GRAMMAR EFFICIENCY OF PARTS-OF-SPEECH SYSTEMS

A thesis submitted
to Kent State University in partial
fulfillment of the requirements for the
degree of Master of Science

by
Barbara Miller
May 2011
Thesis written by

Barbara Miller

B.S., University of Akron, USA 2006

Approved by

___________________________________ , Chair, Master Thesis Committee
Dr. Relja Vulanović

___________________________________ , Members, Master Thesis Committee
Dr. Stephen Gagola

___________________________________
Dr. Austin Melton

Accepted by

___________________________________, Chair, Department of Mathematics
Dr. Andrew Tonge

___________________________________, Dean, College of Arts and Sciences
Dr. Timothy Moerland
# TABLE OF CONTENTS

TABLE OF CONTENTS ........................................................................................................ iii

TABLE OF TABLES ........................................................................................................ vi

TABLE OF FIGURES ......................................................................................................... viii

CHAPTER 1 Introduction ................................................................................................. 1

1.1 Problem Description and Motivation ......................................................................... 1

1.2 Research Aim and Methods ...................................................................................... 2

1.3 Organization of the Thesis ........................................................................................ 3

CHAPTER 2 Word Classes ............................................................................................... 5

2.1 Introduction to Word Classes .................................................................................... 5

2.2 Word Classes in Linguistics ..................................................................................... 5

2.2.1 Introduction .......................................................................................................... 5

2.2.2 Classification of Nouns, Verbs, and Adjectives ...................................................... 6

2.2.3 Classification of Adverbs ..................................................................................... 10

2.3 Hengeveld’s Approach to Word Classes .................................................................. 11

2.4 Acceptance of Hengeveld’s Approach .................................................................... 16

CHAPTER 3 Parts-of-speech Systems ............................................................................. 17

3.1 Introduction to Parts-of-speech Systems .................................................................. 17

3.2 The HRS Approach to PoS Systems ....................................................................... 19

3.3 A Formal Description of Parts-of-speech Systems ................................................... 21
3.3.1 Constraint 1........................................................................................................... 21
3.3.2 Constraint 2........................................................................................................... 25

3.4 Formal Grammars with Regulated Rewriting .......................................................... 26

3.4.1 Description of Phrase-structure Grammars with Regulated Rewriting ............... 26
3.4.2 Phrase-structure Grammar when \( n = 4 \) ................................................................. 27
3.4.3 Phrase-structure Grammar with \( n = 3 \) ................................................................. 29
3.4.4 Phrase-structure Grammar for \( n = 2 \) ................................................................. 29
3.4.5 Phrase-structure Grammar for \( n = 1 \) ................................................................. 30

3.5 Complexity in Phrase-structure Grammars with Regulated Rewriting ................. 30

CHAPTER 4 Grammar Efficiency and Complexity ......................................................... 33

4.1 Description of Grammar Complexity and Efficiency ............................................. 33
4.2 Parsing .................................................................................................................... 34

4.2.1 Description of Parsing and Rules ...................................................................... 34
4.2.2 Parse 3.4Rr and 3.4MM ..................................................................................... 36
4.2.3 Parsing of 3.4Rr ............................................................................................... 36
4.2.4 Parsing of 3.4MM ............................................................................................. 37
4.2.5 Calculations for all PoS system types ............................................................... 38

4.3 The Efficiency Formula ......................................................................................... 40

4.4 Max \( Q \) .................................................................................................................. 42

4.5 Min \( Q \) .................................................................................................................. 43

4.5.1 Reason for using Min \( Q \) .................................................................................. 43
4.5.2 Finding the Min $Q$ of 3.4MM ................................................................. 44
4.5.3 Notes on Min $Q$ ......................................................................................... 45
4.6 Efficiency Values ....................................................................................... 45
4.7 Comparison to Phrase-structure Grammar Efficiency ................................. 47

CHAPTER 5 Statistical Analysis ........................................................................ 48
5.1 Introduction ............................................................................................... 48
5.2 Preparing the Data .................................................................................. 48
5.3 Fitting the Equations ............................................................................... 51
5.4 Analysis of Results ................................................................................ 52

CHAPTER 6 Conclusion .................................................................................. 57
6.1 Main Results ............................................................................................ 57
6.2 Future Research Directions ..................................................................... 58

REFERENCES .................................................................................................. 60
TABLE OF TABLES

Table 2.1: Semantic Classification of Nouns, Adjectives, and Verbs ........................................... 7
Table 2.2: Labeling of Grammatical Slots .................................................................................. 13
Table 3.1: Examples of Rigid PoS Systems .................................................................................. 17
Table 3.2: Examples of Flexible PoS Systems ........................................................................... 19
Table 3.3: HRS PoS System Classification .................................................................................. 20
Table 3.4: Possible Combinations of Propositional Slots ......................................................... 22
Table 3.5: Position of HRS PoS Systems in the Two-number Classification ......................... 22
Table 3.6: Comparison of Attested 2.4 PoS Subtypes ............................................................... 23
Table 3.7: PoS System Types ................................................................................................... 23
Table 3.8: PoS System Type 2.4 Subtypes ............................................................................... 23
Table 3.9: PoS System Type 3.4 Subtypes ............................................................................... 24
Table 3.10: PoS System Type 2.3 Subtypes ........................................................................... 24
Table 3.11: PoS System Types Satisfying C1 and C2 .............................................................. 25
Table 3.12: Rules for 3.4Rr and 3.4rp ..................................................................................... 31
Table 4.1: PoS System Values ................................................................................................... 39
Table 4.2: κ Values .................................................................................................................... 43
Table 4.3: Efficiency Values of the PoS System Types ............................................................. 46
Table 5.1: Minimum, Midpoint, and Maximum Efficiency Values .......................................... 49
Table 5.2: 2D Representation of Min $Eff$ .......................................................... 50
Table 5.3: 2D Representation of Max $Eff$ .......................................................... 50
Table 5.4: 2D Representation of Mid $Eff$ .......................................................... 50
Table 5.5: Results for the Sigmoidal Equation .................................................... 52
Table 5.6: Results for the Linear Equation .......................................................... 52
Table 5.7: Results for Equations (3) and (4) ....................................................... 55
### Table of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 2.1</td>
<td>Theoretically Possible Word Classes with 4 Propositional Functions</td>
<td>14</td>
</tr>
<tr>
<td>Figure 2.2</td>
<td>Theoretically Possible Word Classes with 3 Propositional Functions</td>
<td>15</td>
</tr>
<tr>
<td>Figure 5.1</td>
<td>The Nonlinear Fit of Min $\text{Eff}$</td>
<td>54</td>
</tr>
<tr>
<td>Figure 5.2</td>
<td>The Nonlinear Fit of Max $\text{Eff}$</td>
<td>54</td>
</tr>
</tbody>
</table>
CHAPTER 1

Introduction

1.1 Problem Description and Motivation

The aim of this thesis is to determine the validity of the efficiency formula for the two-number parts-of-speech (PoS) classification system and to determine the validity of this method of classification by testing for correlation between the efficiency of a PoS system type and its two-number label. The PoS system’s efficiency is thought of in the sense of machine efficiency: the ratio of useful output to useful input, where the grammar of the language type is the machine. This efficiency does not take the human processor into account, just the rules of the PoS system’s grammar.

In 2004, Hengeveld, Rijkhoff, and Siewierska introduced an abstract PoS system classification based on a PoS classification of word classes and a 50 language sample. This classification system takes into account only the propositional functions and word classes present in a language. Propositional functions refer to whether a word functions as the head or modifier of the predicate phrase or referential phrase. This PoS system classification includes seven main types and six intermediate types. The proposed system does not classify all attested languages, as shown by Hengeveld and Van Lier (HvL 2010). This inspired a modification to the PoS system classification by Vulanović (Vulanović 2008). The final PoS system classification created is a two-number
classification system which allows for 28 theoretically possible PoS system types (including subtypes). The use of two constraints lowers the number of theoretically possible PoS system types to 17, 12 of which are currently attested.

There exists disagreement among linguists about whether this approach to language classification is acceptable or not. This disagreement traces back to the definition of word class which is used to create the classification system. The classification of word classes is not agreed upon by linguists. This disagreement is due to the existence of multiple classification methods for word classes.

A mathematical analysis of the classification system can be employed to determine the validity of the classification. A grammar efficiency formula based on the defining characteristics of the PoS system types can be used to assign a numerical value (calculated through parsing) to each system. If a correlation exists between the PoS system type and the two-number classification label this will confirm that this form of classification is a reasonable approach to language classification. This, in turn, would give more validity to Hengeveld’s approach to word class classification.

1.2 Research Aim and Methods

This research investigates the efficiency of the 17 PoS system types in the two-number classification. The main focus of the research is twofold. Initially, the research aims to determine whether the efficiency formula created for this PoS system classification is reasonable. This is done by evaluating the efficiency of each PoS system type, including subtypes, using the efficiency formula. The formula is valid if each PoS
system (sub)type has a unique interval of efficiency values except (sub)types which are maximally efficient or have equivalent structures.

If the efficiency formula is found to be valid, then the PoS system classification can be analyzed. The aim is to determine whether there is a correlation between the PoS system type label and the efficiency calculated by the efficiency formula. An efficiency interval will be determined for each PoS system type using the efficiency formula. The efficiency of the PoS system (sub)type is determined with consideration to different word order rules since the efficiency value depends on how free the word order is. The variation of word order rules results in the interval of efficiency values. The interval will include the minimum efficiency and maximum efficiency. The midpoint of the interval will also be considered. Linear and non-linear regression will be used to fit a 3D surface to the data. If the adjusted coefficient of multiple determination value is .70 or higher, then a correlation exists.

1.3 Organization of the Thesis

In Chapter 2, the choice of word class classification is looked at. A general linguistic method is discussed first. This method includes different sets of criteria to classify word classes. Hengeveld’s approach, which is ultimately used, is considered next. Hengeveld’s approach classifies words according to parts-of-speech and can be considered as an extension of one of the sets of criteria presented in the linguistic approach.

In Chapter 3, the PoS system classification is investigated. A brief overview of the development of the classification is presented first. The final form used in this thesis
is then given with criteria which restrict the number of system types. In addition to the PoS system classification, formal grammars with regulated rewriting are also considered. Within the theory for formal grammars, definitions for complexity (and therefore efficiency) already exist; however, these are not useful in relation to this research, as is shown.

In Chapter 4, a previously established efficiency formula is introduced. Each of the variables in the equation and the methods involved for finding these values are discussed. All of the values needed for the efficiency formula are determined. The minimum and maximum efficiency values are then calculated for each PoS system type.

In Chapter 5, a range of efficiency values is determined for each PoS system type as a whole. This set of ranges is then broken into three sets of data: minimum efficiency, maximum efficiency, and midpoint efficiency. Two types of regressions, linear and non-linear, are performed on each of these data sets. The results of the regression analysis are used to determine if a correlation exists between the efficiency of the PoS system types and the two-number labeling.

