Analysis of Korean Predicative Verb Forms in LAG Framework

Soora Kim
Department of Computational Linguistics
University of Erlangen-Nürnberg
Bismarckstr. 12 91054 Erlangen
Republic of Germany
sakim@linguistik.uni-erlangen.de

Abstract

Korean predicative verb forms obligatorily denote the three categories *speech level, mood* and *sentence type* which are not handled by most of the automatic word form recognition systems for this language. These categories are marked by special endings. This paper examines predicative verb forms concentrating on the lexical description of these endings[1] in the framework of Left-Associative Grammar (LAG). Additionally this paper suggests a system to analyse verb forms in these aspects. The results of this study have been implemented using Malaga[2] and integrated into an automatic word form recognition system for Korean called KMM (Korean Malaga Morphology).

1 Theoretical Background

1.1 LA-Grammar

The aspects of natural language signs are strictly 'time linear', that is, reflect processing in real time[3]. The time-linear structure of natural language is so fundamental that a speaker cannot but utter a text sentence by sentence, and a sentence word form by word form. Thereby the time-linear principle suffuses the process of utterance to such a degree that the speaker may decide in the middle of a sentence on how to continue.

Correspondingly, the hearer need not wait until the utterance of a text or sentence has been finished before her or his interpretation can begin. Instaed the hearer interprets the beginning of the sentence without knonwing how it will continue.

The time-linear nature of language can be very well captured by an LA-grammar. The LA-Grammar is a grammar formalism proposed by Hausser (1989a). The same way as speakers and hearers utter and understand sentences and word forms as a linear sequence, one element at a time, LAG produces and analyses sentences or word forms step by step based on the principle of possible continuations: the parts of a sentence or a word form are concatenated from the left to the right, hence the name "Left-Associative Grammar".

LAGs, based on the principle of successive concatenation of categorised surfaces result in the time-linear derivation order, which is cognitively adquate. The time-liearity is inherent to

[1] All the inflectional suffixes are generally called *verbal endings* or *endings* in short in accordance with the traditional terminology.

[2] The Name "Malaga" has two different meanings: on the one hand, it is the name of a special purpose programming language, namely a language to implement grammars for natural languages. On the other hand, it is the name of a program package for development of Malaga Grammars and testing them by analysing words and sentences (see 3). For further description of this language see Beutel (2001).

[3] This formulation may be regarded as a modern version of Ferdinand de Saussure's (1913/1972) second law:

> The designator, being auditory in nature, unfolds soley in time and is characterized by temporal properties: (a) it occupies an expansion, and (b) this expansion is measured in just one dimension: it is a line (F. de. Saussure (1913/1972): p. 103, recited from Hausser (1999): p. 98).

the model of LAG, while the PS-grammars based on the principle of possible substitution can achieve this property only using special algorithms indirectly[4].

The generative capacity of unrestricted LAGs is equivalent to the class of recursive languages. Depending on the degree of ambiguity, subclasses of LAG called C3, C2 and C1 are defined, a complexity hierarchiy which is orthogonal to Chomsky's hierarchy of context-sensitive, context-free and regular grammars. Of particular interest in the LA-hierarchy is the class of C1-languages which parses in linear time and intersects with the class of context-sensitive and context-free languages of the Chomsky hierarchy, which the natural languages are believed to belong to.

The principle of possible continuation can be seen in the rule scheme of LAG:

r_i: cat_1 cat_2 $\Rightarrow cat_3$ rp_i

The rule consists of the name r_i, the category patterns cat_1 cat_2 and cat_3, and the rule package rp_i. The category patterns define a categorial operation which maps a sentence start cat_1 and a next word cat_2 into a new sentece start cat_3. The rule package rp_i lists all rules applicable after the succesful categorial operation of r_i.

1.2 Methods of Automatic Word Recognition

Morphological analysis consists of segmentation, lexical look-up and the concatenation. Possible methods of automatic word form recognition may be distinguished as to whether their analysis lexicon specifies word forms, morphemes or allomorphs. Each method exhibits a characteristic correlation between the recognition algorithm and the associated analysis lexicon.

The *word form method* allows for the simplest recognition algorithm because the surface of the unknown word form simply matches the corresponding key in the analysis lexicon. This method may be useful as a quick and dirty method for toy systems, providing lexical lookup without much programming effort. In the long run this method is costly, however, because of the production, the size and the basic finiteness of its analysis lexicon.

