# A GENERIC SYNTACTIC PARSER
# FOR TURKIC LANGUAGES

by

Harun Reşit ZAFER

A thesis submitted to

the Graduate Institute of Sciences and Engineering

of

Fatih University

in partial fulfillment of the requirements for the degree of

Master of Science

in

Computer Engineering

August 2011
Istanbul, Turkey

# APPROVAL PAGE

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Assoc. Prof. Veli HAKKOYMAZ
Head of Department

This is to certify that I have read this thesis and that in my opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Assist. Prof. Atakan Kurt
Supervisor

Examining Committee Members

Assist. Prof. Atakan Kurt ______________________

Assoc. Prof. Veli HAKKOYMAZ ______________________

Assist. Prof. Kalmamat KULAMSHAEV ______________________

It is approved that this thesis has been written in compliance with the formatting rules laid down by the Graduate Institute of Sciences and Engineering.

Assoc. Prof. Nurullah Arslan
Director

August 2011

# A GENERIC SYNTACTIC PARSER FOR TURKIC LANGUAGES

Harun Reşit ZAFER

M. S. Thesis - Computer Engineering
August 2011

Supervisor: Assist Prof. Dr. Atakan KURT

## ABSTRACT

Syntax defines how to construct word groups with words and sentences with word groups. These words and/or word groups are called syntactic parts. A tool that analyzes a sentence into its syntactic parts is called syntactic parser. Parsing is one of the major tasks of natural language processing and parsers have a wide area of use in real world applications. In this thesis, we developed a generic parser for Turkic languages. The parser depends on CFG (context-free grammar) rules, morphological analysis and validation rules. System works with any Turkic language if required files of the language such as CFG and validation rules and a morphological analyzer are provided. We studied Turkish and Turkmen grammar from a computational point of view. Then we implemented the system for Turkish and slightly modified it for Turkmen.

**Keywords**: Natural Language Processing, Syntactic Parsing, Context Free Grammar, Turkic Languages

# TÜRKÎ DİLLER İÇİN
# UYARLANABİLİR SÖZDİZİMSEL AYRIŞTIRICI

Harun Reşit ZAFER

Yüksek Lisans Tezi – Bilgisayar Mühendisliği
Ağustos 2011

Tez Danışmanı: Yrd. Doç. Dr. Atakan KURT

## ÖZ

Sözdizimi kelimelerin kelime gruplarını, kelime gruplarının da cümleleri nasıl oluşturacaklarını belirler. Bu kelime ve/veya kelime gruplarına sözdizimsel ögeler denir. Cümleyi sözdizimsel ögelerine ayıran araçlara ise sözdizimsel çözümleyici adı verilir. Sözdizimsel çözümleme doğal dil işlemenin temel konularından biridir ve sözdizimsel çözümleyicilere gerçek dünya uygulamalarında yaygın olarak ihtiyaç duyulur. Bu tez çalışmasında Türkî diller için genel bir cümle çözümleyici geliştirildi. Geliştirilen çözümleyici bağlamdan-bağımsız gramer (CFG) kurallarına, morfolojik çözümlemeye ve geçerlilik kurallarına dayanmaktadır. Geliştirilen system CFG ve geçerlilik kurallarını içeren dosyalar ile bir morfolojik çözümleyiciye sahip olan her Türkî dil için çalışmaktadır. Türkçe ve Türkmencenin grameri bilgisayısal açıdan incelenmiş ve sistem Türkçe için hayata geçirilmiştir. Daha sonra kısmi değişiklikler ile Türkmenceye uyarlanmıştır.

**Anahtar Kelimeler:** Doğal Dil İşleme, Sözdizimsel Ayrıştırma, Bağlamdan Bağımsız Dilbilgisi, Türkî Diller

*To Mahmud al-Kashgari (Kâşgarlı Mahmud)*

# ACKNOWLEDGEMENT

Firstly, I would like to thank sincerely to Asst. Prof. Dr. Atakan KURT and Prof. Dr. Mehmet KARA for their guidance and insight throughout the research.

Then it is a pleasure to thank those who made this thesis possible with their valuable work:

Thank you, Leyla Karahan, for writing such a great syntax book of Turkish.

Thank you, Zemberek team, for writing the only available and free morphologic analyzer for Turkish.

Thank you, Scott Martin, for his great Earley parser implementation in Java and his support by answering my emails.

Thank you, my dear friend Musa Topkaya, for preparing the list of transitive and non-transitive verbs of Turkish.

I express my thanks and appreciation to my family for their understanding, motivation and patience. Lastly, but in no sense the least, I am thankful to all colleagues and friends who made my stay at the university a memorable and valuable experience.

# TABLE OF CONTENTS

# LIST OF FIGURES

FIGURE

# LIST OF SYSMBOLS AND ABBREVIATIONS

| SYMBOL/ABBREVIATION | |
|---|---|
| 1pl, 2pl, 3pl | First, second, third person plural |
| 1sg, 2sg, 3sg | First, second, third person singular |
| Abl | Ablative Case |
| Acc | Accusative Case |
| adj | Adjective |
| Adv | Adverb Constituent |
| adv | Adverb |
| AP | Adverbial Phrase |
| CFG | Context-Free Grammar |
| CG | Conjunction Group |
| Comp | Complement |
| CV | Compound Verb Group |
| Dat | Dative Case |
| DObj | Definite object |
| Excl | Excluded Constituent |
| Gen | Genitive Case |
| IG | Interjection Group |
| Ins | Instrumental Case |
| IObj | Indefinite object |
| intj | Interjection |
| JC | Adjective Complement |
| JP | Adjectival Phrase |
| Loc | Locative Case |
| NC | Noun Complement |
| NG | Number Group |

| | |
|---|---|
| noun | Noun |
| NP | Noun Phrase |
| p1p, p2p, p3p | Possessive suffixes first, second and third person plural |
| p1s, p2s, p3s | Possessive suffixes first, second and third person singular |
| PG | Postposition Group |
| plu | Plural |
| PNG | Person Name Group |
| posp | Postposition |
| Poss | Possessive Suffix |
| Pre | Predicate |
| pron | Pronoun |
| RG | Reduplication Group |
| Sub | Subject |
| TG | Title Group |
| VAG | Verbal Adverb Group |
| VJG | Verbal Adjective Group |
| VNG | Verbal Noun Group |
| VP | Verb Phrase |

# CHAPTER 1

# INTRODUCTION

Human computer communication is established through programming (artificial) languages. These languages are used to define tasks that can be performed by computers. Like human (natural) languages, each programming language has a grammar. This grammar is usually just involves syntactic rules and called syntax of the language. Each syntactically correct expression has a precise meaning (semantics). Thus each expression is an instruction and a set of instructions constitute a computer program.

However a grammatically correct expression in a natural language may not have a meaning or may have many different meanings. Despite natural languages are easy to understand for humans, it is a nontrivial task for computers. Computers need a deeply analysis of natural language in many aspects of linguistics such as morphology, syntax, semantics or pragmatics.

Natural Language Processing (NLP) is an area of computer science and linguistics. Main goal of NLP is analyzing human languages computationally. NLP has many real world applications such as machine translation, speech recognition, information extraction, information retrieval, question answering, automatic summarization, optical character recognition (OCR) etc. Real world applications depend on major tasks of NLP. Morphologic analysis is one of these major tasks which is separating a word into its root and affixes (morphemes). Part of speech tagging is another task which is determining the linguistic category (noun, verb, etc.) of each word in a sentence.

Syntax defines how to construct word groups with words and sentences with word groups. These words and/or word groups are called syntactic parts. Syntactic analysis (parsing) is another major task in NLP. It is the process of analyzing a sentence into its syntactic parts. A tool that analyzes a sentence into its syntactic parts is called syntactic parser or simply parser. Parsers are important because knowledge of syntax is essential for many real world applications such as question answering, information extraction, sentence generation, translation, etc. For instance translating a sentence from source language to target language word by word mostly causes inaccurate translations. The order of words is usually different in target language. Also some words can constitute a word group which is not supposed to be translated separately. Thus a machine translation system requires first syntactic analysis of source language. Then it should arrange the order of words and word groups to construct syntactically correct sentences in target language.

## 1.1 STATEMENT OF THE PROBLEM

The main purpose of this study is design and implementation of a syntactic analyzer (parser) which can efficiently be adapted to different Turkic languages. The implementation phase consists of developing parser as a software tool and adapting this tool to Turkish and Turkmen languages which are two of official Turkic languages.

Turkic languages are a language group of approximately thirty five languages which is spoken by 250 million people, including nonnative speakers. They are accepted as a subfamily of Altaic language family. Turkic languages are spoken across a vast area from Eastern Europe and the Mediterranean to Siberia and Western China.

Turkish is the most commonly spoken Turkic Language. During its nearly 1200 years old journey around a very large area from central Asia to east Europe, it has been influenced from many cultures. Since it is the official language of a remarkable world power, Turkish Republic, it is the dominant language among these languages and also the most spoken Turkic Language as a second language. The official languages of five other Turkic republics are Azeri, Kazakh, Turkmen, Uzbek and Kirghiz and they are accepted to be the main Turkic Languages.

There is an appreciable similarity between these languages in terms of both syntax and morphology. They also have a large common vocabulary. However the mutual intelligibility of Turkic Languages is not high. DİLMAÇ (Shylov, 2008) is a generic morphologic analyzer which makes use of this similarity. It can easily be adapted to any Turkic language by providing a word list, an XML file that includes morphologic rules and another XML file that includes orthographic rules. DİLMAÇ also has a translation module that performs word by word translation among these languages.

Especially in terms of syntax, Turkic languages are significantly similar. Providing that each language can be analyzed morphologically, it is feasible to develop a generic parser for Turkic languages. Similar to DİLMAÇ, a parser can perform syntactic analysis for any Turkic language whose syntactic information is provided with language specific files.

In this work the first generic parser for Turkic languages is implemented. Turkish and Turkmen syntax has been researched and context-free grammars for these languages are designed and written. CFG rules and other necessary syntactic information are prepared in the suitable form which the parser accepts. The parser has been tested with both Turkish and Turkmen phrases.

It is believed that there are several contributions of this work:

1. Besides this is the first known attempt to develop a generic parser for Turkic languages, this is also the first parser for Turkmen. There is no known study about developing parser for any Turkic language other than Turkish. Thus it is likely that the parser will also be the first for others with the adaptation of these languages.

2. For Turkish there have been similar studies for the last 20 years. However there is no freely or commercially available parser tool for Turkish yet. As a future work it is planned to make this parser available online to provide an open tool for linguists and other researchers that are interested in Turkic languages.

3. The source code, grammar rules and other implementation information of previous work is also not available. This work will continue as an open source project. We believe that it will be very beneficial for future researches.

4. As mentioned in the following chapters syntactic parsing is highly dependent to morphologic analysis. In this work, from a computational view we show that Turkic languages are both morphologically and syntactically very similar.

## 1.2 DESCRIPTION OF REMAINING CHAPTERS

Following paragraphs explain rest of the chapters briefly:

In Chapter 2, a general review of literature is given. The first part explains briefly linguistic studies for Turkish and Turkmen. The second part summarizes previous work for parsing Turkish.

Chapter 3 provides the necessary background information for the next chapters. It explains the notion of syntactic parsing and context-free grammar. Then in the last section, how parsing is strongly related to morphologic analysis for Turkic languages is discussed.

Chapter 4 explains Turkish and Turkmen grammar from point of natural language processing's view, especially regarding context-free grammar.

Chapter 5 gives design and implementation details of system. Firstly, how morphologic analyzers and the parser work together is explained. Then main CFG rules for Turkish and Turkmen are given. Finally validation of parse trees is described.

