ENGLISH TO TURKISH MACHINE TRANSLATION USING SYNCHRONOUS GRAMMARS

ONUR GÖRGÜN

IŞIK UNIVERSITY
JUNE, 2022

ENGLISH TO TURKISH MACHINE TRANSLATION USING SYNCHRONOUS GRAMMARS

ONUR GÖRGÜN
B.S., Information Technologies, Işık University, 2005
M.Sc., Information Technologies, Işık University, 2008

Submitted to the School of Graduate Studies in partial fulfillment of the requirements for the degree of
Doctor of Philosophy
in
Computer Engineering

IŞIK UNIVERSITY
JUNE, 2022

IŞIK UNIVERSITY
SCHOOL OF GRADUATE STUDIES

ENGLISH TO TURKISH MACHINE TRANSLATION USING SYNCHRONOUS GRAMMARS

ONUR GÖRGÜN

APPROVED BY:

Assist. Prof. Ayşegül Tüysüz Erman (Thesis Supervisor)	Işık University	_____
Prof. Olcay Taner Yıldız (Thesis Co-Supervisor)	Özyeğin Üniversity	_____
Assist. Prof. Nilgün Güler Bayazıt	Yıldız Technical University	_____
Assist. Prof. İlknur Karadeniz	Işık University	_____
Assoc. Prof. Arzucan Özgür	Boğaziçi University	_____
Assist. Prof. Reyhan Aydoğan	Özyeğin University	_____
Assist. Prof. Rahim Dehkharghani	Işık University	_____

APPROVAL DATE: 14/06/2022

ENGLISH TO TURKISH MACHINE TRANSLATION USING SYNCHRONOUS GRAMMARS

ABSTRACT

Machine translation (MT) has been one of the hot topics in NLP research over recent years. However, most of the related studies have been done for specific languages, and there are a limited number of comprehensive studies for languages with free word order, such as Turkish. English-Turkish is also one of the least frequently studied language pairs in translation due to the morphological and syntactic gaps between the two languages. This also makes it hard to build parallel corpora, which is crucial for the machine translation task.

This thesis aims to be the first statistical syntax tree-based machine translation approach to the English-Turkish language pair, as well as a parallel corpus for translation tasks. We construct an English-Turkish parallel treebank of approximately 17K sentences by following a three-phased approach: manual transformation of English trees from Penn Treebank (PTB) by constraining the translated trees to the reordering of the children and gloss replacement; morphological analysis of the translated gloss; and morphological enrichment of the target tree. For translation consistency, we also developed a set of tools. We also apply the transformation schema to the closed-domain and build 8.3K sentences corpus.

We employ both corpora on machine translation task. In our experiments, we obtained a 12.8 BLEU score in the open-domain and a 26.8 BLEU score in the closed-domain. We also evaluate both corpora intrinsically through perplexity analysis. The results show that our studies on making a corpus can be repeated, and studies on machine translation using the small corpus look promising.

Key words: Syntax tree, tree-based translation, synchronous grammars, statistical machine translation

EŞ ZAMANLI DİLBİLGİSİ İLE İNGİLİZCE'DEN TÜRKÇE'YE MAKİNE ÇEVİRİSİ

ÖZET

Makine Çevirisi, son yıllarda Doğal Dil İşleme araştırma araştırmalarında en önde gelen araştırma alanlarından biri olmaktadır. Ancak, ilgili çalışmaların büyük bir bölümü belirli diller için yapılmış olup, Türkçe gibi serbest sözcük dizilişine sahip diller için sınırlı sayıda kapsamlı çalışma bulunmaktadır. İngilizce ve Türkçe, iki dil arasındaki biçimbilimsel ve sözdizimsel farklılıklar sebebi ile daha az çalışılan dil çiftlerinden biridir. Bu durum aynı zamanda makine çevirisi alanının en önemli bölümünü oluşturan paralel derlem çalışmalarını da zorlaştırmaktadır.

Bu tez, İngilizce-Türkçe dil ikilisine yönelik ilk istatistiksel sözdizimi ağacı tabanlı makine çevirisi yaklaşımı olmayı amaçlamakta ve makine çevirisi uygulamaları için paralel derlem oluşturma çalışmalarını sunmaktadır. Üç aşamalı bir yaklaşım izleyerek 17000 cümle boyutunda bir İngilizce-Türkçe paralel derlem oluşturduk. İzlenen adımlar: çevrilmiş ağaçların alt ağaçlarının yeniden sıralanması ve kelime değişimi ile sınırlandırarak, İngilizce ağaçların Penn Treebank'tan (PTB) el ile dönüştürülmesi; çevrilmiş kelimelerin morfolojik analizi ve hedef ağacın morfolojik olarak zenginleştirilmesi olarak belirtilmiştir. Çeviri tutarlılığı amacı ile bir yazılım araçları seti de geliştirdik. Ağaç dönüşümü yaklaşımımızı teknik alana da uygulayarak kapalı-alan için 8300 cümleden oluşan başka bir derlem daha oluşturduk.

Her iki derlemi de makine çevirisi çalışmalarında kullandık. Denemelerimizde, açık-alan için 12.8 BLEU puanı ve kapalı-alan için 26.8 BLEU puanı elde ettik. Ayrıca, karmaşıklık anazili aracılığı ile her iki derlemi de öz değerlendirmeye tabi tuttuk. Sonuçlar göstermektedir ki derlem oluşturma çalışmalarımız tekrarlanabilir olup, oluşturulan kısıtlı derlem ile yapılan makine çevirisi çalışmalarının umut verici olduğunu göstermektedir.

Anahtar kelimeler: Sözdizim ağacı, ağaç-temelli çeviri, eşzamanlı dilbilgisi, istatistiksel makine çevirisi

ACKNOWLEDGEMENTS

I would like to acknowledge and give my warmest thanks to my thesis supervisor, Assist. Prof. Ayşegül Tüysüz Erman. The completion of this study could not have been possible without her support.

I would like to express my deep and sincere gratitude to my ex-supervisor and co-supervisor, Prof. Olcay Taner Yıldız. His guidance, motivation, enthusiasm, advice, and immense knowledge carried me through all the stages of my thesis. This dissertation would not have been possible without his guidance and persistent help.

Besides my advisors, I would also like to thank the rest of my thesis committee. I would like to give a special thanks to Assist. Prof. Nilgün Güler Bayazıt for all the support, patience, and encouragement she gave me during my research. I'm very thankful to Assist. Prof. İlknur Karadeniz for how much she cared about me, helped me, and gave me advice at every stage of my research.

I would like to thank my dearest committee members, Assoc. Prof. Arzucan Özgür, Assist. Prof. Reyhan Aydoğan, and Assist. Prof. Rahim Dehkharghani for their valuable time and insightful comments.

Finally, I would like to thank my wife for her love, patience, and support throughout my journey. I am extremely grateful to my parents for their love, prayers, care and sacrifices for preparing me for life.

The technical domain treebank creation effort presented in this thesis was funded by the Scientific and Technological Research Council of Turkey (TÜBİTAK) Technology and Innovation Grant Programs Directorate (TEYDEB) 1501 Industrial R&D Projects Grant Program (3140986).

To my family. . .

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVATIONS

BLEU	**Bi**ling**u**al **E**valuation **U**nderstudy
CFG	**C**ontext **F**ree **G**rammar
EBMT	**E**xample **B**ased **M**achine **T**ranslation
EM	**E**xpectation **M**aximization
FAHQMT	**F**ully **A**utomatic and **H**igh **Q**uality **M**achine **T**ranslation
FST	**F**inite **S**tate **T**ransducer
GPL	**G**eneral **P**ublic **L**icense
HMM	**H**idden **M**arkov **M**odel
LDC	**L**inguistic **D**ata **C**onsortium
LHS	**L**eft **H**and **S**ide
MT	**M**achine **T**ranslation
NLP	**N**atural **L**anguage **P**rocessing
NT	**N**on **T**erminal
POS	**P**art **O**f **S**peech
PTB	**P**enn **T**ree **B**ank
RBMT	**R**ule **B**ased **M**achine **T**ranslation
RHS	**R**ight **H**and **S**ide
SCFG	**S**ynchronous **C**ontext **F**ree **G**rammars
SMT	**S**tatistical **M**achine **T**ranslation
SOV	**S**ubject **O**bject **V**erb
SVO	**S**ubject **V**erb **O**bject
TPTB	**T**ransformed **P**enn **T**ree **B**ank

CHAPTER 1

1. INTRODUCTION

Machine Translation (MT) is defined as the process of translating the text given as input in the source language into the target language. In general, the Machine Translation process and its outputs are expected to show parallelism with the human translator's experience in the translation process. However, although the automatic translation process may seem simple during execution, it is a very complex process. As expected from the manual translation process, the Translation Learning system is expected to have a deep knowledge of both grammar and syntax as well as semantics in both languages.

From a chronological perspective, the idea of Machine Translation actually dates back to the 17th century. The transformation of the process that started as an idea into realistic applications was possible with the computers that emerged in the 19th century. The idea of automatic translation, (Weaver, 1947), which began to be pronounced as a realistic idea at the end of the 1940s, was handled from a different perspective by Warren Weaver, who also had the idea, especially during the Second World War. Considered as a cryptographic subject, (Weaver, 1955) is defined as: re-encoding a text encoded in a different alphabet into a known alphabet. The interesting point is that although the subject is handled from a different angle, it has the same basic principles as when the idea was first launched, and these principles, especially in modeling, are the baseline for today's machine translation approaches.

The idea of automatic translation systems, which started to be implemented

with the advent of computer systems, was realized with the first machine translation system implemented in partnership with IBM and George Town University. These early attempts at the Machine Translation task are dictionary-based efforts. However, these studies were insufficient as they did not meet the syntax and semantic requirements of the translation. Seeing this shortcoming in translation, researchers tried to turn to intelligent systems that tried to include semantic information in the process as of the 1960s. Again, IBM and this time with the cooperation of Washington University, they developed the Mark II system. However, the results obtained did not have the desired effect in the field. What followed was to be the beginning of a decline in interest in the field, particularly in the United States due the major fund decrease. The ALPAC report, which emphasizes that the efforts to be made in machine translation are unnecessary, did not affect the continuation of the studies in countries other than the United States, such as Germany, Canada and France. Many studies have been conducted in the field and these studies stand out as example-based studies in closed-domain. The main factors driving the work in this period can be grouped into two main groups: (i) commercial needs; (ii) administrative needs. Some of these studies conducted after 1970s can be listed as follows:

- SYSTRAN (*1968*); founded in 1968, and first used during the Cold War in Russian-English by United States Air Force (USAF).
- METEO (*1976*); developed by Montreal University for translation of weather bulletins.
- LOGOS (*1979*); developed to server the industrial needs for German-English-France.
- METAL MT (*1985*); developed by University of Texas for Dutch-French-English.

The earlier approaches to the machine translation task were the rule-based translation systems (RBMT). In RBMT, source and target language structures are studied to create a target language generator with the help of language duo

specific lexicon. Rule-based approaches are grouped under three main categories: (i) word-to-word translation by dictionary lookup (direct translation); (ii) building the language-dependent representation of target language to create target sentence (transfer-based); (iii) generating target sentence out of language-independent form (interlingua). It has been observed that direct-translation approaches do not give successful results, especially for syntactically and morphologically distant language pairs. The Turkish-English language duo exemplifies this situation due to the great syntactic and morphological differences between them. These differences cause the rule extraction process to result in poor performance. On the other hand, interlingua and transfer-based models yield better results because they involve both syntax and morphology.

Researchers turned to smarter and more automated approaches to address the gaps in rule-based machine translation approaches. These studies are grouped under two main categories: (i) example-based machine translation (EBMT); (ii) statistical machine translation (SMT). Both approaches share the following characteristics:

- includes syntax and morphology into the model,
- requires expertise on both languages and a lot of manual labor work,

As a result, new translation systems emerged especially for EMBT after 1970s: Eurotra by European Commission (1972-1992) and SUSY (1977). In 1994, IBM developed the first SMT system which is called CANDIDE (DellaPietra and DellaPietra, 1994). CANDIDE approaches the machine translation problem as an optimization problem. Even requires huge amount of parallel data, SMT approaches yields much better results than rule-based translation models.

Large volumes of parallel data are critical to the performance of statistical machine translation systems (SMT). However, parallel data acquisition is a very challenging process, especially for languages with free word order such as Turkish. Although the difficulty of the process has also affected the machine translation studies on Turkish, the studies carried out in the field have increased especially in recent years. The first studies in Turkish were also approaches made

for the English-Turkish language pair (Sagay, 1981). Some important studies on English-Turkish language pairing in the following period can be listed as follows: translation-based models (Turhan, 1997); rule-based models (Hakkani et al., 1998); one of the earliest statistical models (Durgar El-Kahlout, 2009).

Statistical approaches require huge amount of parallel data to build successful translation models. The aim of the system is to produce the best probable translation for the given sentence employing the trained model. Even requires huge amount of parallel data, SMT approaches yields much better results than rule-based translation models. With the guidance of industry needs and the involvement of software giants such as Google and Microsoft, the work in the field of machine translation has gained momentum. The online translation systems of these companies have increased the interest in the field with the development they have shown in both language diversity and translation success.