The *morpheme method*, on the other hand, uses the smallest possible analysis lexicon consisting of analysed morphemes. Compared to the word form method, it has the advantage that neologisms may be analysed and recognised during run-time using a rule-based segmentation and concatenation of complex word forms into their elements (morphemes). The only requirement is that the elements are lexically known and their mode of composition can be handled correctly by the rules.

The morpheme method is related to transformational grammar. It doesn't treat allomophrs as fully analysed grammatical entitites, but rather as the quasi-adulterated surface reflexions of the 'underlying' morphemes, which are regarded as 'real' entities of the theory. Concatenation takes place at the level of morphemes. For this reason, this method violates the principle of surface compositionality[5] of LAG. Also, because the morpheme method tries to compose the morphemes as much as possible as constituents, it collides with the principle of time linearity. Mathematically and computationally, the morpheme method is of high complexity, another disadvantage of this method, because the system must check the surface for all possible phenemena of allomorphy.

The *allomorph method* combines the respective advantages of the word form and the morpheme method by using a simple recognition algorithm with a small analysis lexicon. Based on its rule-based analysis, the allomorph method can also recognise nelogisms during run-time.

[4]For the further discussion see Hausser (1999): p. 163ff.

[5]An analysis of natural language is surface compositional if

> it uses only concrete word forms as the building blocks... (Hausser (1999): p. 80)

what has the methodological consequence that syntactic analyses are concrete because no kind of zero surface or underlying form may be used.

Before run-time, the analysis lexicon is derived automatically from an elementary lexicon by means of allo-rules. The elementary lexicon consists of the analysed elementary base forms of the open word classes, the analysed elements of the closed word classes, and the allomorphs of the grammatical forms as needed in inflection, derivation and composition.

During run-time, the allomorphs of the analysis lexicon are available as precomputed, fully analysed forms providing the basis for a maximally simple segmentaton: the unknown surface is matched from left to right with suitable allomorphs - *without any reduction to morphemes.* Concatenation takes place on the level of analysed allomorphs by means of so-called combi-rules. This method is in concord with the principles of surface compositionality and time linear derivation order which is inherent to LAG.

Of the three methods, the allomorph method is suited best. It is of low mathematical complexity, describes morphlogical phenomena of concatenation and allomorphy in a linguistically transparent, rule-based manner, handles neologisms during run-time, may be applied easily to new languages, is computationally space and time efficient, and can be easily debugged and scaled up. This method is especially compatible with LAG. The allomorph method has been developed and implemented within the framework of LAG as a system called LA-Morph and has been successfully applied to the automatic word form recognition for various natural languages like English, German, Italian and Spanish.

1.3 LA-Morph Concept

Linguistically LA-Morph is based on the following components[6]

An elementary lexicon containing elementary base forms. A lemma of an elementary lexicon is represented as ordered triples consisting of the surface, the grammatical category and the semantic representation as in the following example

("cwu" (nom acc v) cwu)[7]

The category (nom acc v) characterises the entry as a verb which takes a nominative and an accusative as arguments.

An allo-rule which takes a lemma of the elementary lexicon as input and derives allomorphs from it. The input and output is defined in terms of patterns. The basic structure of the allo-rule is as follows:

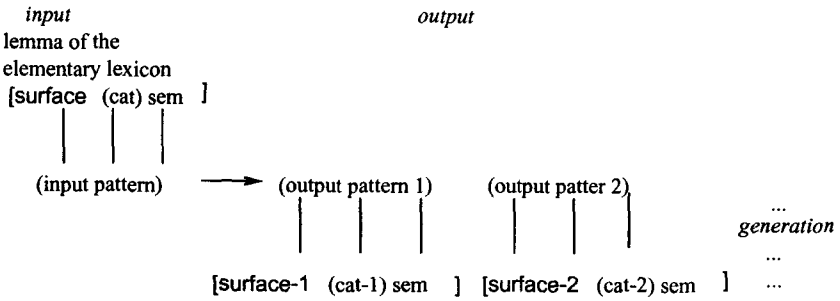

Figure 1. The basic format of the allo-rule

A set of combi-rules which take a start of a word form and a next allomorph as input and map them into a new word form start. It ensures that only grammatically correct forms are recognised. The format of a combi-rule is as follows:

[6] see Hausser (1999):

[7] The list-based structure of the lexical entry can be also presented using the attribute-value structure, as we will see on figure 5 and 6. For the discussion of the equivalence of these variants see 3.