In Chapter 6, a brief overview of the work performed is given. The conclusions of the research are then stated. Suggestions for future research on the topic are also presented.
CHAPTER 2

Word Classes

2.1 Introduction to Word Classes

Word classes are a classification of words within a language into groups with similar features. Word classes are also known as parts-of-speech when the classification is based on syntactic methods. In this chapter, two approaches are described: a standard linguistic approach, which uses a variety of methods, and Hengeveld’s approach, which classifies words classes as parts of speech.

2.2 Word Classes in Linguistics

2.2.1 Introduction

In linguistics word classes can be categorized by semantic criteria, morphological criteria, and syntactic criteria (Givón, 1993: 51-83). The semantic criteria categorize words by the general meaning coded into the words. The morphological criteria categorize words by the types of bound morphemes associated with the word. The syntactic criteria categorize words by the typical position the words in the class take in a sentence. The classification of word classes can include one set of criteria or all three sets of criteria. The approach taken in Givón’s (1993) English Grammar uses all three sets of criteria.
In English, the main word classes are nouns, verbs, adjectives, and adverbs. There are also ten minor word classes: prepositions, conjunctions, subordinators, pronouns, determiners, quantifiers, numerals, ordinals, auxiliaries, and interjections. These word classes can be broken down into more precise subclasses. For example, verbs in English can be divided into the following subclasses based on which words or phrases need to follow them: intransitive, one-part transitive, two-part transitive, the verb ‘be’, and linking (Morenberg 2002: 1). For the purpose of this thesis, only the four main word classes are considered. After this chapter, the subcategories of each word class are not considered.

The standard word classes in linguistics are based on natural classes. This means that the word classes are “created by preconceptions and cognition of living organisms” (Givón, 1993: 52). Within each word class, a majority of the members are considered to be “prototypical” members of the class. Some words within the word class vary from the prototype and can even be considered a member of multiple word classes. This is because natural classes have fuzzy boundaries which allow the word classes to overlap with each other. The word classes can be described by a frequency distribution curve where the prototypical members are those that fall within a certain range of the population mean.

2.2.2 Classification of Nouns, Verbs, and Adjectives

The semantic classification of word classes can be defined by four main criteria. The criteria are

a) Temporal stability (rate of change over time)

b) Concreteness (physicality)
c) Compactness (degree of spatial scatter)
d) Complexity (number of defining features)

These criteria specify the prototypical word within each word class (Givón, 1993: 54).
This list of criteria only works for three of the four main word classes. Table 2.1 offers a comparison of nouns, verbs, and adjectives using the semantic criteria. The classification of adverbs is more complex and is described later.

<table>
<thead>
<tr>
<th></th>
<th>Temporal Stability</th>
<th>Concreteness</th>
<th>Compactness</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Nouns</strong></td>
<td>Slow Changing</td>
<td>Physical</td>
<td>Compact</td>
<td>Multi-featured</td>
</tr>
<tr>
<td><strong>Adjectives</strong></td>
<td>Slow Changing and</td>
<td>Physical or Abstract</td>
<td>Drawn out</td>
<td>Single-featured</td>
</tr>
<tr>
<td></td>
<td>Temporary</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Verbs</strong></td>
<td>Fast Changing</td>
<td>Motion or Action</td>
<td>Compact or</td>
<td>Intermediate</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Drawn out</td>
<td></td>
</tr>
</tbody>
</table>

Table 2.1: Semantic Classification of Nouns, Adjectives, and Verbs

To further explain the meaning of Table 2.1, consider nouns. The prototypical noun is slow changing, physical, compact, and multi-featured. “Slow changing” means that a few characteristics of the noun may change over time, but the majority remains the same. For example, water may change physical state over time from gas to liquid to solid, but it remains water. “Physical” means that the noun is something that can be touched or seen. “Multi-featured” means that the classification of the noun (or noun class) is based on many characteristics.

The definition of a noun taught in primary school of “a person, place, or thing” describes the prototypical noun. Nouns such as man, library, and table fall under this definition. These nouns are slow changing, physical, and multi-featured. There are many nouns which do not fit nicely into the primary school definition. Nouns which deviate from the prototype in the sense of concreteness are abstract and temporal instead of
physical (Givón, 1993: 55). Love and size are abstract nouns, i.e., they have no physical entity associated with them. Evening and anniversary are temporal nouns, i.e., they only exist in time.

The morphological criteria deal with the type of bound morphemes that attach to the words within each word class. A morpheme is the smallest unit of a language that has meaning (Fromkin et al. 2003: 69). Morphemes are either grammatical or derivational, see below. Bound morphemes, which can be affixes or separate words, must be attached to a word. By the morphological criteria, words are organized into classes according to which morphemes they use and how the morpheme changes the word in meaning or function (Givón, 1993: 58).

Grammatical morphemes specify characteristics such as the number and tense of the word (Givón, 1993: 59). Nouns have grammatical morphemes such as plural markers, gender markers, and prepositions which mark the role of indirect object. Verbs have grammatical morphemes such as tense markers, negation, and subject agreement. Adjectives have grammatical morphemes such as comparative and superlative markers. Examples of grammatical morphemes for nouns in English are:

1. Plural marker
   a. table-s
   b. ox-en

2. Prepositions
   a. on the table
   b. by the car
More grammatical morphemes exist; this is just a brief overview.

Derivational morphemes are used to create new words from existing words (Givón, 1993: 59). Attaching a derivational morpheme to a pre-existing word can result in the new word fitting into a different word class than the pre-existing word. For example, there are morphemes which change nouns, verbs, and adjectives into nouns. Examples of such derivational morphemes are:

1. Verb-to-noun:
   create → creat-ation

2. Adjective-to-noun:
   narrow → narrow-ness

3. Noun-to-noun:
   king → king-dom

Verbs and adjectives can also be classified by derivational morphemes. More derivational morphemes exist than those stated above.

The syntactic criteria consider which position in the sentence a word fits into (Givón, 1993: 57). The positions are called syntactic slots or grammatical slots. Where a word class fits into a sentence is called its propositional function. Each slot can be filled by either a single word from a specific word class or a phrase. Nouns can fill slots designated for subject, direct object, indirect object, and predicate. Adjectives can fill slots for predicate adjectives or modifying adjective. Verbs fill the slot designated as the core of the sentence, but are further classified by what fills the remaining slots in a sentence (Givón 1993: 90).
An example of a sentence with the four main word classes is

The giant\textsubscript{a} dog\textsubscript{N} slept\textsubscript{V} peacefully\textsubscript{m}.

In the sentence above, only words from the specified classes (in English) can fill the designated slots. The slot marked with subscript a can only use words (or phrases) that function as adjectives. The slot with subscript N uses words (or phrases) that function as nouns. The slot with subscript V uses words (or phrases) that function as verbs. The slot with subscript m can use words (or phrases) that function as manner adverbs. This represents the most complex sentence structure that is dealt with in this thesis.

2.2.3 Classification of Adverbs

The word class of adverbs is the least homogenous word class and, as a result, classification of words into the entire class cannot be determined by a list of set criteria. This makes adverbs difficult to define (Givón 1993: 71). Generally, adverbs modify the verb in a sentence. Some adverbs, however, modify the sentence as a whole. Some of the subclasses of adverbs are: manner, time, frequency, epistemic, evaluative, and emphatic. This paper considers only manner adverbs because they modify only the verb.

Manner adverbs modify the meaning of the verb by adding further description to the action of the verb. The way in which an adverb modifies a verb depends on the meaning of the verb. Because of this, adverbs have a wide semantic range (Givón 1993: 71). Manner adverbs may be a single word (1.) or constructed as a prepositional phrase (2.):

1. She sang \textit{softly}.

2. He dropped the glass \textit{on purpose}.
They are also syntactically heterogeneous and can occupy different positions in a sentence.

1. Beginning of the clause:
   
   *Easily* she opened the jar.

2. In front of the verb:
   
   She *easily* opened the jar.

3. After the verb:
   
   She opened the jar *easily*.

Manner adverbs can also be derived from adjectives with the suffix *–ly*.

1. Adjective-to-adverb:

   weak \(\rightarrow\) weak-*ly*

### 2.3 Hengeveld’s Approach to Word Classes

Hengeveld’s approach to classifying word classes is a variation of the syntactic criteria classification described above. It is based on an abstract and simplified sentence structure in which only four propositional slots are considered (HRS 2004: 530). These slots are the obligatory heads of the predicate phrase and referential phrase and the optional modifiers of the predicate phrase and referential phrase. This simplification makes the classification of parts-of-speech systems (discussed later) easier to manage.

The following sentence, similar to the previously introduced one, can be used to illustrate the four propositional slots:

Giant dogs slept peacefully.
The referential phrase in this sentence is *giant dogs*; its head is *dogs* and its modifier is *giant*. The predicate phrase in this sentence is *slept peacefully*; its head is *slept* and its modifier is *peacefully*.

Generically, the head of the predicate phrase is a verb, the modifier of the predicate phrase is a manner adverb, the head of the referential phrase is a noun, and the modifier of the referential phrase is an adjective (HRS 2004: 530). As mentioned above, only manner adverbs are considered because they modify only the verb in the sentence and not the entire sentence. This labeling is accurate in certain languages such as English, which can be used as a good descriptive example. However, many languages of the world have different word classes than English. Many of these languages do not have a separate word class for each propositional slot.

As an illustration of a language with a word class that fills multiple propositional slots, consider Turkish. It has a wide class of words that can be used as a noun, an adjective, or an adverb. The Turkish word for “beauty”, *güzel*, is such a word. The following examples of its usage are taken from Hengeveld & van Lier (2010: 130):

*güzel-im*

beauty-1POSS

‘my beauty’

*güzel bir kopek*

beauty ART dog

‘a beautiful dog’
The four grammatical slots, or propositional slots, can be viewed as the following matrix, which also shows the notation used for the propositional slots (HRS 2004: 530).

<table>
<thead>
<tr>
<th></th>
<th>Head</th>
<th>Modifier</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predicate Phrase</td>
<td>P</td>
<td>p</td>
</tr>
<tr>
<td>Referential Phrase</td>
<td>R</td>
<td>r</td>
</tr>
</tbody>
</table>

Table 2.2: Labeling of Grammatical Slots

The above $2 \times 2$ matrix of grammatical slots can be used to describe all possible word classes. Each element in the matrix is optional. For example, a word class can contain only words that function as the head of the predicate phrase, or it can contain words functioning just as modifiers. There are $\binom{4}{1} + \binom{4}{2} + \binom{4}{3} + \binom{4}{4} = 4 + 6 + 4 + 1 = 15$ different, theoretically possible word classes when all four grammatical slots are considered. These 15 word classes are, with names where applicable, presented in Figure 2.1. These matrices also introduce the notation which will be used for word classes from now on.