Chapter 6 discusses the contribution that is made by this study. Explains the limitations both arise from the structure of Turkic grammar and lack of morphologic analyzers for Turkic languages. Finally it explains the future work to be done.

# CHAPTER 2

# A REVIEW OF LITERATURE

In this study Turkish and Turkmen are selected to be the initial languages of the system. Due to time limitations other Turkic languages are left as future work. Thus this section is focused on these two languages. Nevertheless it is known that from linguistic perspective there are many researches and resources for Turkic languages. However from point of computational linguistics, studies about syntax and parsing are only existed for Turkish.

## 2.1 RELATED WORK FROM LINGUISTICS PERSPECTIVE

The first systematic studies about Modern Turkmen grammar has been done in Soviet Union starting with 20th century (Clark, 1998). After collapsing of Soviet Union, the attention to Turkmen and other Turkic languages spoken in this territory increased rapidly. Especially in Turkey researchers studied Turkmen from point of Turkish grammarians.

Kara's book (Kara, 2005) is one of these studies about Turkmen grammar. Biray (Biray, 1985) presented a comparison of Turkish and Turkmen syntax. Karasandık's study (Karasandık, 1998) is another comprehensive work about Turkmen sentence structure.

The first known grammar book about Anatolian Turkish has written in 1530. From that time, there have been many studies about Turkish syntax from a linguistic perspective (Özçam, 1997). However this thesis mostly depends on the study of Karahan, "Türkçede Söz Dizimi" (Karahan, 2008). Karahan's work covers all previous studies and includes many examples from Turkish literature.

There are also several studies involving syntax of other Turkic languages. Karaörs (Karaörs, 2005) compiled articles comparing Turkish Syntax with Kazakh and Azeri. Ertuğrul (Ertuğrul, 2000) made a comparison of Turkish syntax with Uzbek.

## 2.2 RELATED WORK FROM PERSPECTIVE OF COMPUTATIONAL LINGUISTICS

In terms parsing Turkish presents characteristic challenges such as free order of constituents or long dependencies. As explained in section 3.3, syntactic analysis cannot be done without help of a complete morphological analysis of each word in the sentence. Natural language processing studies about Turkish syntax has been studied for the last 20 years. Researchers tried to apply different grammar formalisms to model Turkish grammar.

One of the earliest works about parsing Turkish has been done by Güngördü (Güngördü, 1993). In this study a lexical-functional grammar has been presented for Turkish. Güngördü used 7 categories such as noun, adjectival, adverbial and verb phrases in addition to sentences, dependent clauses and lexical rules. The grammar consisted of more than 200 rules.

Hoffman, in several studies proposed combinatory categorical grammar approach to handle the word order freeness in Turkish (Hoffman, 1992, 1994, 1995).

Göçmen, Şehitoğlu and Bozşahin has summarized Turkish Syntax independent from a particular linguistic theory (Göçmen et al., 1995). The study may be a point of start for NLP researches who want to study Turkish syntax.

Bozşahin and Göçmen proposed a uniform grammatical architecture based on combinatory categorical grammars (Bozşahin & Göçmen, 1996). They used an architecture that integrates morphology, syntax and semantics instead of accepting them as separate modules. A parser based on this architecture has been implemented and tested on Turkish causatives.

Şehitoğlu presented a HPSG (Head-driven Phrase Structure Grammar) for Turkish. Turkish syntax has been analyzed in 5 main categories that are noun, postposition, adjective, adverb and verb (Şehitoğlu, 1996). A parser based on this model was implemented with ALE (Attribute Logic Engine).

Temizsoy and Çiçekli addressed the problem of ambiguity for syntactic analysis and proposed an ontology-based approach (Temizsoy & Çiçekli). Their study was mostly based on semantic analysis which accepts or rejects ambiguous syntactic analyses. System's output was TRM (Text Meaning Representation) of sentences instead of parse trees.

As distinct from context-free and combinatory approaches, Oflazer et al. proposed dependency parsing for Turkish (Oflazer, 2003), (Eryiğit et al., 2006, 2008). In dependency parsing there is no phrasal structures. Instead, a word (head) linked with its dependents constitutes structures. It is claimed that dependency parsing is more suitable for languages with free word order such as Turkish.

Oflazer et al. built a Turkish Treebank which has been still using by researchers to build or evaluate NLP tools (Oflazer et al., 2003). The METU-Sabancı Turkish Treebank includes more than 7000 morphologically and syntactically annotated sentences which are taken from METU Turkish corpus. The rule based dependency parser is used for pre-annotation then sentences were annotated by humans. The Treebank is particularly designed for dependency parsing and doesn't contain POS (part of speech) tags (Oflazer & Atalay, 2003). Thus it is not efficient to use the Treebank with context free-based tools.

Depending on this Treebank, Eryiğit et al. applied statistical methods to rule based dependency parser (Eryiğit et al., 2006, 2009). It is used for both training and testing the statistical parser.

Different than the syntax theories above İstek presented a link grammar for Turkish (İstek, 2006).

# CHAPTER 3

# BACKGROUND

## 3.1 PARSING OR SYNTACTIC ANALYSIS

Syntax defines how to construct word groups with words and sentences with word groups. These words and/or word groups are called syntactic parts or syntactic constituents. The process of analyzing a sentence into its syntactic parts is called syntactic analysis or simply parsing.

There are many linguistic theories (grammars) to model constituent structure in natural languages. Accuracy and performance of parsing process highly depends on the model used for the language to be parsed. The compatibility of the natural language and the grammar is important. The selected grammar must have the ability to handle all characteristics of the language. The existence of efficient algorithms to parse the selected grammar is also important for the performance of the system.

It is a fact that CFG is not very suitable with the languages having complex morphology such as Turkish. However it is known with its simplicity and there are many efficient algorithms and parsers based on these algorithms. Besides many more complex grammars such as lexical functional grammar or head driven phrase structure grammar are extensions of CFG. That’s why a complete CFG grammar of Turkish presented in this work will be a good point of start for future studies.

## 3.2 CONTEXT FREE GRAMMAR

Context-free grammar (also called **phrase structure grammar**) is the most commonly used mathematical system for describing the structure of syntactic parts in natural languages (Jurafsky & Martin, 2009). As mentioned in the previous section there are many extensions of CFG such as LFG, HPSG, and construction grammar.

The notation used to express CFGs is called Backus–Naur Form, or BNF. A CFG consists of a rule set and a lexicon of symbols. A rule has a nonterminal symbol at the left hand side and terminals and/or nonterminal symbols at the right hand side:

S ⟶ SS
S ⟶ (S)
S ⟶ ()

Here S is the nonterminal symbol and () is the terminal one. This grammar produces strings with well-formed parenthesis such as:

(), ()(), (()), (()(()))

The following grammar produces strings that have equal number of a symbol on left hand side to number of symbol b on right hand side.

S ⟶ aSb
S ⟶ ab

This grammar can generate strings like:

ab, aabb, aaabbb, aaaabbbb, ….

Following example is a very simple CFG for handling SOV type Turkish sentences with the words “Ali”, “büyük”, “kitabı”, “verdi”.

S ⟶ VP
S ⟶ OBJ VP
S ⟶ SUB OBJ VP

SUB → Ali
SUB → büyük
Adj → büyük
OBJ → Adj kitabı
OBJ → kitabı
VP → verdi

The following sentence can be generated by this grammar:

‘Ali büyük kitabı verdi’ (Ali gave the big book). The parse tree of this sentence is as follows:

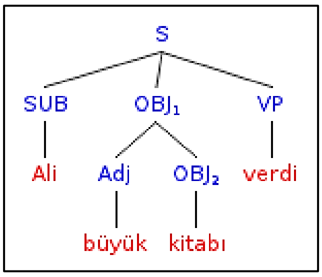

**Figure 1:** Sample parse tree for the sentence "ali büyük kitabı verdi"

Another way of expressing a parse tree is using bracketed notation:

```
[S[SUB[ali]][OBJ[Adj[büyük]][OBJ[kitabı]]][VP[verdi]]]
```

The sentence “büyük kitabı verdi” has more than one parse trees:

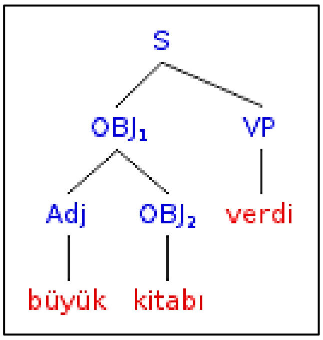

**Figure 2:** The first parse tree for the phrase "büyük kitabı verdi"

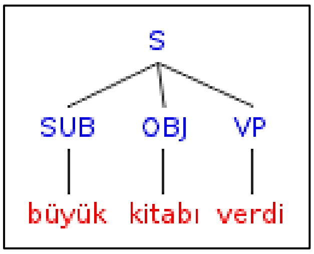

**Figure 3:** The second parse tree for the phrase "büyük kitabı verdi"

Since there are more than one parse tree for the sentence, it is ambiguous.

Mainly there are two approaches to parsing. Top-down parsing tries to build the tree starting from the root node S and continue construction down to the leaf nodes. Bottom-up parsing on the other hand, tries to build the tree staring from input tokens up to the root node.

Besides each method has its advantages and disadvantages, both methods are not efficient because of ambiguity during the construction of tree. To overcome this, some methods use dynamic programming. CYK algorithm is one of these which implements bottom-up approach. On the other hand, Earley algorithm uses dynamic programming to implement top-down approach.

### 3.3 RELATION BETWEEN MORPHOLOGIC ANALYSIS AND SYNTACTIC PARSING

Parsing process is determining the relation and dependency of constituents in a sentence or a phrase. In languages which have rather simple morphology, constituents are related to each other mostly according to their position. However in agglutinative languages, constituents are dependent to each other mostly with suffixes. This makes changing the position of constituents possible and the language more flexible.

Turkic languages are typical examples to this phenomenon. Replacing the position of constituents does not change the meaning but emphasis. In the following example the first sentence has the most general constituent order in Turkish.  All 6 arrangements of 3 words are possible:

| | |
|---|---|
| - ben eve gittim | ‘I went to home’ |
| - eve ben gittim | |
| - ben gittim eve | |
| - … (3 more combinations) | |

If morphologically analyze the first sentence, we saw that constituents are connected to each other with suffixes.

ben/pron ev/noun +e/dat git/verb +ti/tense +m/1sg

The pronoun “ben” is connected to the verb “git” with the 1st singular person suffix “+m” and the noun “ev” is connected to the verb with the dative suffix “+e”. As long as words have the suffixes which connect them to each other, they can freely change their order. Following figure shows the parse trees for 3 versions of the sentence:

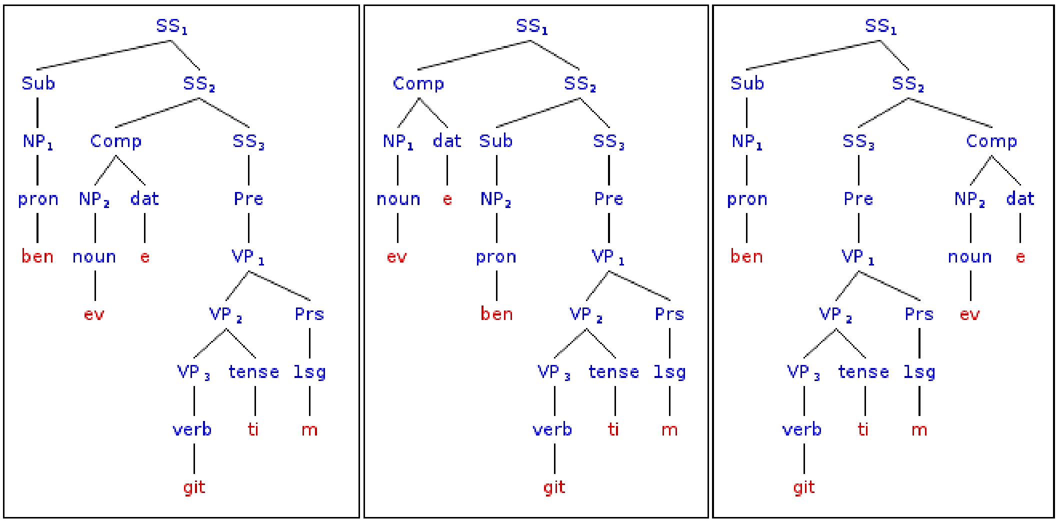

**Figure 4:** Parse trees for 3 versions of the sentence ‘ben eve gittim’

The sentence above is just a little example to show that suffixes are more important than words in terms of syntax. That’s why before parsing process, a complete morphologic analysis for each word in the sentence is vital.