r_i: (pattern of start) (pattern of next) $\Rightarrow cat_3$ rp_i (pattern of new start)[8]

2 Predicative Verb Forms in Korean

The agglutinative nature of the Korean language is distinctly reflected in its abundance of endings attached to verb stems. It is necessary to distinguish between terminal and non-terminal endings, in that the former occur at the end of the verb form and have to be present in order for a verb or adjective to stand independently while the latter do not. Thus, for instance, the verb stem *cwu-* ('to give') is never used alone in utterances. In order for a verb stem to be used predicatively it must be followed by the so called sentence ender. Examples are ***cwu-pnita***, ***cwu-nunya*** etc. Sentence enders take the syntactic role of relating the whole preceding clause to the main clause and denoting the *sentence type*. They also take the semantic and pragmatic role of reflecting the *speaker's relationship with the addressee*(speech level) and *speaker's attitude toward the content of the utterance* (mood), which are important informations at the higher level of the natural language processing.

Consider the following paradigm each consisting of a verb stem and a sentence ender:

cwu-pnita cwu-ptita cwu-psita

The members of the above paradigm differ only in terms of mood. From this sematical opposition between the paradigm members we can infer that the element responsible for the different moods of the members is *ni, ti* and *si*. We can further identify the element *p* of which meaning cannot identified yet. Based upon the observation that the following paradigm members differ only in terms of speech level, and this element appears only in one speech level, namely deferential, as will be discussed later, we can suppose that the semantic feature of this element contributes to denoting this speech level.

cwu-pnita cwu-nunya cwu-nunka

By means of contrasting the different meanings of the following verb forms we can also identify the forms *ta* and *kka* as the resulting in different sentence type:

cwu-pnita cwu-pnikka

Based on the observation so far we can conclude that the sentence ender can consist of up to three the elements denoting differential speech level, mood and sentence type. Of the three categories within a sentence ender the last one is morphologically realised in every predicative verb form, while the other two are not realised in some speech levels and moods. Each category can be realised by more than one variants, as we can see from the paradigm cwu-pnita cwu-nunya nwu-nunka in this case *ni* and *nu*.

Similar observations are found in Sohn (1999). He analyses a sentence ender as consisting of three "slots" which can be filled with endings denoting the categories *addressee honorificcation* (AH), *mood* (MOOD) and *sentence type* (ST), so that the verb forms ***cwu-pnita***, ***cwu-nunya*** must be actually analysed as ***cwu-p-ni-ta***, ***cwu-Ø-nu-nya***[9]. In addition to the mood supposed by Sohn (1999) this paper supposes the mood category Exclamative. Each category has one or more members, as schematised in figure 2, or can be empty with exception to the last element as we will see in the table 2.

[8]Because the allomorphs of LA-morphology and the word forms of LA-syntax are similar in structure(ordered triples) their respective time-linear composition is based on the same general rule mechanism of LAG.

[9]The notation Ødoes not stand for a null allomorph but a slot which is not filled with any grammatical forms.

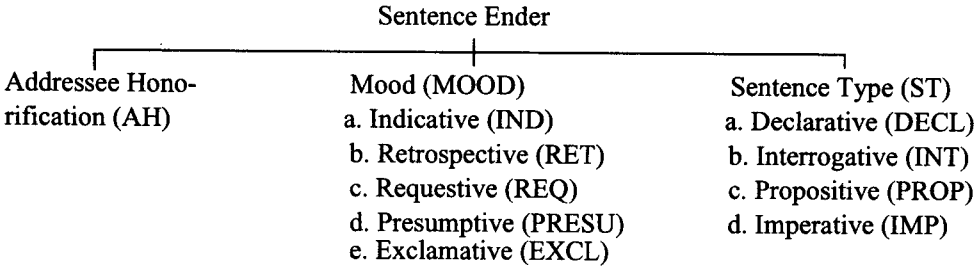

Figure 2. The structure of the sentence ender

The sentence ender furthermore denotes one (or more) of the six speech levels namely *plain* (PLN), *intimate* (INTIM), *familiar* (FML), *blunt* (BLN), *polite* (POL) and *deferential* (DEF) and *Exclamative* (EXCL). This paper introduces one more speech level so-called addresee-insensitive(AIS) as in the verb form *cha-la*'it is cold!'. We call the endings denoting the categories addressee honorification, mood and sentence type AH-marker, MOOD-marker, and ST-marker respectively. Each category can be morphologically realised by more than one form, which to choose is not arbitrary but depends on the speech level. Table 1 - 4 give a simplified description of how the verb stem *cwu-* ('to give'), *ka-* ('to go'), *cap-* ('to catch'), *yeppu-* ('to be pretty') and *mek-* ('to eat') conjugate in their predicative use. For the sake of brevity, four out of six speech levels are chosen.