The word classes for verbs, nouns, adverbs, and adjectives perform as they do in English. The word class Heads can fill the propositional slots designated for the head of the phrase, Predicates can fill any slot in the predicate phrase and Nominals can fill any slot in the referential phrase. Contentives are a class of words that can fill any propositional slot. Non-verbs can fill any propositional slot that is not designated for the head of the predicate phrase. The variable X is just a place holder, it does not stand for
the same word class in each matrix in which it is used. The word classes described in matrices (9), (10), and (12-14) are unattested and, according to the linguistic custom, are marked with a ‘*’.

<table>
<thead>
<tr>
<th>V</th>
<th>-</th>
<th>-</th>
<th>N</th>
<th>m</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>N</td>
<td>-</td>
</tr>
<tr>
<td>V</td>
<td>-</td>
<td>-</td>
<td>N</td>
<td>m</td>
</tr>
</tbody>
</table>

(1) Verbs
(2) Nouns
(3) Manner Adverbs

<table>
<thead>
<tr>
<th>-</th>
<th>-</th>
<th>-</th>
<th>H</th>
<th>-</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>a</td>
<td>-</td>
<td>H</td>
<td>-</td>
</tr>
</tbody>
</table>

(4) Adjectives
(5) * Heads
(6) Predicatives

<table>
<thead>
<tr>
<th>-</th>
<th>-</th>
<th>X</th>
<th>-</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>-</td>
<td>-</td>
<td>X</td>
</tr>
</tbody>
</table>

(7) Nominals
(8) Modifiers
(9) *

<table>
<thead>
<tr>
<th>-</th>
<th>Y</th>
<th>-</th>
<th>-</th>
<th>Λ</th>
</tr>
</thead>
<tbody>
<tr>
<td>Y</td>
<td>-</td>
<td>-</td>
<td>Λ</td>
<td>Λ</td>
</tr>
</tbody>
</table>

(10) *
(11) Non-Verbs [4]
(12) *

<table>
<thead>
<tr>
<th>X</th>
<th>X</th>
<th>-</th>
<th>-</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>X</td>
<td>-</td>
<td>-</td>
<td>Z</td>
</tr>
</tbody>
</table>

(13) *
(14) * Non-Nouns [4]
(15) Contentives [4]

Figure 2.1: Theoretically Possible Word Classes with 4 Propositional Functions

The 15 word classes described above are the word classes possible when all four propositional slots are present. However, there are languages which have fewer than four propositional slots. Tagalog, for instance, is missing the slot for the modifier of the predicate phrase (HvL 2010: 143). For languages such as these, there are four additional word classes to consider, which are presented in Figure 2.2.
Therefore, there are three versions of Contentives: C[4] in Figure 2.1 and C[3p] and C[3r] in Figure 2.2. In this notation the number in brackets is the number of propositional slots and the letter after the 3 indicates which modifier slot is present in the language. However, from now on, only C will be used for all versions of Contentives. This is because they all have the same function in the sense that they can be used in any existing propositional slot. Similar reasoning applies to the two versions of non-verbs and the two versions of non-nouns.

Languages with only two propositional slots (which have to be the two mandatory heads) do not provide any additional word classes. The only word classes that they can use are Verbs, Nouns, and Heads.

When a language is missing a modifier slot, the corresponding head can still be modified. The modification is performed periphrastically, that is, the modification is derived from a word or phrase that fulfills a different propositional slot. The language Krongo only has the propositional slots for the head of the predicate and referential phrase; it has no modifier slots. To modify the head of the referential phrase, a relative clause is formed from a member of the verb class (HvL 2010: 131). This is illustrated by the following two sentences.
Álímì                                      biitì.

MASCULIN.IMPERFECTIVE.be.cold water

‘The water is cold.’

biitì ŋ-álímì

water CONNECTIVE-MASCULIN.IMPERFECTIVE.be.cold

‘cold water’ (lit. ‘water that is cold’)

2.4 Acceptance of Hengeveld’s Approach

Hengeveld’s approach to the classification of word classes is not universally accepted. This is mainly because linguists do not agree on how to define and classify word classes. The linguistic method presented above is only one of many classification methods. For example, linguists such as Baker do not consider adverbs to be separate from adjectives (Vulanović 2008: 53). From their point of view, certain word classes described by Hengeveld simply do not exist. However, Hengeveld’s approach can be viewed as a variation or expansion of Givón’s syntactic description of word classes. Therefore, Hengeveld’s approach is a reasonable option and is used throughout this thesis.
CHAPTER 3

Parts-of-speech Systems

3.1 Introduction to Parts-of-speech Systems

Different languages have different structures and use word classes, or parts of speech (PoS), in different ways to fill the four propositional slots. A PoS system is defined by the propositional slots existing in a language and by the word classes that the language uses to fulfill these slots. As mentioned previously, sentences of the form

Little$_a$ dogs$_N$ run$_V$ quickly$_m$

are standard in English, which has a differentiated PoS system. All languages with a differentiated PoS system have a unique word class for each of the four propositional slots (HRS 2004: 531). This implies that there is only one form of the differentiated PoS system. English is also known as a rigid PoS system since each word class fills only one propositional slot (HRS 2004: 532). There are three other possible rigid PoS systems, each with less than four propositional slots. Examples of languages with rigid PoS systems are described in the table below.

<table>
<thead>
<tr>
<th>Language</th>
<th>Head of Predicate Phrase</th>
<th>Head of Referential Phrase</th>
<th>Modifier of Referential Phrase</th>
<th>Modifier of Predicate Phrase</th>
</tr>
</thead>
<tbody>
<tr>
<td>English</td>
<td>V</td>
<td>N</td>
<td>a</td>
<td>m</td>
</tr>
<tr>
<td>Wambon</td>
<td>V</td>
<td>N</td>
<td>a</td>
<td>-</td>
</tr>
<tr>
<td>Krongo</td>
<td>V</td>
<td>N</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>unattested</td>
<td>V</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 3.1: Examples of Rigid PoS Systems
The unattested language in Table 3.1 has an exceptional structure because the head of the referential phrase is missing while both of the heads are obligatory otherwise. The language Tuscarora is the closest attested language. The following Tuscarora sentences from (HRS 2004) illustrate how verbs can be modified periphrastically to function in the other propositional slots:

```
ra-kwatihs
M-young
‘boy’ (‘He-is-young.’)
```

```
ka-teskr-ahs
NHUM-stink-ASP
‘goat’ (‘It stinks.’)
```

```
ra-kwatihs wa-hr-arketnoʔ ka-teskr-ahs
M-young PAST-M-look.at-ASP NHUM-stink-ASP
‘The boy looked at the goat.’ (‘He is young, he looked at it, it stinks.’)
```

We can therefore say that the unattested language of Table 3.1 is partially attested in Tuscarora.

Some languages, such as Turkish (see example in chapter 2.3), do not have separate word classes for each propositional slot. These languages are said to have flexible PoS systems (HRS 2004: 532). Languages with a flexible PoS system have at
least one word class which can function in two or more propositional slots. However, these languages do not need to have all four propositional functions present (an example of this being Tagalog). Unlike rigid PoS systems, flexible PoS systems have some variation with respect to the word classes present and the propositional functions they fulfill. Table 3.2 presents some examples of languages with flexible PoS systems.

<table>
<thead>
<tr>
<th>Language</th>
<th>Head of Predicate Phrase</th>
<th>Head of Referential Phrase</th>
<th>Modifier of Referential Phrase</th>
<th>Modifier of Predicate Phrase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Samoan</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Warao</td>
<td>V</td>
<td>Λ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ngiti</td>
<td>V</td>
<td>N</td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>Tagalog</td>
<td>C</td>
<td></td>
<td></td>
<td>-</td>
</tr>
</tbody>
</table>

Table 3.2: Examples of Flexible PoS Systems

3.2 The HRS Approach to PoS Systems

The original PoS system classification set up by HRS is based on the word classes described in chapter 2.3 and the propositional slots fulfilled by these word classes. The PoS system classification does not consider word order as a defining feature. The system types are based only on the propositional functions which are included (or missing) in the languages and the word classes that fulfill these functions.

HRS showed that the way languages have or do not have word classes to fill the propositional slots is based upon the hierarchy:

H1: \( P > R > r > p \)

Within the hierarchy, P, R, r, p are as in Table 2.2. The further to the left that a propositional function is in this hierarchy, the more priority it has and the more likely it is that there will be a separate word class for that slot. All PoS Systems must include the
slot for head of the predicate phrase and a word class that functions as the head of the predicate phrase. If a rigid language has a word class which fulfills the modifier of a referential phrase, then it will have separate word classes which fulfill the heads of the referential and predicate phrases. However, if a rigid language is lacking a word class which fulfills the modifier of the referential phrase, it will also lack a word class to fulfill the modifier of the predicate phrase. For flexible languages, if a word class fulfills any of the propositional functions, the same word class is likely to fulfill all propositional functions to the right. (HRS 2004: 533).

H1 specifies which slots are filled by the word classes present in the language. A table of PoS system types can be created using the hierarchy (cf. HRS 2004: 537). All PoS system types are attested except for HRS7 (as indicated by the asterisk in Table 3.3), which is only partially attested (Vulanović 2008: 54), as is mentioned in Chapter 3.1.

<table>
<thead>
<tr>
<th>Parts-of-speech System</th>
<th>HRS type</th>
<th>Head of Predicate Phrase</th>
<th>Head of Referential Phrase</th>
<th>Modifier of Referential Phrase</th>
<th>Modifier of Predicate Phrase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flexible</td>
<td>1</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>V</td>
<td>Λ</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>V</td>
<td>N</td>
<td>a</td>
<td>m</td>
</tr>
<tr>
<td>Rigid</td>
<td>4</td>
<td>V</td>
<td>N</td>
<td>a</td>
<td>m</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>V</td>
<td>N</td>
<td>a</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>V</td>
<td>N</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>*7</td>
<td>V</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 3.3: HRS PoS System Classification

Some languages cannot be classified as any of the PoS system types described in the table. These languages fall in-between the system types. The classification described in Table 3.3 was extended to include these intermediate systems (HRS 2004: 538). The intermediate PoS system types are labeled as 1/2, 2/3, 3/4, 4/5, 5/6, and 6/7. This labeling
indicates the PoS system types that the language in question has features in common with (they are always contiguous). For a language to be an intermediate flexible PoS system, it must first have a word class that functions as two or more propositional slots. One or more of these propositional slots must also be fulfilled by a smaller (closed) word class. For a language to have an intermediate rigid PoS system, it must have a small closed word class for the rightmost propositional slot fulfilled. The number of propositional slots in the intermediate system is determined by the PoS system type with the most slots. Intermediate PoS systems are not considered in the remaining analysis.