# CHAPTER 4

# TURKISH AND TURKMEN SYNTAX

Linguists who had studied Turkish syntax have different views and classifications about syntactic constituents of Turkish. Karaörs explains different approaches of cardinal Turkish linguists and makes brief comparisons of their views (Karaörs, 2005). This study is mostly based on the book written by Karahan (Karahan, 2008). The book explains the Turkish syntax very well with many examples that are mostly from Turkish literature. Also studies for other Turkic languages use same approach.

## 4.1 TURKISH SYNTAX

This section is mostly adopted from Karahan's study (Karahan, 2008) and most of the phrases are taken from there.

In Turkish, a sentence consists of words and word groups. A word group has two or more words. Each word group acts as a single word in the sentence and has a part of speech that can be noun, verb, adjective and adverb.

- Dallarda uzanan hışırtılar/noun, ağaçtan ağaca sürüklenerek/adverb, ormanın kızıl derinliklerinde/noun kayboluyordu/verb.

Words and word groups are dependent to each other with case suffixes. The case suffix in the last word of a word group actually belongs to the word group, not the word.

- Çalışan insan/gr1, kendi varlığında hüküm süren aheng+i/gr2 bütün kainat+a/gr3 nakleder/verb.

In this sentence the second and third word groups are connected the verb with accusative (+i) and dative (+a) case suffixes.

Word groups with more than two words have nested word groups.

- küçük odadaki mumun/gr1      soluk ışığı/gr2
- candle+gen in the small room      dim light+p3s
- 'the dim light of the candle in the small room'

In group 1 there is another group:

- Küçük odadaki/gr1   mum
- Small room+loc+rel   candle
- 'The candle in the small room'

### 4.1.1 Noun Complement

A noun complement consists of two nominal elements. The first constituent is called completing element (Cg) and the second one is called completed element (Cd). Completing element may have a genitive case suffix. If so, the noun complement is called definite. Otherwise it is called indefinite noun complement. The completed element always takes a possessive suffix.

nominal [+gen] + nominal + Poss

- köy   yolu      (indefinite NC)
- village   road+p3s
- 'village road'

- bizim sesimiz      (definite NC)
- biz+gen ses+p1p
- 'our voice'

A definite noun complement’s completing element can also be a noun phrase.

- elbisenin yakasının/Cg    düğmesi/Cd
- collar of dress+gen    stud+p3s
- ‘collar stud of dress’

- elbisenin/Cg    yakası/Cd
- dress+gen    collar+p3s
- ‘collar of dress’

The part of speech of a noun complement can be noun, adjective or adverb in a sentence.

### 4.1.2 Adjective Complement

An adjective complement (JC) is constructed by an adjectival and a nominal element. Nominal element is described by the adjectival element.

- verimli/adj    topraklar/noun
- fertile    soil+plu
- ‘fertile soils’

In an adjective complement, elements can also be word groups.

- verimli topraklı/adj büyük çiftlik/nom    ‘big farm with fertile soil’
- verimli/adj toprak/noun    ‘fertile soil’ (JC) +lI
- büyük/adj çiftlik/noun    ‘büyük çiftlik’ (JC)

- siyah/adj bir ekmek parçası/nom    ‘a black piece of bread’
- bir/adj ekmek parçası/nom    ‘a piece of bread’ (JC)
- ekmek/noun parçası/noun    ‘piece of bread’ (NC)

The part of speech of an adjective complement can be noun, adjective or adverb.

### 4.1.3 Verbal Adjective Group

Verbal adjective group (VJG) is constructed from a verbal adjective (VJ) and one or more complements (completing constituents) connected to it. The constituents are subject, object, indirect object or adverb. The verbal adjective in the group acts as predicate but doesn’t express a judgment.

- dünyayı/Obj kurtaran/VJ (adam)
- world+acc save+An (man)
- ‘the man who saves the world’

- bütün hayalleri/Obj yıkılmış/VJ (insanlar)
- their all dreams+acc crush+tense (indefinite past) (people)
- ‘the people whose all dreams had been crushed’

The order of constituents in VJG may change freely. The part of speech of a verbal adjective group can be noun, adjective or adverb.

### 4.1.4 Verbal Noun Group

Verbal noun group (VNG) is constructed from a verbal noun and one or more complements connected to it. When a verb takes one of the suffixes –mAk, -mA or -Uş, it becomes a verbal noun. The complements (constituents) are subject, object, indirect object or adverb. The verbal noun in the group acts as predicate but doesn’t express a judgment.

- seni/Obj buraya/Comp getirmek/VN
- you+acc here+dat bring+mAk
- ‘bringing you here’

- Uzun bir ayrılıktan sonra/Adv sılaya/Comp dönüş/VN
- After a long separation home+dat come+Uş back
- ‘Coming back to home after a long separation’

The part of speech of a verbal noun group is noun.

### 4.1.5 Verbal Adverb Group

Verbal adverb group (VAG) is constructed from a verbal adverb and one or more complements connected to it. The complements are subject, object, indirect object or adverb. The verbal adverb in the group acts as predicate but doesn't express a judgment.

| | | |
|---|---|---|
| • adam/Sub | evine/Comp | giderken/VA |
| • man | house+Poss+dat | go+tense+kAn |
| • 'while the man is going to his house' | | |

| | |
|---|---|
| • sen/Sub | gülünce/VA |
| • you | smile+UncA |
| • 'when you smile' | |

Part of speech of a verbal adverb group is always adverb in a sentence.

### 4.1.6 Reduplication Group

Reduplication Group consists of a base word and a reduplicant. Base and reduplicant can be same word or have same part-of speech. They are related in terms of both form and meaning.

Some examples of reduplications with same word:

| | |
|---|---|
| • mışıl mışıl (uyumak) | 'soundly asleep' |
| • yavaş yavaş | 'slowly' |
| • koşa koşa | 'as running' |

Examples of reduplications with words having close meanings:

| | |
|---|---|
| • doğru dürüst | 'properly' |
| • eğri büğrü | 'twistedly, scratchy' |

Examples of reduplications with words having opposite meanings:

| | |
|---|---|
| • ileri geri | 'back and forth' |
| • ölüm kalım | 'do-or-die' |

Examples reduplications using a meaningful and a meaningless word:

| | |
|---|---|
| çer çöp | ‘the small fry’ |
| çoluk çocuk | ‘offspring’ |

In Turkish by adding an “m” sound to the base word new reduplications can be derived:

| | |
|---|---|
| kalem malem | ‘no pencil no nothing’ |
| kitap mitap | ‘no book no nothing’ |

Reduplication group can take inflectional suffixes. In this case the suffix is affixed to both words.

| | | |
|---|---|---|
| çoluğu | çocuğu | (getirmek) |
| çoluk+acc | child+acc | (bring) |
| ‘bringing the offspring’ | | |

| | | |
|---|---|---|
| sağa | sola | (bakmak) |
| right+dat | left+dat | (look) |
| ‘look around’ | | |

Part of speech of a reduplication group can be noun, adjective, adverb or verb.

### 4.1.7 Postposition Group

Postposition group (PG) consists of a nominal element and a postposition.

| | |
|---|---|
| yaşadığım/nom | gibi/posp |
| live+DUk+Poss | as, like |
| ‘as I have lived’ | |

- çocuklar için
- children for
- ‘for children’

PG always starts with the nominal element and postposition always takes place at the end.

- bir demet çiçek/nom ile/posp
- a flower bouquet with
- ‘with a flower bouquet’

The nominal element can take a suffix according to the type of postposition.

- yaşamak için
- live to (no suffix)
- ‘to live’

- sen**in** gibi
- you+Poss like (possessive suffix)
- ‘like you’

- deniz**e** doğru
- sea+dat towards (dative)
- ‘towards sea’

- bun**dan** dolayı
- this+abl because of (ablative)
- ‘because of this’

Nominal element of the group can also be a word group. Part of speech of a postposition group can be noun, adjective, adverb.

### 4.1.8 Conjunction Group

Conjunction group (CG) consists of two or more nominal elements connected with conjunctions.

- kırmızı ve siyah
- red and black

- ‘red and black’

| | | |
|---|---|---|
| babalar | ile | oğulları |
| father+plu | with | sons+plu+p3p |

- ‘fathers and their sons’

| | | |
|---|---|---|
| olmak | veya | olmamak |
| be+mAk | or | be+neg+mAk |

- ‘to be or not to be’

| | | |
|---|---|---|
| güzel | ama | küstah |
| beautiful | but | insolent |

- ‘beautiful but insolent’

If there are more than two nominal in the group, comma is used between the elements until the last two. The postposition “ve” is used between the last two elements.

| | | | | |
|---|---|---|---|---|
| Okumak | , | anlamak | ve | uygulamak |
| Read+mAk | cm | understand+mAk | and | apply+mAk |

- ‘reading, understanding and applying’

There are also conjunctions like “ne… ne…”, “hem… hem…”, “…da …da” etc.

| | | | |
|---|---|---|---|
| ne | sevinç | ne | üzüntü |
| what | joy | what | sorrow |

- ‘neither joy nor sorrow’

| | | | |
|---|---|---|---|
| eli | de | ayağı | da |
| hand+p3s | too | foot+p3s | too |

- ‘both his hand and his foot’

In a CG with this structure the number of nominal elements is equal to the number of postpositions.

| | | | | | |
|---|---|---|---|---|---|
| Ya | şunu | ya | bunu | ya da | onu |
| or | that+acc | or | this+acc | or and | it+acc |
| ‘that or this or it’ | | | | | |

Nominal elements of the group can be word groups.

| | | | | |
|---|---|---|---|---|
| aklın | ziyası/NC | ve | kalbin | nuru/NC |
| mind+gen | light+p3s | and | heart+gen | glory+p3s |
| ‘light of mind and glory of heart’ | | | | |

The part of speech of a conjunction group can be noun, adjective, adverb.

### 4.1.9 Title Group

Title group consists of a person name and a title or a relationship name.

| | |
|---|---|
| Fevziye | hanım |
| Fevziye | Ms/Mrs |
| ‘Ms/Mrs Fevziye’ | |

| | |
|---|---|
| Osman | dayı |
| Osman | uncle |
| ‘uncle Osman’ | |

Person name always comes before the title or relationship name.

| | |
|---|---|
| *Çağrı Bey* | *‘Lord Çağrı’* |
| *Mehlika Sultan* | *‘Sultan Mehlika’* |
| *Oğuz Han* | *‘Oghuz Khan’* |
| *Enver Paşa* | *‘General Enver’* |

Person name can be a compound proper name.

| | |
|---|---|
| *Mehmet Akif Bey* | *‘Mr Mehmet Akif’* |
| *Kazım Karabekir Paşa* | *‘General Kazım Karabekir’* |

The part of speech of title group is always noun in the sentence.