AH	MOOD			ST		example
Ø	IND	after the adjective stem	Ø	DECL	nta	ka-Ø-Ø-n̄ta
					ta	yeppu-Ø-Ø-ta
				INT	nya	yeppu-Ø - Ø-nya
		after verb with a closed last syllable	nu	DECL	nta	mek -Ø-nu-nta
				INT	nya	mek-Ø-nu-nya
	RET		te	DECL	la	ka-Ø-te-la
				INT	nya	yeppu-Ø-te-nya

Table 1: Plain Speech Level

AH	MOOD	ST			example
Ø	Ø	DEC/INT/PROP/IMP	after open a or e	Ø	ka-Ø-Ø-Ø
			after closed a	a	cap-Ø-Ø-a
			after closed e	e	mek-Ø-Ø-e

Table 2: Intimate Speech Level

This paper proposes a system to implement the process to analyse the predicative verb forms, which will be discussed in the following section.

AH	MOOD			ST		example
Ø	ney (MOOD: IND, ST: DEC)					ka-Ø-ney
	tey (MOOD: RET, ST: DEC)					ka-Ø-tey
	sey (MOOD: REQ, ST: PROP)					ka-Ø-sey
	key (MOOD: REQ, ST: IMP)					ka-Ø-key
	IND	after verb	nu	INT	nka	ka-Ø-nu-nka
		after adjective or copula	Ø			yeppu-Ø-Ø-nka
	RET		te			nka-Ø-te-nka

Table 3: Familiar Speech Level

AH	MOOD		ST		example
p	IND	ni	DEC	ta	ka-p-ni-ta
			INTER	kka	ka-p-ni-kka
	RET	ti	DEC	ta	ka-p-ti-ta
			INTER	kka	ka-p-ti-kka
p	REQ	si	PROP	ta	ka- p-si-ta
(p)			IMP	o	ka- (p-) si-o

Table 4: Deferential Speech Level

3 Implementing the Analysis of Predicative Verb Forms

As already mentioned, the ST-marker is an obligatory part of every sentence ender. It reflects, possibly in cooperation with the AH-marker and MOOD-marker, the speech level as well as the sentence type. Some are merged with MOOD-markers (the single-syllable declarative ender *-ney*, see table 3). ST-markers like *-nka* occur only in one speech level denoting only one sentence type, namely interrogative in familiar speech level (see table 3), while ST-markers like *-la* denote different sentence types in different speech levels, namely so-called neutralised imperative in plain speech level, declarative in plain speech level, exclamative in addressee-insensitive speech level: endings like this are, before being used in a concrete utterance, ambiguous in these categories. It has at least three different sets of agreement properties and resulting categories, which may be represented as three different lexical readings.

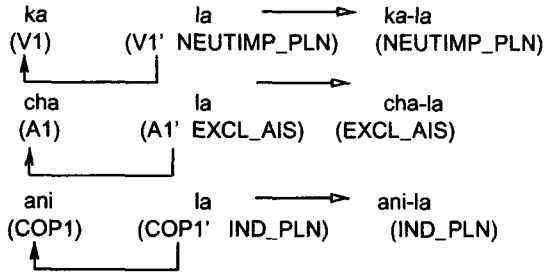

Figure 3. Combinatorics of the ending "la" (simplified). The ending categories consist of two segments of which the first determines each the word class and its form it can be attached to (in the above example the base form of verb, adjective and copula denoted by V1', A1' and COP1'), the second the result. The category NEUTIMP_PLN stands for 'neutralised imperative in plain speech level', the category EXCL_AIS 'exclamative in addressee-insensitve speech level' the category IND_PLN 'indicative in plain speech level'.

From the viewpoint of computational linguistics, these three different categories can be handled either by means of three different lexical entries with equal surfaces as

[la (V1' NEUTIMP_PLN) ENDING]
[la (A' EXCL_AIS) ENDING]
[la (COP' IND_PLN) ENDING]

or one lexical entry with one surface and three alternative categories as

[la (V1' NEUTIMP_PLN) (A' EXCL_AIS) (COP' IND_PLN) ENDING].