In the scope of this thesis, we aim to build a statistical machine translation model as the theory suggests: (1) an adequate amount of parallel data as training input; (2) learning algorithm; and (3) a decoder to generate the target sentence. In contrast to phrase-based translation efforts for English-Turkish, this study aims to integrate both morphology and syntactic annotation using constituency tree structure into the parallel data.

1.1 Motivation of the thesis

The vast majority of related work has been done for specific languages, and there has been little work done on free word-order languages like Turkish. There has been very little work in literature, particularly for English to Turkish translation.

Turkish and English are morphologically and syntactically very different languages. Furthermore, there is a scarcity of parallel data for the English-Turkish pair. This thesis describes a constituency treebank construction strategy and a tree-based translation model that aims to be the first to apply statistical machine translation to the English-Turkish language pair.

1.2 Contributions of the thesis

This thesis presents the results of a tree-based statistical machine translation approach from English to Turkish. Indeed, this thesis proposes a machine translation corpus generation schema and an approach for machine translation that utilizes the corpus. There are several points that motivates the study presented in the scope of this thesis as follows:

- We construct the first tree-based English-Turkish corpus with syntactic and morphological annotations using the Penn Treebank corpus. We finally construct a corpus of nearly 17K sentences of varying lengths.
- We extend our corpus generation strategy from open-domain to closed-domain (telecommunication domain) and demonstrate the applicability of our methodology across domains. Finally, we were able to compile another 8.3K sentence corpus in the technical domain.
- We both evaluate our corpora both intrinsically through perplexity and extrinsically through the machine translation task, which constitutes the core aim of this thesis. We obtain 12.8 BLEU score in the open-domain and a 26.8 BLEU score in the closed-domain.
- We produce the entire tool set within the scope of this study: an FST-based morphological analyzer; a statistical morphological disambiguator; and a visual corpus annotation framework for human annotators.

1.3 Thesis Outline

The following is the outline of the thesis.

In Chapter 2, we begin with a quick overview of the history of machine translation. We present the fundamental concept underlying statistical machine translation (SMT). The many techniques of SMT, including word-based, phrase-based, and tree-based models, are then described.

In Chapter 3, we offer a brief comparison of the English and Turkish languages. We examine data alignment issues as well as morphological and syntactical distinctions between the English and Turkish languages.

In Chapter 4, we investigate the parallel treebank creation efforts for Turkish and other languages. We list the challenges to create an English-Turkish treebank. We also present our corpus construction strategy in detail. We illustrate the construction process by introducing our reordering criteria and highlighting the suffixation order in Turkish while introducing our morphological annotation. Moreover, we provide statistics about our corpora, which provide insights. We also apply our parallel corpus creation process to closed-domain corpora and provide statistics as well.

In Chapter 5, we express our translation approach for English-to-Turkish statistical machine translation. This is a two-level translation approach: leaf reordering and gloss replacement. We also define intrinsic evaluation experiments based on perplexity analysis. Lastly, we conclude the chapter with experimental results.

In Chapter 6, we conclude with contributions and future work.

CHAPTER 2

2. MACHINE TRANSLATION OVERVIEW

The aim of this section is to categorize machine translation approaches from past to present. This categorization is based on Chomsky's linguistic theory (Chomsky, 1957). Chomsky's linguistic theory is known as a cornerstone in the field of machine translation. The ideas presented in the theory have also guided studies on the categorization of machine translation approaches (Vauquois, 1968). The study, which makes a classification according to the depth of analysis made during the translation model and the language factors included in the process, and reveals this hierarchically (see Figure 2.1), also shed light on subsequent studies. In this study, statistical machine translation models among these approaches were examined in depth.

The earlier approaches in machine translation were generally based on the rule-based approaches. These studies were basically in limited areas, such as weather forecasting domain (Thouin, 1981) or technical document translation domain. Although the rule-based model outputs are in line with the idea of "Fully Automatic and High-Quality Translation", these results were obtained on restricted domains where rule definition is relatively easy. The partnership of IBM and Georgetown University, which emerged as a work put forward as a direct-translation approach, is also cited as an example in this category as a dictionary-based translation approach. These approaches, excluding syntax and morphology, are at the lowest level of the Vauquois hierarchy (see Figure 2.1).

The transfer-based machine translation method is an approach to address this shortcoming. In the transfer-based translation approach, the text given in the

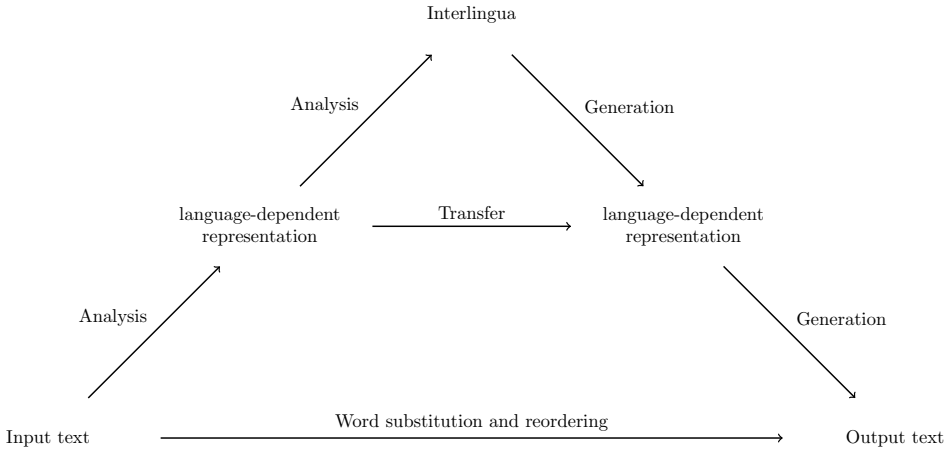

Figure 2.1 The Vauquois Triangle by Hutchins & Somers, 1992.

source language goes through the analysis stage and is translated into an intermediate presentation form. This presentation, which is a language-specific form, is actually a rule-based translation approach, these rules are known as transfer rules. The transition from language-dependent and transfer rule-based models to language-independent and abstract presentation approach has been possible with the emergence of interlingua approaches. Using this language-dependent presentation, text in the target language is generated.

Statistical machine translation models are data-driven translation approaches as opposed to rule-based translation models. Example-based statistical machine translation systems (EMBT) are the precursors of statistical models. As a basic principle, they translate using similar translation memory between source and target languages. The basic steps of the EBMT model, which requires large volumes of parallel data, are defined as follows: (i) separation of source sentence words and phrases; (ii) replacing each word and phrase with its target language equivalent.

The focus of research on statistical translation models in machine translation studies started with the reporting of the superiority of the IBM CANDIDE system over rule-based approaches. The biggest reason for the success was the emergence of computer systems with increased data processing performance.

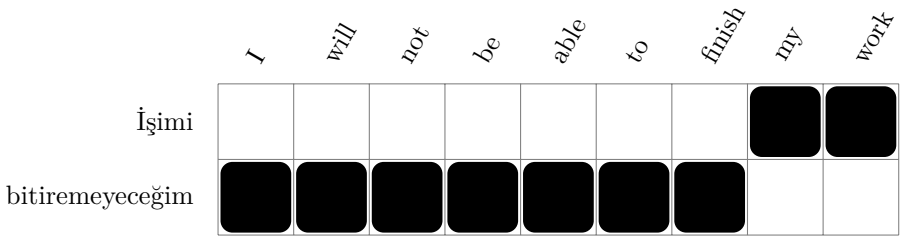

Figure 2.2 An example word alignment matrix.

2.1 Statistical Machine Translation Approaches

The Statistical Machine Translation system aims to ensure both the adequacy and the fluency of the translation. While adequacy is defined as the accuracy and precision of the translation, fluency is described as the ease of the translation. In this section, we aim to describe the statistical machine translation approaches based on the translation unit used. We also aim to explain how to create an appropriate parallel data to accomplish the machine translation goals.

2.2 Word-based models

Word-based models use words as the main translation unit and perform translation to the target language given the source sentence. More formally, the word-based model replaces the source gloss with its equivalent in the target language. However, this replacement requires an additional process which is called word alignment and is illustrated in the word alignment matrix (see Figure 2.2). There are different approaches to extracting the word alignments from the given parallel text.

The fundamental approach for extracting the word alignments is the expectation maximization (EM) algorithm. The expectation-maximization algorithm tries to get the most probable word replacement among all possible word alignments. Besides the word alignment, the early approach to the word-based model also combines two fundamental blocks to maximize the translation results: the translation model justified by noisy-channel for adequacy and the language model (fluency).

The language models are important to ensure fluency. The most fundamental approach to building the language model is led by IBM Models 1-5 which relies on n-gram probabilities. The IBM Model-1 is the basic model that calculates lexical translation probability distributions using maximum-likelihood estimation from parallel data. The model lacks different fertilities in both source and target languages. The model was also chosen in the early word-based statistical machine translation approaches. The next generation IBM Models brings the following solutions to the Model-1 problems (from Model-2 to Model-5, respectively): (1) absolute alignment (lexical translation and lexical alignment); (2) fertility (filling the gaps in lexical alignment by null insertion); (3) relative alignment (lexical alignment depending on the surrounding words); (4) deficiency (keeping track of unaligned words). In addition to the basic models, there are studies to expand the IBM Models. Some of these studies are listed as follows: Hidden Markov Model (HMM) to address the relative alignment problem but not fertility (Vogel et al., 1996); HMM approach to IBM Model-4 (Och and Ney, 2003).

In order to build the language models from parallel text, there is a need for the proper tooling, especially for a large amount of data. In literature, there exist proven tools to produce the language models such as GIZA++ and NATools (Natural Alignment Tools). Both tools are under GPL-Licence and publicly available.

2.3 Phrase-based models

The phrase-based models treat the chunk of words, namely phrase, as the main translation unit, unlike word-based models. Moreover, word-based models still lack different fertilities and long-distance reordering. For fertility issues, it is not possible to align a single Turkish word “gidiyorum” with multiple English words “I am going”, but a single English word “going”. This alignment is only possible if the Turkish words are expressed in terms of lexical (stem and suffixes). In English-Turkish machine translation, lexical and sub-lexical structures are investigated in the literature (Durgar El-Kahlout, 2009). Moreover, phrase-based

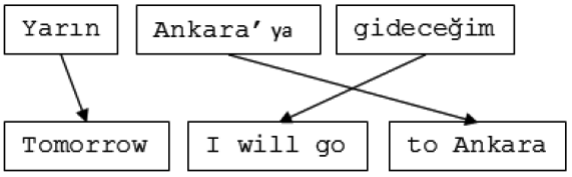

Figure 2.3 Example phrase-based alignment.

models are useful tools to address the localization effect caused by the lack of context in word-based alignments.

In a typical phrase-based translation model, the sentence-aligned parallel text is divided into a chunk of words (phrase) sentence-by-sentence. The extracted phrases in the source language are aligned to the corresponding phrase in the target language. For fluency, the order of the phrases needs to be changed, if required. In Figure 2.3, a sample phrase alignment is illustrated. In-phrase word order is also maintained by a word-based language model.

To keep phrase translation probabilities, a phrase table is built and maintained. When estimating the likelihood of a phrase being translated, word-level lexical translation probabilities are utilized, and the translation probabilities are maintained in a phrase translation table. Word alignment probabilities are used to keep the phrase translation table (Tillmann, 2003; Zhang et al., 2007). Phrase-based statistical machine translation models are successful translation models and are frequently used for various language pairs (Och and Weber, 1998; Och et al., 1999; Och, 2002; Och and Ney, 2004). There are also studies that employ factored statistical translation models for Turkish-English pair (Yeniterzi and Oflazer, 2010).

2.4 Tree-based Models

Word-based and phrase-based models lack syntactic annotation. However, for a successful translation system designed for free word-order languages, it is proven that syntactic annotation plays a crucial role. Indeed, syntactic annotation provided via constituency tree structure is perfectly aligned to the grammatical

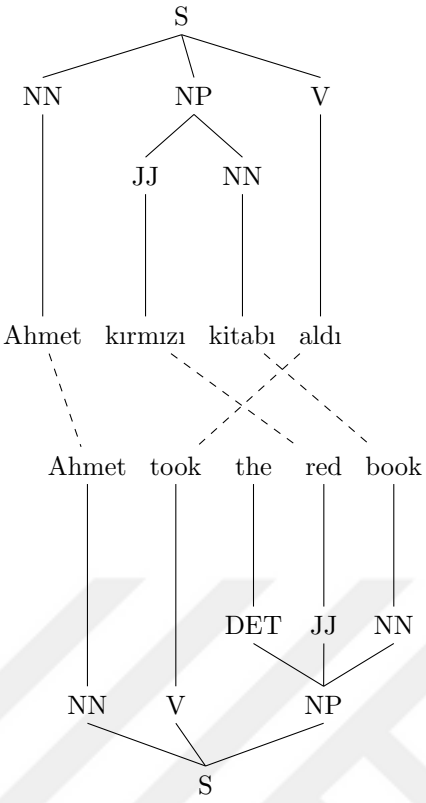

Figure 2.4 An example of word aligned constituency tree structure.

structure of language. The syntactic annotation also brings value to the machine translation for the reasons listed:

- Ensuring the subject-object-verb order in the target sentence.
- Ensuring better fluency by keeping the function words in the translation model.
- Paying attention to the context for better translation of words with multiple translations and syntactically related words.
- Integrating syntactic language model to resolve the complex sentence structures.

