Parsing Turkish using the Lexical Functional Grammar Formalism

Zelal Güngördü[1]
Centre for Cognitive Science
University of Edinburgh
2 Buccleuch Place
Edinburgh, EH8 9LW Scotland, U.K.
gungordu@cogsci.ed.ac.uk

Kemal Oflazer
Department of Computer Engineering
and Information Science
Bilkent University
Ankara, 06533, TURKEY
ko@cs.bilkent.edu.tr

Abstract This paper describes our work on parsing Turkish using the *lexical-functional grammar* formalism. This work represents the first effort for parsing Turkish. Our implementation is based on Tomita's parser developed at Carnegie-Mellon University Center for Machine Translation. The grammar covers a substantial subset of Turkish including structurally simple and complex sentences, and deals with a reasonable amount of word order freeness. The complex agglutinative morphology of Turkish lexical structures is handled using a separate two-level morphological analyzer. After a discussion of the key relevant issues regarding Turkish grammar, we discuss aspects of our system and present results from our implementation. Our initial results suggest that our system can parse about 82% of the sentences directly and almost all the remaining with very minor pre-editing.

1 Introduction

As part of our ongoing work on the development of computational resources for natural language processing in Turkish, we have undertaken the development of a parser for Turkish using the lexical-functional grammar formalism for use in a number of applications. Although there have been a number of studies of Turkish syntax from a linguistic perspective (e.g., [8]), this work represents the first approach to the computational analysis of Turkish. Our implementation is based on Tomita's parser developed at Carnegie-Mellon University Center for Machine Translation [15, 16]. Our grammar covers a substantial subset of Turkish including structurally simple and complex sentences, and deals with a reasonable amount of word order freeness. This system is expected to be a part of the machine translation system that we are planning to build as a part of a large scale natural language processing project for Turkish, supported by NATO [12].

Turkish has two characteristics that have to be taken into account: agglutinative morphology, and rather free word order with explicit case marking. We handle the complex agglutinative morphology of the Turkish lexical structures using a separate morphological processor based on the two-level paradigm [1, 11] that we have integrated with the lexical-functional grammar parser. Word order freeness, on the other hand, is dealt with by relaxing the order of phrases in the phrase structure parts of lexical-functional grammar rules by means of generalized phrases.

[1] This work was done as a part of the first author's M.Sc. degree work at the Department of Computer Engineering and Information Science, Bilkent University, Ankara, 06533 Turkey.

2 Lexical-Functional Grammar

Lexical-functional grammar (LFG) is a linguistic theory which fits nicely into computational approaches that use *unification* [14]. A lexical-functional grammar assigns two levels of syntactic description to every sentence of a language: a *constituent structure* and a *functional structure*. Constituent structures (c-structures) characterize the phrase structure configurations as a conventional phrase structure tree, while surface grammatical functions such as *subject, object*, and *adjuncts* are represented in functional structures (f-structures). Because of space limitations we will not go into the details of the theory. One can refer to Kaplan and Bresnan [5] for a thorough discussion of the LFG formalism.

3 Turkish Grammar

In this section, we would like to highlight two of the relevant key issues in Turkish grammar, namely highly inflected agglutinative morphology and free word order, and give a description of the structural classification of Turkish sentences that we deal with.

3.1 Morphology

Turkish is an agglutinative language with word structures formed by productive affixations of derivational and inflectional suffixes to root words [11]. This extensive use of suffixes causes morphological parsing of words to be rather complicated, and results in ambiguous lexical interpretations in many cases. For example:

(1) **çocukları**
çocuk+lar+ı

a.	child+PLU+3SG–POSS	his children
b.	child+3PL–POSS	their child
c.	child+PLU+ACC	children (accusative)
	çocuk+ları	
d.	child+(PLU)+3PL–POSS	their children

Such ambiguity can sometimes be resolved at phrase and sentence levels by the help of agreement requirements though this is not always possible:

(2a)	**O+nlar+ın**	**çocuk+ları**	**gel+di+ler.**	Their children came.
	it+PLU+GEN (they)	child+PLU+3PL-POSS	come+PAST+3PL	

(2b)	**Çocukları**		**geldiler.**	
	Çocuk+lar+ı		**gel+di+ler.**	
	child+PLU+3SG–POSS		come+PAST+3PL	His children came.
	Çocuk+ları		**gel+di+ler.**	
	child+(PLU)+3PL–POSS		come+PAST+3PL	Their children came.

For example, in (2a) only the interpretation (1d) (i.e., *their children*) is possible because:

- the agreement requirement between the modifier and the modified parts in a possessive com-

pound noun eliminates (1a),[2]

- the facts that the verb *gel- (come)* does not subcategorize for an accusative marked direct object, and that in Turkish the subject of a finite sentence must be nominative (i.e., unmarked) rule out (1c),
- the agreement requirement between the subject and the verb of a sentence eliminates (1b).[3]

In (2b), on the other hand, both (1a) (i.e., *his children*) and (1d) (i.e., *their children*) are possible since the modifier of the possessive compound noun is a covert one: it may be either *onun* (*his*) or *onların* (*their*). The other two interpretations are eliminated due to the same reasons as in the case of (2a).

3.2 Word Order

In terms of word order, Turkish can be characterized as an *subject–object–verb (SOV) language* in which constituents at some phrase levels can change order rather freely. This is due to the fact that morphology of Turkish enables morphological markings on the constituents to signal their grammatical roles without relying on their order. This, however, does not mean that word order is immaterial. Sentences with different word orders reflect different pragmatic conditions, in that topic, focus and background information conveyed by such sentences differ.[4] Besides, word order is fixed at some phrase levels such as postpositional phrases. There are even severe constraints at sentence level, some of which happen to be useful in eliminating potential ambiguities in the semantic interpretation of sentences.

One such constraint is related to the existence of case marking on direct objects. Direct objects in Turkish can be both accusative marked and unmarked (i.e., nominative). Case marking generally correlates with a specific reading of the object. The constraint is that nominative direct objects can only appear in the immediately preverbal position in a sentence, which determines that *mutluluk* is the subject and *huzur* is the direct object in (3):[5]

(3)	**Mutluluk**	**huzur**	**getir+ir.**	Happiness brings peace of mind.
	happiness	peace of mind	bring+PRES(+3SG)	*Peace of mind brings happiness.