The latter is the so-called multicat notation first explored in the LAP-system, an implementation of LA-morphology developed by Schüller (1994). This multicat solution has the advantage that only one lexical lookup is necessary. Furthermore, instead of branching immediately into three different parallel paths of syntactic analysis, the alternatives coded in the multicat may be processed in one branch until the result segments come into play.

The use of multicats requires that format and implementation of the combi-rules in morphology and syntax be extended to handle the alternatives coded in the new kind of categories. On the one hand, such an extension of the combi-rules capabilities leads to a version of LA-grammar which differs from the original LAP-system. On the other hand, an LA-grammar using multicats can always be reformulated as one in the orignal format using several lexical readings. Thus the use of multicats does not change the theoretical status as compared to a correspongding version without multicats.

Another subtheoretical variant of LAG was the definition of categories and rule patterns as attribute-value structures written by Beutel (1996), called Malaga. In Malaga the list-based patterns of LAP are replaced by hierarchically structured patterns. Malaga uses multicats and attribute-value structures at the same time. The list-based and the feature-based presentation are equivalent. For the theoretical presentation of the combinational process the original list-based format was used (see figure 3). The automatic word form recognition system for Korean KMM has been also implemented using Malaga. For this reason the feature-based format will be used in the following exmaples of lexical entries (see figure 4).

This paper proposes a processing mechanism using several attributes in order to represent the relevant categories denoted by the parts of the sentence ender: the attribute MOOD relates to the MOOD-markers and can have a value IND, RET, REQ and PROP and EXCL. The attribute ST (sentence type) refers to the sentence type denoted by a ST-marker and its value is either an atomary symbol like DECL[10]. The attribute SL (= speech level) is used for all parts of a sentence ender and its value is either an atomary symbol like INTIM. The proposed lexical entries of the parts of the sentence ender *-p-*, *-ni-*, *-ta* as followed.

```
[Surface: "p",
 POS: <Ending>,
 Pre: <[Yield: [Type: AH,
                SL: DEF],
        Check: [StemForm: base,
                Syllable: open,
                LastPOS: <Verb, Copula, Adj>]]>,
 POS: Ending,
 AlloMorph: "p"]
[Surface: "ni",
 POS: <Ending>,
```

[10]For further discussion about various value types in MALAGA see Beutel (2001).

```
 Pre: <[Yield: [Type: IND_DEF,
                Mood: IND,
                ST: INT,
                SL: DEF],
        Check: [LastPOSType: <AH>]]>]
[Surface: "ta",
 POS: Ending,
 Pre: <[Yield: [Type: DECL_DEF,
                ST: Decl,
                SL: DEF,
                SESlot: <<AH, IND_DEF>, <AH, RET_DEF>>],
        Check: [LastPOSType: IND_DEF&RET_DEF]],
       [Yield: [Type: PROP_BLN,
                ST: PROP,
                SL: BLN,
                SESlot: <<AH, REQ_BLN&REQ_DEF>>],
        Check:  [LastPOSType: REQ_BLN&REQ_DEF]]>]
```

Figure 4. The simplified Lexical Entries for the Endings "p", "ni", and "ta". The presentation is based on the attribute-value structure and applies multicat notion discussed above. The attribute Pre holds a list of eventually more than one syntactical readings of the respective endings.

At the last stage of analysis the multicat values of the attribute SL of the parts of the sentence ender are to be disambiguated so that finally only the "non-ambiguous" value remains. According to the fact that the ST-Marker is the obligatory part of the sentence ender, this paper uses an additional attribute SESlot (sentence ender slot). It checks, after a verb form is completely analysed, whether each slot of the sentence ender are filled with the ending of appropriate category. Its value is a list of properties like ⟨AH, RET_DEF⟩ or ⟨e⟩ that must be established by the preceeding parts of the sentence ender.

The result of the analysis of the verb form "cwu-p-ni-ta" is shown in the following figure.