As in the morphologic annotation process, syntactic enrichment is also delivered with the help of language-specific tools called syntactic parsers. There are some available tools that provide syntactic annotation. In Turkish, there is no syntactic parser that produces constituency tree structure, but a phrase structure Turkish model for MaltParser (Eryiğit et al., 2008).

The word-based and phrase-based models failed to transfer the recursive structure of the language, as they treat the sentence as a flat sequence of words and phrases. The use of tree-based syntactic structures in translation systems provides high performance. In this part of the study, we will start depicting how a constituency tree is structured. Next, we explain the fundamental model, synchronous grammar formalism, to extract grammar rules from syntax trees, and utilize those rules to form the target sentence.

2.4.1 Synchronous Grammars

Context-free grammars (context-free grammars (CFG)) are the common way of presenting syntax in machine translation. Translation with context-free grammar is performed by parsing the constituency trees. As a result of parsing, context-free grammar rules are extracted from the parallel corpora. In a typical constituency tree structure, there are constituents in a hierarchy, such as Noun Phrase (NP), Prepositional Phrase (PP), VP (Verbal Phrase), Sentence (S), etc. In context-free formalism, grammar rules contains terminal symbols (words) and non-terminal symbols (phrase labels and POS tags) (see Equation 2.1 and 2.2),

$$NT \rightarrow [NT,\ T]+ \tag{2.1}$$

$$NT \rightarrow [NT]+ \tag{2.2}$$

where NT and T identify the non-terminal and the terminal symbols, respectively. The synchronous grammar rule is identified as the pair of rules defined for both source and target language, respectively. In Figure 2.4 an example constituenct tree structure for the Turkish sentence “Ahmet kırmızı kitabı aldı” (“Ahmet took the red book”) is illustrated. The following rules are extracted for noun phrase (NP) out of English-Turkish sentence pair (see Equation 2.3 and 2.4).

$$NP \rightarrow DET\ \ JJ\ \ NN\ \ (English) \tag{2.3}$$

$$NP \rightarrow JJ \;\; NN \;\; (Turkish) \tag{2.4}$$

and a combined synchronous grammar rule is defined for the sample constituency tree as follows (see Equation 2.5):

$$NP \rightarrow DET_1 \;\; JJ_1 \;\; NN_1 \mid JJ_2 \;\; NN_2 \tag{2.5}$$

where index numbers stands for the corresponding non-terminals in both languages. It is possible to define rules for terminal symbols. Rule definitions for the terminal symbols are defined as follows (see Equation 2.6 and 2.7):

$$JJ \rightarrow red \mid kırmızı \tag{2.6}$$

$$NP \rightarrow the \;\; red \;\; book \mid kırmızı \;\; kitap \tag{2.7}$$

or combination of both representation as follows (see Equation 2.8):

$$NP \rightarrow the \;\; JJ_1 \;\; book \mid JJ_2 \;\; kitap \tag{2.8}$$

When the grammar rule extraction is done, the grammar rule set contains rules that reflect the nested structure of the constituency tree. Therefore, there are rules that contain subtrees represented by their phrase label explicitly. Hence, the extracted rule set allows us to build the target sentence in the correct order even the long-distance ordering is required. Similar to the other statistical machine translation approaches, the probability distribution for each rule is calculated by the conditional probability of the left-hand side (English) given the right-hand side (Turkish). The method proposed in this thesis handles both rule types, rules for terminals and non-terminals, separately. Non-terminal rules are used to build the constituency tree on the Turkish side, and terminal rules are used to build gloss level translation model.

2.4.2 Learning and Decoding

There are some fundamental models (Chiang, 2005) that learn from the synchronous grammar rules in the literature. Based on the learning schema, synchronous grammar learning depends on the word-based alignments and the way hierarchical phrase-based models are acquired. Formally, given the word-aligned bilingual corpora, phrase translation rules which are compliant with word alignments are extracted. Then, the conditional probability is estimated where t and e correspond to Turkish and English sentence pairs (see Equation 2.9):

$$\phi(t|e) \tag{2.9}$$

The phrase translation rules for synchronous grammar are defined as follows (see Equation 2.10):

$$Y \rightarrow f \mid e \tag{2.10}$$

As discussed in synchronous grammar formalism, no rule contains non-terminal symbols on the left-hand side of the rule definition. In the constituency tree structure, it is expected to have mixed rules that contain both non-terminal and terminal symbols on the right-hand side. Non-terminal rules on the right-hand side are symbolized by some variable X and correspond to a subtree in the constituency structure. The following rule can be written as follows (see Equation 2.11):

$$Y \rightarrow X \ \ kitabım \mid my \ \ X \ \ book \tag{2.11}$$

variable X may correspond to a non-terminal symbol so that a subtree also contains sub-trees. The situation also reveals the complexity of the learning process depending on the depth of the constituency tree. In order to reduce the computational complexity, the number of non-terminal symbols and the number of words can be limited (Chiang, 2005). Even though the learning process brings complexity, the learning schema yields better results than phrase-based models.

In the literature, there exists some proven rule learning approaches (Koehn, 2010) from the bilingual text. The learning algorithm is defined as follows: assume that $g \in T$ is a governing node for the words $w_1; w_2; ...; w_n$ if and only if the leaf nodes of the sub-tree under g are exactly the words $w_1; w_2; ...; w_n$. The governing node can contain subtrees as its children. It is possible to define synchronous grammar rules for each translation phrase pair $(\bar{e}, \bar{f})$ if;

- there are governing nodes for phrases $\bar{e}, \bar{f}$,
- there are governing nodes for all phrases $\bar{e}, \bar{f}$ of non-terminals.

More formally, general formalism is defined obtained as follows (see Equation 2.12):

$$LHS \rightarrow RHS_t | RHS_e \tag{2.12}$$

The rule extraction phase can end up with a large number of rules for a huge bilingual corpus. Next, probability scores are calculated for the entire rule set. There are several alternatives which serve as the scoring function:

- Joint rule probability $p(LHS; RHS_t; RHS_e)$,
- Rule application probability $p(RHS_e; RHS_t | LHS)$,
- Direct translation probability $p(RHS_t | RHS_e; LHS)$,
- Noisy-channel translation probability $p(RHS_e | RHS_t; LHS)$,
- Lexical translation probability $\prod_{e_i} \in RHS_e p(t_i | RHS_e; a)$ where a is word alignment from word-translation probability

For the decoding phase, we separate the terminal and non-terminal rules and create two different components for our translation model: constituency tree translation and ordering model, and gloss translation model. We utilize the combined translation model for decoding and building the target sentence. We explain the decoding phase in the upcoming sections where the translation approach is explained.

2.5 Parallel Data

With regards to statistical theory, a reliable statistical model requires data of good quality and in the desired volume. For this reason, a high-quality and FAHQMT compliant "parallel corpus" is required in order to build a successful statistical machine translation system. A parallel corpus is identified as the collection of sentences in both source and target languages. In the literature, there are several parallel corpora for different language pairs including English-Turkish. The LDC (Linguistic Data Consortium) corpus (large set of text for Arabic-English, Chinese-English, and French-English language pairs), the Europarl corpus (collection of proceedings of the European Parliament in 11 different languages), the OPUS corpus (collection of open-source software documentation), the Acquis Communautaire corpus (collection of legal documents signed by European Parliament countries), and the European Pat corpus (Täger, 2011) can be listed as some well-known corpora. For Turkish and English pairs, there is also a morphological annotated corpus to address the machine translation problem (Oflazer and Durgar El-Kahlout, 2007).

Prior to the parallel corpus generation, parallel text acquisition is also a crucial and complex process. There are different ways to capture the parallel text from different resources such as crawling web pages from bilingual resources or extracting data from bilingual technical documentation for closed-domain approaches. However, the document extraction process reveals some important problems as well: document alignment and sentence alignment. Document alignment is the initial step in the data acquisition process and involves the alignment of relevant document parts in both languages. Subsequently, sentence alignment is performed on the document-aligned corpus to match the sentence pairs or to combine any fragmented sentences. The sentence alignment process for Turkish-English pair is defined more formally as in the following: given the English sentences $e_1; e_2; ...; e_n$ and the Turkish sentences $t_1; t_2; ...; t_n$, the sentence alignment S, also known as the set of sentence pairs $s_1; s_2; ...; s_n$, matches each English sentence with one or more matching candidates in the target language. The sentence

alignment quality for candidate pairs is determined by the matching function (see Equation 2.13).

$$score(S) = \prod_{i}^{n} match(s_i) \tag{2.13}$$

where s_i is ith translation pair.

There are several studies conducted by researchers to address the sentence alignment problem from different perspectives: aligning sentences based on the word count (Brown et al., 1991; Gale and Church, 1991, 1993); aligning sentence pairs based on their character counts (Church, 1993); using rare and identically spelt words (Enright and Kondrak, 2007).

For the translation language pairs where the syntactic and morphological gap is huge such as English and Turkish, the size and quality of the parallel data play an important role in the training process. When successful machine translation approaches are examined, it is seen that the amount of parallel data used is the size of a hundred thousand or even a million sentences. However, there is no data in that size for the Turkish-English pairs. Consequently, researchers try to solve the data sparsity issue by examining different lexical representations. Undoubtedly, the absence of Turkish-specific syntactic annotation tools and the difficulty of implementing these tools due to the complexity of Turkish also explain this lack of data problem and constitutes another motivation for this thesis.

CHAPTER 3

3. COMPARATIVE ANALYSIS OF LANGUAGES

In this chapter, we summarize the language- and data-specific obstacles that served as the primary motivation for writing this thesis. This chapter begins with a concise introduction to Turkish morphology. The section then continues with a comparative examination of the Turkish-English language pair. The difficulty of the English-Turkish translation problem is largely determined by the morphological and syntactic differences between the two languages. We try to show these differences that make translating from English to Turkish both interesting and hard.

3.1 Turkish Morphology

In Turkish morphology, suffixes are used to construct word forms. Attaching the correct affix to a given root word is an example of suffixation. The suffixation process in Turkish is governed by various suffix categories and rules that determine suffix order. The majority of Turkish words are complex and comprise multiple syllables. In some cases, the way Turkish words are made can lead to very long words that are the same length as whole English sentences.

başarı+sız+laş+tır+ıcı+laş+tır+ıver+ebil+ecek+ler+imiz

'those who we cannot make one easily a maker of unsuccessful ones'

The suffixation process is based on the vowel harmony and consonant agreement, so each vowel and consonant in the suffix is dependent on the suffix that came before it. Regarding vowel harmony and consonant agreement, various

forms of each suffix are possible. For instance, the plural has two forms: "+lar" and "+ler", and the narrative case has four: "+mış", "+miş", "+muş", and "+müş". The meta representations of these various suffix forms (surface forms) are displayed (lexical forms). Capitalization is used to indicate vowels that can change due to vowel or consonant agreement ("+lAş", "+DHr", etc.).

başarı +sız +laş +tır +ıcı +laş +tır +ıver +ebil +ecek +ler +imiz
başarı +sHz +lAş +DHr +HcH +lAş +DHr +HvAr +AbHl +AcAk +lAr +HmHz
'those who we cannot make one easily a maker of unsuccessful ones'

In Turkish, there are two types of affixes: (i) derivational affixes and (ii) inflectional affixes. Affixes of derivation are used to produce new forms that correspond to distinct word classes. There are five word classes: nominal (noun NN, pronoun PNON, adjective ADJ, and adverb ADV), verb VB, postposition PP, conjunction CONJ, and interjection INTJ. Derivational suffixation begins with a root from one of the word classes and concludes with a suffix from another word class. Turkish contains an extensive inventory of derivational affixes based on the transition from one word class to another. Following is a list of some derivational affixes organized by type of derivation.

