Local Grammar-based Lexical Stemmer for Korean Language

ChangYeol LEE

Korea Education and Research Information Service(KERIS), lcy@ms.keris.or.kr

Abstract

Stemming is to find basic forms of words. It enhances the performance of the information retrieval systems. Many systems utilize statistical approaches including Successor Variety and N-gram to stem.

In this paper, a lexical stemming method, which is based on Local Grammar, for Korean language is proposed. Korean words can be expressed by Local Grammar, and each node of Local Grammar corresponds to Korean lexicons. Each entry of the lexicons is manually checked in advance to prevent non-deterministic transitions of the local grammatical expression. It is because there are synonymous or other ambiguous data among the lexicons. The stemming procedures are to follow the nodes of the transitions with input word, and then to extract the nominal basic form of the word. If the ambiguous transitions are encountered, the pre-checked data are available for correction.

1. Introduction

Stemming is to find a basic form of words. In NLP system, stemming is a procedure for morphological analysis, and used in IR system to improve retrieval performance. Several stemming methods such as, Table Lookup, Affix Removal, Successor Variety, and N-gram have been used. In this paper, a lexical stemming method based on Local Grammar is proposed. Local grammar is a kind of finite states automata as is defined at LADL[1][1][3][5]. The local grammar has an advantage of allowing a very efficient treatment of the great number of the irregular grammatical phenomena and the implementation of robust, large-scale NLP systems[7][8].

Korean words that include nominal elements can be expressed by Local Grammar. Each node of the local grammar corresponds to lexicons. If the transition processing in the transition diagram of the grammar successfully arrived at the finial states, then it returns the grammatical categories of the input word. When the analyzed grammatical categories are ambiguous, further information is needed. For this information, all ambiguous transitions corresponding to the grammar expression are manually checked in advance. The stemming procedures are to follow the transitions of the local grammar with input words, and then the transitions system return the basic form of the word. When ambiguous transitions are encountered, then the pre-checked data are available for the transition correction.

Korean words are rich in multi-word expression. If they are not recognized as complex lexical units like German language, they cannot be properly understood in NLP system. It makes difficult to extract correct basic forms of Korean words without the grammatical information about complex lexical units. (1) is an example of Korean sentence that includes 4 Korean words

(1) 그는 그 학교에서 연구한다.

Pronoun-Postposition Article Noun-Postposition Noun-Predicate

[1] Laboratoire d'Automatique Documentaire et Linguistique, Université Paris 7

He-**nominative** the school-**locative** research-**st**

Most grammatical markers are attached to verbs (such as tense suffixes and modal suffixes) and to nouns (such as nominative postpositions and genitive postpositions) without blank. *He* and *school* of (1) respectively suffixed with grammatical markers, **nominative** and **locative** in Korean. Also the 4th Korean word, '연구-한다' of (1) consists of a Noun and a Predicate.

(2) *그는 그 학교에서 연구를 한다.*
Pronoun-Postposition Article Noun-Postposition Noun-Postposition Predicate
He-**nominative** the school-**locative** research-**acc** do-**st**
(He does a research in the school.)

(2) has the same meaning with (1), although they have different grammatical structures. The predicative term in (1) is simple verb '연구하다(to research), derived from a noun '연구'(the research) by means of a derivational suffix '하다', where as that in (2) is a syntactical phrase containing an accusative object '연구를'(research-**acc**) and a simple verb '하다'(to do). Nouns which can be found in the 'Noun-Predicate' and 'Noun-**acc** Predicate' forms are called Predicative Noun[6].

In IR system, stemming is confined to the words including nominal elements, because the other categories are very frequent in texts, and considered as the common words(or stop words). Stemming using Local Grammar highly depends on lexicon structures, because each node of the local grammar corresponds to lexicons. The Korean lexicons, which will be discussed in this paper, were constructed with Lexicon-Grammar by the Korean linguists of LADL[4][6]. The lexicons are called simple lexicons, because each grammatical unit of the lexicons is corresponding to simple verbs(SV), simple adjectives(SA) and simple nouns(SN). It is similar with the lexeme as a finitely enumerable set of lexical elements[2], because the simple words are also finite sets of the simple lexicons. The simple word is the most obvious units of meaning, and described in the dictionary[3].