Another constraint is that nonderived manner adverbs[6] always immediately precede the verb or, if it exists, the nominative direct object. Hence, *iyi* can only be interpreted as an adjective that modifies the accusative direct object *yemeği* in (4a), whereas in (4b), it is an adverb modifying the verb *pişirdin*. In (4c), on the other hand, it can either be an adjective modifying the nominative direct object *yemek*, or an adverb modifying the verb *pişirdin*:

[2]The agreement of the modifier must be the same as the possessive suffix of the modified with the exception that if the modifier is third person plural, the possessive suffix of the modified is either third person plural or third person singular.

[3]In a Turkish sentence, person features of the subject and the verb should be the same. This is true also for the number features with one exception: in the case of third person plural subjects, the verb may sometimes be marked with the third person singular suffix.

[4]See Erguvanlı [3] for a discussion of the function of word order in Turkish grammar.

[5]This example is taken from Erguvanlı [3].

[6]These adverbs are in fact qualitative adjectives, but can also be used as adverbs. Examples are *iyi 'good/well', hızlı 'fast', güzel 'beautiful/beautifully'.*

Table 1: Percentage of different word orders in Turkish.

Sentence Type	Children Speech	Adult Speech
SOV	46%	48%
OSV	7%	8%
SVO	17%	25%
OVS	20%	13%
VSO	10%	6%
VOS	0%	0%

(4a)	**İyi**	**yemeğ+i**	**pişir+di+n.**	You cooked the good meal.
	good	meal+ACC	cook+PAST+2SG	*You cooked the meal well.
(4b)	**Yemeğ+i**	**iyi**	**pişir+di+n.**	You cooked the meal well.
	meal+ACC	well	cook+PAST+2SG	
(4c)	**İyi**	**yemek**	**pişir+di+n.**	You cooked a/some good meal.
	good/well	meal	cook+PAST+2SG	You cooked well.

The flexibility of word order in general applies to the sentence level, resulting in different discourse conditions. The data in Table 1 from Erguvanlı [3], shows the percentages of different word orders in discourse. We will not go into details of the pragmatic conditions conveyed by different word orders, but will rather provide some examples for such conditions. (See Erguvanlı [3] for a thorough discussion of those conditions.)

For instance, a constituent that is to be emphasized is generally placed immediately before the verb. This affects the places of all the constituents in a sentence except that of the verb:[7]

(5a)	**Ben**	**çocuğ+a**	**kitab+ı**	**ver+di+m.**	I gave the book to the child.
	I	child+DAT	book+ACC	give+PAST+1SG	
(5b)	**Çocuğ+a**	**kitab+ı**	**ben**	**ver+di+m.**	I gave the book to the child.
	child+DAT	book+ACC	I	give+PAST+1SG	
(5c)	**Ben**	**kitab+ı**	**çocuğ+a**	**ver+di+m.**	I gave the book to the child.
	I	book+ACC	child+DAT	give+PAST+1SG	

(5a) is an example of the typical word order whereas in (5b) the subject, *ben*, is emphasized. In (5c), on the other hand, the indirect object, *çocuğa*, is emphasized.

In addition, the verb itself may move away from its typical place, i.e., the end of the sentence. Such sentences are called *inverted sentences* and are typically used in informal prose and discourse. The reason behind using an inverted sentence is sometimes to emphasize the verb:

(6)	**Gel+me**	**bura+ya!**	Don't come here!
	come+NEG(+IMP+2SG)	here+DAT	

[7]The underlined words in Turkish examples show the constituent that is emphasized and the ones in English translations show the word marked with stress phonetically.

3.3 Structural Classification of Sentences

- **Simple Sentences:** A simple sentence contains only one independent judgment. The sentences in (2), (3), (4), (5), and (6) are all examples of simple sentences.
- **Complex Sentences :** In Turkish, a sentence can be transformed into a construction with a *verbal noun*, *participle* or *gerund* by affixing certain suffixes to the verb of the sentence. Complex sentences are those that include such dependent (subordinate) clauses as their constituents, or as modifiers of their constituents. Dependent clauses may themselves contain other dependent clauses, resulting in embedded structures like (7):

(7)

Bura+da	**iç+il+ebil+ecek**	**su**	
here+LOC	drink+PASS+POT +FUT–PART	water	
bul+ama+yacağ+ım+ı	**zannet+mek**	**doğru**	**ol+maz+dı.**
find+NEG–POT+FUT–PART +1SG–POSS+ACC	think+INF	right	be+NEG–AOR +PAST(+3SG)

It wouldn't be right to think that I wouldn't be able to find drinkable water here.

The subject of (7) (*burada içilebilecek su bulamayacağımı zannetmek – to think that I wouldn't be able to find drinkable water here*) is a nominal dependent clause whose accusative object (*burada içilebilecek su bulamayacağımı – that I wouldn't be able to find drinkable water here*) is an adjectival dependent clause which acts as a nominal one. The nominative object of this accusative object (*içilebilecek su – drinkable water*) is a compound noun whose modifier part is another adjectival dependent clause (*içilebilecek – drinkable*), and modified part is a noun (*su – water*).

It should be noted that there are other types of sentences in the classification according to structure, for which we will not provide any examples here because of space limitations. (See Şimşek [2], and Güngördü [4] for details.)

4 System Architecture and Implementation

We have implemented our parser in the grammar development environment of the Generalized LR Parser/Compiler developed at Carnegie Mellon University Center for Machine Translation. No attempt has been made to include morphological rules as the parser lets us incorporate our own morphological analyzer for which we use a full scale two-level specification of Turkish morphology based on a lexicon of about 24,000 root words[1, 11]. This lexicon is mainly used for morphological analysis and has limited additional syntactic and semantic information, and is augmented with an argument structure database.[8]

Figure 1 shows the architecture of our system. When a sentence is given as input to the program, the program first calls the morphological analyzer for each word in the sentence, and keeps the

[8]The morphological analyzer returns a list of *feature–value* pairs. For instance, for the word *evdekilerin* (of those (things) in the house/your things in the house) it returns:

1. `((*CAT* N)(*R* "ev")(*CASE* LOC)(*CONV* ADJ "ki")(*AGR* 3PL)(*CASE* GEN))`
2. `((*CAT* N)(*R* "ev")(*CASE* LOC)(*CONV* ADJ "ki")(*AGR* 3PL)(*POSS* 2SG))`