- Derivations producing nouns:
  - Noun-to-Noun: *göz-NN +DA (gözde-NN),*
    arka-NN +dAş (arkadaş-NN), kitap-NN +lHk (kitaplık-NN).
  - *Verb-to-Noun: del-VB +gAC (delgeç-NN), bildir-VB +gA (bildirge-NN), kork-VB +H (korku-NN).*
- Derivations producing adjectives:
  - Noun-to-Adjective: *insan-NN +CA (insan-ADJ),*
    insan-NN +cHl (insancıl-ADJ), sayı-NN +sAl (sayısal-ADJ).
  - Verb-to-Adjective: *ol-VB +mAdHk (olmadık-ADJ), utan-VB +gAç (utangaç-ADJ), gör-VB +sAl (görsel-ADJ).*
  - Adjective-to-Adjective: *kuru-ADJ +(A)k (kurak-ADJ)*
- Derivations producing verbs:

  - Noun-to-Verb: *tuz-NN +lA (tuzla-ADJ).*
  - Verb-to-Verb: *bak-VB +(H)n (bakın-VB), bak-VB +(H)ş (bakış-VB).*
  - Adjective-to-Verb: *korkak-ADJ +CA (korkakça-ADJ).*

- Adverbials:
  - Noun-to-Adverbial: *insan-NN +CA (insanca-ADV davran-VB), sürat-NN +lA (süratle-ADV yürü-VB).*
  - Verb-to-Adverbial: *dur-VB +dHkçA (durdukça-ADV dur-VB), vur-VB +dHkçA (vurdukça-ADV vur-VB).*

başarı +sHz +lAş +DHr +HcH +lAş +DHr +HvAr +AbHl +AcAk +lAr +HmHz
'those who we cannot make one easily a maker of unsuccessful ones'

The case, person, and tense relationships between nouns and verbals are expressed by inflectional affixes. Number (singular "araba (car.NN)" or plural "araba +lAr (car.PL)), possession that indicates the possessor ("+(H)m" (1st singular), "+(H)n" (2nd singular), "+(s)H" (3rd singular), "+(H)mHz" (1st plural), "+(H)nHz" (2nd plural), and "+lArH" (3rd plural)).The suffixation is in number-possession-case order.

The morphological analysis process is to resolve the given word form as stem and suffixes attached to it. However, another problem still needs to be solved for the final solution: morphological disambiguation. As a result of the morphological analysis, more than one possible analysis result is usually produced for the majority of the words and choosing the right one among these analyzes is a problem. The process is very important in finding the right root word and its suffixes to fill the gap between Turkish and English. There are several studies which address the morphological disambiguation problem: n-gram based statistical approach (Hakkani-Tür et al., 2002); rule-based approach which utilizes Greedy Prepend algorithm as decision list learner (Yuret and Türe, 2006); stochastic approach to the morphological disambiguation (Sak et al., 2007); and a classification model (Görgün and Yildiz, 2012).

3.2 Turkish Syntax

Similar to syntactically complex languages, Turkish sentences can be categorized as simple (single clause) (1) or complex (a main clause and one/many subordinate clauses)(2).

(1) Dün okula gitmedim (I didn't go to school yesterday)

(2) Dün [okula giderken] arkadaşımı gördüm. (Yesterday, [as I was going to school], I ran into a friend).

Subordinate clause in (2) is marked with square brackets.

In Turkish, a sentence has two main constituents: subject and predicate. The subject is not overtly expressed within the sentence. If the subject is expressed, it is always a noun phrase (NP). Noun phrase can be in form of single pronoun (3), or can be indetified by determiners, numerals or adjectives (4).

(3) Ben dün okula gitmedim. (I didn't go to school yesterday).

(4) Bugün iki sınav iptal oldu. (Two exams have been cancelled today).

A predicate expresses an event or process that the subject is involved in. In example (1) "gitmedim"' is the predicate. Sentences are investigated under two main topics with respect to the type of predicate they have: verbal sentences (predicates are finite verbs) and nominal sentences (does not contain an overt verb or contains a verb in form of copula, e.g. to be, to become).

Nominal sentences are categorized as linking and existential. Existential sentences are in form of "A has B" (possessive existential) or "There is an A in B" (locative existential). Linking sentences stick to the pattern ("A is B") and contain the following elements;

- a subject, overtly expressed or not,
- a subject complement as part of the predicate (provides description about location, identification, characterization, and state of the subject),
- a copular marker (person, number marking of the predicate is attached to copular marker),

- negated by “değil” “not” by placing it between the subject complement and copular marker.

In Turkish, even the subject is overt or not, predicate agrees with subject in terms of person and number. For first and second person, person suffixes must be added to predicate for agreement. Third person suffix does not have an overt marking. However, third person plural suffix must be added to predicate, if the plural subject is not expressed by an overt noun phrase (e.g. “Dün gittiler”).

3.3 Turkish vs. English

English, a member of the Indo-European language family, and Turkish, a member of the Ural-Altaic language family, are both syntactically and morphologically separate languages. Especially, syntactic (sentence-level) and morphological (word-level) gaps between two languges require additional processing in Turkish side. We aim to highlight those differences in the following section:

- *Language Family*: Turkish belongs to the Ural-Altaic language family and is identified as an agglutinative language. In contrast, English is an isolating language and part of the Indo-European language family.
- *Word Order*: While the elements of sentence are particularly in order of subject, object, and predicate order (SOV), in English elements are listed in subject, predicate, and object order. Even the word order is subject to change and allow one to create inverted sentences, this is not very common in English. The role of the arguments is identified in Turkish (Turhan, 1997). For example, “Bugün okula gideceğim” (Turkish) sentence can be built as “Okula gittim bugün” (Turkish). In English sentence “I will go to the school, today”, it is not possible to change position of “school” within the sentence.
- *Word Derivation*: Turkish is an agglutinative language and the new words are formed by suffixation. Turkish suffixes are used to build new word forms and change the current word form into an another one as well. In theory,

it is possible to create an unlimited number of words by suffixation. Unlike Turkish, suffixation is used in a very limited context in English. The level of suffixation creates an huge gap between English and Turkish.

- *Verbal Structures*: The differences in building verbal structures are listed in the following four categories:
  - Turkish verbs are always in regular form. However, in English irregularity exists in present, past, and participle forms for English.
  - Gerunds are used in verbal structures in Turkish. Contrary, English uses gerunds in verbal structures only in present continuous form.
  - Copula "to be" (+DHr) in Turkish can be omitted depending on the context. English never neglects the copular marker.
  - In Turkish, personal pronouns and tense affixes are suffixed to the verbal stem to express the tense and propositions. In English, tense and propositions are provided as separate gloss. This difference also creates a morphological gap between the two languages.
- *Nominal Structures*: A typical noun phrase consists of head and one or more modifiers. In Turkish modifiers of noun preceed the head which we call pre-nominal modifiers. In English, modifiers can preceed the head (pre-nominal modifier), or follow the head (post-nominal modifier) at the same time.
- *Definite and Indefinite Determiners*: In Turkish, there is no definite article "the", unlike English.
- *Personal Pronouns and Gender*: In Turkish, gender is only expressed in the third person singular, contrary to English. Moreover, personal pronouns are attached to both verbal and nominal gloss in Turkish. In English, personal pronouns are expressed explicitly as a gloss.
- *Singularity vs. Plurality*: In English, there is a concept of countable and uncountable for nominals. In Turkish , plurality is expressed by suffixation to the nominal word, such as "kitap" (singular) and "kitap+lar" (plural) .

CHAPTER 4

4. SYNTAX-BASED STATISTICAL MACHINE TRANSLATION

In the last decade, statistical machine translation has made significant progress. Since the IBM model's superior performance to traditional rule-based approaches, machine translation research has flourished with increasingly complex statistical models. As computational power and parallel corpora become more accessible, researchers are shifting from manually crafted linguistic models to empirically learned statistical models, from string/word-based models to tree/tree-based models.

Early statistical machine translation approaches are word-based models. These models insert, delete, and rearrange source and target words as their primary translation unit. Formally, the target word(s) should be aligned with the corresponding word in the target language(s). On the other hand, because word-based models utilize words as the fundamental translation units, the alignment obtained by these models does not always correspond to the actual alignment. For languages with varying fertilities, such as English and Turkish, certain words are not aligned. For instance, "geleceğim" (I will come) should be translated as a verbal phrase. However, word-based models typically extract "come" as its English match. Consequently, the translation does not accurately convey the meaning of the Turkish phrase.

The phrase-based model have been used as model for English-Turkish machine translation (Oflazer and Durgar El-Kahlout, 2007). As a next step, a factored phrase-based translation model that defines complex custom syntactic tags

on the English side and inputs the augmented sentences into a phrase-based translation system is proposed (Yeniterzi and Oflazer, 2010). Alternatively, there is an alternative study that examines the effects of various sub-lexical representational structures (Durgar El-Kahlout, 2009).

Integration of syntactic structure into models of machine translation has been shown to enhance performance. Specifically, tree-based models are effective at incorporating the recursive structure of language and provide superior alignment results (Koehn, 2010). In language pairs such as English and Turkish, where the unmarked order varies across the grammar and the latter has numerous permutations, ordering problems are more challenging.

4.1 Challenges

4.1.1 Syntactic Parsing of Turkish

The concept of parsing derives from the idea of rearranging words into larger units, and these larger units, particularly grammars, are the essential building blocks of many NLP applications. Due to the unrestricted word order in Turkish, syntactic parsing causes difficulties, and makes the Turkish treebank creation is complicated.

The history of Turkish syntactic studies dates back 20 years, and researchers have proposed numerous grammar formalisms to address the issue. Within the concept of syntactic parsing, research focuses primarily on two aspects: constituency and dependency. These are the two main concepts of phrase structure and dependency structure in parsing. Turkish NLP research on syntactic parsing focuses on these two structures, with dependency parsing taking precedence.

The breakthrough study on dependency parsing starts with the dependency-based treebank creation efforts of METU, the so-called METU Treebank, consisting of 2 million words. This dependency structure explicitly represents the head-dependent relations and functional categories but is annotation-free. In order to adapt the corpus written in 1990's Turkish to further studies, a subset of the corpus of the size of 10K sentences was annotated morphologically and syntactically

(Atalay et al., 2003). The METU Treebank is used in many Turkish dependency parsing efforts, such as; the statistical parsing method that uses different representational units for parsing (Eryiğit and Oflazer, 2006), machine learning approach that uses a decision list learner to decide the existence, direction, and type of link between the pair of words (Yüret, 2006), linear programming formulation of the dependency parsing problem to enforce linguistics constraints; (Riedel et al., 2006), so called Integer Linear Programming, which is an efficient non-projective Maximum Spanning Tree algorithm than other dependency parsing models; (Çakıcı and Baldridge, 2006), rule-based probabilistic approach; (Eryiğit et al., 2006), a data-driven model which emphasizes the morphological structures of the words for finding syntactic relations (Eryiğit et al., 2008).

There are parsers that generate data for dependency structures, but there are no constituency structure parsers. In this study, we present a Penn Treebank based data generation methodology to address this problem. Our method transforms Penn Treebank sentences into their Turkish equivalents without changing the English tags. The primary advantage of our method is that it requires only good knowledge of Turkish grammar and does not necessitate advanced structural precision.

4.1.2 Annotation Tools

In the absence of a fully-fledged parsing tool that generates constituency structure, annotation tools play a crucial role in the construction of a treebank. Depending on the objective of the study, the toolkit may include sentence chunking and part-of-speech tagging as well as morphological analyzers and disambiguators. Additionally, visual aids are required to simplify the process for the end-user.

In the absence of a fully-pledged parsing tool that produces constituency structure, the annotation tools play an important role in building a treebank. Based on the purpose of the study, the set of tools varies from sentence chunking and part-of-speech tagging to morphological analyzers and disambiguators.

Moreover, visual aids are also required to make the process simpler for the end-user.

Within the scope of this thesis, we propose a variety of software tools to help human annotators. The tree transformation process that we introduce is a semi-automatic process that requires a huge amount of manual hand-work and a good understanding of Turkish syntax and morphology. Naturally, manual annotation is a repetitive and error-prone process. We offer a set of tools so-called NLP Toolkit [1] to help the human annotators during the process (Yıldız et al., 2014). Currently, NLP Toolkit is expanded to cover topics such as semantic role labeling, word sense disambiguation, sentiment analysis, etc. Within the scope of this thesis, we implement the core libraries and the following functionality to transform Penn Treebank trees. Each module is provided in executables with user interface and core Java libraries as well.