Each affix is obligatorily attached to a stem consisting of simple words. Derivational affixes produce new items, when they are attached to simple nouns, where as inflectional affixes do not produce new items. We call the set of the derived and inflected words the complex words. The words are similar with multi-word lexemes in German language[1].

The complex words can be expressed by Local Grammar. The local grammar consists of nodes and link sets. It is a kind of finite states automata. Korean words are checked following the each node of the automaton, but it may generate ambiguous grammatical analysis. In other words, the automaton is non-deterministic. The non-deterministic transition information of the automaton is checked in advance by the manual work. Stemming procedures using the local grammar and the pre-checked information may be highly efficient, because it has the advantages of the automata expression. Also the ambiguity information is replaced by the pre-checked information. As a result, the local grammar with the pre-checked information has deterministic transitions.

2. Lexicons

All Korean simple words were extracted by Korean linguists in LADL and kept into the simple lexicons. They were constructed not only by using existing commercial dictionaries and large Korean corpus, but also combinatorial methods based upon explicitly defined lexical categories.

PreFixes(PF) and SuFfixes(SF) can be attached to the SN as affixes. Nominal PostPositions(NPP) play a role of

the grammatical function markers such as, nominative, accusative, dative, and locative which are optionally linked to nouns without any blanks. The number of the entries in the lexicons are shown in <Table 1>:

<Table 1> The number of entries in Korean lexicons

SN	SA	SV	PF	SF	NPP
14,000	5,000	8,000	950	900	1,500

3. Word Structure

A Korean word which includes nominal elements can be expressed by the regular expression, because the grammatical elements of the words correspond to lexicons and the lexicon is set which are the collection of the lexicon entries.

Korean words which include nominal elements can be defined as **Kn**. Predicative nouns are defined as 'Np'. The regular expression of **Kn** is '$PF^{*}SN^{+}SF^{*}NPP^{?}$ | Np (SV | SA)'.

3.1. Noun Structure

CN(Complex Noun) is defined as '$PF^{*}SN^{+}SF^{*}$'. NPP can be attached to CN as a functional marker such as, CN $NPP^{?}$.

All instances of '$PF^{*}SN^{+}SF^{*}$' are not CN, because the expression may generate unusable and ambiguous data. It means that the stemming of words may generate incorrect basic forms. Therefore the possible ambiguous data must be extracted from the expression, and then put into the correct grammatical categories of the data to the special module, called LD, which will be discussed at <Figure 3>.

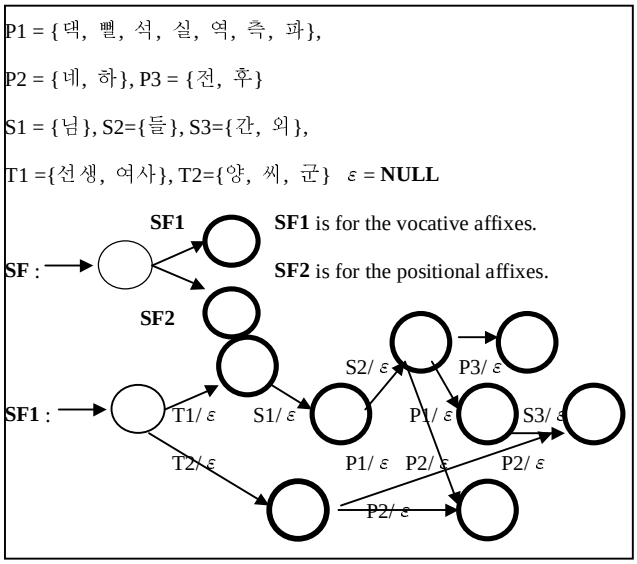
3.1.1. PF structure

In the CN expression, PF can be repetitively generated. It makes difficult to analyses the CN structure. Fortunately, PF is infrequent and the repeats of PF are not frequent as much as PF.

