheme-based Korean part-of-speech tagging model considering *eojeol*-unit contexts. In *Proceedings of the Spring Conference of the Korean Cognitive Science Society*, pages 97–106. (Written in Korean.)

Kim, J. H., C. S. Lim, and J. Seo. 1995. An efficient Korean part-of-speech tagging using a hidden Markov model. *Journal of the Korean Information Science Society*, 22:136–146. (Written in Korean.)

Kupiec, J. 1992. Robust part-of-speech tagging using a hidden Markov model. *Computer Speech and Language*, 6:225–242.

Lee, U. J., K. S. Choi, and G. C. Kim. 1993. Korean text-tagging system. In *Proceedings of the Spring Conference of the Korean Information Science Society*, pages 805–808. (Written in Korean.)

Merialdo, B. 1994. Tagging English text with a probabilistic model. *Computational Linguistics*, 20:155–171.

Mikheev, A. 1996. Unsupervised learning of word-category guessing rules. In *Proceedings of the 34th Annual Meeting of the Association for the Computational Linguistics*, pages 327–334.

Nagata, M. 1994. A stochastic Japanese morphological analyzer using a forward-DP backward-A$^*$ N-best search algorithm. In *Proceedings of the 15th International Conference on Computational Linguistics*, pages 201–207.

Oflazer, K. and G. Tür. 1996. Combining hand-crafted rules and unsupervised learning in constraint-based morphological disambiguation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 69–81.

Sproat, R. 1992. *Morphology and Computation*. MIT Press, Cambridge, MA.

Tapanainen, P. and A. Voutilainen. 1994. Tagging accurately—don't guess if you know. In *Proceedings of the Conference on Applied Natural Language Processing*, pages 149–156.

Voutilainen, A. 1995. A syntax-based part-of-speech analyzer. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics*, pages 157–164.

Weischedel, R., M. Meteer, R. Schwartz, L. Rawshaw, and J. Ralmucci. 1993. Coping with ambiguity and unknown words through probabilistic models. *Computational Linguistics*, 19:359–382.