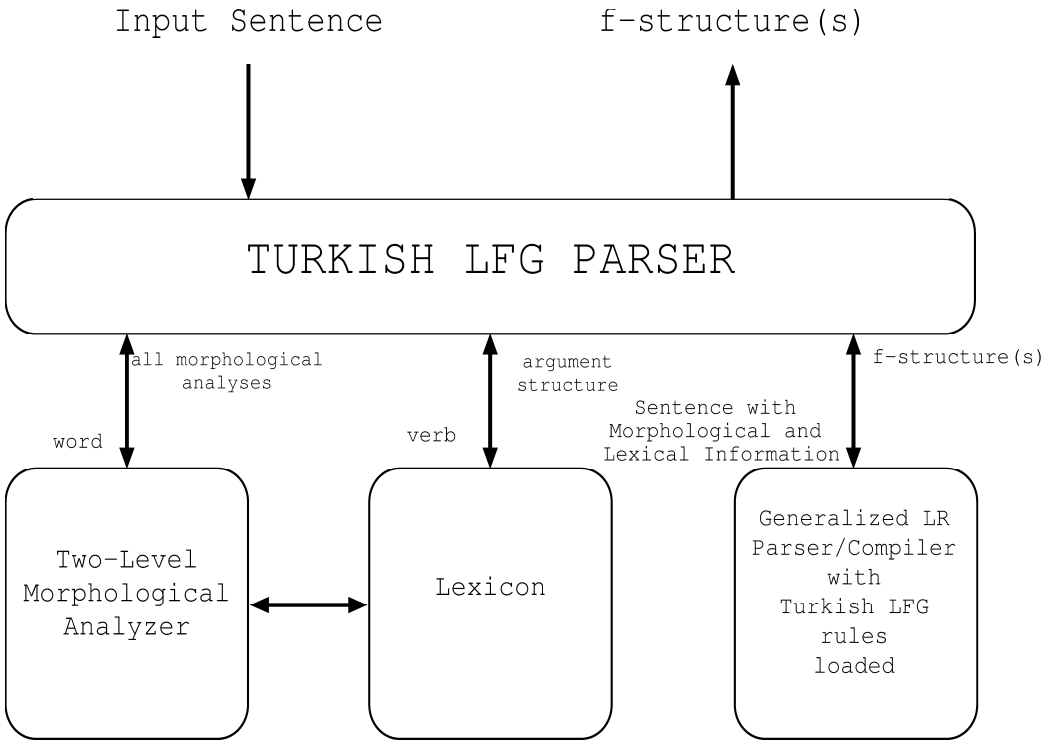

Figure 1: The system architecture.

results of these calls in a list to be used later by the parser.[9] If the morphological analyzer fails to return a structure for a word for any reason (e.g., the lexicon may lack the word or the word may be misspelled), the program returns with an error message. After the morphological analysis is completed, the parser is invoked to check whether the sentence is grammatical. The parser performs bottom-up parsing. During this analysis, whenever it consumes a new word from the sentence, it picks up the morphological structure of this word from the list. If the word is a finite or non-finite verb, the parser is also provided with the subcategorization frame of the word. At the end of the analysis, if the sentence is grammatical, its f-structure is output by the parser.

5 The Grammar

In this section, we present an overview of the LFG specification that we have developed for Turkish syntax. Our grammar includes rules for *sentences, dependent clauses, noun phrases, adjectival phrases, postpositional phrases, adverbial constructs, verb phrases*, and a number of *lexical look up rules.*[10] Table 2 presents the number of rules for each category in the grammar. There are also some intermediary rules, not shown here.

Recall that the typical order of constituents in a sentence may change due to a number of reasons. Since the order of phrases is fixed in the phrase structure component of an LFG rule, this rather

[9]Recall that there may be a number of morphologically ambiguous interpretations of a word. In such cases, the morphological analyzer returns all of the possible morphological structures in a list, and the parser takes care of the ambiguity regarding the grammar rules.

[10]Recall that no morphological rules have been included. The lexical look up rules are used just to call the morphological analyzer.

Table 2: The number of rules for each category in the grammar.

Category	Number of Rules
Noun phrases	17
Adjectival phrases	10
Postpositional phrases	24
Adverbial constructs	50
Verb phrases	21
Dependent clauses	14
Sentences	6
Lexical look up rules	11
TOTAL	153

free nature of word order at sentence level constitutes a major problem. In order to keep from using a number of redundant rules we have adopted the following strategy in our rules: We use the same place holder, <XP>, for all the syntactic categories in the phrase structure component of a sentence or a dependent clause rule, and check the categories of these phrases in the equations part of the rule. In Figure 2, we give a grammar rule for sentences with two constituents, with an informal description of the equation part.[11]

Recall also that a nominative direct object should be placed immediately before the verb, and that nonderived manner adverbs always immediately precede the verb or, if it exists, the nominative direct object (cf. Section 3.2). In our grammar, we treat such objects and adverbial adjuncts as part of the verb phrase. So, we do not check these constraints at the sentence or dependent clause level.

6 Performance Evaluation

In this section, we present some results about the performance of our system on test runs with four different texts on different topics. All of the texts are articles taken from magazines. We used the CMU Common Lisp system running in a Unix environment on SUN Sparcstations at Centre for Cognitive Science, University of Edinburgh.[12]

In all of the texts there were some sentences outside our scope. These were:

- sentences with finite sentences as their constituents or modifiers of their constituents,
- conditional sentences,
- finite sentences that were connected by conjunctions, and

[11] Note that *x0*, *x1*, and *x2* refer to the functional structures of the sentence, the first constituent and the second constituent in the phrase structure, respectively.

[12] We should, however, note that the times reported are exclusive of the time taken by the morphological analyzer, which, with a 24,000 word root lexicon, is rather slow and can process about 2 lexical forms per second. However, we have ported our morphological analyzer to the XEROX Twol system developed by Lauri Karttunen [6] and this system can process about 500 forms a second. We intend to integrate this to our system when it is completed and tested.

```
(<S> <==> (<XP> <XP>)
  1) if x1's category is VP then
        assign x1 to the functional structure of the verb of the sentence
     if x2's category is VP then
        assign x2 to the functional structure of the verb of the sentence


  2) for i = 1 to 2 do
     (use if, not else if, since there may be ambiguous parses)
       if xi has already been assigned to the functional structure of the verb then
          do nothing

       if xi's category is ADVP then
          add xi to the adverbial adjuncts of the sentence

       if xi's category is NP and xi's case is nominative then
          assign xi to the functional structure of the subject of the sentence

       if xi's category is NP then
          (coherence check)
          if the verb of the sentence can take an object with this case
             (consider also the voice of the verb)
             add xi to the objects of the verb

     (completeness check)
  3) check if the verb has taken all the objects that it has to take

  4) make sure that the verb has not taken more than one object with the same
     thematic role

  5) check if the subject and the verb agree in number and person:
       if the subject is defined (overt) then
         if the agreement feature of the subject is third person plural then
            the agreement feature of the verb may be either third person singular
                                                    or third person plural

         else
            the agreement features of the subject and the verb must be the same

       else if the subject is undefined (covert) then
         assign the agreement feature of the verb to that of the subject
```

Figure 2: An LFG rule for the sentence level given with an informal description of the equation part.