From the 950 PF entries, the complete connected data were listed out which have the expression PF^{n} (n = 2). The maximum number of the cases is 902,500(= 950 * 950). The most of the data are unusable. Using the exhaustive way, we manually extracted the usable data from the cases. There are 26 cases that can be combined with SN. The other possibility such as, PF^{n} (n > 2), doesn't exist. The list was inserted into PF lexicon. Hereinafter, PF lexicon will contain 950 elements of <Table 1> and additional 26 elements. Therefore, the expression CN is redefined as $PF^{?}$ $SN^{+}SF^{*}$. (3) is an sample instance including the repetitive expression of PF:

(3) 청녹색
bluish-greenish-color
PF-PF-SN
(bluish green)

3.1.2. SF Structure



<Figure 1> Local Grammar for SF

There are 900 entries in SF lexicon. SF has different repetition style comparing with PF, although SF is also

infrequent. For the vocative cases and positional cases, SF can be repetitively attached to SN. The repetitive structures are heuristically extracted from SF^{*}.

The structure of SF is described by Local Grammar at <Figure 1>. Details for the positional cases, which are defined as **SF2**, are not mentioned in here.

If we use the local grammar editor, developed by LADL(GUI editor), the edited grammar is automatically saved as a regular expression, and verified by the corpus data[7]. It generates the grammatical elements from the each Local Grammar. Therefore, the grammar expression using the local grammar editor is convenient tool for linguists to use the finite-state expression.

3.2. Predicative Noun Structure

Np denotes the predicative nouns. It is from the fact that the cases are small and all the cases can be enumerated, it is because the nouns is proper subset of SN. Korean linguists of LADL insisted that the number of Np is about 5,000(from 14,000 SN) [4] for the form 'Np-하다'. They, without any exceptions, can be found in the form 'Np-**acc** 하다'. Also there are other similar cases concerning 'Np-Adjective', such as 'Np-있다', 'Np-없다', 'Np-같다'. Korean predicates consist of verbs and adjectives. Adjectives also should be followed by the IS(Inflectional Suffixes). They indicate all grammatical functions of adjectives whereas it is a copulative verb such as '*be*' or equivalent verbs in English that take the markers indicating grammatical functions of adjectival strings. <Table 2> describes the samples of the predicative types.

<Table 2> Predicative types in Korean

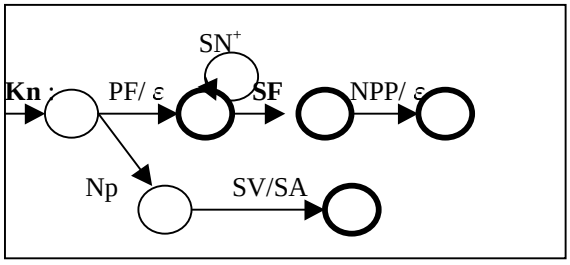
POS	English Example	Korean Ex.	Tagging
Verb	John sleeps	존은 잔다	N-NPP V-IS
Adj.	John is tall	존은 크다	N-NPP Adj-IS
Np	John makes a joke(=John jokes)	존은 농담한다	N-NPP N-V-IS

4. Stemmer

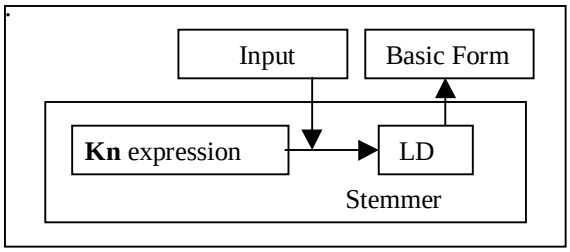
4.1. Local Grammar for Korean words

From the section 3.1 and 3.2, the definition of Korean words were derived that include nominal elements. The constituent **SF** is defined in <Figure 1>. **Kn** can be redefined as $PF^{?}$ SN^{+} **SF** $NPP^{?}$ | Np (SV | SA). Also, the local grammar of **Kn** is described in <Figure 2>.