3.3 A Formal Description of Parts-of-speech Systems

3.3.1 Constraint 1

While the HRS approach to PoS systems is a good starting point, it leaves out some attested languages (HvL 2010). In an effort to amend this, Vulanović introduced two new constraints and a new labeling system. The first constraint, a modification of H1, is:

\[ C1: \quad P > R > \text{mod} \]

where “mod” stands for r, p, or both r and p. It states that if a language has a PoS system which includes a modifier, then it must include both of the heads. It does not, however, need to include the other modifier. Unlike the original HRS hierarchy, C1 is not meant to refer to word classes, just to the propositional functions (Vulanović 2008: 55). Table 3.4 shows the possible combinations of propositional slots that a PoS system may have according to C1.
Table 3.4: Possible Combinations of Propositional Slots

<table>
<thead>
<tr>
<th>n</th>
<th>P</th>
<th>R</th>
<th>r</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>+</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>3</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>3</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>2</td>
<td>+</td>
<td>+</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>1</td>
<td>+</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

The new constraint allows for more variations within the flexible and rigid PoS systems, which prompts the need for a new labeling system. Instead of the PoS system types being referred to as HRS1-7, they are now referred to using the two-number classification $l.n$ where $l$ is the number of word classes present in the language and $n$ is the number of propositional slots (Vulanović 2008: 53). Table 3.5 shows how the HRS classification fits within the new classification. The symbol $^\wedge$ indicates that the HRS PoS system is a subtype within the new classification and the + indicates where new PoS systems are present.

Table 3.5: Position of HRS PoS Systems in the Two-number Classification

<table>
<thead>
<tr>
<th>$l \backslash n$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HRS7</td>
<td>+</td>
<td>+</td>
<td>HRS1</td>
</tr>
<tr>
<td>2</td>
<td>-</td>
<td>HRS6</td>
<td>+</td>
<td>HRS2$^\wedge$</td>
</tr>
<tr>
<td>3</td>
<td>-</td>
<td>-</td>
<td>HRS5$^\wedge$</td>
<td>HRS3$^\wedge$</td>
</tr>
<tr>
<td>4</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>HRS4</td>
</tr>
</tbody>
</table>

Unlike the HRS classification, this version of the classification allows for flexible PoS systems with less than four propositional slots such as Tagalog. The system also allows for the inclusion of subtypes within each PoS system. For example, the 2.4 PoS system covers Kayardild as well as Warao. Kayardild is referred to as a 2.4HM PoS system and Warao is referred to as a 2.4P PoS system. Table 3.6 illustrates the difference between the two languages.
<table>
<thead>
<tr>
<th>Language</th>
<th>Head of Predicate Phrase</th>
<th>Head of Referential Phrase</th>
<th>Modifier of Referential Phrase</th>
<th>Modifier of Predicate Phrase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Warao</td>
<td>V</td>
<td>Λ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kayardild</td>
<td>$\emptyset$</td>
<td>Λ</td>
<td>$\emptyset$</td>
<td>$\emptyset$</td>
</tr>
</tbody>
</table>

Table 3.6: Comparison of Attested 2.4 PoS Subtypes

The PoS classification using C1 and the two-number system is shown below in Tables 3.7-10. These tables include all 28 theoretically possible PoS systems based on C1. Unattested PoS system types are marked with a *. The HRS classification is also indicated.

<table>
<thead>
<tr>
<th>PoS Type</th>
<th>P</th>
<th>R</th>
<th>r</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.4 (HRS1)</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.4</td>
<td>5 subtypes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.4</td>
<td>4 subtypes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.4 (HRS4)</td>
<td>V</td>
<td>N</td>
<td>a</td>
<td>m</td>
</tr>
<tr>
<td>1.3r</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*1.3p</td>
<td>C</td>
<td></td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>*2.3</td>
<td>3 subtypes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.3r (HRS5)</td>
<td>V</td>
<td>N</td>
<td>a</td>
<td>-</td>
</tr>
<tr>
<td>*3.3p</td>
<td>V</td>
<td>N</td>
<td>-</td>
<td>m</td>
</tr>
<tr>
<td>1.2</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.2 (HRS6)</td>
<td>V</td>
<td>N</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>*1.1 (HRS7)</td>
<td>V</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 3.7: PoS System Types

<table>
<thead>
<tr>
<th>2.4 Subtypes</th>
<th>P</th>
<th>R</th>
<th>r</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>H</td>
<td>P (HRS2)</td>
<td>V</td>
<td>Λ</td>
<td></td>
</tr>
<tr>
<td>*R</td>
<td>Z</td>
<td>N</td>
<td>Z</td>
<td></td>
</tr>
<tr>
<td>*M</td>
<td>*r</td>
<td>X</td>
<td>a</td>
<td>X</td>
</tr>
<tr>
<td></td>
<td>*p</td>
<td>X</td>
<td>m</td>
<td></td>
</tr>
<tr>
<td>*HH = MM</td>
<td>H</td>
<td>M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HM</td>
<td>$\emptyset$</td>
<td>$\emptyset$</td>
<td>$\emptyset$</td>
<td>$\emptyset$</td>
</tr>
<tr>
<td>*HxM</td>
<td>X</td>
<td>Y</td>
<td>X</td>
<td>Y</td>
</tr>
</tbody>
</table>

Table 3.8: PoS System Type 2.4 Subtypes
In the first column of table 3.8, the H and M refer to whether it is a head or a modifier which is fulfilled by a separate word class than the other propositional functions, which are fulfilled by one word class. In all of the tables, HH, HM, and H×M indicate which propositional functions are paired together and fulfilled by the same word class. HM indicates that one of the word classes functions as the head and modifier of either the referential phrase or the predicate phrase. H×M means that one of the word classes functions as either the head of the predicate phrase and the modifier of the referential phrase or the head of the referential phrase and the modifier of the predicate phrase. A further specification of which propositional functions are indicated in the first column is given in the second column, if needed.
3.3.2 Constraint 2

A second constraint was introduced by Vulanović to eliminate most of the unattested PoS system types present in the previous tables (Vulanović 2010: 242). This constraint is

\[ C2: \text{ C, A, and M are the only word classes that can have both predicative and referential functions.} \]

The following table shows all of the PoS system types with the new labeling system which satisfy the two constraints.

<table>
<thead>
<tr>
<th>PoS System type</th>
<th>P</th>
<th>R</th>
<th>r</th>
<th>p</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.4</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>C</td>
</tr>
<tr>
<td>2.4P</td>
<td>V</td>
<td>Λ</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.4HM</td>
<td>Ψ</td>
<td>Ψ</td>
<td></td>
<td>Ψ</td>
</tr>
<tr>
<td>*3.4Pp</td>
<td>Ψ</td>
<td>N</td>
<td>a</td>
<td>Ψ</td>
</tr>
<tr>
<td>3.4Rr</td>
<td>V</td>
<td>Ψ</td>
<td>m</td>
<td></td>
</tr>
<tr>
<td>3.4MM</td>
<td>V</td>
<td>N</td>
<td>M</td>
<td></td>
</tr>
<tr>
<td>4.4</td>
<td>V</td>
<td>N</td>
<td>a</td>
<td>m</td>
</tr>
<tr>
<td>1.3r</td>
<td>C</td>
<td>C</td>
<td>C</td>
<td>-</td>
</tr>
<tr>
<td>*1.3p</td>
<td>C</td>
<td>C</td>
<td>-</td>
<td>C</td>
</tr>
<tr>
<td>*2.3P</td>
<td>V</td>
<td>Ψ</td>
<td></td>
<td>-</td>
</tr>
<tr>
<td>*2.3R</td>
<td>Ψ</td>
<td>N</td>
<td>-</td>
<td>Ψ</td>
</tr>
<tr>
<td>*2.3Rp</td>
<td>V</td>
<td>Λ</td>
<td>-</td>
<td>Λ</td>
</tr>
<tr>
<td>3.3r</td>
<td>V</td>
<td>N</td>
<td>a</td>
<td>-</td>
</tr>
<tr>
<td>*3.3p</td>
<td>V</td>
<td>N</td>
<td>-</td>
<td>m</td>
</tr>
<tr>
<td>1.2</td>
<td>C</td>
<td>C</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2.2</td>
<td>V</td>
<td>N</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>*1.1</td>
<td>V</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

Table 3.11: PoS System Types Satisfying C1 and C2

This final PoS classification is the version which is used throughout the remainder of the thesis. It includes the 10 attested PoS systems as well as 7 unattested PoS systems which are logically possible. For instance, PoS system 1.1 is not attested. However, it is indirectly confirmed by a language which falls between 2.2 and 1.1 (a 6/7 PoS system on
the HRS classification) (HRS 2004: 539). This indirect confirmation is discussed above in chapter 3.1. Two other types, 2.3Rp and 3.3p, can also be found as parts of more complex PoS systems. Only four types have not been attested yet in any form: 3.4Pp, 2.3R, 2.3Rp, and 1.3p.

3.4 Formal Grammars with Regulated Rewriting

The above representation of PoS systems is not the only way of describing them. A method from formal language theory can also be employed. Phrase-structure grammars can be used to accurately describe the same PoS systems.

3.4.1 Description of Phrase-structure Grammars with Regulated Rewriting

A phrase structure grammar is defined as $G = (V, \Sigma, P, S)$. $V$, the total vocabulary, is a finite nonempty set containing all the symbols used in the grammar. $\Sigma \subseteq V$, also nonempty, is the terminal alphabet. $S \in N = V - \Sigma$ is the start symbol and $N$ is known as the set of non-terminals. The set $P$ contains the rules of the language for rewriting the non-terminals (Harrison 1978: 13). To compare this to the PoS systems described above, $\Sigma$ would be the set of word classes and $N$ would be the variables such as $<\text{Pred Ph}>$ (indicating the predicate phrase) and $P$. The set $P$ contains rules such as $<\text{Pred Ph}> \rightarrow Pp$ $| pP$, which means the symbol $<\text{Pred Ph}>$ can be rewritten as either the string $Pp$ or $pP$.

All of the PoS system types can be described using context free phrase-structure grammars. Context-free means that the non-terminals can be rewritten according to a rule in $P$ regardless of where they are located in a string (Harrison 1978: 18). To make a phrase-structure grammar regulated, a set of restrictions or rules can be set in place
before the final rules are applied. The regulation can specify that certain strings must be included in the final language or set a minimum of strings that are required.

A separate phrase structure grammar can be created for each number of propositional phrases possible, that is, \( n = 4, 3, 2, 1 \). A general phrase structure grammar to cover all 4 cases can be created as well, but is unnecessary. Some of the PoS systems don’t need an elaborate phrase-structure grammar, as is shown below.