- *Visual Tree Transformator*: Penn Treebank is provided in hierarchical square bracketed format, which makes it really challenging to edit. With respect to our transformation methodology, human annotators need to perform sub-tree reordering and gloss replacement, initially. We build a visual tree editor that allows annotators to perform the required actions easily. The initial version of the user interface just allows you to perform the required actions on trees (see Figure 4.1). The Visual Tree Transformation tool also keeps track of translations for the same word and displays these words with their number of occurences. Hence, the annotator can choose any of them that suits or can enter his/her own translation by typing it.
- *Morphological Analyzer*: Even though we ensure the syntactic annotation in Turkish, we still lack morphological annotation due to the morphological gap between English and Turkish. Morphological analysis is expected to analyze the given word and chunk the given word into lexical units: stem and a list of inflectional or derivational suffixes. The morphological analyzer is created based on the Turkish dictionary of root words and possible state

[1] Available on https://github.com/olcaytaner/NlpToolkit

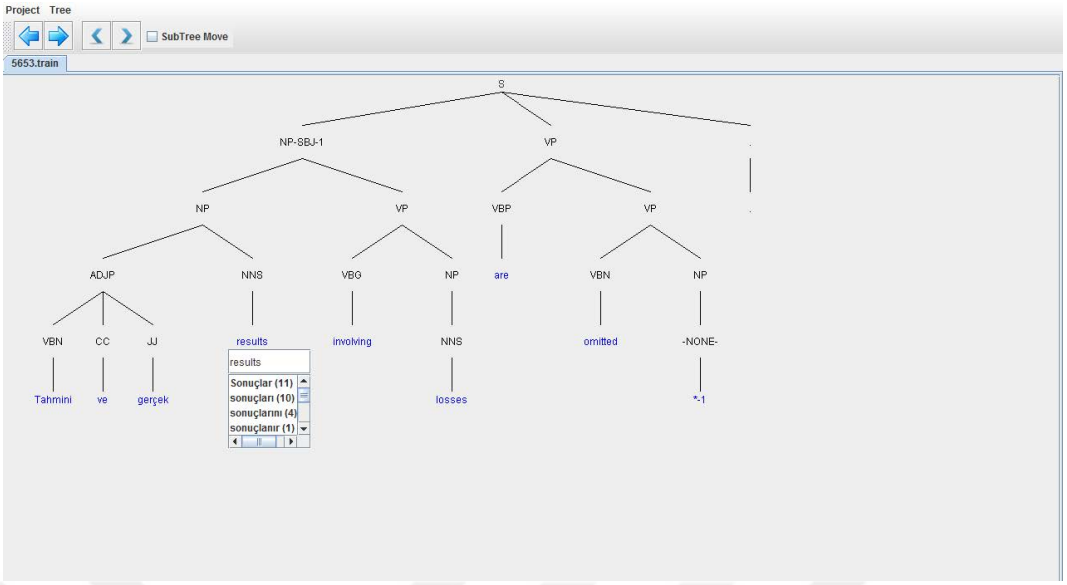

Figure 4.1 A screenshot of Visual Tree Transformation tool.

transitions by suffixation (Oflazer, 1994). We integrate the morphological analyzer into the visual editor (see Figure 4.2) to allow the annotator to list all possible analysis results. Hence, the annotator is able to select the appropriate one from the list. The current version of the morphological analyzer, called Dilbaz (Yıldız et al., 2019) is also part of the NLP Toolkit.

- *Morphological Disambiguator*: Morphological analysis generally yields more than one result for Turkish words. Selecting the correct analysis is identified as another problem to be solved, and the selection process requires expertise in the field. We use the statistical morphological disambiguator (Görgün and Yildiz, 2012) library to help the annotator and link it to a morphological analyzer user interface as option "AutoDisambiguation" so that the process can be automatized.
- *Meta Morpheme Movement*: Meta morphemes are the sublexical structures and source of morphological enrichment of a given word. Once we have the correct morphological analysis, the annotator may need to move these morphemes to their corresponding leaves in the tree so that he or she can fill the gap created by the transformation process. We provide an additional

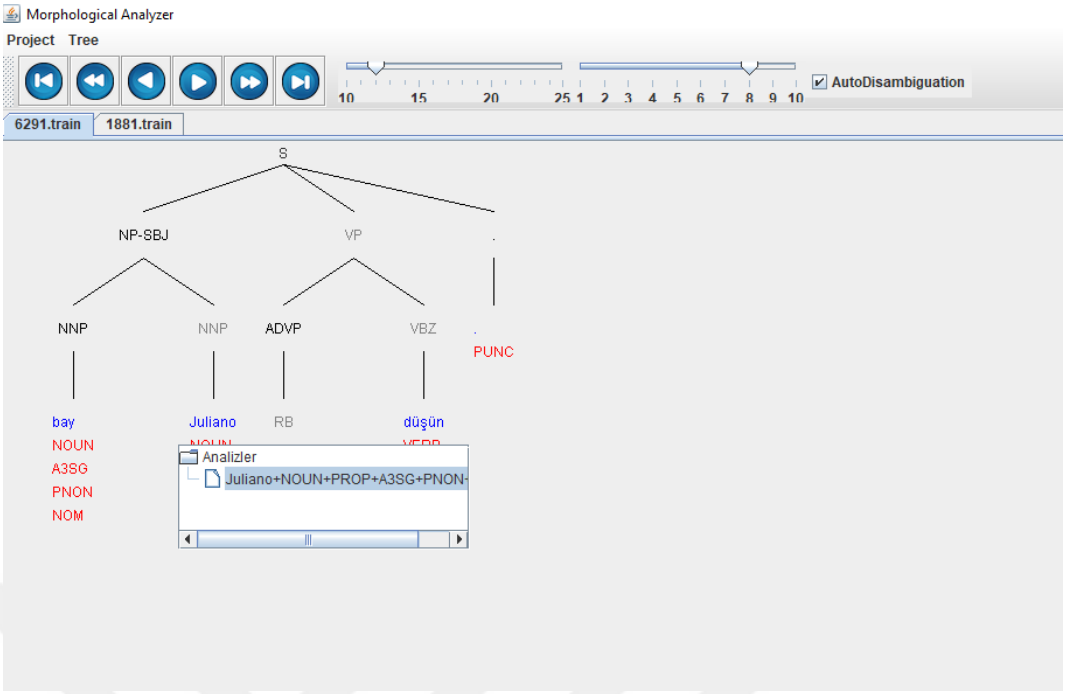

Figure 4.2 A screenshot of Morphological Analyzer tool.

user interface for the user to execute the necessary movement actions (see Figure 4.3).

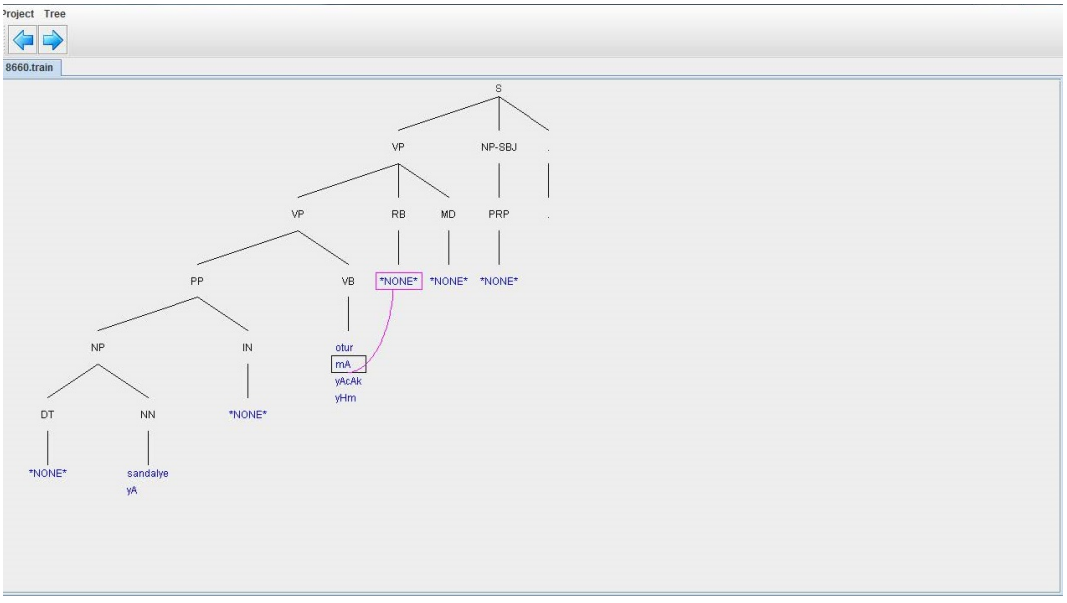

Figure 4.3 A screenshot of Meta Morpheme Movement tool.

4.1.3 Parallel Corpora

Natural language processing research requires a vast amount of linguistic data for different purposes. The data is used for single-language applications such as sentiment analysis or morphological analysis, or multi-language applications such as machine translation. In machine translation cases, parallel data plays an important role in building successful machine translation models. In literature, we notice a variety of language resources for well-known language pairs. However, English and Turkish language pair lacks parallel data.

Syntactic parallel treebanks are one of the most important sources of data for machine translation research. Annotated treebanks can be annotated with constituency or dependency structures. Dependency structures are important to highlight the syntactic and lexical predicate-argument structure. In contrast, constituency treebanks play a much more important role (Chomsky, 1957) to illustrate the structural ambiguities clearly by handling dependencies between multiple nodes. There are various treebank creation efforts for different languages in the literature for both types of treebanks. For well-known and frequently studied languages, studies in both constituency and dependency structures can be found. Turkish NLP research is directed to treebanks with dependency structures due to the rich morphological and complex syntactical nature of Turkish: sentence reordering based on the discourse or determining the constituent syntactic functions (Kornfilt, 1997).

For machine translation tasks, bilingual parallel treebanks are crucial inputs to the translation model. Numerous studies have been conducted to build parallel treebanks both in constituency and dependency structure in the literature:

- ENPC (Oksefjell, 1999): English-Norwegian Parallel Corpus.
- ISJ-ELAN (Erjavec, 2002): Slovene-English Parallel Corpus.
- EuroParl (Koehn, 2002, 2005): One of the largest parallel corporas, including European languages. Parallel data is extracted from the proceedings of the European Parliament.

- FuSe (Cyrus et al., 2003): Parallel treebank for English-German pair with constituency tree structure.
- LinES (Ahrenberg, 2007): English-Swedish parallel treebank in constituency structure.
- Stockholm Treebank (Smultron) (Gustafson-Čapková et al., 2007): Trilingual parallel corpus for English-Swedish-German.
- Prague Trebank (Čmejrek et al., 2005): Czech-English parallel treebank with dependency structure.
- Prague-English Treebank (Hajič et al., 2012): Parallel treebank build on Penn Treebank - Wall Street of Journal with dependency structure and semantics labelling.

For well-studied languages like English, the Penn Treebank (Marcus et al., 1993) is the most comprehensive treebank with constituency structure. There are also different treebanks for various languages: French Treebank for romance languages (Abeillé et al., 2003), TIGER treebank (Brants et al., 2002) for German, Penn Treebank versions for Arabic (Maamouri et al., 2004) and Chinese (Xue et al., 2005), and Finnish as an agglutinative language (Haverinen et al., 2014).

For Turkish, there has been an increasing number of monolingual and multilingual studies recently. However, all of these studies are based on the dependency structure. The most important treebank creation efforts are listed as follows:

- METU-Sabancı Treebank (Atalay et al., 2003): The earliest monolingual treebank with dependency structure for Turkish consisting of syntactically and morphologically annotated 10K sentences. METU-Sabancı Treebank has been a subject to various studies approaching the corpus as a source for solving other NLP problems such dependency parsing (Eryiğit and Oflazer, 2006; Eryiğit et al., 2006; Eryiğit et al., 2008; Yuret and Türe, 2006; Riedel et al., 2006; Çakıcı and Baldridge, 2006)
- Phrase-based parallel data for phrase-based translation approaches (Durgar El-Kahlout, 2009; Yeniterzi and Oflazer, 2010)

- Swedish-Turkish parallel treebank (Megyesi et al., 2008): Syntactically (dependency structure) and morphologically annotated bilingual treebank.
- English-Swedish-Turkish parallel treebank (Megyesi et al., 2010): A trilingual treebank with dependency structure which uses Swedish as transient language.
- ParGram Parallel Treebank (Sulger et al., 2013): Multilingual treebank with both dependency and constituency structures covering ten different languages from six different language families.

4.2 Syntactic Tree Transformation

As a contribution of this thesis, we propose a tree-transformation-based annotation schema to create a parallel treebank with constituency structure. To address the lack of a syntactic parser for Turkish, we offer a method based on the original Penn Treebank (Marcus et al., 1993) trees to generate a Turkish Treebank (TPTB). Our methodology is dependent on the original Penn Treebank POS tags, and we do not propose any Turkish-specific POS tags. Consequently, based on the number of nodes and leaves on both sides, the two trees are similar.

We choose around 17,000 sentences of varying lengths from Penn Treebank (see Table 4.1 for corpus statistics). We translate English trees into their Turkish equivalents using a three-step schema (Yıldız et al., 2014). We want to emulate the broad method followed by human translators using the toolset we have developed within the scope of this thesis. Proposed 3-step annotation strategy is described as follows:

- **Gloss Replacement and Reordering** (*Step-1*) : For any given English tree, the annotator does the flat sentence translation. After flat sentence translation, the annotator discovers two aspects: (i) the number of words in English and Turkish sentences differs; and (ii) the order of words in English and Turkish sentences differs. To solve the first issue, the annotator puts the translated Turkish word instead of its English equivalent. For the words which do not have any direct Turkish translation at word level, they are

marked as *NONE*. For the second problem, the annotator is supposed to perform sub-tree movement actions to obtain the correct order in Turkish. The annotator keeps the word order and morphotactics in consideration while performing the reordering. The visual tree editor allows the annotator to do the reordering of subtrees by just moving them left and right at the same level in the tree.

- **Morphological Analysis and Disambiguation** (*Step-2*): After Step-1, Turkish trees have leaves with *NONE* glosses that reveal the morphological gap between English and Turkish. In order to fill this gap, we perform: (i) mophological analysis; (ii) morphological disambiguation. The final morphological analysis result is expressed as the Turkish stem and the Turkish suffixes in their lexical forms.
- **Filling the Morphological Gap** (*Step-3*): As is known, a Turkish word may correspond to more than one English word during the translation. The mismatched English words are appended to the Turkish stem as suffixes and expressed in their lexical forms. The annotator is expected to fill the gap that is created in Step-1 using Step-2 output in lexical forms.