The stemming of a word including the nominal elements follows the routine in <Figure 2>. Each node of the expression receives a part of the word which corresponds to any lexicons such as, PF, SN, SF, Np, SV, or SA, and then transits. If it successfully consumes the all part of the word, it returns the grammatical categories of the input word. Because of the synonymous or other ambiguous data among the lexicons, the local grammar generates ambiguous transitions. For the data correction, it needs further information about the ambiguous instances. LD(Lexical Dis-ambiguity) module of <Figure 3> contains the ambiguity-free checked data. Details of LD will be discussed in the next section.



<Figure 2> Local Grammar for **Kn**



<Figure 3> Stemming procedure of Korean words.

The stemming procedures of <Figure 3> are described at

(4):

(4) - Analyses input word using local grammar for **Kn**.
- If the analyzed data is ambiguous, then correct the data with LD information.
- Extracts a basic form from the analyzed data.

4.2. LD module

A word, which is classified as CN, may be ambiguous, because it consists of many lexicons' entries. To find out the ambiguous data, the possible combinatorial structures among the lexicons will be defined first. (5) is the structure, where X_i of the definition denotes the i[th] syllable of X.

(5) **Kn** = $\prod_{k=0}^{j} PFk \prod_{i=j+1}^{m} SNi \prod_{p=m+1}^{o} SFp \prod_{r=o+1}^{n} NPPr$,

if $0 \le j \le m \le \sigma \le n$

The ambiguous structures among the lexicons of **Kn** are defined at (6), (7), (8), and (9). After automatically extracting the possible ambiguous instances of each structure, we manually check the data. But some of them shall be continuously ambiguous.

In the following structures, low case alphabets(x, y, z) denote lexicon's entries, and the concatenation operator ' · ' denotes the string attachment operator likes the standard library function ***strcat*** of **UNIX** system, and upper case alphabets(X, Y, Z) denote lexicons. The structures can generate ambiguous instances, although they are grammatically correct.

(6) P1 structure : $(\exists x)$ $(x \in X$ and $x \in Y)$ $(X \neq Y)$
x is an entry of lexicon X and Y. For example, '-가' '(*-ga*) is an entry of lexicon SF and NPP.

(7) P2 structure : $(\exists x)(\exists y)(\exists z)((x \cdot y \in X \wedge z \in Y)$ and $(x \in X \wedge y \cdot z \in Y))$

(8) P3 structure : $(\exists x)(\exists z)(\exists x \cdot y)(\exists y \cdot z)((x \cdot y \in X \wedge z \in Y)$ and $(x \in X \wedge y \cdot z \in Z))$
This structure can only be generated between SN-SF and SN-NPP.

(9) P4 structure : $(\exists x)(\exists z)(\exists x \cdot y)(\exists y \cdot z)((x \in X \wedge y \cdot z \in Y)$ and $(x \cdot y \in Y \wedge y \cdot z \in Z))$
This structure can only be generated between PF-SN and SN-SF, and between PF-SN and SN-NPP.

4.2.1. P1 ambiguity

P1 ambiguity structure generated 262 ambiguous data between PF and SN. Also 253 P2 data are generated from SF and SN. To correct the data, it needs further information on the syntactic or semantic level. Therefore, it can not be provided information about these data.

There are 13 data which generate P1 ambiguities between SF and NPP. When the number of the ambiguities is small, the usage of the data can be checked manually. The entries of SF or NPP must require SN entries. It is natural that all data can generate P1 ambiguities between SN-SF, and SN-NPP is less than 182,000(equal to the multiplication of the number of SN by the 13 data). We manually extracted the correct data from the 182,000 data. The information was inserted to LD module. It is that some of the data are continuously ambiguous in the morphological level.