### 4.1.10 Person Name Group

PNG consists of proper nouns coming together to be a person name. It involves basically first names and a last name.

- Ziya Gökalp
- Yahya Kemal Beyatlı
- Necip Fazıl Kısakürek

The part of speech of personal name group is always noun in the sentence.

### 4.1.11 Interjection Group

Interjection group consists of an interjection (exclamation) and a nominal constituent.

- ey Türk gençliği !
- o Turkish youth+p3s excl
- ‘Turkish youth!’ or ‘Oh, Turkish youth’

The interjection (ey, ay, hey, bre, a, ya, yahu etc.) always comes before the nominal.

- a canım
- Oh life+p1s
- ‘Oh my dear!’

- be kardeşim
- excl brother+p1s
- ‘Oh man!’ or ‘come on man!’

The nominal constituent can be a word group.

- Ey/intj mavi göklerin beyaz ve kızıl süsü/NC

- Oh blue sky+plu+p3s white and crimson ornament+p3s
- ‘Hey, the crimson and white ornament of blue skies’

This group is used as salutation in sentences but it is not accepted as a part of sentence. The group’s part of speech is always interjection.

### 4.1.12 Number Group

This group covers numbers. Number nouns are ordered according to the digit system. They get bigger from end to beginning. They join together without taking any suffix.

| | |
|---|---|
| altmış dört | ‘sixty four’ |
| seksen dokuz | ‘eighty nine’ |

- yüz yetmiş beş
- hundred seventy five
- ‘one hundred and seventy five’

- beş yüz on iki
- five hundred ten two
- ‘five hundred and twelve’

Main numbers are accepted as adjective complement, others are accepted as number group.

| | | |
|---|---|---|
| iki yüz | ‘two hundred’ | JC |
| beş bin | ‘five thousand’ | JC |
| otuz milyon | ‘thirty million’ | JC |

| | | |
|---|---|---|
| on bir | ‘eleven’ | NG |
| doksan iki | ‘ninety two’ | NG |
| yüz elli dört | ‘one hundred and fifty four’ | NG |

Numbers less than a million have two elements, millions may have three, and billions may have four elements.

- Yedi yüz/JC elli/num
- Seven hundred fifty
- ‘seven hundred and fifty’

- Üç milyon/JC sekiz yüz bin/NG dokuz/num
- Three million eight hundred thousand nine
- ‘three million, eight hundred thousand, nine’

A Number Group’s part of speech can be noun or adjective.

### 4.1.13 Compound Verb Group

A compound verb group corresponds or describes an action. That’s why they are analyzed in two categories.

#### 4.1.13.1 Compound Verbs that Correspond an Action

They consist of a nominal element and a verb. The verb can be either a helper (auxiliary) verb or a verb being used in a different meaning than its dictionary meaning.

(A) Compound Verbs with helper verbs

Verbs such as “et-, ol-, yap-, eyle-, kıl-, bulun-” are used as auxiliary verb in this group. Auxiliary verb is always at the end.

- dost ol
- friend be
- ‘to be friends’

- yok et
- nonexistent do
- ‘destroy’

The nominal constituent can be a verbal adjective. If so the auxiliary verb can only be “ol-“  and “bulun-“.

- gelmez   oldun.
- come+neg+tense   be+tense+2sg
- ‘you never come anymore’

- yapmış   bulundu.
- do+tense   attend+tense+3sg
- ‘he/she had done it already’

The nominal constituent of this group can be word group.

- gürültü ve   yaramazlık/nom   ederdim/verb.   (nom: CG)
- nosie   and   naughtiness   do+tense+1sg
- ‘I used to be noisy and naughty’

- duyar gibi/Nom oldum/verb.  (nom: PG)
- hear+tense   be+tense+1sg
- ‘I seemed to hear’

(B) Compound Verbs with other verbs

These types of compound verbs are actually idiomatic phrases. The only possible way to determine and parse these groups is using a dictionary.

| | |
|---|---|
| • gönül vermek | ‘to set one's heart on’ |
| • yol almak | ‘to move forward’ |
| • ümit kesmek | ‘to abandon hope’ |

### 4.1.13.2 Compound Verbs that Describe an Action

This group consists of a verb which takes one of the verbal adverb suffixes ‘–A’, ‘-I’, or ‘-Ip’ and an auxiliary describing verb.

- koş +a +bil/aux = koşabil
- run +A able
- ‘be able to run’

- yaz +ı +ver/aux = yazıver
- write +I give
- ‘write up’

- gez +ip +dur/aux = gezip dur
- stroll Ip +stop
- ‘stroll away’ or ‘move about’

The auxiliary verbs “bil-, ver-, dur-, gel-, git-, kal-, koy, gör-, yaz-” adds the meaning of ability, possibility, duration, astonishment etc.

bil- ability, possibility
ver-: rapidness, easiness
git-, koy-, gör-, dur-: duration
gel-, kal: astonishment

## 4.2 DIFFERENCES OF TURKMEN SYNTAX FROM TURKISH

Turkmen (Tk) has a very similar syntax to Turkish (Tr). That’s why this section covers important differences of two. This section is mostly adopted from Biray’s study (Biray, 1985).

### 4.2.1 Noun Complement

Like in Turkish in Turkmen Noun Phrase are constructed with same formula.

nominal [+gen] + nominal + Poss

| Tk | | Tr | |
|---|---|---|---|
| erkinliğin | aşığı | hürriyet aşığı | |
| liberty+gen | lover+p3s | liberty lover+p3s | |
| lover of liberty | | | |

| | | | |
|---|---|---|---|
| bäägüliñ | açılışı | gülün | açılışı |
| rose+gen | bloom+Iş+p3s | rose+gen | bloom+Iş+p3s |
| ‘blooming of the rose’ | | | |

Some Noun Phrases in Turkish correspond to Adjective Phrases in Turkmen. The second element doesn’t take a suffix. Instead the first nominal element takes a suffix which changes its part-of-speech to adjective. This especially occurs with the dates.

| | | | |
|---|---|---|---|
| 1943-nci yıl (JC) | 1943 | yılı | (NC) |
| 1943rd year | 1943 | year+p3s | |
| ‘the year 1943’ | | | |

| | | | |
|---|---|---|---|
| 1962-nci yılda (JC) | 1962 | yılında | (NC) |
| 1962nd year+loc | 1962 | year+p3s+loc | |
| ‘in the year 1962’ | | | |

There are also other examples of this fact.

| | | | |
|---|---|---|---|
| agşamlık | nahar (JC) | akşam | yemeği |
| evening+lIk | meal | evening | meal+p3s |
| ‘dinner’ or ‘meal for the evening’ | | | |

Especially with usage of the reflexive pronoun, definite noun phrase is used instead of indefinite noun phrase.

| | | | |
|---|---|---|---|
| özümiñ | dirligim (def) | kendi | sağlığım (indef) |
| own+gen | health+p1s | own | health+p1s |
| ‘my own health’ | | | |

- yeriñ yüzü yer yüzü
- earth+gen face+p3s earth face+p3s
- ‘surface of earth’

Also there are some cases where indefinite noun phrase is used instead of definite noun phrase.

- Göz önüñ (indef) gözünün önü (def)
- Eye front+p2s göz+p2s+gen front+p3s
- ‘under one’s eyes’

### 4.2.2 Adjective Complement

Although Adjective Phrase of Turkmen is very similar to Turkish, there are minor differences.

(A) In some cases an adjective phrase corresponds to a definite noun phrase. This especially happens with the adjectives having the “-DUk” suffix in Turkish. In Turkmen the “-An” suffix is used with adjectives and the second constituent of the group takes a possessive suffix.

- duran yerinden durduğu yerden
- stand+An place+p3s+abl stand+DUk+p3s place+abl
- ‘from the place where he/she stands’

(B) In Turkmen words like “yok, bar, gerek” are used as adjective. In Turkish same words are used with different part-of-speech.

- onuñ bar/adj maksad onun var/nom olan (bütün) maksadı
- his exist aim his exist be (all) aim
- ‘all his aim’

- gerek/adj yerinde gerektiği/VJ yerde
- required place require+DUk+p3s place
- ‘when required’

### 4.2.3 Verbal Adjective Group

About verbal adjectives the most remarkable difference of Turkmen and Turkish is that in Turkmen the participle suffix “-An” is used instead of the participle suffix “-DUk” in Turkish.

| | | | |
|---|---|---|---|
| heleyiñ | bilyeni | kadının | bildiği |
| women+gen | know+An | women+gen | know+Duk |
| ‘what the women knows’ | | | |

| | | | |
|---|---|---|---|
| ondan | eşidenlerim | ondan | işittiklerim |
| him+abl | hear+An+plu+p1s | him+abl | hear+An+plu+p1s |
| ‘(the things that) I heard from him’ | | | |

### 4.2.4 Verbal Adverb Group

(A) The negative verbal adverb “-madan, -meden” in Turkish corresponds to “-man, -men” in Turkmen.

| | |
|---|---|
| açılman | açılmadan |
| open+mAn | open+mAdAn |
| ‘without opening’ | |

(B) Instead of the negative form of verbal adverb suffix “-Ip” in Turkish, “-man, -men” is used in Turkmen.

| | | | |
|---|---|---|---|
| gürrüñ | etmäñ | gürültü | etmeyip |
| noise | do+mAn | noise | do+neg+Ip |
| ‘without making noise’ | | | |

(C) Instead of the verbal adverb suffix “-IncA” in Turkish, the participle “-AndA” is used in Turkmen.

| | | | | | |
|---|---|---|---|---|---|
| gar | sırap | başlanda” | kar | yağmaya | başlayınca |
| snow | rain | start+AndA | snow | rain | start+IncA |

- ‘when it started to snow’

(D) Instead of phrases like “birinci olarak”, “ikinci olarak”, In Turkmen ordinal numbers like “birinci”, “ikinci” are used with ablative suffix.

| | | |
|---|---|---|
| birinciden | birinci | olarak |
| first+DAn | first | be+ArAk |
| ‘firstly’ | | |

### 4.2.5 Repetition Group

Turkmen also have similar repetition groups to Turkish.

| | | |
|---|---|---|
| böreñ böreñ | gürül gürül | ‘with a gurgling sound’ |
| aar naamıs | ar namus | ‘purity and honor’ |
| baarı yooğı | varı yoğu | ‘all that he has’ |

(A) In some repetition groups the suffix is only affixed to the last constituent.

| | | | |
|---|---|---|---|
| yüz | gözleri | yüzleri | gözleri |
| face | eye+p3p | face+p3p | eye+p3p |
| ‘their faces and eyes’ | | | |

(B) The repetitions constructed with ordinal numeral adjectives are constructed with cardinal numbers in Turkmen.

| | | |
|---|---|---|
| iki-iki | ikişer | ikişer |
| two-two | two+şAr | two+şAr |
| ‘two by two’ | | |

| | | |
|---|---|---|
| üç-üç | üçer üçer | ‘three by three’ |

(C) some other exceptional cases

| | | | |
|---|---|---|---|
| yıl- | yıldan | “yıldan | yıla” |
| year- | year+abl | year+abl | year+dat |

- ‘from year to year’

| | | |
|---|---|---|
| az-azdan | az az | ‘little by little’ |
| gün-günden | günden güne | ‘day by day’ |

### 4.2.6 Postposition Group

(A) In Turkmen the vocal gerundive “deyip” is used Instead of “diye” in Turkish.

| | | | |
|---|---|---|---|
| uçsam | diyip | uçsam | diye |
| fly+sAm | saying | fly+sAm | saying |
| ‘in order to fly’ | | | |

(B) In Turkmen the vocal gerundive “soñ” is used Instead of postposition “sonra” in Turkish.

| | | |
|---|---|---|
| düşünensoñ | düşündükten | sonra |
| think+An+soñ | think+DUk+abl | after |
| ‘after he thinks/thought’ | | |

### 4.1.7 Conjunction Group

(A) In Turkmen the conjunction “hem” is widely used instead of “ve” in Turkish.

| | | | | | |
|---|---|---|---|---|---|
| işçiler | hem | gullukcular | işçiler | ve | ameleler |
| worker+plu | and | laborer+plu | worker+plu | and | laborer+plu |
| ‘workers and laborers’ | | | | | |

(B) In some cases instead of using a conjunction like “ve”, the suffix “-DUr” is affixed to the first element of the group

| | | | | |
|---|---|---|---|---|
| geçidir | tilkiniñ | keçinin | ve | tilkinin” |
| goat+Dur | fox+gen | goat+gen | and | fox+gen |
| ‘goat’s and fox’s’ | | | | |

### 4.2.8 Compound Verb Group

(A) In Turkish the ability auxiliary verb “-bil” is used in positive meaning. However in negative ability “-yAmA” suffix is used. In Turkmen “–bil” suffix is used for both occasions. And the ability auxiliary is used as a separated word.