### 3.4.2 Phrase-structure Grammar when \( n = 4 \)

\[
G = \{ V, \Sigma, P, S \}
\]

\[
N = \{ S, \text{<Pred Ph>}, \text{<Ref Ph>}, P, R, \text{p}, \text{r}, \text{p}, \text{r} \}
\]

\[
\Sigma = \{ \text{word classes present in the PoS system, } \lambda \text{ (the two-sided identity element = empty string)} \}
\]

\( P \) is as follows:

\[
S \rightarrow \text{<Pred Ph>} \text{<Ref Ph>} | \text{<Ref Ph>} \text{<Pred Ph>}
\]

\[
<\text{Pred Ph}> \rightarrow P \text{ p} | \text{ p} P \\
\text{ p} \rightarrow \text{ p} | \lambda
\]

\[
<\text{Ref Ph}> \rightarrow R \text{ r} | \text{ r} R \\
\text{ r} \rightarrow \text{ r} | \lambda
\]

This phrase structure grammar produces the strings contained in the following four sets:

\[
R_2 = \{ \text{PR, RP} \}
\]

\[
R_{3r} = \{ \text{PRr, PrR, RrP, rRP} \}
\]

\[
R_{3p} = \{ \text{PpR, pPR, RPP, RpP} \}
\]

\[
R_4 = \{ \text{PpRr, PprR, pPRr, pPrR, RrPp, RrpP, rRPp, rRpP} \}
\]
Since some languages use less than the above 18 strings of propositional functions, a set of restrictions or rules can be set in place before the final rules (below) are applied. This regulates the phrase-structure grammar further. For example, the regulation can be as follows.

**Regulation:** At least one string from each of the above sets has to be generated.

This gives that the minimum number of strings generated is 4:

{RP, RrP, RPp, RrPp}.

The maximum number of strings, which would be all possible strings, is 18. A variation of this example with the same regulation applies to each of the phrase structure grammars presented.

The final rules can now be applied to give the possible sentences. This phrase-structure grammar works for differentiated and flexible PoS systems and so these rules will vary depending on the PoS system being created.

For languages like English, the final rules would be

\[ P \rightarrow V \quad R \rightarrow N \quad r \rightarrow a \quad p \rightarrow m \]

Using the above regulation, the sentences would be

{NV, NaV, NVm, NaVm}.

For the PoS system of type 1.4, the rules would be

\[ P \rightarrow C \quad R \rightarrow C \quad r \rightarrow C \quad p \rightarrow C \]

This would give the following sentences:

{CC, CCC, CCCC}.
This shows that the phrase-structure method described above is much too complicated for PoS system 1.4 since the rules could simply be
\[ S \rightarrow CC | CCC | CCCC. \]

3.4.3 Phrase-structure Grammar with \( n = 3 \)
This phrase-structure grammar will be presented with \( n = 3 \) and for the propositional slots \( P, R, p \). The phrase-structure grammar for propositional slots \( P, R, r \) follows similarly.
\[
G = \{V, \Sigma, P, S\}
\]
\[
N = \{S, <Pred Ph>, P, R, \_p, p\}
\]
\[
\Sigma = \{\text{word classes present in the PoS system, } \lambda\}
\]
\( P \) is as follows:
\[
S \rightarrow <Pred Ph> R | R<Pred Ph>
\]
\[
<Pred Ph> \rightarrow P \_p | \_p P \quad \_p \rightarrow p | \lambda
\]
The phrase-structure grammar produces the following strings:
\[
R_2 = \{PR, RP\}
\]
\[
R_3 = \{PpR, pPR, RPp, RpP\}
\]
The final rules vary between the subtypes.

3.4.4 Phrase-structure Grammar for \( n = 2 \)
\[
G = \{V, \Sigma, P, S\}
\]
\[
N = \{S, P, R\}
\]
\[
\Sigma = \{\text{word classes present in the PoS system}\}
\]
\( P \) is as follows:
This phrase-structure grammar produces the following strings:

\[ R_2 = \{ PR, RP \} \]

The final rules vary between the subtypes.

### 3.4.5 Phrase-structure Grammar for \( n = 1 \)

This phrase-structure grammar is for the unattested PoS system type 1.1 = HRS7.

\[ G = \{ V, \Sigma, P, S \} \]

\[ N = \{ S \} \]

\[ \Sigma = \{ V \} \]

\( P \) is as follows:

\[ S \rightarrow V \]

The only sentence produced is V.

### 3.5 Complexity in Phrase-structure Grammars with Regulated Rewriting

In phrase-structure grammars with regulated rewriting, the complexity of a PoS system type is determined by comparing the cardinalities of non-terminal symbols or the cardinality of production rules in the system with other systems (Dassow & Păun 1989: 170). A third method is also mentioned there, but is not discussed in depth. It compares the total number of occurrences of letters in \( P \). For all of these methods, the higher the cardinality, the more complex the system is. While this may be a good measure of complexity for some studies, it does not work for the analysis presented in this thesis.
Comparing the complexity of two subsystems within 3.4, we find a fault in these methods for calculating complexity. PoS system type 3.4Rr has \( l = 3, n = 4 \), and word classes: V, m, and R. PoS system type 3.4rp has \( l = 3, n = 4 \), and word classes: V, N, and M. Both languages have the phrase-structure grammar described in chapter 3.4.2 with the following final rules:

<table>
<thead>
<tr>
<th>3.4Rr</th>
<th>3.4MM</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P \rightarrow V )</td>
<td>( P \rightarrow V )</td>
</tr>
<tr>
<td>( R \rightarrow R )</td>
<td>( R \rightarrow N )</td>
</tr>
<tr>
<td>( r \rightarrow N )</td>
<td>( r \rightarrow M )</td>
</tr>
<tr>
<td>( p \rightarrow m )</td>
<td>( p \rightarrow M )</td>
</tr>
</tbody>
</table>

Table 3.12: Rules for 3.4Rr and 3.4rp

By the first complexity method, comparing the cardinality of non-terminal symbols, both PoS system types have the same complexity since they both have 9 non-terminal symbols (\( S, \) <Pred Ph>, <Ref Ph>, \( P, R, \) \( \overline{p}, \overline{r}, p, r \)). By the second complexity method, both PoS system types also have the same complexity since they both have 14 production rules. By the third complexity method, the total number of letters in the production rules is 34 for both PoS system types. All three methods for testing complexity indicate that the two PoS system types have the same complexity.

These methods of determining the complexity of a PoS system type do not work with the analysis performed in this thesis. Only the PoS system types or subtypes with similar structures may have the same complexity. For instance, 3.3r and 3.3p have similar structures, the only difference is that 3.3r has \( r \rightarrow a \) and 3.3p has \( p \rightarrow m \), the other final production rules being equal: \( P \rightarrow V \) and \( R \rightarrow N \). However, PoS system types 3.4Rr and 3.4MM should have different complexities since they have different structures. That
these structures are essentially different is shown by the different ways sentences in 3.4Rr and 3.4MM are parsed, see Chapter 4. Due to this, the way complexity is measured in phrase-structure grammars is not the best option when analyzing PoS system (sub)types.
CHAPTER 4

Grammar Efficiency and Complexity

4.1 Description of Grammar Complexity and Efficiency

Grammar complexity is a comparative measure which means little without the context of the complexity measure of other grammars. There are multiple ways to define grammar complexity. It can be defined based on only the rules and forms of the grammar, known as absolute complexity. It can consider the rules and forms of the grammar along with the information conveyed, known as information-relative complexity (Vulanović 2007: 2). Complexity can also be defined relative to the difficulty of processing, acquisition, and learning. This is known as relative complexity (Vulanović 2007: 3).

The PoS system classification described above restricts the type of grammar complexity measure which can be used. The classification does not give information about the human processor, so relative complexity is not an option. The classification does provide information about the rules of the language. The parsing technique described below uses the classification to give information on the information conveyed by language. This implies that the definitions of absolute complexity and information-relative complexity can be used.

The more complex a grammar is, the less efficient it is. Thus, grammar efficiency and complexity can be defined as reciprocals of one another. The approach taken here to
define grammar efficiency, and thus grammar complexity, is inspired by machine efficiency. Machine efficiency is defined as the ratio of useful output to useful input; in this case the grammar is the machine (Vulanović 2007: 2). A more descriptive explanation of the efficiency used, including the formula, is presented after a description of calculating the useful input and output.

4.2 Parsing

4.2.1 Description of Parsing and Rules

The main purpose of the parsing here is to determine all possible ways a string of words can be interpreted with respect to the rules of the language. The rules include a mapping from the word classes to the propositional slots they can fill. The mapping is reversed from the previous chapter because the efficiency formula follows machine efficiency and the sentences are the input, the analysis (the strings of propositional functions) is the output. The rules also include any restrictions on word order. While the classification of PoS systems doesn’t take word order into consideration, the calculation of grammar efficiency does. Word order can either be restricted or free, where every possible word order can be considered. To retain the simplicity of the classification, a restriction is placed on the allowable word orders considered: phrases cannot be broken. That is, the head and modifier of a predicate phrase or referential phrase must remain contiguous (Vulanović 2008: 58).

Except for PoS system type 1.1, the strings of words being parsed must include those that function as the heads of the predicate phrase and referential phrase. This means
that for all PoS system types, other than type 1.1, the minimum length for the string of
words is two. The maximum length for a string for words is $n$, which is the number of
propositional functions present in the PoS system being analyzed.

The parsing of strings results in successful and unsuccessful parses as well as
ambiguous parses and unambiguous parses. A successful parse is one where every word
in the string is successfully mapped to a propositional function in an allowable sequence;
unsuccessful means that a rule is broken. Unsuccessful parses are marked with a “-” at
the end of the parsing attempt. An unambiguous parse means that the string can only be
interpreted in one way; ambiguous means that there is more than one interpretation. The
specific amount of ambiguity is determined depending on the number of successful
parses.

The parsing process utilizes the mapping $\Phi$ from word classes to propositional
functions. During parsing, each string is considered one at a time. Within each string,
each word, or element, is looked at one at a time, left to right. The first word of the string
is compared to the mapping $\Phi$. The appropriate mapping is then applied. This is repeated
until the end of the string. If $\Phi$ allows for a word to be mapped to more than one
propositional function, then the parsing must be attempted for each possible mapping,
resulting in more than one parsing of that string. If the choice of mappings leads to an
incomplete sentence (that is, both heads are not present) or a non-contiguous
prepositional phrase or referential phrase, the parsing is halted and “-” is written at the
end.
The following quantities can be determined from the parsing process and are used later in the grammar efficiency formula:

- $\rho^* = \text{the total number of all parses attempted when each permutation of each possible sentence is parsed},$
- $\rho_A = \text{the greatest possible number of unambiguous parses (this only depends on the mapping } \Phi \text{ which assigns propositional functions to word classes; the index } A \text{ in } \rho_A \text{ comes from the word “assignment”).}$

### 4.2.2 Parse 3.4Rr and 3.4MM

To illustrate the parsing process, the examples from chapter 3.5 are parsed. Later, the efficiency of each these subtypes will be calculated to show that the efficiency formula is reasonable.

### 4.2.3 Parsing of 3.4Rr

The 3.4Rr PoS system type has four propositional functions and three word classes. The word classes are $V$, $R$, and $m$. The mapping $\Phi$ is defined as follows:

$$\Phi: V \rightarrow P \quad R \rightarrow R, r \quad m \rightarrow p$$

The possible strings are the permutations of elements in the following strings (keeping $R \in R$ together and $Vm$ together):

$VmR, VR, VmR, VR$

giving a total number of $4 + 2 + 4 + 2 = 12$ possible strings.

