Proceedings of the Mongolian Academy of Sciences *Vol. 52 No 04 (204) 2012*

USING THE TWO-LEVEL MORPHOLOGY ON MODERN MONGOLIAN LINGUISTICS

Uuganbaatar Dulamragchaa[1], Guanglai Gao[2], Byambasuren Ivanov[1], Nergui Baasan[1]
[1] Informatics and RS Institute, Mongolian Academy of Science
[2] College of computer science, Inner Mongolia university

Abstract.

This study compiles primarily the word structure of Modern Mongolian language and further more focused on the possibilities of description of Mongolian language in PC KIMMO, a two level processing method of morphological parsing. The rules file and lexicon presented in the paper describe the morphology of Mongolian words. A lexicon containing the root words of contemporary Mongolian is used in the testing. As a result the two-level morphology is determined as completely possible to be used for Mongolian linguistics. In addition PC-KIMMO description of traditional Mongolian script is considered as being possible.

Key words: *Two level morphology, Mongolian language, Mongolian grammar, Morphology, NLP, PC-KIMMO, computational linguistics*

Introduction

In the twenty first century of rapid development of science and information technology, many nations and countries are converting and developing their language scripts and grammars into computer programming. So our Mongolians need to pay attention on that development and to apply the advanced development of computer programming in our linguistic studies. Therefore It is important to study the two-level morphology model which is applied in present linguistics broadly. By this study I tried to conclude how the advancement of this program could be used in Mongolian language. There are some researching works have been done in regard with usage of two-level morphology method in Mongolian linguistics.

In this study we generated the description of modern Mongolian grammar in PC-KIMMO based on two- level morphology model by creating the main components such as lexicon file and grammatical file. Based on the created description, the process of Mongolian word insertion and recognition proceeded as well as showing results of morphological parsing.

Word structure of Mongolian language

Linguistics consists of four major lexical units including text (composition), sentence, word and morpheme. Morphology is a word analysis that studies word by its description and structure with the relation of the meaning. Morphology is a sub-discipline of linguistics which studies morpheme structure and its role or position in the word and sentence structure. Morpheme is defined as the minimal unit which includes root, suffix, prefix, article, particle as well as super segments. Morphemes are divided into word root and affix by meaning and role. A root keeps the most original meaning of the word and is a basic unit of a word.

Mongolian language is an agglutinative language with its grammar structure and form, therefore the morphemes have certain consistency and coherence of morphology. Morphemes of word structure have canonical places and particularity and separations are appeared clear. The canonical places for morphemes of Mongolian words are divided into word root, derivational and inflectional suffixes.

- For inflectional model of noun stem, places and particularity are appears as follows in case there is one in each morpheme.

Root + noun building suffix + plural suffix + case suffix + possessive suffix

- For inflectional model of verb stem, places and particularity are appears as follows in case there is one in each morpheme.

Root + verb building suffix + aspect suffix + verb conjunctive suffix + ending suffix

In some cases, morphemes of Mongolian words are infracted.

Several inflectional suffixes (mainly 2-5) are conjugated rarely to some words. For example:

хэл +лц +үүл +г (3)

үйл +д +вэр +л +л (4)

хам +т +р +л +ж +уул +лт (6) etc.

A detailed examination of rules and order of such conjugation of several inflectional suffixes will be an important criterion to identify words and terms from the linguistic side. For the structure of one word, there is no conjugation of same types of suffixes, whereas two aspect suffixes of verb are conjugated rarely.

Word root keeps original meaning of the word and suffixes have lexical abstract meaning. In addition, word stems at the beginning of the word are more than derivational suffix and derivational suffixes are more than inflectional suffixes.

Although number of inflectional suffixes are fewer and express abstract meaning, one suffix can be conjugated to number of word roots. Whereas, a derivational suffix (even it is productive) can be conjugated only a few roots and stems. Moreover, one word root can be appeared afore only a few derivational suffixes.