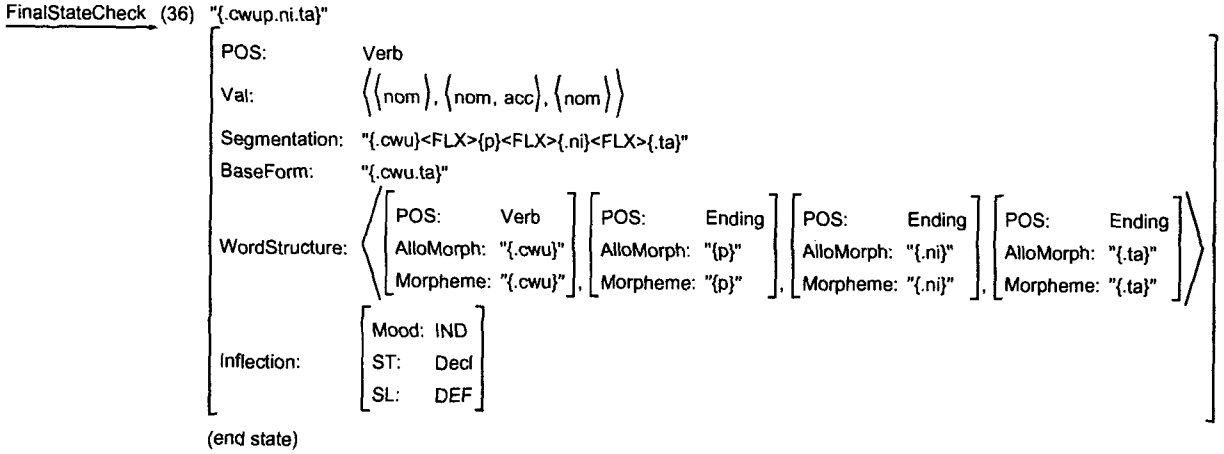

Figure 6. The analysis of the verb form "cwu-p-ni-ta".

This system works modularly in that it is only necessary to check the values of the respective attributes of each ending in order to find out the speech level, mood and sentence type, while

the attribute SESlot is used to check whether all the slots are appropriately filled with.
For evaluation purposes, 100 predicative verb forms were chosen randomly from the manually tagged corpus of KAIST and were analysed by the KMM applying this system. Since there are - as known to the author so far - no corpora which are morphologically analysed in these aspects, the results were manually evaluated by the author. The results are seen in table 5.

Analysis Result	%
correctly analysed	85
syntactically correct, incorrect segmentation	2
ambiguous with correct and incorrect reading	6
syntactically incorrect	1
non-existant ambiguity	1
not recognised	6

Table 5: Results of the Analysis of 100 Randomly Chosen Predicative Verb Forms from the manually tagged KAIST Corpus.

4 Conclusions

The predicative verb forms in Korean denote the speech level, the mood and the sentence type, what requires a complicated system to analyse them. This paper proposed a lexical description of predicative verbal endings which may work without requiring complicated system, and a system that uses this description to analyse predicative verb forms. On a subset of 100 predicative verb forms randomly chosen from the manually tagged KAIST corpus, it analyses 85% correctly. An evaluation based on a more extensive data is planned. A greater part of the ambiguous and incorrect analyses are expected to be eliminated by improving the lexical description of the endings. That is the work that is get to be done.

References

Beutel, Björn (2001) *Malaga 5.6. User's and Programmer's Manual.* (HTTP://www.linguistik.uni-erlangen.de/bjoern/malaga.html)

Hausser, Roland (1989a) *Computation of Language. An Essay on Syntax. Semantics and Pragmatics in Natural Man-Machine Communication.* Springer press. Symbolic Computation: Artificial Intelligence.

Hausser, Roland (1989b) "Principles of Computational Morphology," Laboratory of Computational Linguistics. Carnegie-Mellon University.

Hausser, Roland (1999) *Foundations of Computational Linguistics.* Springer Press.

Kim, Chakywun (1999) *Studies on the Temporal Structure and Aspect of Korean Language.* Tayhak Press.

Lee, Heeja (1999) *Lexicon of Korean Verbal Endings.* Hankwukmhunhwa Press.

Leidner, Jochen (1997) *Linksassoziative morphologische Analyse des Englishcne mit stochastischer Disambiguierung.* Master's thesis, Department of Computational Linguistics, Friedrich-Alexander University of Erlangen-Nürnberg.

Lorenz, Oliver (1997) *Automatische Wortformerkennung für das Deutche im Rahmen von MALAGA.* Master's thesis, Department of Computational Linguistics, Friedrich-Alexander University of Erlangen-Nürnberg.

Marlow, Kerstin (1999) *Automatische Wortform Analyse.* Master's thesis, Department of Computational Linguistics, Friedrich-Alexander University of Erlangen-Nürnberg.

Sohn, Ho-Min (1999) *The Korean Language.* Cambridge University Press.