With no additional restrictions on word order, the parsing results with the following:

$VmR \rightarrow PpRr, PprR \quad mVR \rightarrow pPRr, pPRR$
\( \forall \forall Vm \rightarrow RrPp, rRPp \quad \forall \forall mV \rightarrow RrP, rRPP \)

\( V \forall \forall \rightarrow PRr, PrR \quad \forall \forall V \rightarrow RrP, rRP \)

\( Vm \forall \rightarrow PpR, Ppr- \quad \forall Vm \rightarrow RPP, r- \)

\( mV \forall \rightarrow pPR, pPr- \quad \forall mV \rightarrow RPP, r- \)

\( V \forall \rightarrow PR, Pr- \quad \forall V \rightarrow RP, r- \)

From this parsing process, \( \rho^* = 24 \) and \( \rho_A = 12 \).

### 4.2.4 Parsing of 3.4MM

The 3.4MM PoS system type has four propositional functions and three word classes. The word classes are V, N, and M. The mapping \( \Phi \) is:

\[ \Phi: \quad V \rightarrow P \quad N \rightarrow R \quad M \rightarrow r, p \]

The possible strings are the permutations of the following:

VMNM, VMN, VN

This gives \( 8 + 6 + 2 = 16 \) possible strings.

With no additional restrictions on word order, the parsing results with the following.

\( \forall \forall Vm \rightarrow RrPp, rRPp \quad \forall \forall mV \rightarrow RrP, rRPP \)

\( V \forall \forall \rightarrow PRr, PrR \quad \forall \forall V \rightarrow RrP, rRP \)

\( Vm \forall \rightarrow PpR, Ppr- \quad \forall Vm \rightarrow RPP, r- \)

\( mV \forall \rightarrow pPR, pPr- \quad \forall mV \rightarrow RPP, r- \)

\( V \forall \rightarrow PR, Pr- \quad \forall V \rightarrow RP, r- \)
From this parsing process, $\rho^* = 28$ and $\rho_A = 16$.

### 4.2.5 Calculations for all PoS system types

A computer program created using FORTRAN was used to quickly calculate the variable values for each PoS system type. This table includes all theoretically possible PoS system types using just C1. PoS system types with similar structures are simplified into one PoS system type unless they are one of the 17 PoS system types analyzed later. While all PoS system types are not considered in the final analysis, they are important in calculating the efficiency of the 17 PoS system types considered. The value of main importance obtained from $\rho^*$ and $\rho_A$ is Max $Q$, which is discussed in chapter 4.4.
<table>
<thead>
<tr>
<th>PoS System Type</th>
<th>ρ*</th>
<th>ρ_λ</th>
<th>Max Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.4</td>
<td>32</td>
<td>4</td>
<td>.0625</td>
</tr>
<tr>
<td>2.4P</td>
<td>34</td>
<td>9</td>
<td>.265</td>
</tr>
<tr>
<td>2.4M</td>
<td>37</td>
<td>9</td>
<td>.243</td>
</tr>
<tr>
<td>2.4HH</td>
<td>22</td>
<td>8</td>
<td>.364</td>
</tr>
<tr>
<td>2.4HM</td>
<td>28</td>
<td>8</td>
<td>.286</td>
</tr>
<tr>
<td>2.4H×M</td>
<td>34</td>
<td>12</td>
<td>.353</td>
</tr>
<tr>
<td>3.4HH</td>
<td>24</td>
<td>15</td>
<td>.625</td>
</tr>
<tr>
<td>3.4Pp</td>
<td>24</td>
<td>12</td>
<td>.500</td>
</tr>
<tr>
<td>3.4Rr</td>
<td>24</td>
<td>12</td>
<td>.500</td>
</tr>
<tr>
<td>3.4H×M</td>
<td>30</td>
<td>17</td>
<td>.567</td>
</tr>
<tr>
<td>3.4MM</td>
<td>28</td>
<td>16</td>
<td>.571</td>
</tr>
<tr>
<td>4.4</td>
<td>18</td>
<td>18</td>
<td>1.00</td>
</tr>
<tr>
<td>1.3r</td>
<td>10</td>
<td>2</td>
<td>.200</td>
</tr>
<tr>
<td>1.3p</td>
<td>10</td>
<td>2</td>
<td>.200</td>
</tr>
<tr>
<td>2.3HH</td>
<td>7</td>
<td>4</td>
<td>.571</td>
</tr>
<tr>
<td>2.3P</td>
<td>8</td>
<td>4</td>
<td>.500</td>
</tr>
<tr>
<td>2.3R</td>
<td>8</td>
<td>4</td>
<td>.500</td>
</tr>
<tr>
<td>2.3Rp</td>
<td>10</td>
<td>5</td>
<td>.500</td>
</tr>
<tr>
<td>3.3r</td>
<td>6</td>
<td>6</td>
<td>1.00</td>
</tr>
<tr>
<td>3.3p</td>
<td>6</td>
<td>6</td>
<td>1.00</td>
</tr>
<tr>
<td>1.2</td>
<td>2</td>
<td>1</td>
<td>.500</td>
</tr>
<tr>
<td>2.2</td>
<td>2</td>
<td>2</td>
<td>1.00</td>
</tr>
<tr>
<td>1.1</td>
<td>1</td>
<td>1</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Table 4.1: PoS System Values
4.3 The Efficiency Formula

The formula for defining grammar efficiency, as given in (Vulanović 2008: 57), is

\[ Eff = \kappa Q \frac{n}{l}. \]

The variables \( n \) and \( l \) are the number of propositional functions and the number of word classes, respectively, from the two-number classification system. \( Q \) is a type of absolute grammar efficiency which allows for a comparison within the subtypes of the PoS systems. \( Q \) generally is defined as

\[ Q = \frac{\rho - \rho_o}{\rho^*} \]

where \( \rho_o \) is the number of ambiguous parses and \( \rho^* \) is as defined above. The value \( \rho \) is the number of successful parses of all permitted sentences. This value will vary with the restrictions placed upon word order (Vulanović 2008: 59).

This formula for grammar efficiency is inspired by machine efficiency, which is the ratio of appropriate measures of useful machine output and machine input. This ratio is here represented by \( n/l \) since propositional functions form the output and word classes form the input. The conversion of the input to the output is done through the parsing process which is viewed as the grammar machine. If parsing is more efficient, so is the grammar. The quantity \( Q \), also known as parsing ratio, indicates how efficient parsing is. The definition of \( Q \) takes into account the situations in which PoS system types should be more efficient. Fewer word order rules will allow for a more efficient system and \( \rho \) will be greater. A smaller value of \( \rho_o \) indicates less ambiguity and will result in a more
efficient system. A system that requires fewer parsing attempts will also be more efficient (Vulanović 2008: 59). Moreover, the inclusion of $Q$ in (1) allows for a comparison between PoS system subtypes since the value of $n/l$ would remain the same for each subtype.

The value $\kappa$ is a scaling coefficient which allows for comparison of all PoS system types. The value $\kappa$ is determined so that $Eff = 1$ for all maximally efficient grammars (Vulanović 2008: 57). The value of $\kappa$ remains constant within a PoS system type and may vary between different types. To calculate $\kappa$ for a specific PoS system type, the maximum value of $Q$ (Max $Q$) must be determined for each PoS system subtype out of all the theoretically possibly PoS systems according to C1. The subtype with the highest Max $Q$ value has the most efficient grammar within that PoS system type. After the highest Max $Q$ is determined, $Eff$ in (1) is set equal to 1, the determined value of Max $Q$ is used for the value of $Q$, and the appropriate values of $n$ and $l$ are used. The value of $\kappa$ can then be calculated.

Finding the maximally efficient grammar within a type depends on the following problem: Does there exist a grammar in the $l,n$ PoS system type such that $\rho_0 = 0$ and $Q$ is maximized? (Vulanović 2003: 133). There exist PoS system types which have no solution to this problem (Vulanović 2003: 134). PoS system type 1.4, for example, will always have ambiguity no matter how the word order is restricted for sentences with three words. For PoS system types such as these, the value of $\kappa$ is set to 1. For all other PoS system types, the system with the highest Max $Q$ is used to determine the value of $\kappa$. 
The PoS system subtypes which do not solve the problem are the subtypes which do not produce the maximum Q for the l.n type. These systems are not maximally efficient. Such PoS system types have an efficiency that is less than 1 (Vulanović 2003: 14).

4.4 Max Q

The formula for Max Q is

\[ \text{Max } Q = \frac{\rho_A - \rho_o}{\rho^*} \]

where \( \rho_A \) is the greatest possible number of unambiguous parses, \( \rho_o \) here is the least possible number of ambiguous parses (\( \rho_o = 0 \) in all (sub)types except 1.4), and \( \rho^* \) is the total number of parsing attempts (Vulanović 2008: 60). The value \( \rho_A \), as defined in 4.2.1, is determined from the parsing process and is equal to the total number of strings which are parsed unambiguously.

The values of Max Q are given in Table 4.1. From this, within each PoS system type, the most efficient PoS system subtype can be determined. Within 2.4 it is 2.4HH. Within 3.4 it is 3.4HH. Within 2.3 it is 2.3HH. The remaining PoS system types either have no subtypes or the subtypes have the same Max Q value.

The values of \( \kappa \) for each PoS system type can now be calculated. For PoS system type 2.4, the Max Q value for the entire system type belongs to 2.4HH. For the value of Max Q, \( \rho_A = 8 \), \( \rho^* = 22 \), and \( \rho_o = 0 \). Also, \( n = 4 \), \( l = 2 \), and \( \text{Eff} = 1 \). This gives

\[ \kappa = \text{Eff} \frac{1}{\text{Max } Q} \frac{l}{n} = \frac{1}{(\rho_A - \rho_o)} \frac{l}{n} = \frac{22}{(8 - 0)} \frac{2}{4} = \frac{11}{8} \]

The values of \( \kappa \) are presented in Table 4.2 for each PoS system type.
<table>
<thead>
<tr>
<th>PoS System Type</th>
<th>( \kappa )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.4</td>
<td>1</td>
</tr>
<tr>
<td>2.4</td>
<td>11/8</td>
</tr>
<tr>
<td>3.4</td>
<td>6/5</td>
</tr>
<tr>
<td>4.4</td>
<td>1</td>
</tr>
<tr>
<td>1.3</td>
<td>10/6</td>
</tr>
<tr>
<td>2.3</td>
<td>7/6</td>
</tr>
<tr>
<td>3.3</td>
<td>1</td>
</tr>
<tr>
<td>1.2</td>
<td>1</td>
</tr>
<tr>
<td>2.2</td>
<td>1</td>
</tr>
<tr>
<td>1.1</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4.2: \( \kappa \) Values

4.5 Min \( Q \)

4.5.1 Reason for using Min \( Q \)

The minimum values of \( Q \) are also of interest since they give a range of efficiency for each PoS system type. In this case, restrictions are placed on the word orders allowed. That is, only one order of propositional functions will be permitted. The possible word orders are found in chapter 3.4 for each number of propositional functions. To calculate the value of Min \( Q \), the following equation is used

\[
\text{Min } Q = \frac{\rho_* - \rho_o}{\rho^*}
\]

where \( \rho_* \) is the smallest possible number of parses (which depends on the number of propositional functions in the PoS system type, see 4.5.2 and 4.5.3), \( \rho_o \) is the greatest number of ambiguous parses with fixed word order, and \( \rho^* \) is the same as defined earlier.
Fixing the word order, in general, reduces the amount of ambiguity to zero. All PoS system types except four were found to have no ambiguity regardless of which fixed word order was used. The PoS system (sub)types which have ambiguity with at least one fixed word order are 1.4, 2.4HH, 2.4P, and 3.4MM. An example of finding the ambiguity is presented below for PoS system subtype 3.4MM. The remaining PoS system types were tested using a program written in FORTRAN.