Two level morphology

In linguistic, a two level morphology is a morphological parsing method for computational linguistics. Its main consequence is that introduced the connection of surface level form and lexical level form in morphology by forming a constant grammatical rule for computational linguistics. In 1983 the Finnish Scientist Kimmo Koskenniemi introduced a concept called two-level morphology into computational linguistics. The basic idea was to separate a lexical level and a surface level when considering the morphology of words. It would then be possible to use finite state automata to describe the relationship between every letter on the lexical and the surface side. The prototype of this system was successfully used to analyse Finnish. The system has since been applied on a wide range of different languages. It can be assumed that Koskenniemi's model is also suitable for the problems of the Mongolian language since Mongolian shares a number of common features with Finnish, like rich inventory of suffixes, vowel harmony etc.

A computer program operating on the principle of two-level rules has two basic capacities. Given the surface form of a word, it will be able to recognize its corresponding lexical form , and given the lexical form, it will be able to generate its corresponding surface form.

a. Two level Morphology and Mongolian language

Considering the linguistic goals for Mongolian we can develop the following framework of introducing two-level morphology into our work:

- *Spell-checking system*: In the spell-checking system, the system will accept input from a data stream and compare the words found in the input against its database consisting of roots and morph tactic information.

The database consists of stems, roots, morphemes and essentially contains the definitions of morphological and phonological structure.

- *Lemmatization system*: A lemmatization system operates according to the same mechanism but its output string would contain additional information about morphological properties of all morphemes of a word.
- *Mongolian script converter*: In a script

converting system, the Classical Mongolian form could be defined as lexical form and the Modern Mongolian form could be defined as surface form. In recognizer mode, the system would accept the Modern Mongolian word form and state the Classical word form; in generator mode, it would produce the Modern Mongolian word as the surface form of its lexical form being the word in its Classical Mongolian form.

· *Word form recognition system*: Word form recognition systems do basically the same as lemmatization systems yet their output does not focus on the stem and root of a word but on a grammatical description which is as complete as possible.

Ambiguity solving system: An ambiguity solver accepts input strings in Mongolian writing and recognizes all possible lexical forms that produce this surface string. This information alone, of course, does not solve the ambiguity- it only reveals it. However, a refined system is capable of offering additional information.

b. Two-level rule notation

In its basic form, the formal notation of a two level rules is separable into three elements: a description of a morphophonemic process;

a description of the environment of processing take place; and

a relational operator linking the descriptions of the process and its environment.

The environment consists of an underscore indicating the position of the process and optional left-hand and right hand environment specifications. At least one specification must be present but cases where both sides are specified are also easy to imagine. The operator linking process and environment is one of four:

=> *Context restriction rule*: The correspondence only occurs in the environment L:S=>E. L will be E only in environment S. It is also called *only if rule***.** It shows if there is a correspondence found in a context it is possible to be more correspondence.

For example: if there is vowel, soft consonant or suffix –x of future tense after soft sign, the soft sign will turn as vowel и.

ю:i => __+:0 [VO:0|COv|x]

<= *Surface coercion rule*: The correspondence always occurs in the environment L:S<=E. Surface coercion rule is also called *always but not only*. This correspondence shows that the context will be always relative. When one environment is declared a correspondence will be found, but there is always another environment where is a same correspondence. In other words the rule of the definition for the environment of correspondence will not be only one.

For example: if a masculine gender word takes future tense –x suffix, it will take vowel **-а** before -x. **+:а <= VOf CO (CO)___ x**

<=> *Composite rule*: The correspondence always and only occurs in the environment L:S<=>E. *Composite rule* or *if and only if rule* determines that the correspondence will be found only in one context and the context will require the found correspondence. Antworth named this rule as *always and only*.

For example: vowels **я** or **ё** stands in masculine word as a separate vocal of previos vowel, the hard mark will be written in between.

+: р <=> VOf CO (CO) __ [я, л]

/<= ***Negation rule***: The correspondence never occurs in the environment L:S/<= E

Negation rule shows that the correspondence cannot be included in the given context. The correspondence will never enter to this environment of context, but can be in other context.

For example: an unheard sound before the last consonant of a proper noun can not be missed. **V:0 /<= #Cap VO* CO* ___ CO +:0 VOVO**

PC-KIMMO program

The main difference of finite automata and finite distributer is input alphabet. The finite distributer receives pair letters of alphabets of one formal language into one side and another pairs of letter from another formal language into another side. So the conversion

of those two languages can be executed by finite distributor. There are 12 basic FST in PC-KIMMO.