Table 3: Statistical information about the test runs.

Text	Number of Sentences	Sentences in Scope	Sent. ignored	Sent. after Pre-editing	Avg. Parses per Sentence	Avg. CPU Time per Sentence
1	43	30	0	55	4.28	12.26 sec.
2	51	41	2	62	5.02	8.92 sec.
3	56	48	1	64	4.87	10.28 sec.
4	80	70	0	97	3.25	7.46 sec.
Total	230	189(82%)	3	279	–	–

- sentences where an adverbial adjunct of the verb intervened in a compound noun, causing it to become a discontinuous constituent.[13]

We pre-edited the texts so that the sentences were in our scope (e.g., separated finite sentences connected by conjunctions and commas, and parsed them as independent sentences, and ignored the conditional sentences). Table 3 presents some statistical information about the test runs. The first, second and third columns show the document number, the total number of sentences and the number of sentences that we could parse without pre-editing, respectively. The other columns show the number of sentences that we totally ignored, the number of sentences in the pre-edited versions of the documents, average number of parses per sentence generated and average runtime for each of the sentences in the texts, respectively. It can be seen that our grammar can successfully deal with about 82% of the sentences that we have experimented with, with almost all the remaining sentences becoming parsable after a minor pre-editing. This indicates that our grammar coverage is reasonably satisfactory.

In the rest of this section, we will first discuss the impact of morphological disambiguation on the performance of our parser, and then provide some example outputs from our implementation.

6.1 Impact of Morphological Disambiguation on the Parser

In languages like Turkish with words that are morphologically ambiguous due to ambiguities in the part-of-speech of the root, or to different ways of interpreting the suffixes, using a tagger that relies on various sources of information (contextual constraints, usage statistics, lexical preferences and heuristics) to preprocess the input, can have a significant impact on parsing. We have tested the impact of morphological and lexical disambiguation on the performance of the parser by tagging our input using the tagger that we have developed in a different work [7, 13]. The input to the parser was disambiguated using the tool developed and the results were compared to the case when the parser had to consider all possible morphological ambiguities itself. For a set of 80 sentences considered, it can be seen that (Table 4), morphological disambiguation enables almost a factor of two reduction in the average number of parses generated and over a factor of two speed-up in time.[14]

[13] Again, this is a consequence of the word order freeness in Turkish.

[14] This set of measurements were performed on a slower machine and hence the differences in parsing time.

Table 4: Impact of disambiguation on parsing performance

	No disambiguation		With disambiguation		Ratios	
Avg. Length (words)	Avg. parses	Avg. time (sec)	Avg. parses	Avg. time (sec)	parses	speed-up
5.7	5.78	29.11	3.30	11.91	1.97	2.38

Note: The ratios are the averages of the sentence by sentence ratios.

6.2 Examples

The first example we present is for a sentence which shows very nicely where the structural ambiguity comes out in Turkish.[15] The output for (8a) indicates that there are four ambiguous interpretations for this sentence as indicated in (8b-e):[16]

(8a) **Küçük** **kırmızı** **top** **git+tikçe** **hızlan+dı.**
little red ball go+GER speed up+PAST(+3SG)
kırmız+ı gradually
red paint/insect+3SG–POSS

(8b) The little red ball gradually sped up.
(8c) The little red (one) sped up as the ball went.
(8d) The little (one) sped up as the red ball went.
(8e) It sped up as the little red ball went.

The output of the parser for the first interpretation, which is in fact semantically the most plausible one, is given in Figure 3. This output indicates that the subject of the sentence is a noun phrase whose modifier part is *küçük*, and modified part is another noun phrase whose modifier part is *kırmızı* and modified part is *top*. The agreement of the subject is third person singular, case is nominative, etc. *Hızlandı* is the verb of the sentence, and its voice is active, tense is past, agreement is third person singular, etc. *Gittikçe* is a temporal adverbial adjunct, derived from a verbal root.

Figures 4 through 7 illustrate the c-structures of the four ambiguous interpretations (8b-e), respectively.[17] Note that:

- In (8b), the adjective *kırmızı* modifies the noun *top*, and this noun phrase is then modified by the adjective *küçük*. The entire noun phrase functions as the subject of the main verb *hızlandı*, and the gerund *gittikçe* functions as an adverbial adjunct of the main verb.
- In (8c), the adjective *kırmızı* is used as a noun, and is modified by the adjective *küçük*.[18] This noun phrase functions as the subject of the main verb. The noun *top* functions as the subject of the gerund *gittikçe*, and this non-finite clause functions as an adverbial adjunct of the main verb.

[15] This example is not in any of the texts mentioned above. It is taken from the first author's M.Sc. thesis [4].

[16] In fact, this sentence has a fifth interpretation due to the lexical ambiguity of the second word. In Turkish, *kırmız* is the name of a shining, red paint obtained from an insect with the same name. So, (8a) also means '*His little red paint/insect sped up as the ball went.*' However, this is very unlikely to come to mind even for native speakers.

[17] The c-structures given here are simplified by removing some nodes introduced by certain intermediary rules, to increase readability.

[18] In Turkish, any adjective can be used as a noun.

```
((SUBJ
   ((*AGR* 3SG) (*CASE* NOM)
                (*DEF* -)
                (*CAT* NP)
                (MODIFIED
                  ((*CAT* NP)
                    (MODIFIER
                      ((*CASE* NOM) (*AGR* 3SG)
                                    (*LEX* "kIrmIzI")
                                    (*CAT* ADJ)
                                    (*R* "kIrmIzI")))
                    (MODIFIED
                      ((*CAT* N) (*CASE* NOM)
                                 (*AGR* 3SG)
                                 (*LEX* "top")
                                 (*R* "top")))
                    (*AGR* 3SG)
                    (*CASE* NOM)
                    (*LEX* "top")
                    (*DEF* -)))
                (MODIFIER
                  ((*SUB* QUAL) (*CASE* NOM)
                                (*AGR* 3SG)
                                (*LEX* "kUCUk")))
                (*LEX* "top")))
  (VERB
    ((*TYPE* VERBAL) (*VOICE* ACT)
                     (*LEX* "hIzlandI")
                     (*CAT* V)
                     (*R* "hIzlan")
                     (*ASPECT* PAST)
                     (*AGR* 3SG)))
  (ADVADJUNCTS
      ((*SUB* TEMP) (*LEX* "gittikCe")
                    (*CAT* ADVP)
                    (*CONV*
                      ((*WITH-SUFFIX* "dikce") (*CAT* V)
                                               (*R* "git"))))))
```

Figure 3: Output of the parser for the first the ambiguous interpretation of (8a) (i.e., (8b)).