- durup bilmedi” “duramadı”
- stop+Ip bil+neg+tense+3sg stop+yAmA+tense+3sg
- ‘he couldn’t stop’

- getirip bilerdik getirebilirdik
- bring+Ip bil+tense+1pl bring+Abil+tense+1pl
- ‘we could bring’

(B) In Turkish “-Uver” suffix is used to provide rapidity in meaning. In Turkmen the auxiliary “ber” is connected to the word with the suffix “-Ip”

- çalıp berdi çalıverdi
- play+Ip give+tense+3sg play+Uver+tense+3sg
- ‘he played up’

(C) In Turkish “dur-” is used as continuous auxiliary. In Turkmen “yör-” auxiliary is used and connected with the gerundive “-Ip” suffix.

- gezip yören gezip duran
- pace+Ip stand+An pace+Ip stand+An
- ‘(the one who is) pacing around’

- bakıp yördi bakıp duruyordu
- look+Ip stand+tense look+Ip stand+tense+tense
- ‘he was continuously looking at’

(D) The main verb “başla-” in Turkish is an auxiliary verb in Turkmen. In Turkish the verb connected to a verb with dative suffix “-A”. However in Turkmen the auxiliary verb connected to a verb with the gerundive “-Ip”

- yazıp     başladı     yazmaya     başladı
- write     start+tense     write+MA+dat     start+tense
- ‘he started to write’

# CHAPTER 5

# DESIGN AND IMPLEMENTATION

Briefly the system performs the steps showed in Figure 5. Following sections explain each step detailed. System is implemented in Java. Each module seen in the figure correspond several Java classes or libraries.

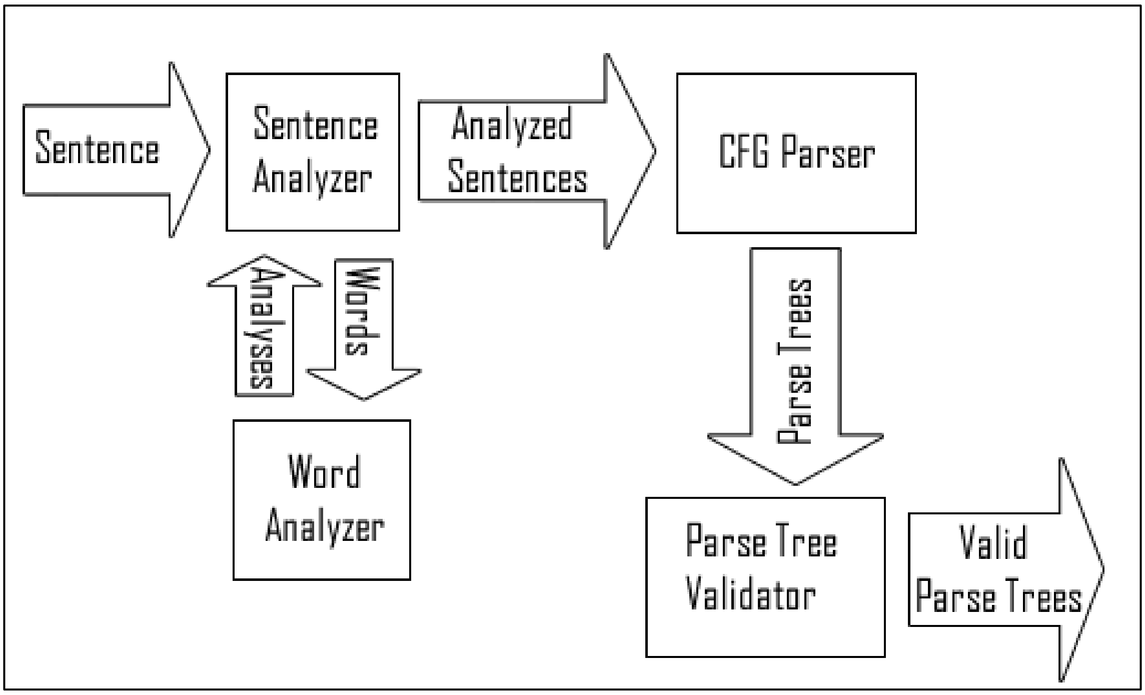

**Figure 5:** System flow chart

## 5.1 MORPHOLOGIC ANALYSIS

Phrases or sentences are split into words by a simple word tokenizer. Words are tokenized by white spaces. After this step each word is analyzed into their morphemes by morphologic analyzer module. After this process, sentence becomes a sequence of morphemes and each morpheme is tagged with suitable part-of-speech. Following example shows this process:

- ben eve gittim
- pron noun dat verb tense 1sg
- ben/pron ev/noun e/dat git/verb ti/tense m/1sg

System can use several analyzers together in given order. Presently only two analyzers are being used. Zemberek analyzer is used for Turkish. Manual analyzer is used for both Turkish and Turkmen. Although Dilmaç can analyze several Turkic languages including Turkish and Turkmen, it isn't designed in an API (application programming interface) manner. Because of its difficulty to be integrated with our system, we couldn't benefit from Dilmaç in this study.

### 5.1.1 Manual Analyzer

Manual analyzer lets us using handmade analyses for particular words. The words which cannot be analyzed by Zemberek are manually analyzed in a file. Manual analyzer has the most priority in both languages. This means if there is an analysis made by human for some word; parser uses that analysis and ignores the other analyzers.

On the other hand as mentioned above there is no suitable analyzer that we could use in this study. That's why all the words in tested Turkmen sentences were analyzed manually. Following example shows manual analyses for the Turkish word "benimki" and Turkmen word "olarıñ".

| | | |
|---|---|---|
| • benimki | ben/pron+im/gen+ki/rel | 'mine' |
| • olarıñ | olar/pron+ıñ/gen | 'their' |

### 5.1.2 Zemberek

Besides it is an open source project, Zemberek (Akın & Akın, 2007) is the best free and publicly available morphologic analyzer for Turkish yet. It is also used in the open source word processor OpenOffice as Turkish spell checker. Following example shows the analyses for the word "izin" by Zemberek:

Solution 1

- root:izin ‘permission’
- pos:ISIM ‘noun’
- suffixes:ISIM_KOK ‘noun’
- ‘permission’

Solution 2

- root:iz ‘trace’
- pos:ISIM ‘noun’
- suffixes:ISIM_KOK, ISIM_TAMLAMA_IN ‘noun, gen’
- ‘of the trace’

Solution 3

- root:iz ‘trace’
- pos:ISIM ‘noun’
- suffixes:ISIM_KOK, ISIM_SAHIPLIK_SEN_IN ‘noun, p2s’
- ‘your trace’

### 5.1.3 Sentence Analyzer

As seen in the previous example one word may have multiple morphologic analyses. In this case all possible combinations are produced as a sentence. If a sentence has three words having the 1, 3, 2 solutions, then 1 x 3 x 2 = 6 different sentences will be produced. Following example shows this all possible solutions for the sentence “ben izin verdim”:

1: ben/pron izin/noun ver/verb di/tense m/1sg
2: ben/pron iz/noun in/gen ver/verb di/tense m/1sg
3: ben/pron iz/noun in/p2s ver/verb di/tense m/1sg
4: ben/noun izin/noun ver/verb di/tense m/1sg
5: ben/noun iz/noun in/gen ver/verb di/tense m/1sg
6: ben/noun iz/noun in/p2s ver/verb di/tense m/1sg

In example above the first combination is the correct one. Presently our system doesn’t make a selection among the combinations. All the sentences produced are being parsed. Nevertheless most of them are eliminated according during the parse process.

## 5.2 AN EARLEY PARSER PEP

PEP (stands for *Pep is an Earley Parser*) is a Java implementation of the Earley's chart-parsing algorithm. It provides a command line interface but it is mostly designed as a library. PEP can parse any left-recursive CFG grammar efficiently. The accepted file format for grammars is XML. The CFG rule “S -> NP VP” is implemented as follows:

```
<rule category="S">
 <category name="NP"/>
 <category name="VP"/>
</rule>
```

We integrated PEP into our system and implemented Turkish and Turkmen CFG grammars in separated XML files.

## 5.3 CONTEXT-FREE GRAMMAR

In this section, main parts of our grammar are presented. All the symbols starting with a capital letter are terminal and others are non-terminal.

### 5.3.1 General Sentence Rules

A sentence (S) may have multiple sub-sentences. Sub-sentences can be connected by commas (cm) or conjunctions (conj).

S ⟶ SS

S ⟶ cm SS

S ⟶ conj SS

A sub-sentence is the phrase which has one and only one predicate. Other constituents are subject, definite object, indefinite object, complement, adverb and the excluded constituent. These constituents may or may not be in a sentence.

SS ⟶ Pre
SS ⟶ IObj Pre
SS ⟶ Sub SS
SS ⟶ SS Sub
SS ⟶ DObj SS
SS ⟶ SS DObj
SS ⟶ SS Comp
SS ⟶ Comp SS
SS ⟶ SS Adv
SS ⟶ Adv SS
SS ⟶ SS Excl
SS ⟶ Excl SS

According to these rules a sentence must have one and only one predicate. This predicate may have an indefinite object on the left hand side.

### 5.3.2 Noun Complement

The main CFG rules for NCs are as follows:

NC ⟶ NP NP Poss
NC ⟶ NP gen NP Poss
Poss ⟶ p1s | p2s | p3s | p1p | p2p | p3p

Following example and figure show the parse of the phrase “köy yolu” in bracketed notation and in tree notation.

[NC[NP[noun[köy]]][NP[noun[yol]]][Poss[p3s[u]]]]

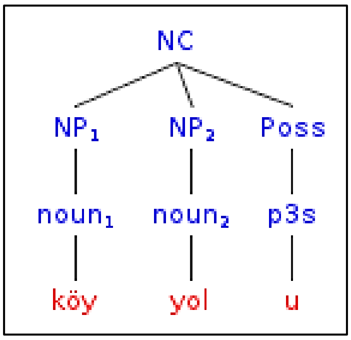

**Figure 6:** Parse tree for NC "köy yolu"

Here is another example for the definite noun complement “ovanın yeşili”:

[NC[NP[noun[ova]]][gen[nın]][NP[adj[yeşil]]][Poss[p3s[i]]]]

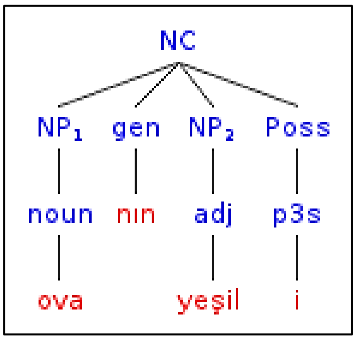

**Figure 7:** Parse tree for NC "ovanın yeşili"

### 5.3.2 Adjective Complement

Main CFG rules for JCs are as follows:

JC ⟶ JP NP

JC ⟶ JP cm JC

JC ⟶ JC cm NP

Here are some examples and figures for adjective complement.