This is a standard technique mostly used in the field of linguistic processing of analysis and morphological parsing. While the parsing is mostly considered as analysis in the sentence level, the morphological analysis of single words required to be contemplated in morphological parsing. A PC-KIMMO description of a language consists of two files provided by the user:

1. a **rules file,** which specifies the alphabet and the phonological (or spelling) rules, and

2. a **lexicon file,** which lists lexical items (words and morphemes) and their glosses, and encodes morph tactic constraints.

Lexical theoretical model used in PC-KIMMO program is characterized as a two-level model of word structure. In which a word is represented as a correspondence between its lexical level form and its surface level form.

Because the PC-KIMMO program is intended to facilitate development of a description, its data-processing capabilities are limited. The primitive PC-KIMMO functions are available as a source code library that can be included in another program. This means that the users can develop and debug a two-level description using the PC-KIMMO program and then link PC-KIMMO's functions into their own programs. KGEN and KTEXT programs are linked to the PC-KIMMO. Further information related with the development process, components and usage of PC-KIMMO are able to read from the source "PC-KIMMO: A Two-level Processor for Morphological Analysis" [3].

DESCRIBING MONGOLIAN LANGUAGE IN PC-KIMMO

Romanization of given natural language script character is necessary for description in relevant files for the analysis in PC-KIMMO. As PC-KIMMO program is written in programming language C, here we have used C language as well.

mongolian	а	б	в	г	д	е	ё	ж	з	и	й	к	л	м	н	о	ө	п
latin	a	b	v	g	d	й	л	j	z	i	н	k	l	m	n	o	ц	p
mongolian	р	с	т	у	ү	ф	х	ц	ч	ш	щ	ъ	ы	ь	э	ю	я	
latin	r	s	t	u	п	f	х	c	з	š	w	р	у	ю	е	ь	я	

This table of transliteration doesn't show an approved standard of trans-letters, although aimed to be used in this study only.

a. Creating Lexicon files of Mongolian language

Lexicon files of Mongolian language can be determined as following form.

```
ALTERNATION   Root    N  V  AJ
ALTERNATION   Suffix  SUFFIX INFL
ALTERNATION   Infl    INFL
ALTERNATION   End     End
…
FIELDCODE     lf  U   ;lexical item
FIELDCODE     lx  L   ;sublexicon
FIELDCODE     alt A   ;alternation
FIELDCODE     fea F   ;features
FIELDCODE     gl  G   ;gloss

INCLUDE   affix.lex   ;file of affixes
INCLUDE   noun.lex    ;file of nouns
INCLUDE   verb.lex    ;file of verbs
INCLUDE   adjectiv.lex
 ;file of  adjectives

END
```

Root alternation shows roots can be a Noun(N), Verb(V) or Adjective(AJ). Suffix alternation shows Suffix and inflectional suffix can be placed after Suffix. End alternation shows that word creating process has finished.

Example of inputs of lexicon initials.

\lf aav	\lf av	\lf +gз
\lx N	\lx V	\lx SUFFIX
\alt Suffix	\alt Suffix	\alt Suffix
\gl1 abu	\gl ab	\fea v/n
\gl2		\gl

b. Creating rule files

In order to process description in PC-KIMMO, all rules should be created true and to be checked consequently. Infinite analysis of rules written in limited range will cause some output shortage and error, however can be used in some cases

Here introduced three forms for creating rules. The first form is suitable for processing in the constant type system of surface form and lexicon form.

The next form of rule creating is focused on analysis of various situational words comparing their structural forms and attempts to get hidden rules of grammar.

This is an attempt to justify own comprehensions of given natural language in grammatical form. If the declared rules return unexecuted data, it should be explained through the program. For instance, the cause can be a unknown/foreign word.

The common forms of rules are:

- Creating rule for alphabet
- Justifying rule based on linguistic capacity
- Inserting data as a table in rule

There are 5 rules which apply to various amplitudes.