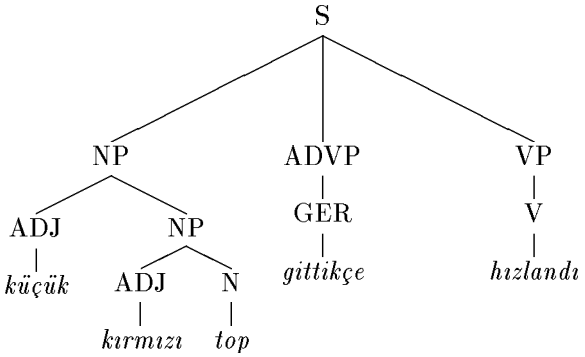

Figure 4: C-structure for (8b).

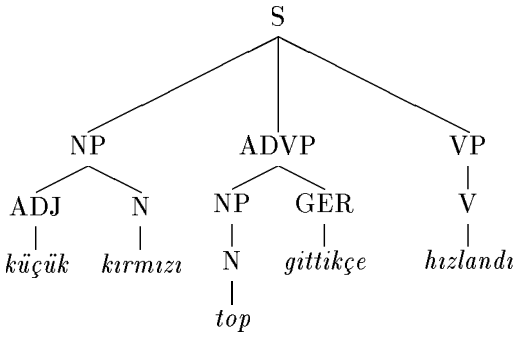

Figure 5: C-structure for (8c).

- In (8d), the adjective *küçük* is used as a noun, and functions as the subject of the main verb. The noun phrase *kırmızı top* functions as the subject of the gerund *gittikçe*, and this non-finite clause functions as an adverbial adjunct of the main verb.
- Finally, in (8e), the noun phrase *küçük kırmızı top* functions as the subject of the gerund *gittikçe* (cf. (8b) where it functions as the subject of the main verb), and this non-finite clause functions as an adverbial adjunct of the main verb. Note that the subject of the main verb in this interpretation (i.e., *it*) is a covert one. Hence, it does not appear in the c-structure shown in Figure 7.

It can be seen that the ambiguities result essentially from the various ways the initial noun phrase can be apportioned into two separate noun phrases, one being the subject of the main sentence, and the other being the subject of the embedded gerund clause. This is possible in this case since all Turkish adjectives can function as nouns effectively modifying a covert third person singular nominal. It is possible to remove some of these ambiguities in a post-processing stage where, for example, parses with the longest noun phrases and/or with overt subjects are preferred.

The second example is for a rather complicated sentence (7) given earlier, which involves embedded dependent clauses. We repeat it here for convenience:

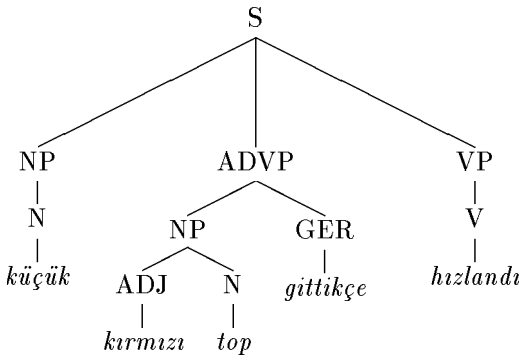

Figure 6: C-structure for (8d).

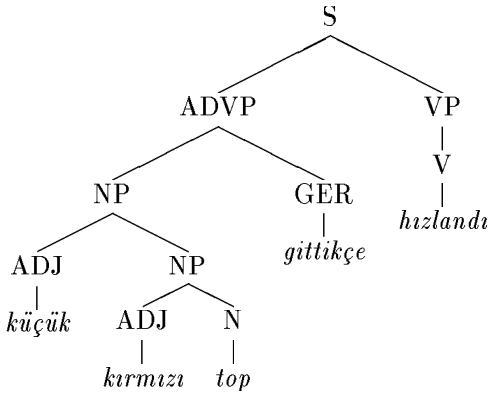

Figure 7: C-structure for (8e).

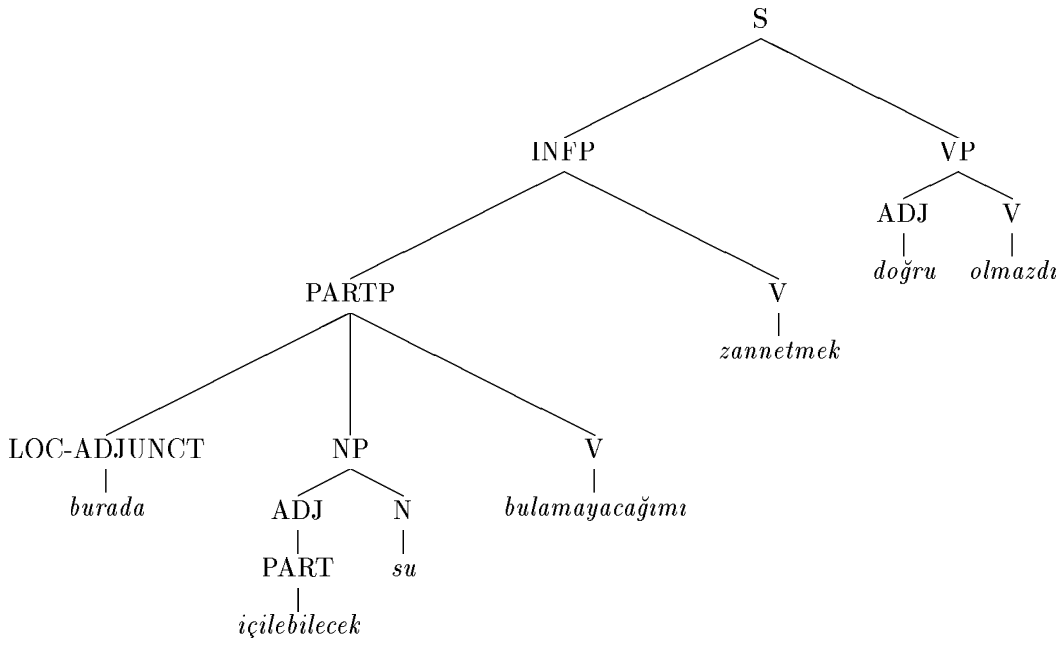

Figure 8: C-structure for the intended interpretation of (7)

(7)	**Bura+da**	**iç+il+ebil+ecek**	**su**	
	here+LOC	drink+PASS+POT +FUT–PART	water	
	bul+ama+yacağ+ım+ı	**zannet+mek**	**doğru**	**ol+maz+dı.**
	find+NEG–POT+FUT–PART +1SG–POSS+ACC	think+INF	right	be+NEG–AOR +PAST(+3SG)

It wouldn't be right to think that I wouldn't be able to find drinkable water here.