We completed the whole treebank in three phases. Initially, we transformed 5K Penn Treebank sentences in the first phase (Yıldız et al., 2014). The selected trees have 15 tokens at most, including punctuation. In the next phase, we extend the initial treebank to 9.5K sentences, including 15 tokens. We increase the number of sentences in the treebank to 17K, including sentences with up to 50 tokens (Görgün and Yildiz, 2022) (see Table 4.1 for statistics after the 3rd phase).

Table 4.1 Number of sentences by number of tokens in open-domain.

	1 to 15	16 to 25	26 to 40	41 to 50
# of samples	9500	3300	1300	3000

To show how the process works, we try to give a running example of it. Let's assume that the annotator is given the following English sentence as input

in tree format: “All other trademarks will be the property of their respective owners” (see Figure 4.4).

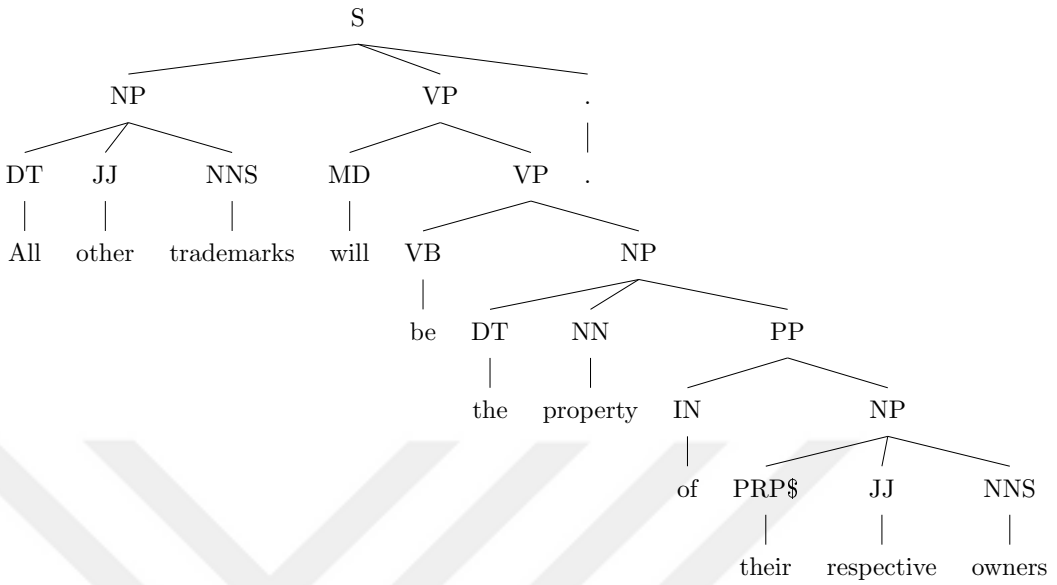

Figure 4.4 A sample English sentence as input

As a first step, the annotator treats the sentence as a flat English sentence and performs the transformation by gloss replacement. For example, the annotator obtains the translated Turkish sentence: “Tüm diğer ticari markalar ol mülk ilgili sahipler” (see Figure 4.5).

Starting from the right-hand side, the annotator checks if the translated words is in their final form or requires any modifications due to the morphological gap between two languages. In Figure 4.6 and 4.7, for noun structure, noun phrase (NP) “their respective owners” is translated to Turkish with two possible suffixations:

- PRP$ (possessive pronoun) “their” to NNS (noun plural) “owners” (PRP$-JJ-NNS) as “sahipler+i” (JJ-NNS-PRP$) (possessive suffix “+i” follows the plural noun);
- IN (preposition) “of” to NNS “their respective owners” as “sahipler+i+nin” (NP-IN) (preposition follows the noun phrase).

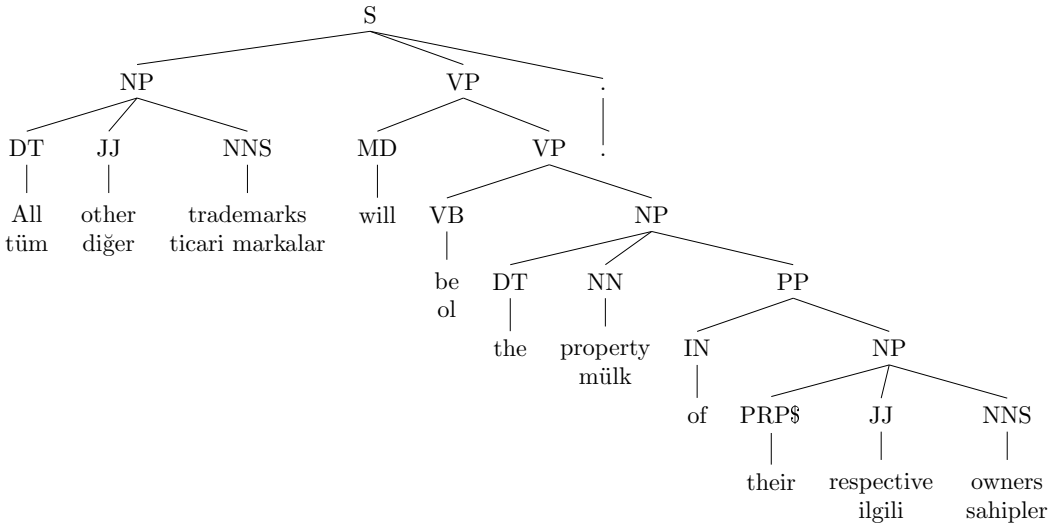

Figure 4.5 A sample English sentence after initial translation

As noticed, if any suffixation is required, it is primarily performed at the same level. (see Figure 4.7).

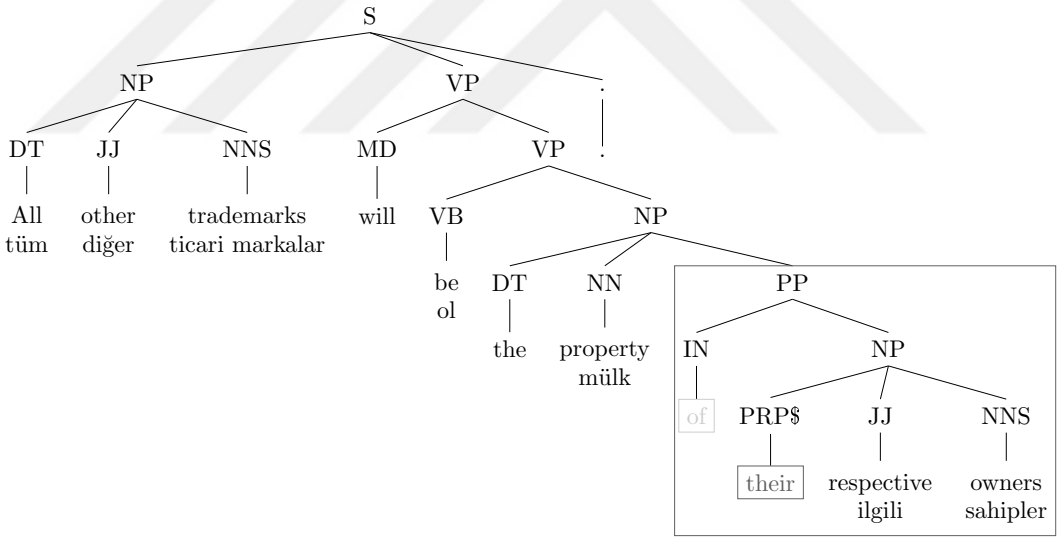

Figure 4.6 Detecting the possible suffixations for the noun phrase.

Once we have the actual gloss with appropriate suffixes in Turkish, the annotator needs to perform the reordering in order to ensure the suffixation order in Turkish tree. In Figure 4.8, the annotator reorder NP children PRP$-JJ-NNS to JJ-NNS-PRP$, and PP children IN-NP to NP-IN. The final word form "sahiplerinin" is set to NNS (see Figure 4.9). English PRP$ and IN leaves do not

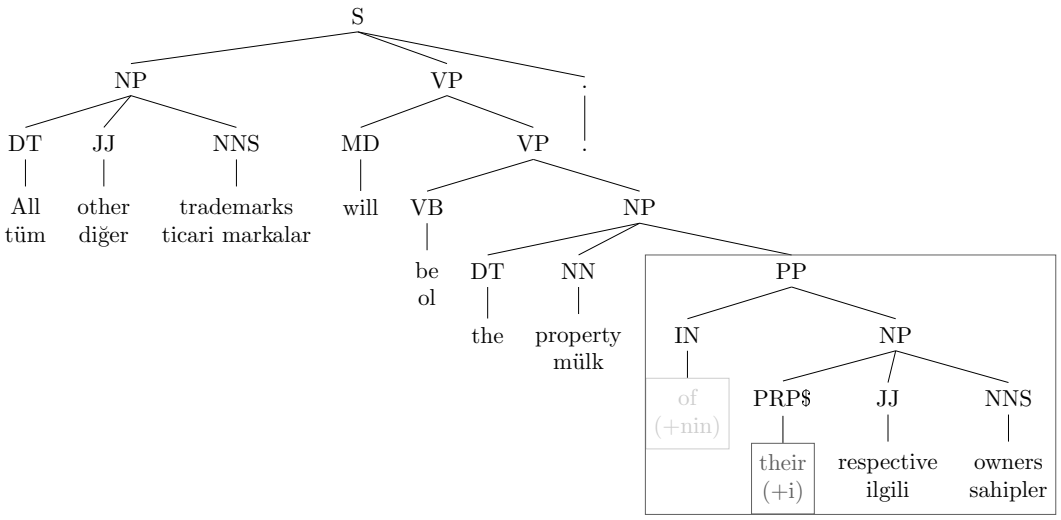

Figure 4.7 Determining the correct suffix for the plural noun (NNS)

have any direct equivalent in Turkish, but they are provided as suffixes to the noun phrase. Therefore, we mark them as *NONE*. In Table 4.2, we present the list of part-of-speech (POS) tags and English words which are marked as *NONE* and in which form they are appended to the corresponding noun.

Table 4.2 English POS tags marked as *NONE* and how they are appended to the NN tag.

POS Tag	Morpheme/Word
('s, VBZ), (is, VBZ)	-DHr
(are, VBP), ('re, VBP)	-DHr
(in, OF), (, POS) -(n)	-nHn
(in, IN), (on, IN), (at, IN)	-DA
(than, IN), (from, IN), (since, IN)	-DAn
(his, *PRP*), (her, *PRP*), (its, *PRP*)	-sH
(our, *PRP*)	-HmHz
(into, IN), (until, IN)	-(n)A
(with, IN), (by, IN)	-(y)lA
(to, TO)	-(y)A
(my, *PRP*)	-Hm
(your, *PRP*)	-Hn
(their, *PRP*)	-lArH

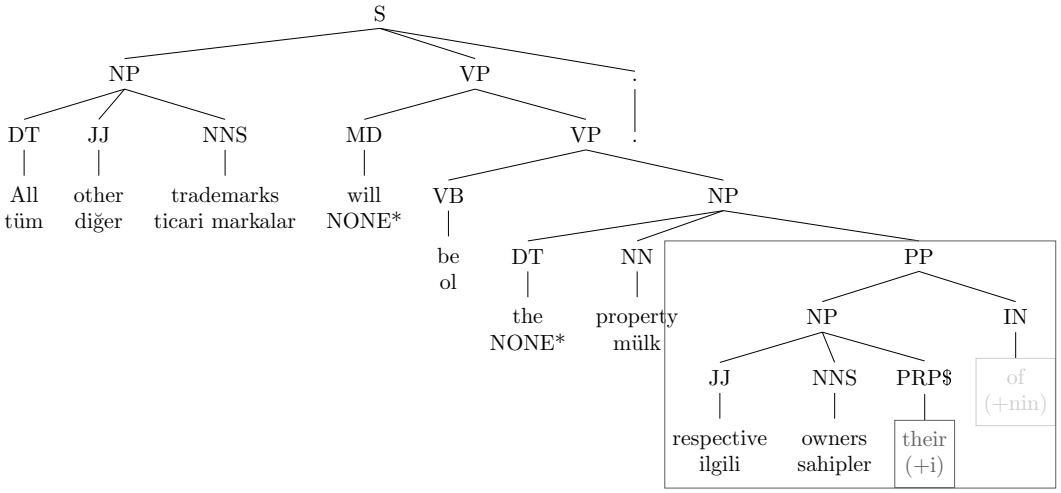

Figure 4.8 Reordering of the leaves and subtrees for the noun phrase (NP) and prepositional phrase(PP).