Phrase: ‘acı gerçek’ ‘the sad truth’

Bracketed: [JC[JP[adj[acı]]][NP[noun[gerçek]]]]

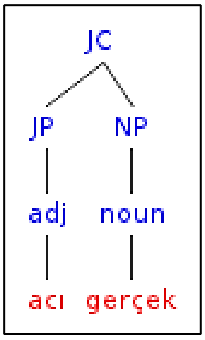

**Figure 8:** Parse tree for JC "acı gerçek"

Phrase: “kırmızılı çocuklar”

Bracketed: [JC[JP[NP[adj[kırmızı]]][+lU[lı]]][NP[noun[çocuk]][plu[lar]]]]

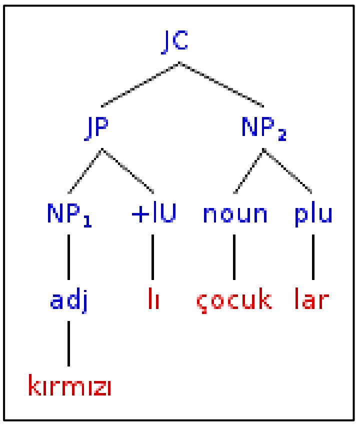

**Figure 9:** Parse Tree for JC "kırmızılı çocuklar"

### 5.3.3 Verbal Adjective Group

The only mandatory constituent of the group is verbal adjective. Thus it is possible to construct a VJG with a single verbal adjective.

VJG ⟶ VJ

VJG ⟶ IObj VJ

Any constituent or constituents can complement the verbal adjective. This means a VJG can consist of a verbal adjective and all possible combinations of other four complements.

VJG ⟶ DObj VJG

VJG ⟶ VJG DObj

VJG ⟶ Sub VJG

VJG ⟶ VJG Sub

…

Following figure shows the parse tree of the verbal adjective phrase “başını öne eğen”:

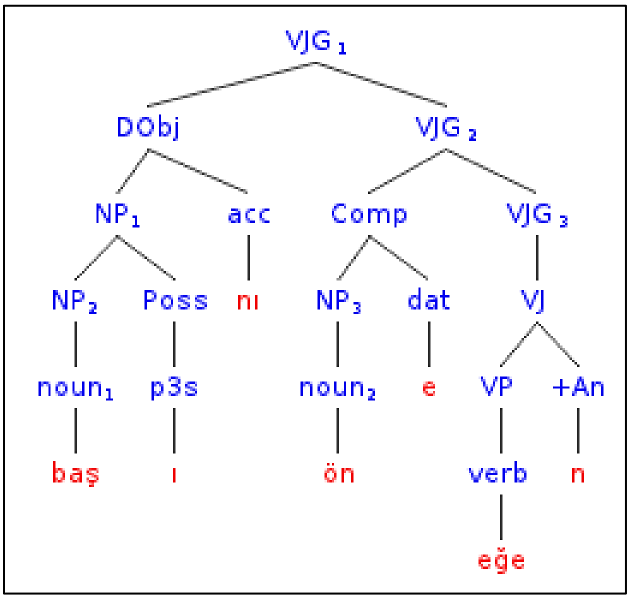

**Figure 10:** Parse tree for verbal adjective group "başını öne eğen"

### 5.3.4 Verbal Noun Group

The mandatory constituent of the group is verbal noun. Thus it is possible to construct a VNG with a single verbal noun.

VNG ⟶ VN

VNG ⟶ IObj VN

Any constituent or constituents can complement the verbal noun. This means a VNG can consist of a verbal noun and all possible combinations of subject, definite object, adverb and complement.

VNG⟶ DObj VNG

VNG⟶ VNG DObj

VNG⟶ Sub VNG

VNG⟶ VNG Sub

…

Following figure shows the parse tree of the verbal adjective phrase “onu eve getirmek”:

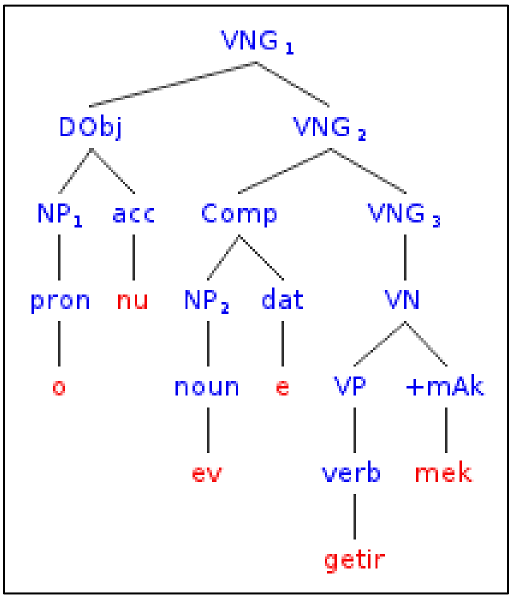

**Figure 11:** Parse tree for verbal noun group “onu eve getirmek”

### 5.3.5 Verbal Adverb Group

Similar to VJG and VNG the mandatory constituent of the group is verbal adverb. Thus it is possible to construct a VAG with a single verbal noun.

VAG ⟶VA

VAG ⟶IObj VA

Any constituent or constituents can complement the verbal adjective. This means a VAG can consist of a verbal adverb and all possible combinations of other four complements.

VAG⟶ DObj VAG

VAG⟶ VAG DObj

VAG→ Sub VAG

VAG→ VAG Sub

…

Following figure shows the parse tree of the verbal adjective phrase “hayalini hatırımdan silince”:

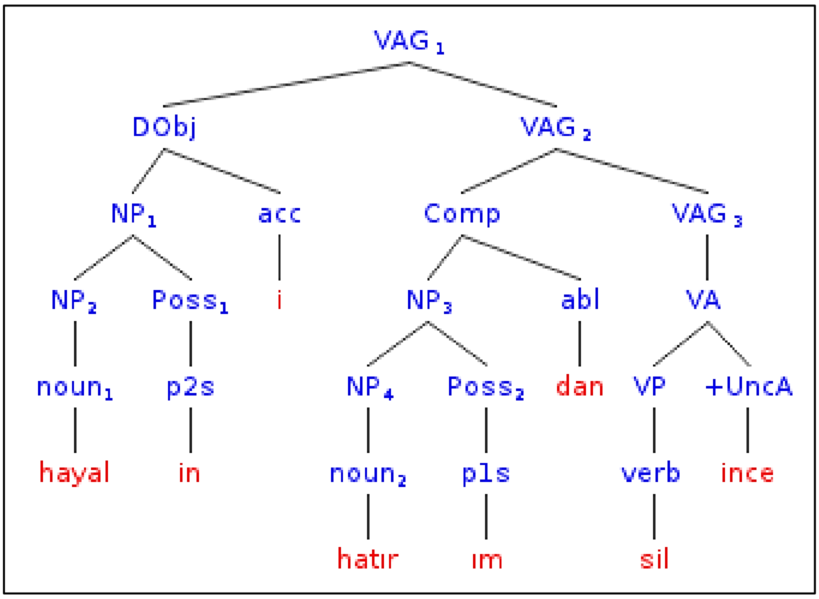

**Figure 12:** Parse tree for verbal adverb group "hayalini hatırımdan silince"

### 5.3.6 Reduplication Group

Reduplications are mostly idiomatic phrases. The only way to determine reduplication is using a lexicon. However in Turkish both base word and the reduplicant take inflectional suffixes and can be in many forms. This makes analyzing reduplication harder than analyzing a word. Analyzers used in this study only works on single words. That’s why presently reduplication groups taken inflectional suffixes are not handled. CFG rules for reduplications group is as following:

RG → base redup

Following figure shows the parse tree of the reduplication phrase “kılık kıyafet”:

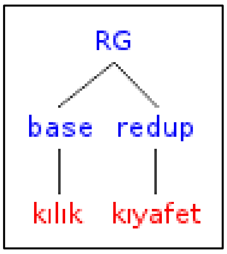

**Figure 13:** Parse tree for reduplication group "kılık kıyafet"

### 5.3.7 Postposition Group

The CFG rules for PPGs are as follows:

PG → NP posp
PG → NP dat posp
PG → NP gen posp
PG → NP abl posp

Following figure shows the parse tree of the postposition phrase “yaşadığım gibi”:

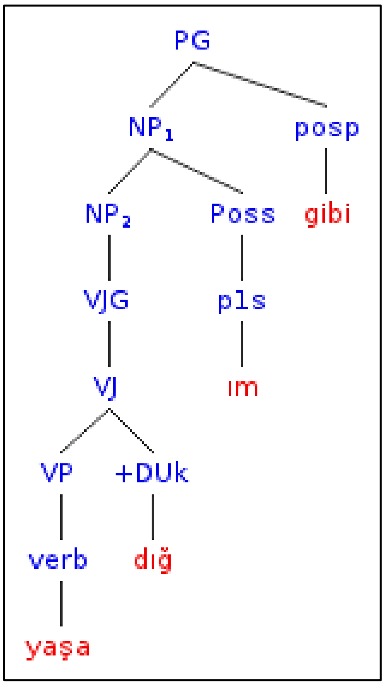

**Figure 14:** Parse tree for the PG "yaşadığım gibi"

### 5.3.8 Conjunction Group

The CFG rules for PPGs are as follows:

CG ⟶ NP conj NP

CG ⟶ NP cm CG

Following rules handle conjunction postpositions like "ne… ne…", "hem… hem…", "…da …da" etc.

CG ⟶ NP 2conj NP 2conj

CG ⟶ conj2 NP conj2 NP

The instrumental case –ylA is actually the suffix form of the conjunction "ile". The following rule handles this situation.

CG ⟶ NP ins NP

Following figure shows the parse tree of the conjunction group "ben, babam ve oğlum":

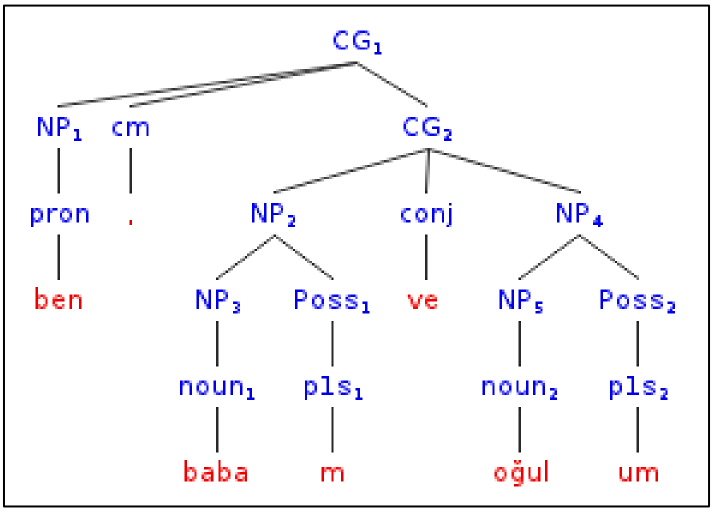

**Figure 15:** Parse tree for the CG "ben, babam ve oğlum"

### 5.3.9 Title Group

In Turkish, title can precede or follow the proper name. The CFG rules for title group are as follows:

TG ⟶ prop ttl

TG ⟶ ttl prop

As explained in 4.1.9, the proper name of the group can also be a word group called the personal name group (PNG).