4.2.2. P2 ambiguity

```
for(j=0; j<Maximum_Length_Of_PF;j++)
    for(i=j+1; i<=Maximum_Length_Of_PF; i++)
      if(Lookup(PF, ΠXj) && Lookup(SN, ΠXm-j) &&
         Lookup(PF, ΠXi) && Lookup(SN, ΠXm-i))
       Printf(%s %s, ΠXj ΠXm-j, ΠXi VXm-i);
```

<Figure 4> P2 data extraction algorithm between PF and SN

P2 candidate lists can be automatically extracted by <Figure 4> algorithm. 2,739 P2 data from the algorithm

with PF and SN were acquired. After the manual checking of the data, 136 unambiguous instances were acquired. The information was kept in LD module. The same method was applied between SN and SF, and 2,272 candidates were acquired. 157 unambiguous instances from the candidates were acquired.

4.2.3. P3 ambiguity

There are two P3 structures in **Kn** expression:

(10) $(\exists x)(\exists z)(\exists x \cdot y)(\exists y \cdot z)((x \cdot y \in \mathrm{SN} \wedge z \in \mathrm{SF})$ and $(x \in \mathrm{SN} \wedge y \cdot z \in \mathrm{NPP}))$

(11) $(\exists x)(\exists z)(\exists x \cdot y)(\exists y \cdot z)((x \cdot y \in \mathrm{SN} \wedge z \in \mathrm{NPP})$ and $(x \in \mathrm{SN} \wedge y \cdot z \in \mathrm{SF}))$

Applying the (10) and (11) structure to (5) expression, 189 P3 candidate were extracted. After the manual checking of the data, 3 data were selected and corrected.

4.2.4. P4 ambiguity

Applying P4 structure to (5) expression, 400,949 data were acquired. It is on-going processing to check of the data.

5. Conclusion

Korean lexicons, discussed in this paper, were constructed with Lexicon-Grammar. The basis of the lexicons is the simple concept. The local grammar is utilized in the canonical expression of the robust and large-scale linguistic works.

Korean words including nominal elements are expressed by the local grammar. The grammar is naturally utilized in stemming steps, but the grammar contains ambiguous instances. In this paper, the structural ambiguity types among Korean lexicons were defined, and then extracted all instances of the each type. The instances were manually checked and saved to LD module. With the **Kn** expression and LD module, the stemmer of this paper easily generates the basic forms of Korean words. Stemming is a simple step of a morphological analyzing and described at (4). Although these procedures are not usual comparing with the traditional stemming approaches, it is efficient method in case of the simple lexicon environment.

Reference

[1] Breidt, Elisabeth, Segond, Frederique, and Valetto, Giuseppe, 1996, Local Grammars for the Description of Multi-Word Lexemes and their Automatic Recognition in Texts, Computational Lexicography 96, Linguistics Institute, Budapest.

[2] Cruse, D.A., 1986, Lexical Semantics, Cambridge Textbooks in Linguistics, p76-p83.

[3] Gross, Maurice, 1984, "Lexicon-Grammar and the Syntactic Analysis of French", in Proceedings of the 10th International Conference on Computational Linguistics, Stanford, p275-282.

[4] Han, Sun-Hae, 1993, "*Sur la construction nominale N0 N1-leul Npréd-leul Hada*", Mémoires du CERIL, No 12, Institute Gaspard Monge, Université de Marne-La-Vallée.

[5] Laporte, Eric, 1994, "Experiences in lexical disambiguration using local grammars", In COMPLEX 94, Papers in Computational Lexicography.

[6] Nam, Jee-Sun. 1994. *Dictionnaire des Noms Simples du Coreen*, TR-46, LADL, Universite Paris VII, 1994.

[7] Silberztein, Max, 1993, Dictionnaires électorniques et analyse automatique de textes – Le système INTEX, Masson, Paris, France.

[8] Walker , Donald E. Zampolli, Antonio. and Calzolari, Nocoletta. 1995. *Automating The Lexicon*, Clarendon Press Oxford, 1995