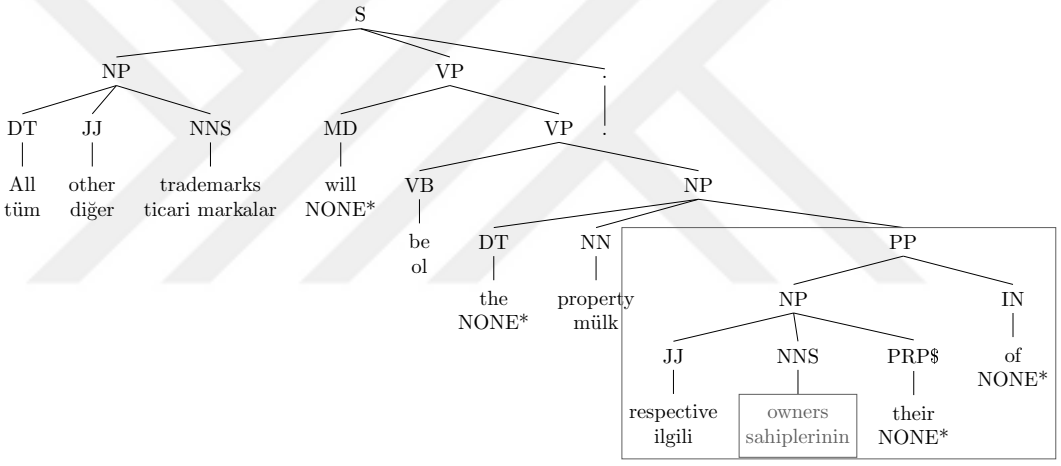

Figure 4.9 Translation for PP and NP subtrees after reordering.

We move to the upper noun phrase NP → DT-NN-PP ("the-property-PP"). One of the prospective translations for the subsentence is "ilgili sahiplerinin mülkü" (see Figure 4.10). In Turkish, determiner (DT) has no direct translation and most of the time it is marked *NONE* in our transformation strategy. However, determiner "the" identifies the noun "property" ("mülk") and is supposed to be added as suffix "ü" to the end of the "mülk" as "mülk+ü" (see Figure 4.11). The annotator preserves the word/suffixation order and reorder the subtree NP as PP-NN-DT ("ilgili sahiplerinin mülkü"), mark the determiner "the" as *NONE*

(see Figure 4.12).

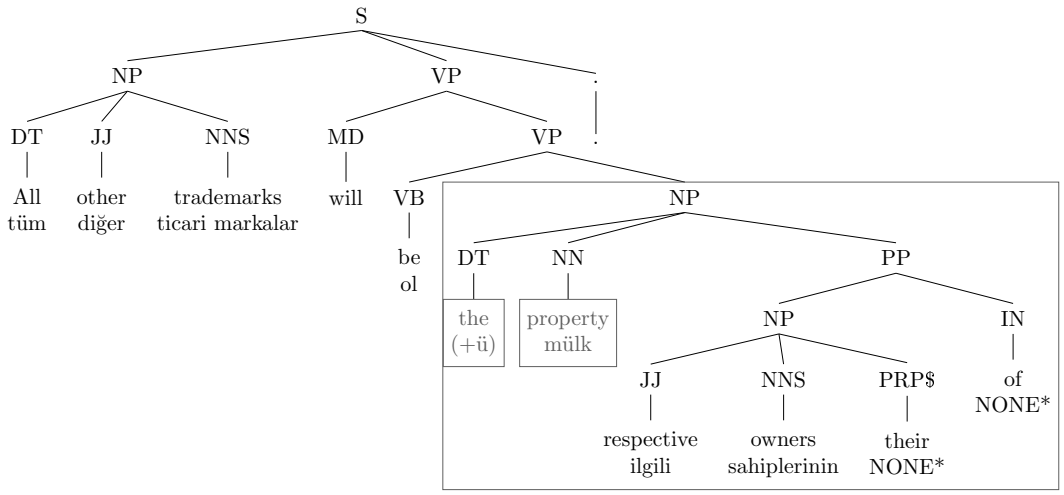

Figure 4.10 Determiner "the" as a suffix for noun phrase.

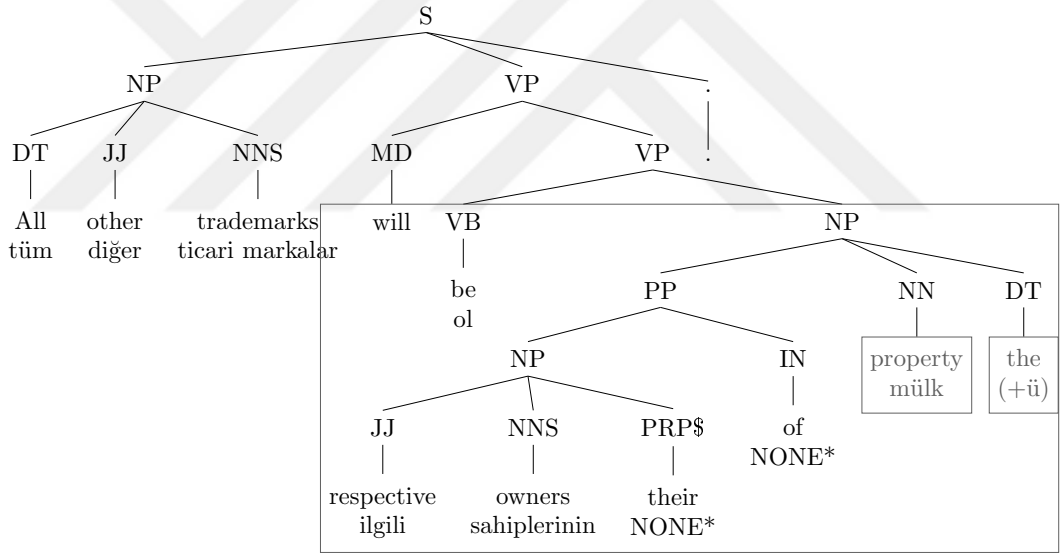

Figure 4.11 Translation for NP after reordering.

In the upper-level, we have the verbal phrase VP → VB-NP ("be-NP") (see Figure 4.13). For verbal structures, we follow the same procedure as we follow for noun structures in terms of suffixation. In Turkish, verb is placed at the end of sentence, if the sentence is not a question sentence or the speaker does not intend to emphasize a specific situation. Since, the sentence given is a regular sentence, verb in base forn (VB) "be" needs to be moved to the end of the sentence. So,

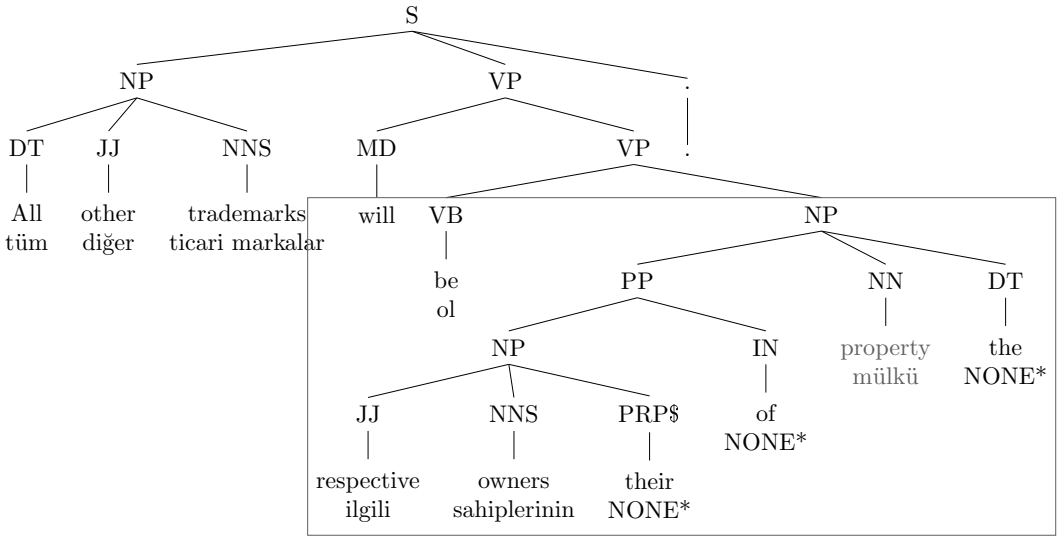

Figure 4.12 Transformation of verbal phrase.

the correct order for the VP → VB-NP is NP-VB ("ilgili sahiplerinin mülkü ol") (see Figure 4.14). In Table 4.3, we present the list of part-of-speech (POS) tags and English words which are marked as *NONE* and in which form they are appended to the corresponding verbal phrase.

Table 4.3 English POS tags marked as *NONE* and how they are appended to the VB tag.

POS Tag	Morpheme/Word
(will, MD)	-(y)AcAk
(will, MD)+(, RB)	-mA + (y)AcAk
(can, MD), (may, MD), (might, MD), (could, MD)	-(y)Abil + Hr
(can, MD)+(, RB), (may, MD)+(, RB), (might, MD)+(, RB), (could, MD)+(, RB)	-mAz
(would, MD), (wo, MD)	-Hyor
(would, MD)+(, RB), (wo, MD)+(, RB)	-m + Hyor
(to, TO)+(have, VB)	-mAlI
(to, TO)+(had, VBD)	-mAlH + (y)DH
(, RB)+(did, VBD)	-mA + DH
(, RB)+(do, VBP), (, RB)+(does, VBZ)	-mAz

Next, we move to the sentence-level (S) verbal phrase VP → MD-VP. Modal (MD) "will" is translated as a suffix "+acak" to Turkish (see Figure 4.15). To preserve the order of suffixation in Turkish (see Figure 4.15), we:

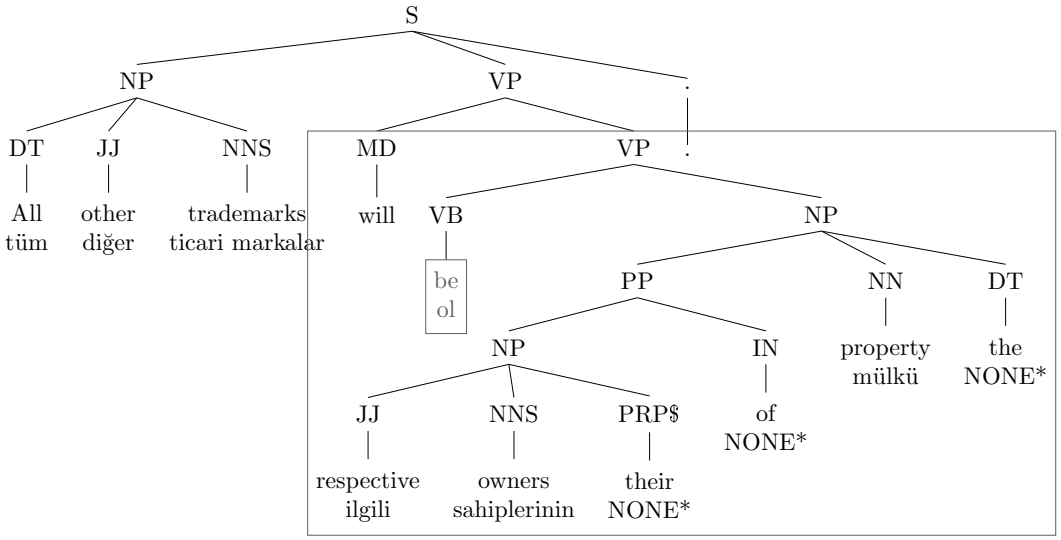

Figure 4.13 Translation for VP after reordering.

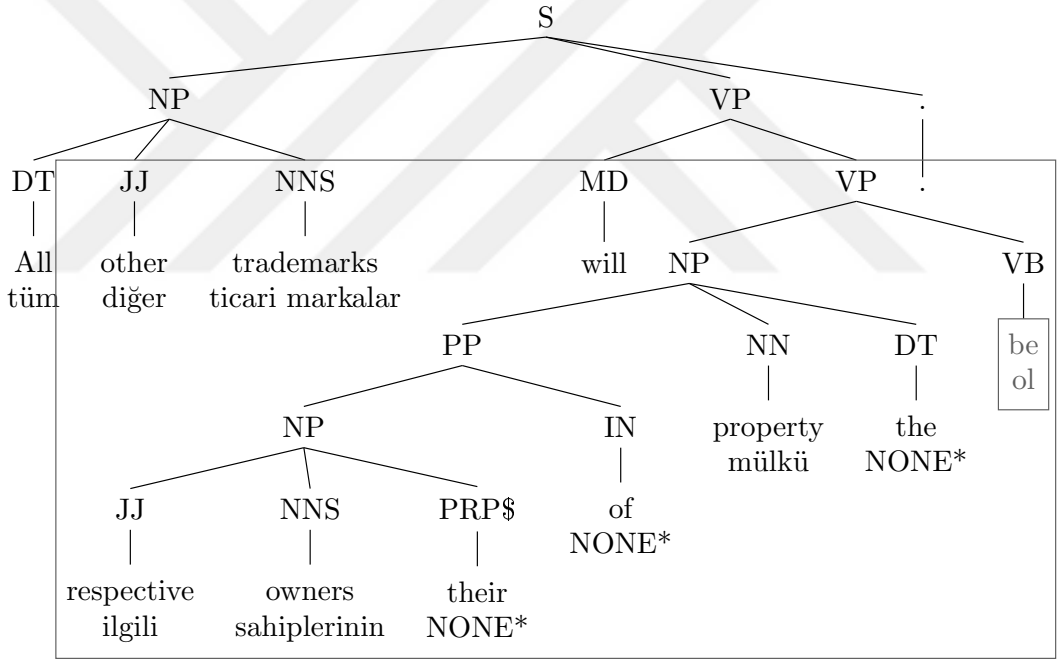

Figure 4.14 The modal "will" as a suffix.