4.5.2 Finding the Min $Q$ of 3.4MM

This PoS system type has been discussed in 4.2.4. It uses the mapping $\Phi$,

$$\Phi: \begin{align*}
V &\rightarrow P \\
N &\rightarrow R \\
M &\rightarrow r, p
\end{align*}$$

The possible orders of all four propositional functions are PpRr, PprR, pPRr, pPR, RrPp, RrpP, rRPP, rRpP. However, PpRr, PprR, pPRr, pPR will give the same results as RrPp, RrpP, rRPP, rRpP, so only 4 orders need to be tested.

PpRr also allows for PrR, PRR, and PR, which means that $p_* = 4$. These 4 strings of propositional functions result from the following respective sentences: VMNM, VMN, VNM, and VN. None of these strings can be parsed with ambiguity under this restricted word order.

PprR also allows for PpR, PrR, and PR. The corresponding sentences are, respectively, VMMN, VMN, VMN, and VN. There are two ambiguous parses under this restricted word order: VMN $\rightarrow$ PpR, PrR.

pPRr also allows for pPR, PRR, and PR. The corresponding sentences are, respectively, MVNM, MVN, VNM, and VN. None of these strings can be parsed with ambiguity under this restricted word order.
pPrR also allows for pPR, PrR, and PR. The corresponding sentences are now, MVMN, MVN, VMN, and VN, respectively. Again, none of these strings can be parsed with ambiguity under the restricted word order.

From this analysis, the maximum ambiguity happens when the restricted word order is PprR (as well as RrpP). Therefore, when calculating the Min $Q$, $\rho_o$ will be set at 2 and

$$\text{Min } Q = \frac{\rho_o - \rho_0}{\rho_o} = \frac{4-2}{28} = .071$$

4.5.3 Notes on Min $Q$

When calculating the Min $Q$ of PoS system types 1.4, 2.4HH and 2.4P, the ambiguity occurs with the restricted word orders PprR and RrpP. These also result in $\rho_o = 2$.

For PoS system types with $n = 4$, like in 4.5.2, the $\rho*$ value is set at 4. Each word order restriction results in four possible propositional function orders. For PoS system types with $n = 3$, the $\rho*$ value is set at 2. This is because each word order restriction results in only 2 propositional function orders. For example, PpR would allow PR and no additional orders. For PoS system types with $n = 2$ or $n = 1$, the $\rho*$ value is set at 1. This is because each string needs P and R (except in the case of $n = 1$, where the string consists of only P).

4.6 Efficiency Values

The efficiency formula (1) can be used to calculate the efficiency of each PoS system type using the Min $Q$ and the Max $Q$. The formula for Max $Eff$ is
\[ (5) \quad \text{Max } \text{Eff} = \kappa\left(\text{Max } Q\right)^{\frac{n}{l}} \]

and the formula for \( \text{Min } \text{Eff} \) is

\[ (6) \quad \text{Min } \text{Eff} = \kappa\left(\text{Min } Q\right)^{\frac{n}{l}} \]

These values are presented in Table 4.3. The values are calculated only for the PoS system types which satisfy both C1 and C2.

<table>
<thead>
<tr>
<th>PoS System Type</th>
<th>Max ( Q )</th>
<th>Min ( Q )</th>
<th>Max ( \text{Eff} )</th>
<th>Min ( \text{Eff} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.4</td>
<td>.0625</td>
<td>.0625</td>
<td>.250</td>
<td>.250</td>
</tr>
<tr>
<td>2.4P</td>
<td>.265</td>
<td>.0588</td>
<td>.726</td>
<td>.162</td>
</tr>
<tr>
<td>2.4HM</td>
<td>.286</td>
<td>.143</td>
<td>.786</td>
<td>.393</td>
</tr>
<tr>
<td>3.4Pp</td>
<td>.500</td>
<td>.167</td>
<td>.800</td>
<td>.267</td>
</tr>
<tr>
<td>3.4Rr</td>
<td>.500</td>
<td>.167</td>
<td>.800</td>
<td>.267</td>
</tr>
<tr>
<td>3.4MM</td>
<td>.571</td>
<td>.0714</td>
<td>.914</td>
<td>.114</td>
</tr>
<tr>
<td>4.4</td>
<td>1.00</td>
<td>.222</td>
<td>1.00</td>
<td>.222</td>
</tr>
<tr>
<td>1.3r</td>
<td>.200</td>
<td>.200</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>1.3p</td>
<td>.200</td>
<td>.200</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>2.3P</td>
<td>.500</td>
<td>.250</td>
<td>.875</td>
<td>.438</td>
</tr>
<tr>
<td>2.3R</td>
<td>.500</td>
<td>.250</td>
<td>.875</td>
<td>.438</td>
</tr>
<tr>
<td>2.3Rp</td>
<td>.500</td>
<td>.200</td>
<td>.875</td>
<td>.350</td>
</tr>
<tr>
<td>3.3r</td>
<td>1.00</td>
<td>.333</td>
<td>1.00</td>
<td>.333</td>
</tr>
<tr>
<td>3.3p</td>
<td>1.00</td>
<td>.333</td>
<td>1.00</td>
<td>.333</td>
</tr>
<tr>
<td>1.2</td>
<td>.500</td>
<td>.500</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>2.2</td>
<td>1.00</td>
<td>.500</td>
<td>1.00</td>
<td>.500</td>
</tr>
<tr>
<td>1.1</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Table 4.3: Efficiency Values of the PoS System Types

The efficiency interval [Min \( \text{Eff} \), Max \( \text{Eff} \)] does not contain all possible efficiency values for the PoS system types represented by the classification system. PoS system types such as 1.4 can have a free word order which results in every successfully parsed string being ambiguous, causing \( \text{Eff} = 0 \). For PoS systems where ambiguity is unavoidable regardless of word order restrictions, the minimum unavoidable amount of
ambiguity is included in the efficiency interval. The efficiency interval does, however, include all efficiency values where no ambiguity is involved.

4.7 Comparison to Phrase-structure Grammar Efficiency

As mentioned in 4.1, grammar complexity and efficiency are reciprocal to each other. Therefore, the tests for phrase-structure grammars in 3.5 show that PoS system subtypes 3.4Rr and 3.4MM have the same efficiency despite having completely different structures. The efficiency formula presented above for the two-number PoS systems assigns different values to the efficiency of 3.4Rr and 3.4MM. In fact, all calculated values are different between these two PoS system subtypes. This verifies that the efficiency formula is adequate for the comparison presented in this thesis. It assigns a unique interval of efficiency values to each PoS system (sub)type, except to systems that have equivalent structures or $\text{Min } \text{Eff} = \text{Max } \text{Eff} = 1$.

The fact that $\text{Eff}$ values form an interval is another difference between definition (1) and the definitions of efficiency in phrase-structure grammars. The complexity measures for the phrase-structure grammars do not take word order into consideration and therefore assign a single value to each PoS system. However, languages use word order rules to convey information and this is why restricting word order should have an effect on the efficiency of a PoS system. This is another advantage of formula (1) over the definitions of efficiency in phrase-structure grammars.
CHAPTER 5

Statistical Analysis

5.1 Introduction

In this chapter, the efficiency values calculated in Chapter 4.6 are used to create a 3D model. This model is used to determine whether a correlation exists between the two-number labeling of PoS system types and their efficiency values. The linear regression is used to fit a plane to the data and the nonlinear regression is used to fit a sigmoidal surface to the data. Three sets of data are used in the regression: the Min $Eff$, the Max $Eff$, and the Mid $Eff$ values. Each Mid $Eff$ value represents the midpoint of the [Min $Eff$, Max $Eff$] interval for the PoS system type under consideration. The fit of the equations is then analyzed to determine whether a correlation exists.

5.2 Preparing the Data

The data presented in Table 4.3 is used for the regression analysis. From this table, an interval of values from Min $Eff$ to Max $Eff$ is created for each PoS system type. The interval is created so that a comparison is allowed between the efficiency values for each PoS system type as a whole, regardless of its possible subtypes. The interval is determined by grouping together all subtypes within a PoS system type, if there are subtypes. The smallest Min $Eff$ value out of the group is used as the Min $Eff$ value for that entire PoS system type. The largest Max $Eff$ is used as the Max $Eff$ for that PoS system
type. If there are no subtypes for a PoS system, the values for that PoS system are used unchanged. The Min $Eff$ and Max $Eff$ values determined this way give a complete range of efficiency values for each PoS system type. The midpoint of the Min $Eff$ and Max $Eff$ range is then calculated for each PoS system type and designated as the Mid $Eff$ value. Table 5.1 shows the values of Min $Eff$, Mid $Eff$, and Max $Eff$ for each PoS system type.