TG ⟶ PNG ttl

TG ⟶ ttl PNG

Following figure shows the parse tree of the conjunction group “Halide Edip Hanım”:

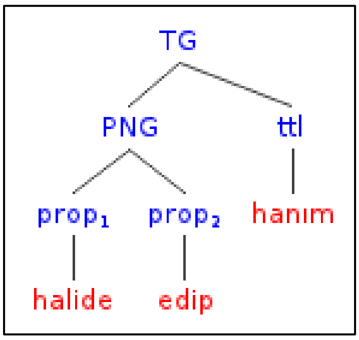

**Figure 16:** Parse tree for the CG "Halide Edip Hanım"

### 5.3.10 Person Name Group

A PNG consist of two or more proper names that constitute a person name.

PNG ⟶ prop prop

PNG ⟶ PNG prop

Following figure shows the parse tree of the conjunction group “Yahya Kemal Beyatlı”:

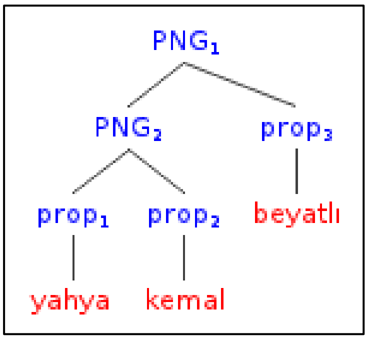

**Figure 17:** Parse tree for the PNG "Yahya Kemal Beyatlı"

### 5.3.11 Interjection Group

The CFG rules for interjection group are as follows:

IG → intj NP

In the group there can be more than one noun phrase separated with comma. This situation is handled by the following rule:

IG → IG cm NP

Following figure shows the parse tree of the Interjection group "Ey tatlı ve ulvi gece":

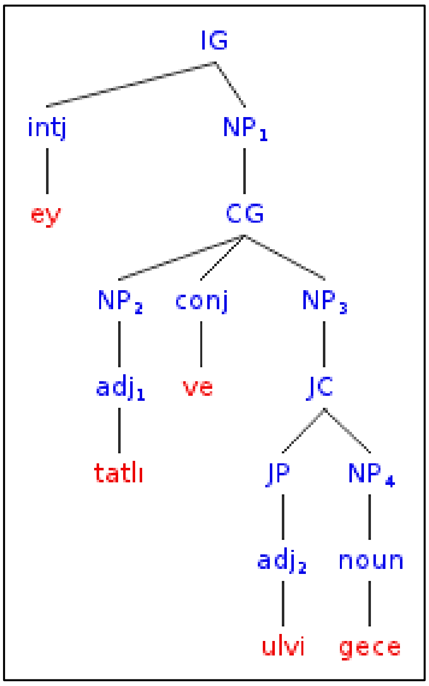

**Figure 18:** Parse tree for the IG "Ey tatlı ve ulvi gece"

### 5.3.12 Number Group

The CFG rules for number group are as follows:

NG ⟶ num num

NG ⟶ NG num

Following figure shows the parse tree of the number group “bin dört yüz elli üç”:

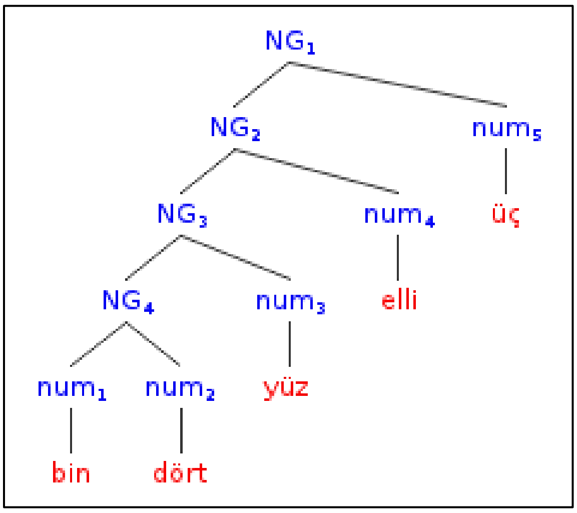

**Figure 19:** Parse tree for the NG "bin dört yüz elli üç"

### 5.3.13 Compound Verb

Most of the compound verbs are idiomatic expressions that their detection requires morphologic analysis. That’s why presently our system only handles compound verbs with helper verbs. CFG rules for compound verb group are as follows:

CVG → NP XVP
XVP → auxv
XVP → auxv psv
XVP → XVP neg
XVP → XVP tense
XVP → XVP Prs
XVP → XVP aux
XVP → XVP mood

Following figure shows the parse tree of the compound verb group “kahvaltı etmemiştim”:

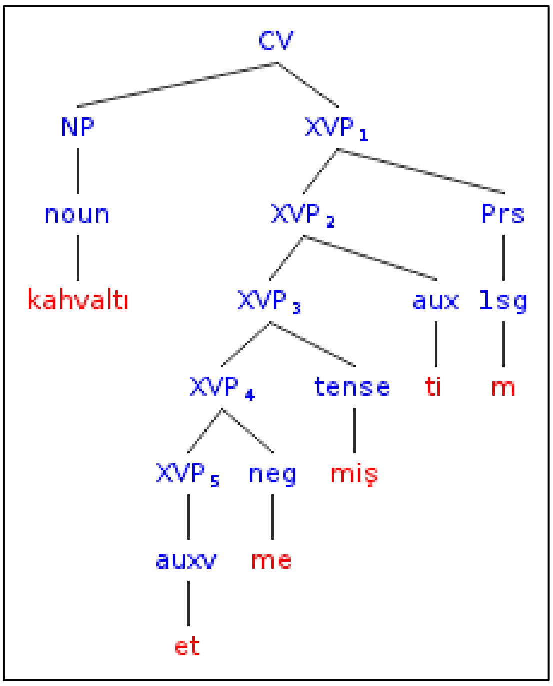

**Figure 20:** Parse tree for the CV "kahvaltı etmemiştim"

### 5.3.14 CFG Differences For Turkmen

As stated in section 4.2 Turkmen syntax is slightly different than Turkish in some cases. Nevertheless these differences are mostly concern of morphologic analysis and do not cause any modification of context-free grammar.

## 5.4 PARSE TREE VALIDATOR

As stated in section 3.1 context-free grammar is not very suitable with languages having rather complex morphology. Instead of handling situations like subject-predicate agreement or free order of constituents with many CFG rules, we designed our grammar as comprehensive as possible. This reduced the number of rules from thousands to a few hundred. Besides correct phrases, our grammar accepts many incorrect phrases too. That’s why we added an extra module that validates parse trees according to some external rules.

Parse tree validator module reads validation rules from an XML file and runs them on each tree. Following entry is from validation rules file. This rule handles the ‘if the predicate is non-transitive, there cannot be any objects in the sentence’ situation.

```
<rule scope="SS" ignore="VJG, VNG, VAG">
        <description>Yüklem geçişiz fiil ise nesne yoktur</description>
        <status type="cond" cat="Pre"        op="hasNonTransitiveVerb"/>
        <status type="rest" cat="IObj"       op="notExists"/>
        <status type="rest" cat="DObj"       op="notExists"/>
</rule>
```

Following XML node is one of the subject-predicate agreement rules:

```
<rule scope="SS" ignore="VJG, VNG, VAG">
        <description> 1. tekil şahıs eki özne-yüklem uyumu</description>
        <status type="cond" cat="Sub" op="equals"
                        str="[Sub[NP[pron[ben]]]]"/>
        <status type="rest" cat="Pre" op="includes"  str="[Prs[1sg["/>
</rule>
```

# CHAPTER 6

# RESULTS AND CONCLUSION

## 6.1 CONCLUSION

In this study we presented a generic syntactic parser for Turkic languages. A context-free grammar that covers most of the Turkish and Turkmen grammar is prepared and tested. The characteristics of Turkish syntax and the ability of CFG to handle these characteristics are discussed. Our study showed that a generic syntactic parser for Turkic languages can be implemented effectively.

We tried to use a simple but effective way to parse sentences. The simplicity of our approach helps us to clearly see the limitations and understand the basics of parsing Turkic languages. We believe that our study offers a good point of start for similar researches.

In contradiction to previous studies, this study is based on a more general grammar theory which is being used today by most of the Turkish linguists.

## 6.2 LIMITATIONS AND FUTURE WORK

There are several limitations we have faced during the development of this parser. In this section we explain each one and offer some solutions as future work.

In an agglutinative language a word usually have more than one morphologic analysis. As mentioned in section 5.1.3, this causes to produce all possible combinations of these analyses as a sentence. Hence if a sentence has 5 words having numbers analyses as 2, 3, 1, 1, 4, this makes 2 x 3 x 4 = 24 sentences. A part-of speech tagger which uses statistical data can be an effective solution to this problem. As a future work we plan to design and implement such a POS tagger. We also plan to present our work as an online parser and collecting training data from the users.

In Turkish and Turkmen there are some words groups such as “geliyor musun”, “çoluğu çocuğu”, “ben de” or “gönül al-” that must be analyzed morphologically as a single word. Nonexistence of such parser presently makes it impossible to parse similar phrases effectively.  We plan to implement a more accurate morphologic analyzer which is capable of analyzing multiple tokens as a single word.

As mentioned in section 5.1 there is no suitable morphologic analyzer for Turkmen that we could use in our system. That’s why system is tested very limited manually analyzed data. Dilmaç’s adaptation to our system is another future work for us. Similarly adaptation of other Turkic languages is waiting to be done.

In Turkish (and most probably other Turkic languages) rarely word groups cross each other. A word that doesn’t belong to a word group can be in between the words of it. Situations like these cannot be handled by context-free grammar. We plan to implement an extra module for handling these exceptional situations.

Finally we plan to continue this study as an open source project. Users will be able to test our system online and access all the source code and other documentation. We believe that this will help us improving the parser faster and more effective.

# REFERENCES

A. A. Akın and M. D. Akın, “Zemberek , an open source NLP framework for Turkic Languages,” *Structure*, 2007.

Atalay N. and K. Oflazer, “The annotation process in the Turkish treebank,” Proc. of the 4th Intern. Workshop on, 2003.

Ayan E., “Kazak Türkçesi ve Türkiye Türkçesinin Cümle Öğeleri Yönünden Karşılaştırılması,” turkishstudies.net, vol. 2/3, 2007, pp. 73-99.

Biray H., “Türkmen Şivesinin Sentaks Bakımından Türkiye Türkçesiyle Karşılaştırılması,” Yüksek Lisans Tezi, Gazi Üniversitesi, Ankara, Türkiye, 1985.

Bozşahin C. and E. Göçmen, “Türkçedeki Dilbilgisel Yapilarin Bilgi Sayisal Dilbilim Modellerinin Tasarimi,” 1996.

Clark L.V., Turkmen reference grammar, Otto Harrassowitz Verlag, 1998.

Eryiğit G. and K. Oflazer, “Statistical dependency parsing of Turkish,” Proc. of EACL-2006, Citeseer, 2006, p. 89–96.

Eryiğit G., E. Adalı, and K. Oflazer, “Türkçe’nin olasılık tabanlı bağlılık ayrıştırması,” itudergi.itu.edu.tr, 2009, pp. 53-64.

Eryiğit G., J. Nivre, and K. Oflazer, “Dependency Parsing of Turkish,” Computational Linguistics, vol. 34, Sep. 2008, pp. 357-389.

Eryiğit G., K. Oflazer, and E. Adalı, “Türkçe cümlelerin kural tabanlı bağlılık analizi,” Elektronik, 2006, pp. 17-24.