Figure 8 shows the c-structure and Figures 9 and 10 show the f-structure generated by the parser, for the intended interpretation.

Although, the gloss above is the intended or preferred interpretation of this sentence where the locative adjunct *burada* is attached to the participle phrase *içilebilecek su bulamayacağımı*, the parser generates additional parses which attach *burada* to each of the other two embedded clauses and the main verb, resulting in three more parses:

1. It would not be right to think that I would not be able find water that could not be drunk *here* (literally – not drinkable here) (where *burada* modifies the participle *içilebilecek*).

2. It would not be right to think *here* that I would not be able find drinkable water (where *burada* modifies the infinitive *zannetmek*).

3. It would not be right *here*, to think that I would not be able to find drinkable water (where *burada* modifies the main verb *olmazdı*).

Furthermore, a number of other parses are generated due to the fact that the participle *bulamayacağımı* can be interpreted as a stand alone adjective, which has been used as a noun. This is

because although the root verb *bul-* (find) is transitive, its object is optional (which is true of almost all Turkish transitive verbs). In this case the preceding noun phrase *içilebilecek su* is not attached as the object noun phrase to this participle, but rather acts as a modifier for its adjectival interpretation, resulting in a syntactically valid compound noun.

This example shows another aspect of Turkish syntax that we deal with in a very limited fashion (though not in this specific example): that of using punctuation information to resolve attachment ambiguities. For instance, a comma after the locative adjunct *burada* would attach it to the main verb *olmazdı* corresponding to the 3^{rd} interpretation above, while the lack of this comma could be taken as a basis to rule out this interpretation.

The third example that we present serves to emphasize our capability in dealing with word order freeness. Our approach to handling word order freeness does not deal with all of the subtle issues involved. We accept a sentence to be grammatically correct if the order of the constituents (at every level) does not violate certain constraints (namely, those that we discuss in Section 3.2) and the argument requirements of the verbs are satisfied.

The example is the following sentence:

(8)

Ben	**kitab+ı**	**ev+den**	**okul+a**	**götür+dü+m.**
I	book+ACC	house+ABL	school+DAT	take+PAST+1SG

I took the book from the house to the school.

Our system processes this as follows:

```
Enter the sentence :  ben kitabI evden okula  gOtUrdUm

("ben" "kitabI" "evden" "okula" "gOtUrdUm")
Total time in Morphological Analyzer =      736 Msecs
Avg/word =      147 Msecs

((((*LEX* "ben") (*CAT* N) (*R* "ben")(*AGR* 3SG)(*CASE* NOM))
  ((*LEX* "ben") (*CAT* PN) (*R* "ben") (*AGR 1SG)(*CASE* NOM)))
 (((*LEX* "kitabI") (*CAT* N) (*R* "kitap")(*AGR* 3SG)(*POSS* 3SG))
  ((*LEX* "kitabI") *CAT* N) (*R* "kitap")(*AGR* 3SG)(*CASE* ACC)))
 (((*LEX* "evden") (*CAT* N) (*R* "ev")(*AGR* 3SG)(*CASE* ABL)))
 (((*LEX* "okula") (*CAT* N) (*R* "okul")(*AGR* 3SG)(*CASE* DAT)))
 (((*LEX* "gOtUrdUm" (*CAT* V) (*R* "gOtUr") (*TENSE* PAST)
                                              (*AGR* 1SG))))

 1 (1) ambiguity found and took 2.454042 seconds of real time
```

The functional structure that is output for this case is the following:

```
;**** ambiguity 1 ***
((SUBJ
   ((*AGR* 1SG) (*CASE* NOM)
                (*CAT* NP)
                (*DEF* +)
                (*LEX* "ben")
```

```
(SUBJ      ((*AGR* 3SG) (*CASE* NOM)
                (*CAT* NP)
                (*DEF* NIL)
                (INFINITIVAL
                  ((*CONV* ((*WITH-SUFFIX* "mak") (*CAT* V)
                                                  (*R* "zannet")))
                    (OBJS
                      ((*DEF* +) (*CASE* ACC)
                                 (ADJUNCT
                                   ((*TYPE* LOCATIVE) (*CAT* NP)
                                                      (*DEF* NIL)
                                                      (*AGR* 3SG)
                                                      (*LEX* "burada")
                                                      (*R* "bura")
                                                      (*CASE* LOC)))
                                 (INFINITIVAL
                                   ((*CONV*
                                      ((*CAT* V) (*WITH-SUFFIX* "yacak")
                                                 (*R* "bul")
                                                 (*SENSE* NEGC)))
                                     (OBJS
                                       ((*DEF* -) (*LEX* "su")
                                                  (*AGR* 3SG)
                                                  (MODIFIER
                                                    ((*CASE* NOM)
                                                      (*CONV*
                                                        ((*CAT* V)
                                                          (*WITH-SUFFIX*
                                                            "yacak")
                                                          (*R* "iC")
                                                          (*VOICE* PASS)
                                                          (*COMP* "yabil")))
                                                      (*AGR* 3SG)
                                                      (ARGS
                                                        (((*CASE* (NOM ACC))
                                                           (*TYPE* DIRECT)
                                                           (*OCC* OPTIONAL)
                                                           (*ROLE* THEME))))
                                                      (*LEX* "iCilebilecek")
                                                      (*CAT* ADJ)))
                                                  (MODIFIED
                                                    ((*CAT* N) (*CASE* NOM)
                                                               (*AGR* 3SG)
                                                               (*LEX* "su")
                                                               (*R* "su")))
                                                  (*CAT* N)
                                                  (*CASE* NOM)
                                                  (*TYPE* DIRECT)
```