1. Rule of appearance of a letter of the most limited scale amplitude.V:0 vs. V:V
2. Rule to recognize if a letter from large scale amplitude to be recognized as surface form changed and keeping place V1: V1 vs. V1: V2
3. Rule of larger scale by influencing environment without affecting the letters. For instance: In mongilain language, a word, ended with consonant **ж, ч, ш,** and **г** will take genitive suffix **–ийн**, however the word is of masculine gender.
4. Rule of larger scale, which applies to one and more amplitudes. This is mostly refers to the rule of vocalic harmony
5. Rule of largest scale of amplitude that applies to all letters from start to end.

c. Describing rules of Mongolian language into virtual form

Let us start with declaring the main elements of of Modern Mongolian language as Alphabet, Subset and rules. *Creating rule files:*

In the first step, create the alphabet. Create a list of informal surface forms and lexical forms of letters under the keyword ALPHABET.

ALPHABET
a b v g d й л j z і н k l m n o ц p r s t u п f x c з š w р у ю е ь я
NULL 0
ANY @
BOUNDARY #

@- any character, # - ambit , start or end of word

Declare the following subsets:

VO – subset of all vowels

CO – subset of all consonants

According to the rule of vocalic harmony, vowels must be divided into at least two subsets.

VOf – all feminine vowels
VOm - masculine vowels
VOn – neutral vowels
VOmn – primary vowels

Consonants can be divided into few subsets.

CO – subset of all consonants
COs – consonants that only used in foreign words
COv – vocalized consonants
COp – non vocalized consonants
Si – sign letters

SUBSET VO а е і о u ц п я й л ь н у
; vowels
SUBSET VOmn а е і о u ц п
; main vowels

```
SUBSET VOau  я й л ь н у
        ; auxiliary vowels
SUBSET VOm   а о ө ы я л у
        ; male    + ьө
SUBSET VOf    е ц п й
        ; female  + ьп
SUBSET VOn    і н
        ; neuter
SUBSET VOd    аа ее оо өө ыы цц пп ін
        ; dual
SUBSET VOp    ан ен он ын пн
        ; pair
SUBSET CO     b v g d j z k l m n p r s t f x c з š w
        ; consonants
SUBSET COv     b v g l m n r
        ; voweled consonants
SUBSET COp     d j z s t x c з š
        ; partitive consonants
SUBSET COs     k p f w
        ; special consonants
SUBSET COr     j з š
        ; used for rule 27
SUBSET Si        р ю
        ; signs

RULE
“Defaults  1”  1  38
a b v g d й л j z і н k l m n о ц р r s t u п f x c
з š w р у ю е ь я - + @
a b v g d й л j z і н k l m n о ц р r s t u п f x c
з š w р у ю е ь я - 0 @
1: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
;uppercase letters and period

RULE
“Defaults 2” 1 37
A B V G D Й Л J Z I H K L M N O Ц P R S
T U П F X C З Š W Р Ү Ю Е Ь Ӳ . @
A B V G D Й Л J Z I H K L M N O Ц P R S
T U П F X C З Š W Р Ү Ю Е Ь Ӳ . @
1: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

Let us take example of making a rule of genitive suffix. Genitive suffixes are refers to following surface forms: **+ы, +ын, +ий, +ийн** and **+н**. Let us consider that the vowel selection will depend on the vowel type of feminine or masculine and create a rule as normal as follows.

If the vowel before last consonant belongs to subset **Vom**, vowel will turn to **ы.**

If the vowel before last consonant belongs to subset **Vof,** the vowel will turn to **ий.**

For instance, if we take the words **гар** and **гэр**, the rules will be described as гар+ын, гэр+ийн, which are correct. Lets create a rule as:

```
I:i   <=   VOf  C* ___
I:y   <=   VOm  C* ___
```

Also, lets take an example of a word that will not be applied to above rules, for instance - **хорины**. The vowel before last consonant of this word doesn’t belong to subsets **VOm** and **Vof**. In this case we need to consider the previous vowel. It is not possible to create a rule which applies to take **+ы** after **+I** of previous syllable. If **+ i-** stands before another vowel there is no functional relevance between **+ i-** and suffix.

Therefore the following rule is cannot be created. * I:y <= i CO* ___

Nevertheless if **+ i** is a sole vowel of the word, it should be described by rule. In this case lets consider **i** as a element of **VOf** subset. If we comprise it with the above rule the new rule will be as follows:

```
I:i   <=  # CO* VOf  C* i CO ___
I:y   <=  # CO* VOm  C* i CO ___
```

We can find lot of words which finish by **i** and p. For example: сургууль, анги etc. In this case **-ы** will never be used, but **–i**. (сургуулийн,ангийн) So the rule is as below

```
I2:i <= [i| p |COr] +:0  I1:i ___
```

In order to check the rule insert a word as багш and see the error if the result is false as **багшын** instead of **багшийн**. Some consonant are irregular and contrasts with rule of vocalic

harmony. Those kind of consonants are listed in COr = {j з š g} subset. Note that consonant **g** can be seen after consonant **n**. Some words end with hidden (phonetical) **g** which doesn't appear in surface form. (For example: сан). Therefore the root of such a word ending by **n** will not belong to **Cor** subset, however the suffixed form is **сангийн**.

The rule of this can be extended as follows. VO:0 / <= @:n @:g ___

If comprise them:

```
RULE "I2:y =>VOm:@ (CO) +:0 I1:0 ___" 4 6
RULE "I2:i <= [i|p|COr] +:0 I1:i ___" 4 8
RULE " I1:i <=> ___I2:i " 3 4
RULE " I2:0 <=> i +:0 I1:0 ___" 4 6
RULE "Pure Vowel Deletion Correspondences" 1 8
а о е и цпі @
0 0 0 0 0 0 0 @
1: 1 1 1 1 1 1 1
```

```
RULE "Vp:0 <= CO ___+:0 I1:@ I2:@" 5 7
RULE " ю:0 <= CO ___+:0 I1:i I2:i " 5 7
RULE "VO:0 /<=# (CO) ___" 2 4
RULE "VOp:0 /<=VO ___" 2 3
RULE "VO:0 /<= ___VOp" 2 3
```

RULE "VO:0 /<=@:n @:g___" 3 4.

We can describe all the rules of Mongolian language as shown above.

Conclusion

By this research work, we studied the principles of two level morphology method in accordance with linguistic theories as well as including creation of rule files and lexicon files of Mongolian language. The description of mongolian language in PC-KIMMO program was successfully processed and tested. The study shows that the two level morhology method is possible to be used consequently in computaional linguistics of Mongolian. In addition PC-KIMMO description of traditional Mongolian script is considered as being possible.

The importance of the study is concluded with regard of our objectives to help linguist students and scientists, while attempting to develop computational linguistic applications such as for parsing of mongolian language structure and converter of native language .

References:

1. *Oliver Corff, Nergui.B, Urgamal, Dorj and Sambuudorj. "Usage of a two-Level morphological theory", Ulaanbaatar, 1996.*
2. *Unurbayan.Ts. "Modern Mongolian Morphology", School of Mongolian Studies, National Educational University of Mongolia, 2004.*
3. *PC-KIMMO: A two-level processor for morphological analysis. November 1995, see the link http://www.sil.org/.*
4. *Evan L. Antworth. PC-KIMMO: A two-level Processor for Morphological Analysis, volume Number 16 of Occasional Publications in Academic Computing. Summer Institute of Linguistics, International Academic Bookstore, Summer Institute of Linguistics, 7500 W. Camp Wisdom Road, Dallas, Texas 75236, 1990.*
5. *LauriKarttunen, KimmoKoskenniemi, and Ronald M. Kaplan. A compiler for two-level phonological rules. Number CSLI-87-108. Center for the Study of Language and Information, Stanford University, 1987.*
6. *Chagnaa Altangerel, Baasanjav Adiyatseren. Two level Rules for Mongolian language. ISSN 1975-4736 MITA 2011.*
7. *Uyanga.B, Suld-Erdene.G, Narangerel.Sh, Khishigjargal.J and Oyun.D. "Orthography Dictionary of Mongolian Cyrillic", Ulaanbaatar, 2011.ISBN 987-99962-0-342-8*
8. *Bayarsaikhan.B. "Mongolian Language Dictionary", Ulaanbaatar, 2011.*