Göçmen E., O.T. Şehitoğlu, and C. Bozşahin, “An outline of turkish syntax,” Ms. Department of Computer Engineering, 1995, pp. 1-36.

Güngördü Z. and K. Oflazer, “Parsing Turkish using the lexical functional grammar formalism,” Machine Translation, vol. 10, 1995, pp. 293-319.

Güngördü Z., “A lexical-functional grammar for turkish,” Master’s thesis, Bilkent University, Ankara Turkey, 1993.

Hoffman B., “A CCG approach to free word order languages,” Proceedings of the 30th annual meeting on Association for Computational Linguistics -, 1992, p. 300.Dd

Hoffman B., “Generating context-appropriate word orders in Turkish,” Proceedings of the Seventh International Workshop on Natural Language Generation - INLG ’94, 1994, pp. 117-126.

Hoffman B., “Integrating free word order syntax and information structure,” Proceedings of the seventh conference on European chapter of the Association for Computational Linguistics, Morgan Kaufmann Publishers Inc., 1995, p. 245–252.

İstek Ö., “A Link Grammar for Turkish,” Master’s thesis, Bilkent University, Ankara Turkey, 2006.

Jurafsky D. and Martin H. James. 2009. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition and. 2nd ed. Pearson Education.

Kara M., Türkmen Türkçesi Grameri, ANKARA: Gazi Kitabevi, 2005.

Karahan L., Türkçede Söz Dizimi, Ankara: Akçağ, 2008.

Karaörs M.M., Türk Lehçelerinde Karşılaştırmalı Şekil ve Cümle Bilgisi, ANKARA: Akçağ, 2005.

Kayasandık A., “Türkmen Türkçesinde Cümle Yapısı,” Yüksek Lisans Tezi, Erciyes Üniversitesi, Kayseri, Türkiye, 1998.

Oflazer K., “Dependency Parsing with an Extended Finite-State Approach,” Computational Linguistics, vol. 29, Dec. 2003, pp. 515-544.

Oflazer K., B. Say, D.Z. Hakkani-Tür, and G. Tür, “Building A Turkish Treebank,” Treebanks: Building and, 2003, pp. 1-17.

Özçam Ç., “Türkiye Türkçesi ile ilgili Gramer Çalışmaları,” Türk Dünyası Araştırmaları, vol. 110, 1997, pp. 121-163.

Shylov M., “Turkish and Turkmen Morphological Analyzer and Machine Translation Program,” Master’s thesis, Fatih University, İstanbul Turkey, 2008.

Şehitoğlu O., “A sign-based phrase structure grammar for Turkish,” Arxiv preprint cmp-lg/9608016, 1996.

Temizsoy M. and I. Cicekli, “An Ontology-Based Approach to Parsing Turkish Sentences,” Machine Translation and the Information Soup, 1998.

Yaman E., Türkiye Türkçesi ve Özbek Türkçesinin Söz Dizimi Bakımından Karşılaştırılması, ANKARA: TDK Yayınları, 2000.

# APPENDIX A

In this section some output examples of system for Turkish and Turkmen sentences are given.

Sentence: ben eve gittim, geldim

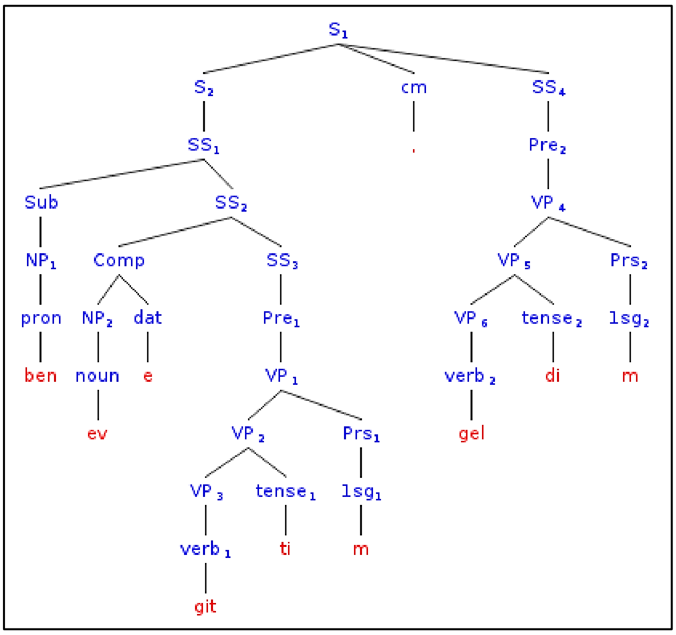

**Figure 21:** Parse tree for sentence "ben eve gittim, geldim"

Sentence: Zeynep bu romanı okudu

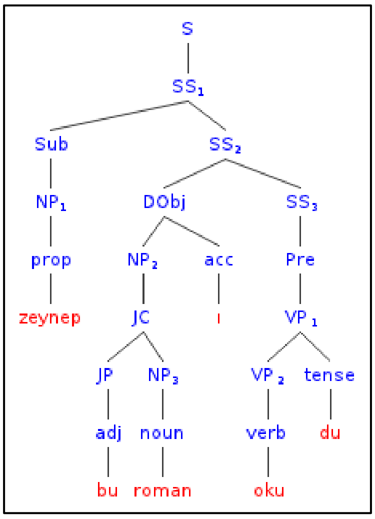

**Figure 22:** Parse tree for sentence "Zeynep bu romanı okudu"

Sentence: Sahipsiz olan memleketin batması haktır

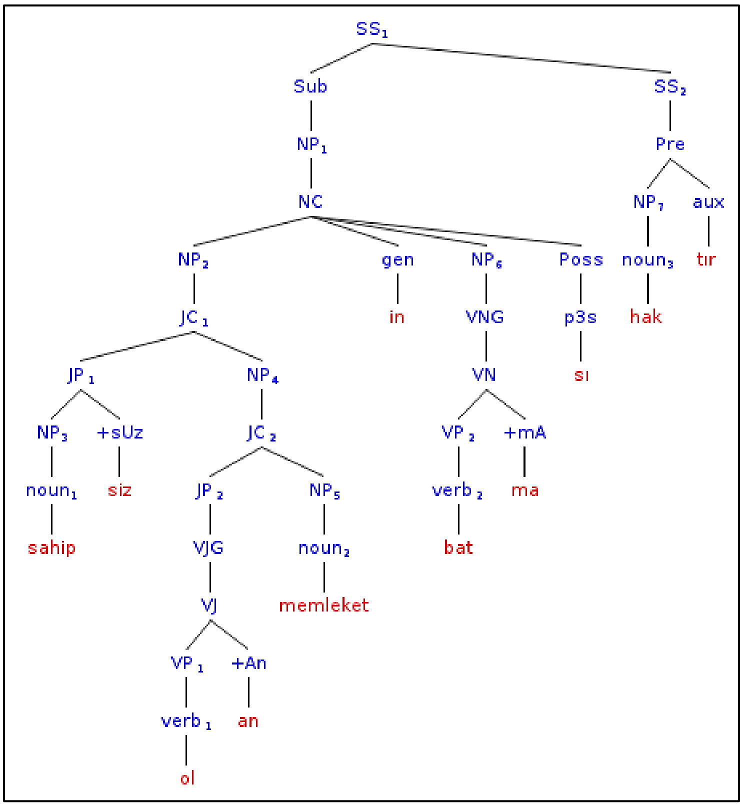

**Figure 23:** Parse tree for sentence "Sahipsiz olan memleketin batması haktır"

Sentence: Muayene odasının kapısı açılır

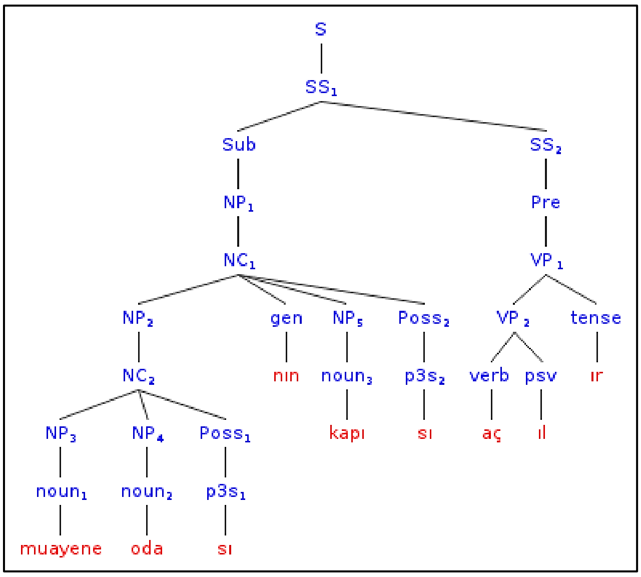

**Figure 24:** Parse tree for sentence " Muayene odasının kapısı açılır"

Sentence: yağtı dünyäämize yañadaan geldim (aydınlık dünyamıza yeniden geldim)

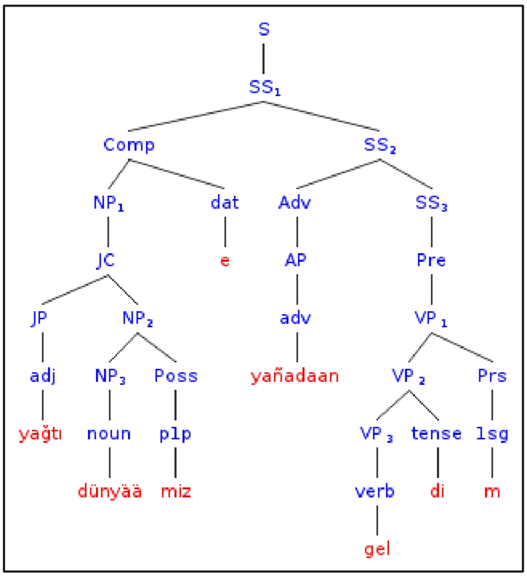

**Figure 25:** Parse tree for sentence "yağtı dünyäämize yañadaan geldim"

Sentence: bu obada yaşulı yok (bu köyde yaşlı yok)

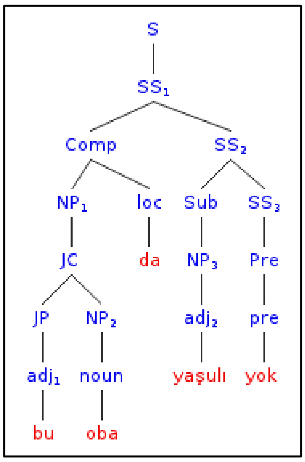

**Figure 26:** Parse tree for sentence " bu obada yaşulı yok "

Sentence: bilbil ovazını övyää şahırlar (bülbül sesini övüyor şairler)

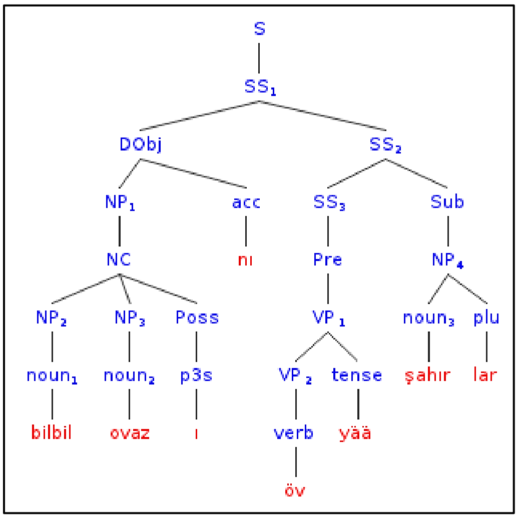

**Figure 27:** Parse tree for sentence “bilbil ovazını övyää şahırlar”