Figure 9: Output of the parser for the intended interpretation of (7)

```
                                        (*ROLE* THEME)))
                                (*AGR* 3SG)
                                (ARGS
                                  (((*CASE* (NOM ACC)) (*TYPE* DIRECT)
                                                       (*OCC* OPTIONAL)
                                                       (*ROLE* THEME))))
                                (*LEX* "bulamayacaGImI")
                                (*CAT* ADJ)
                                (*POSS* 1SG)
                                (*CASE* ACC)))
                            (*AGR* 3SG)
                            (*CAT* NP)
                            (*TYPE* DIRECT)
                            (*ROLE* THEME)))
                    (*CASE* NOM)
                    (*AGR* 3SG)
                    (ARGS
                      (((*CASE* (NOM ACC)) (*TYPE* DIRECT)
                                           (*OCC* OPTIONAL)
                                           (*ROLE* THEME))))
                    (*LEX* "zannetmek")
                    (*CAT* INF))))

(VERB
  ((*CAT* VP) (*TYPE* VERBAL)
              (*VOICE* ACT)
              (*LEX* "olmazdI")
              (*R* "ol")
              (*SENSE* NEG)
              (*ASPECT* AOR)
              (*TENSE* PAST)
              (*AGR* 3SG)))
(ADVCOMPLEMENTS
  ((*SUB* QUAL) (*AGR* 3SG)
                (*LEX* "doGru")
                (*CAT* ADJ)
                (*R* "doGru"))))
```

Figure 10: Output of the parser for the intended interpretation of (7)(continued)

```
              (*R* "ben")))
(VERB
  ((OBJS
     (*MULTIPLE*
       ((*CASE* DAT) (*R* "okul")
                     (*LEX* "okula")
                     (*AGR* 3SG)
                     (*DEF* NIL)
                     (*CAT* NP)
                     (*TYPE* OBLIQUE)
                     (*ROLE* GOAL))
       ((*CASE* ABL) (*R* "ev")
                     (*LEX* "evden")
                     (*AGR* 3SG)
                     (*DEF* NIL)
                     (*CAT* NP)
                     (*TYPE* OBLIQUE)
                     (*ROLE* SOURCE))
       ((*DEF* +) (*CASE* ACC)
                  (*R* "kitap")
                  (*LEX* "kitabI")
                  (*AGR* 3SG)
                  (*CAT* NP)
                  (*TYPE* DIRECT)
                  (*ROLE* THEME))))
    (*CAT* VP)
    (*TYPE* VERBAL)
    (*VOICE* ACT)
    (ARGS
      (((*CASE* (NOM ACC)) (*TYPE* DIRECT)
                           (*OCC* OPTIONAL)
                           (*ROLE* THEME))
       ((*CASE* DAT) (*TYPE* OBLIQUE)
                     (*OCC* OPTIONAL)
                     (*ROLE* GOAL))
       ((*CASE* ABL) (*TYPE* OBLIQUE)
                     (*OCC* OPTIONAL)
                     (*ROLE* SOURCE))))
    (*LEX* "gOtUrdUm")
    (*R* "gOtUr")
    (*TENSE* PAST)
    (*AGR* 1SG))))
```

Note that at this point we are not able to extract discourse-related information like topic, focus, background information, which is mostly marked using the constituent order.

Note also the following summary of outputs, which show what our approach can handle in terms of word-order freeness.

```
Enter the sentence:  evden okula ben kitabI gOtUrdUm
  2 (2) ambiguities found and took 2.128624 seconds of real time

Enter the sentence :  evden ben okula kitabI gOtUrdUm
 1 (1) ambiguity found and took 1.650397 seconds of real time

Enter the sentence :  evden kitabI okula ben gOtUrdUm
 1 (1) ambiguity found and took 1.906963 seconds of real time

Enter the sentence :  okula evden kitabI ben gOtUrdUm
 1 (1) ambiguity found and took 1.749944 seconds of real time

Enter the sentence :  okula kitabI ben evden gOtUrdUm
 1 (1) ambiguity found and took 2.176758 seconds of real time

Enter the sentence :  evden kitabI ben okula gOtUrdUm
 1 (1) ambiguity found and took 1.713014 seconds of real time

Enter the sentence :  kitabI okula ben evden gOtUrdUm
 1 (1) ambiguity found and took 1.842986 seconds of real time

Enter the sentence :  gOtUrdUm ben okula evden kitabI
 1 (1) ambiguity found and took 1.439124 seconds of real time

Enter the sentence :  okula gOtUrdUm ben evden kitabI
 1 (1) ambiguity found and took 1.370975 seconds of real time

Enter the sentence :  ben kitap gOtUrdUm evden okula
 1 (1) ambiguity found and took 1.487312 seconds of real time

Enter the sentence :  kitap ben gOtUrdUm evden okula
 failed

Enter the sentence :  ben kitap evden okula gOtUrdUm
 failed
```

A few points of clarification are needed here. In the first example above, there is a syntactically correct second interpretation due to the lexical ambiguity of the word *ben* (pronoun *I*, or noun *mole*). The second interpretation when followed by a noun with the compound marker (CM) (*kitabı* - whose surface form is the same as its accusative form) forms a syntactically valid compound noun *ben kitabı*, in which case the subject of the whole sentence is assumed to be covert and just marked with the agreement suffix in the verb:

(9)	**Ev+den**	**okul+a**	**ben**	**kitab+ı**	**götür+dü+m.**
	house+ABL	school+DAT	I	book+ACC	take+PAST+1SG
			mole	book+CM	

I took the book from the house to the school.
I took a mole book from the house to the school.

The last two examples in the summary above display cases where the position of the nominative direct object *kitap* has strayed from the immediately preverbal position rendering these sentences ungrammatical (cf. the constraint on nominative direct objects given in Section 3.2).

Finally, consider the following example regarding the constraints on word order that we mention in Section 3.2. In the case of (10), the parser generates two ambiguities where, in the first one the adjective *hızlı* modifies the succeeding noun *araba*, and in the second one it acts as an adverbial adjunct modifying the verb *götürdüm*:

(10)

Ben	**ev+den**	**okul+a**	**hızlı**	**araba**	**götür+dü+m.**
I	house+ABL	school+DAT	fast	car	take+PAST+1SG

I took a fast car from the house to the school.
I quickly took a car from the house to the school.

```
Enter the sentence :  ben evden okula hIzlI araba gOtUrdUm

("ben" "evden" "okula" "hIzlI" "araba" "gOtUrdUm")
Total time in Morphological Analyzer =      925 Msecs
Avg/word =      154 Msecs
....
2 (2) ambiguities found and took 5.820933 seconds of real time
```

If, however, *hızlı* appears in the immediately preverbal position, the sentence becomes ungrammatical and is rejected by the parser since the nominative direct object *araba* does not immediately precede the verb:

```
Enter the sentence :  ben evden okula araba hIzlI gOtUrdUm

("ben" "evden" "okula" "araba" "hIzlI" "gOtUrdUm")
Total time in Morphological Analyzer =      880 Msecs
Avg/word =      146 Msecs

failed
```

On the other hand, had the direct object *araba* been accusative (with the surface form *arabayı*) then we would have a grammatical sentence even when the adverb was preverbal:

(11)

Ben	**ev+den**	**okul+a**	**araba+yı**	**hızlı**	**götür+dü+m.**
I	house+ABL	school+DAT	car+ACC	fast	take+PAST+1SG

I quickly took the car from the house to the school.

```
Enter the sentence :  ben evden okula arabayI hIzlI gOtUrdUm
```

```
("ben" "evden" "okula" "arabayI" "hIzlI" "gOtUrdUm")
Total time in Morphological Analyzer =      871 Msecs
Avg/word =      145 Msecs
 1 (1) ambiguity found and took 2.938792 seconds of real time
.....
```

7 Conclusions and Suggestions

We have presented a summary and highlights of our current work on parsing Turkish using a unification-based framework. This is the first such effort for constructing a computational grammar for Turkish with such a wide coverage and is expected to be used in further machine translation work involving Turkish in the context of a larger project. The rather complex morphological analyses of agglutinative word structures of Turkish are handled by a full-scale two-level morphological specification implemented in PC-KIMMO [1].

We have a number of directions for improving our grammar and parser:

- Turkish is very rich in terms of non-lexicalized collocations where a sequence of lexical forms with a certain set of morpho-syntactic constraints is interpreted from a syntactic point as a single entity with a completely different part of speech. For instance any sequence like:

  verb+AOR+3SG verb+NEG+AOR+3SG

  with both verbal roots the same, is equivalent to the manner adverbial "by verb+ing" in English, yet the relations between the original verbal root and its complements are still in effect. We currently deal with these in the parser, but our tagger [7, 13] can successfully deal with these and we expect to integrate this functionality to relieve the parser from dealing with such lexical problems at syntactic level.

- We are currently working on extending our domain to make it cover the types of sentences other than structurally simple and complex ones as well.

- Turkish verbs have typically many idiomatic meanings when they are used with subjects, objects, adverbial adjuncts with certain lexical, morphological and semantic features. For example, the verb *ye-* (*eat*), when used with the object:

  - *para* (*money*) with no case and possessive marking, means to accept bribe,
  - *para* with obligatory accusative marking and optional possessive marking, means to spend money,
  - *kafa* (*head*) with obligatory accusative marking and no possessive marking, means to get mentally deranged,
  - *hak* (*right*) with optional accusative and possessive marking, means to be unfair to somebody,
  - *baş* (*head*) (or a noun denoting a human) with obligatory accusative and possessive marking (obligatory only with *baş*), means to waste or demote a person.

  Clearly such usage has impact on thematic role assignments to various role fillers, and even on the syntactic behavior of the verb in question. For instance, for the second and third

cases, a passive form would not be grammatical. We have designed and built a verb lexicon and verb sense and idiomatic usage disambiguator [17] to deal with this aspect of Turkish explicitly and are in the process of integrating it into the parser. This verb lexicon is inspired by the CMU-CMT approach [9, 10] and in addition uses an ontological database represented in the LOOM system for evaluating complex selectional constraints.

8 Acknowledgments

We would like to thank Carnegie-Mellon University, Center for Machine Translation for making available to us their LFG parsing system. We would also like to thank Elisabet Engdahl and Matt Crocker of Centre for Cognitive Science, University of Edinburgh, for providing valuable comments on an earlier version of this paper. This work was done as a part of a large scale NLP project and was supported in part by a NATO Science for Stability Grant TU-LANGUAGE.

References

[1] E. L. Antworth, *PC-KIMMO: A Two-level Processor for Morphological Analysis.* Summer Institute of Linguistics, 1990.

[2] R. Şimşek, *Örneklerle Türkçe Sözdizimi (Turkish Syntax with Examples).* Kuzey Matbaacılık, 1987.

[3] E. E. Erguvanlı,. *The Function of Word Order in Turkish Grammar.* PhD thesis, Department of Linguistics, University of California, Los Angeles, 1979.

[4] Z. Güngördü, "A lexical-functional grammar for Turkish," M.Sc. thesis, Department of Computer Engineering and Information Sciences, Bilkent University, Ankara, Turkey, July 1993.

[5] R. Kaplan and J. Bresnan, *The Mental Representation of Grammatical Relations*, chapter Lexical-Functional Grammar: A Formal System for Grammatical Representation, pp. 173–281. MIT Press, 1982.

[6] L. Karttunen and K. R. Beesley,. "Two-level rule compiler,". Technical Report, XEROX Palo Alto Research Center, 1992.

[7] İ. Kuruöz, "Tagging and morphological disambiguation of Turkish text," M.Sc. thesis, Department of Computer Engineering and Information Sciences, Bilkent University, Ankara, Turkey, July 1994.

[8] R. H. Meskill, *A Transformational Analysis of Turkish Syntax.* Mouton, The Hague, Paris, 1970.

[9] I. Meyer, B. Onyshkevych, and L. Carlson,. "Lexicographic principles and design for knowledge-based machine translation,". Technical Report CMU-CMT-90-118, Carnegie-Mellon University, Center for Machine Translation, 1990.

[10] S. Nirenburg, J. Carbonell, M. Tomita, and K. Goodman, *Machine Translation: A Knowledge-based Approach.* Morgan Kaufman Publishers, 1992.

[11] K. Oflazer, "Two-level description of Turkish morphology," in *Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics*, April 1993. A full version appears in *Literary and Linguistic Computing*, Vol.9 No.2, 1994.

[12] K. Oflazer and C. Bozşahin, "Turkish Natural Language Processing Initiative: An overview," in *Proceedings of the Third Turkish Symposium on Artifical Intelligence*. Middle East Technical University, 1994.

[13] K. Oflazer and İ. Kuruöz, "Tagging and morphological disambiguation of Turkish text," in *Proceedings of the Fourth Conference on Applied Natural Language Processing*, pp. 144–149. ACL, 1994.

[14] S.M. Shieber, *An Introduction to Unification-Based Approaches to Grammar*. CSLI Lecture Notes 4, 1986.

[15] H. Musha T. Mitamura and M. Kee,. *The Generalized LR Parser/Compiler Version 8.1: User's Guide*. Carnegie-Mellon University – Center for Machine Translation, April 1988.

[16] M. Tomita, "An efficient augmented-context-free parsing algorithm," *Computational Linguistics*, vol. 13, 1-2, pp. 31–46, January-June 1987.

[17] O. Yılmaz, "Design and implementation of a verb lexicon and a verb sense disambiguator for Turkish," M.Sc. thesis, Department of Computer Engineering and Information Sciences, Bilkent University, Ankara, Turkey, September 1994.