<table>
<thead>
<tr>
<th>PoS System</th>
<th>Min $Eff$</th>
<th>Mid $Eff$</th>
<th>Max $Eff$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.4</td>
<td>.250</td>
<td>.250</td>
<td>.250</td>
</tr>
<tr>
<td>2.4</td>
<td>.162</td>
<td>.474</td>
<td>.786</td>
</tr>
<tr>
<td>3.4</td>
<td>.114</td>
<td>.514</td>
<td>.914</td>
</tr>
<tr>
<td>4.4</td>
<td>.222</td>
<td>.611</td>
<td>1.00</td>
</tr>
<tr>
<td>1.3</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>2.3</td>
<td>.350</td>
<td>.612</td>
<td>.875</td>
</tr>
<tr>
<td>3.3</td>
<td>.333</td>
<td>.667</td>
<td>1.00</td>
</tr>
<tr>
<td>1.2</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>2.2</td>
<td>.500</td>
<td>.750</td>
<td>1.00</td>
</tr>
<tr>
<td>1.1</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>

Table 5.1: Minimum, Midpoint, and Maximum Efficiency Values

For each efficiency value, a 2D graphical table can be created. This gives an idea about how the data will be fit to a 3D graph. These are presented in Tables 5.2, 5.3, and 5.4. Each table can be viewed as a 2D graph where the horizontal axis represents the $l$ value and the vertical axis represents the $n$ value; each is labeled from 1 to 4. The values in the boxes represent the designated efficiency value for the corresponding $l,n$ PoS system type. The shaded regions indicate $(l,n)$ pairs that do not represent a PoS system type.
Table 5.2: 2D Representation of Min $Eff$

<table>
<thead>
<tr>
<th>n</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>.250</td>
<td>.162</td>
<td>.114</td>
<td>.222</td>
</tr>
<tr>
<td>3</td>
<td>1.00</td>
<td>.350</td>
<td>.333</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1.00</td>
<td>.500</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.3: 2D Representation of Max $Eff$

<table>
<thead>
<tr>
<th>n</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>.250</td>
<td>.786</td>
<td>.914</td>
<td>1.00</td>
</tr>
<tr>
<td>3</td>
<td>1.00</td>
<td>.875</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5.4: 2D Representation of Mid $Eff$

<table>
<thead>
<tr>
<th>n</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>.250</td>
<td>.474</td>
<td>.514</td>
<td>.611</td>
</tr>
<tr>
<td>3</td>
<td>1.00</td>
<td>.612</td>
<td>.667</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1.00</td>
<td>.750</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1.00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
5.3 Fitting the Equations

As mentioned before, the regression analysis is performed twice for each set of data; once as linear regression and once as nonlinear regression. To determine which equations to fit to the data, consider Tables 5.2-4. The efficiency value for each \((l,n)\) pair can be considered the height of the surface. The data can simply be seen as points close to a plane, which the linear regression models. The general linear equation used in the linear regression is

\begin{equation}
(1) \quad y = an + bl + c.
\end{equation}

The equation for the nonlinear regression is the following 3D generalization of the sigmoid equation

\begin{equation}
(2) \quad y = \frac{1}{1 + e^{an + bl + c}}.
\end{equation}

This is used because the possible values of efficiency range from 0 to 1, which implies that the data can be seen as points on a transition between two plateaus. The plateaus are the completely non-efficient PoS systems \((Eff = 0 = y)\) and maximally efficient PoS systems \((Eff = 1 = y)\). The transition between two plateaus is what the equation for a sigmoidal surface represents in the nonlinear regression.

This type of non-linear regression is also known as logistic regression and the equation for the sigmoidal surface is called a logistic equation (Devore 2004: 572). In mathematics, this equation is used to model population growth in situations where there is a cap to the population size (Stewart 1999: 652). In linguistics, the 2D sigmoid, which gets its name from the “S” shape of the curve, is used to model how certain linguistic
properties change over time, which is known as the Piotrowski or Piotrowski-Altmann Law in Quantitative Linguistics (Vulanović and Köhler 2009: 304).

The program NLREG was used to fit the equations to the data. The values for $a$, $b$, $c$, and $R^2_a$ are presented in Tables 5.5 and 5.6. $R^2_a$ is the adjusted coefficient of multiple determination. $R^2_a$ is used in place of $R^2$ (the coefficient of multiple determination) due to the use of more than one parameter and a low number of data-points; the use of more parameters can artificially inflate the value of $R^2$. In this analysis, 3 parameters and 10 data points are used. The $R^2_a$ value represents the total variation of data points that is explained by the fitted equation (Devore 2004: 580). The higher the value of $R^2_a$, the better the data fits the graph. $R^2_a$ also gives an idea about how well the model will predict points outside of the data set. The value of $R^2_a$ is calculated in a similar manner for both linear and non-linear regression (Devore 2004: 594).

<table>
<thead>
<tr>
<th>Data Set</th>
<th>$a$</th>
<th>$b$</th>
<th>$c$</th>
<th>$R^2_a$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min $Eff$</td>
<td>1.21228672</td>
<td>1.26233643</td>
<td>-5.88724159</td>
<td>.7396</td>
</tr>
<tr>
<td>Mid $Eff$</td>
<td>-0.1720781</td>
<td>1.35746579</td>
<td>-4.86758983</td>
<td>.5947</td>
</tr>
<tr>
<td>Max $Eff$</td>
<td>-2.23810112</td>
<td>5.34037628</td>
<td>-18.0746256</td>
<td>.9507</td>
</tr>
</tbody>
</table>

Table 5.5: Results for the Sigmoidal Equation

<table>
<thead>
<tr>
<th>Data Set</th>
<th>$a$</th>
<th>$b$</th>
<th>$c$</th>
<th>$R^2_a$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min $Eff$</td>
<td>-0.13106667</td>
<td>-0.20966667</td>
<td>1.38233333</td>
<td>.6615</td>
</tr>
<tr>
<td>Mid $Eff$</td>
<td>0.0138</td>
<td>-0.197</td>
<td>1.2513</td>
<td>.5244</td>
</tr>
<tr>
<td>Max $Eff$</td>
<td>0.15853333</td>
<td>-0.18426666</td>
<td>1.11823333</td>
<td>.4300</td>
</tr>
</tbody>
</table>

Table 5.6: Results for the Linear Equation

5.4 Analysis of Results

For the purpose of this analysis, an acceptable value range for $R^2_a$ is 0.7 or greater for both the linear regression and the nonlinear regression. The sigmoidal equation in the nonlinear regression fit the data points best for Max $Eff$ according to $R^2_a$; however the
linear fit is not acceptable. The fit for Min Eff using nonlinear regression is acceptable and the linear fit is not acceptable. The fit for Mid Eff using either regression is below the acceptable range.

These results imply that there is a strong correlation between the Max Eff values of a PoS system and the two-number label assigned to it. There is also a decent correlation between the Min Eff values and the two-number labels. Both of these correlations are based on the nonlinear regression results. There is not much of a correlation between the Mid Eff and the two-number system. Unlike Max Eff and Min Eff, whose values are taken from either a PoS system type or subtype, Mid Eff does not represent any PoS system described by the classification. Mid Eff can be viewed as an artificial representation of a PoS system type with a semi-free word order. However, most languages tend to have a completely free word order or a variation of a fixed word order (Comrie 1989: 88). There is no guarantee that such a PoS system type is even theoretically possible with the corresponding Mid Eff value.

The graphs for the nonlinear fit of Min Eff and Max Eff are presented below in Figures 5.1 and 5.2. In the graph for Min Eff, the data points which are visible do not appear to be very close to the sigmoidal surface. In the graph for Max Eff, five of the data points are visible and the remaining points are hidden by the sigmoidal surface. In this case the points appear relatively close to the sigmoidal surface which agrees with the high $R^2$ value.
Figure 5.1: The Nonlinear Fit of Min $Eff$

Figure 5.2: The Nonlinear Fit of Max $Eff$
The program used to graph Figures 5.1 and 5.2 cannot accurately show the domain in which \( l \leq n \). When looking at the graphs, the part above the \( l \times n \) plane where \( l > n \) should be ignored since it does not represent a PoS system type in the classification system. \( (l > n \) to the right of the diagonal between the square vertices \((1, 1, 0)\) and \((4, 4, 0)\) in Fig. 5.1, and between \((1, 1, 0.2)\) and \((4, 4, 0.2)\) in Fig. 5.2.)

In view of the definition of \( Eff \) in 4.3(1), it also seems natural to perform a regression analysis using the equations

\[
(3) \quad y = \frac{an}{l} + b
\]

and

\[
(4) \quad y = \frac{an+b}{l+c}.
\]

Both of these equations are a generalization of \( y = \text{const} \cdot n/l \). The regression analysis results are presented in Table 5.7. Equation (3) is not a good fit for any of the efficiency values. Equation (4) is a good fit for Min \( Eff \) but not Mid \( Eff \) and is not even close for Max \( Eff \). This shows that the coefficient \( \kappa Q \) in the grammar efficiency does not behave like a constant, but rather depends on \( n \) and \( l \) in an intricate, nontrivial way. Thus, the use of the linear and sigmoidal equations in the analysis above is further justified.

<table>
<thead>
<tr>
<th></th>
<th>Min ( Eff )</th>
<th>Mid ( Eff )</th>
<th>Max ( Eff )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Equation (3)</td>
<td>-.1180</td>
<td>-.0251</td>
<td>.5357</td>
</tr>
<tr>
<td>Equation (4)</td>
<td>.7920</td>
<td>.6279</td>
<td>.0017</td>
</tr>
</tbody>
</table>

Table 5.7: \( R^2 \) values for Equations (3) and (4)
The three efficiency values that the surfaces were fit to depend only on the values of $l$ and $n$. The exact dependence is not known explicitly, so the sigmoidal equation is a good explicit model of how Min Eff and Max Eff depend on $l$ and $n$. 
CHAPTER 6

Conclusion

6.1 Main Results

The aim of this thesis was twofold. First, the validity of the efficiency formula for the two-number PoS system classification was determined. The next step was to determine the validity of the two-number PoS system classification by testing for correlation between the efficiency of a PoS system type and its two-number label. To do so, all of the theoretically possible PoS system types allowed by the PoS system classification had to be considered. The final analysis only focused on the 17 PoS system (sub)types which satisfied two constraints. The first constraint regulated the propositional phrases used in each PoS system and the second constraint restricted the word classes which could be used in a PoS system type.

The efficiency values for each PoS system type, including subtypes, have been determined using the efficiency formula. Each PoS system type has a unique interval of efficiency values except those systems which have equivalent structures and those which are maximally efficient. The difference in efficiency intervals for PoS systems which are not maximally efficient nor have similar structures implies the validity of the efficiency equation.
The sigmoidal equation is a good explicit model of how Min $Eff$ and Max $Eff$ depend on $l$ and $n$ based on the correlations between the Max $Eff$ and the two-number label and the Min $Eff$ and the two-number label. A strong correlation was determined between the Max $Eff$ values and two-number labeling of the PoS system types presented in the classification system. A reasonable correlation exists between the Min $Eff$ and the two-number labeling of the PoS system types. The correlation only exists for Max $Eff$ and Min $Eff$, which are the two directly calculated values based on the classification system. There is no strong correlation for Mid $Eff$. However, the Mid $Eff$ values may not even represent theoretically possible PoS system types.

The strong correlation between the efficiency values and the two-number labeling system means that the two-number PoS system classification system created by Vulanović is a reasonable method for language classification. This justification of the two-number classifications system also indirectly gives a stronger justification to Hengeveld’s PoS-based classification of word classes.

### 6.2 Future Research Directions

With the positive results concluded from this research, further research can be performed on the PoS system classification as a whole. A similar study can be conducted with the intermediate PoS system types. The efficiency formula can be modified to account for these intermediate PoS system types. Whether a correlation exists between the efficiency values and the two-number classification of the entire PoS system classification, including intermediate types, can be determined.
A modification to the classification system can be made to differentiate between the subtypes by including some suitable, third, numerical characteristic in addition to $l$ and $n$. This would give a three-number classification whose correlation with grammar efficiency could be investigated in a similar manner as the two-number classification.

The efficiency formula itself can also be looked at to see if any modifications would allow for a stronger correlation between the label of the PoS system types and their efficiency values.

The PoS system classification as a whole, once verified for intermediate PoS system type, can be made more complex by allowing more propositional functions to be considered in the classification. This would allow for a more specific classification of languages within the system.
REFERENCES