- append the "+acak" to VB "ol" to obtain ol+acak (see Figure 4.16),
- mark model(MD) "will" as *NONE* (see Figure 4.17).

The final word order also preserves the sentence-level ordering of phrases NP-VP, since in Turkish verbals are placed at the end of the sentence.

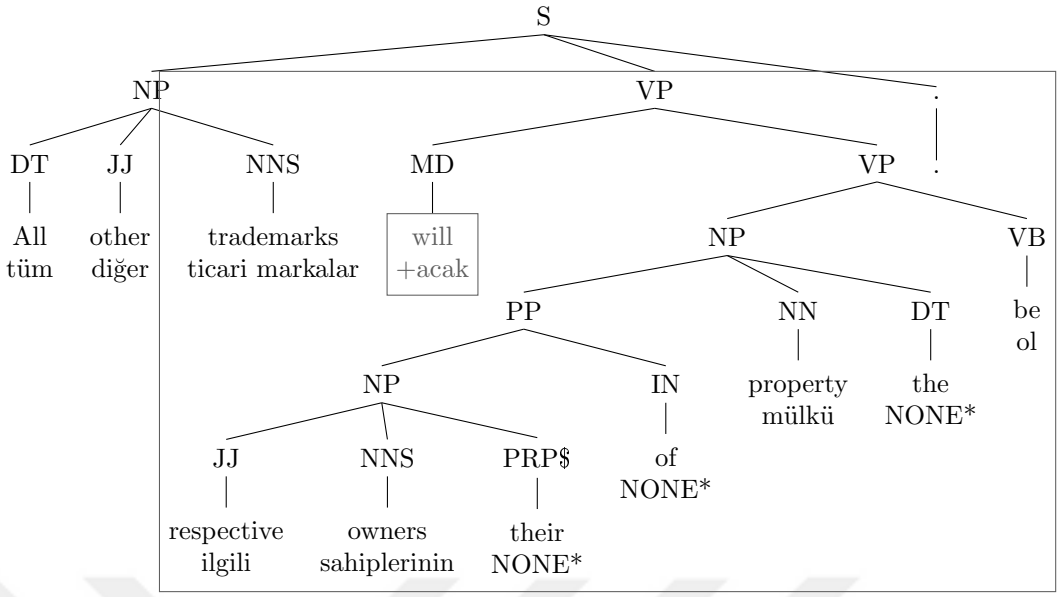

Figure 4.15 Verbal structure after reordering.

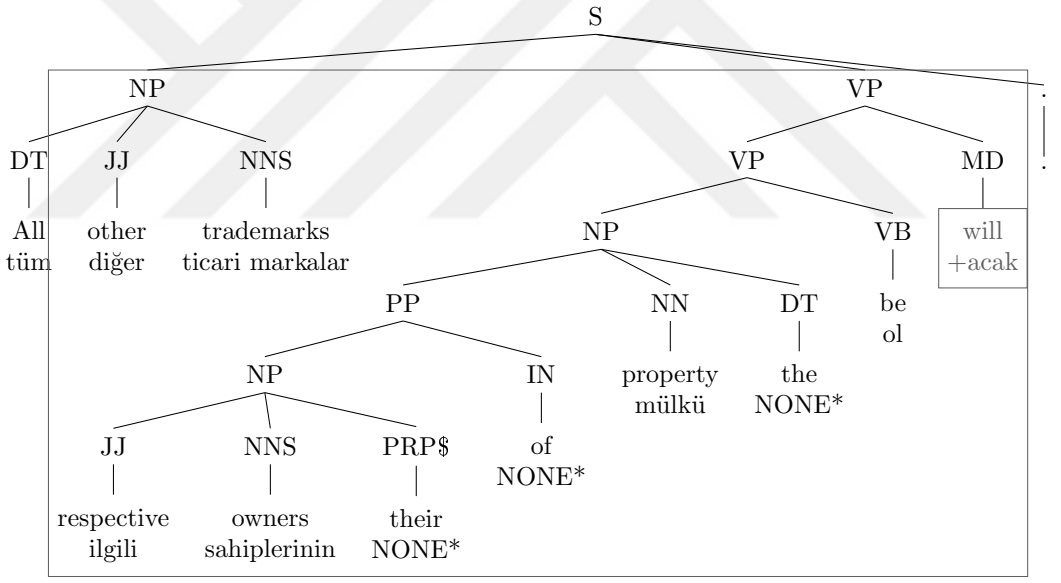

Figure 4.16 The modal "will" as a suffix for VB.

For the noun phrase NP → DT-JJ-NNS ("all-other-trademarks"), the annotator can perform different translations. The annotator is free to do the translation as long as it sounds naturally. For the noun phrase, both "tüm diğer markalar" (see Figure 4.18) or "diğer tüm markalar" (see Figure 4.19) are possible, since we have no visibility on the context. The copular marker ("tır") in

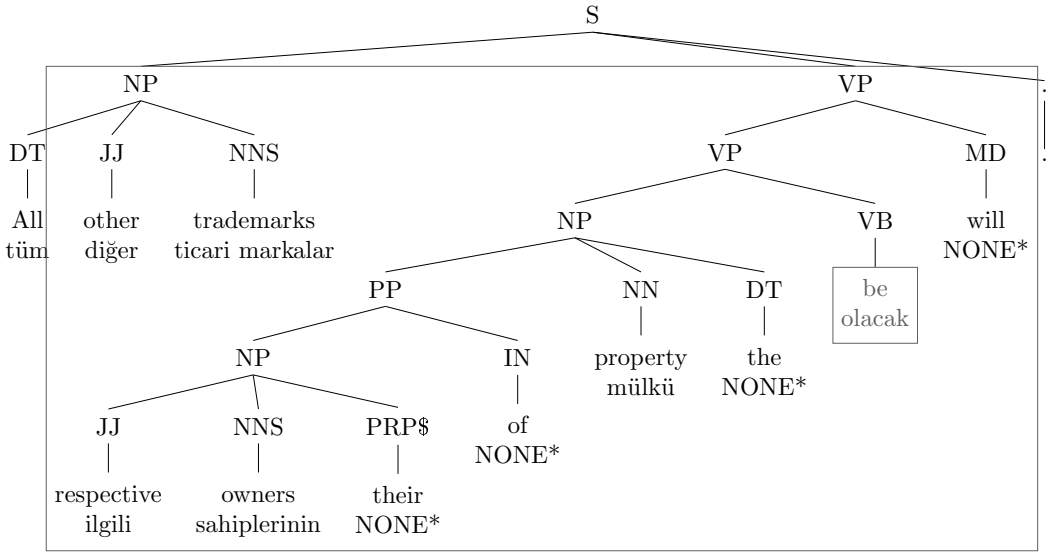

Figure 4.17 Translation of sentence level VP after reordering.

Figure 4.19 is also possible and can be suffixed to the verbal (olacak(tır)). We generally drop the copular marker for simplification.

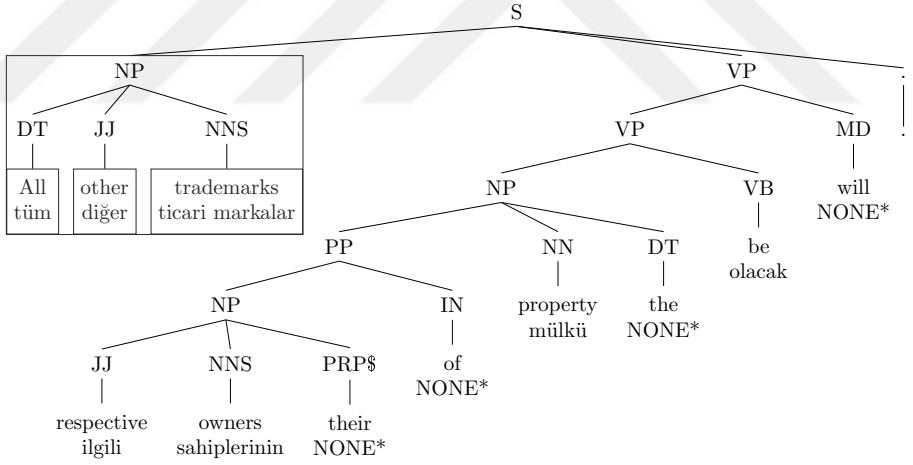

Figure 4.18 Alternative translation for noun phrase ("all other trademarks").

After the last reodering, we obtain the final translated sentence "Diğer tüm ticari markalar ilgili sahiplerinin mülkü olacak" in Figure 4.19. According to the procedure we have defined, we completed Step 1. In the next step, we do morphological analysis and disambiguation to get the right analysis for the translated

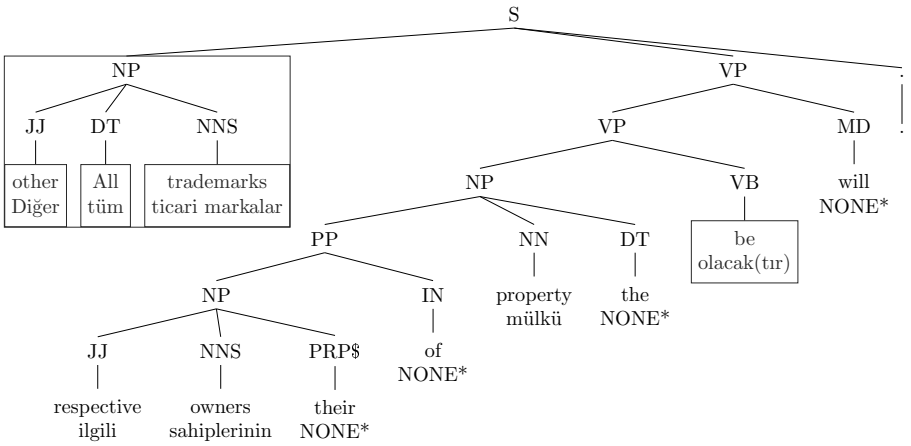

Figure 4.19 Alternative translation for noun phrase ("all other trademarks") and the role of copular marker "+DHr".

glosses. In fact, the annotator is expected to do this analysis during the leaf/-subtree reordering. However, this analysis generally is done in surface level. For statistical purposes, we provide the analysis results in lexical form. For example, in Figure 4.20, you see the result fo morphological anaylsis and disambiguation, and suffixes are provided in their lexical forms (e.g. (y)AcAk, (y)H).

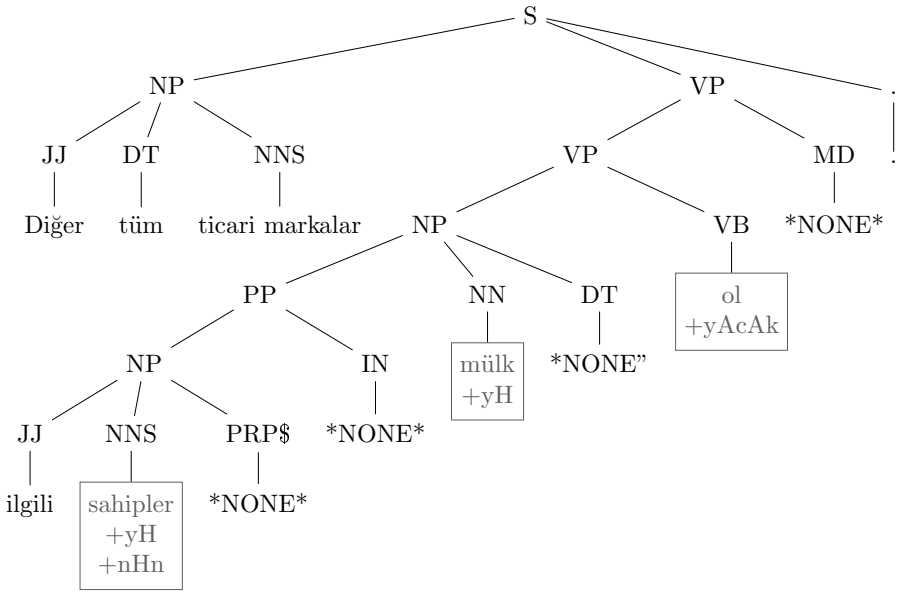

Figure 4.20 Transfomed Turkish tree after morphological anaylsis.

The purpose of the morphological analysis step is to identify the potential morphemes to fill the gaps, *NONE* leaves, that are created during Step-1 in transformation process. In Figure 4.4, we provide the statistics about total *NONE* leave count with the top-3 morphemes replaced with those *NONE* leaves. As we preserve the suffixation order during the transformation, we just move the morphemes to their corresponding positions (see Figure 4.21) and obtain the resulting tree in Figure 4.22.

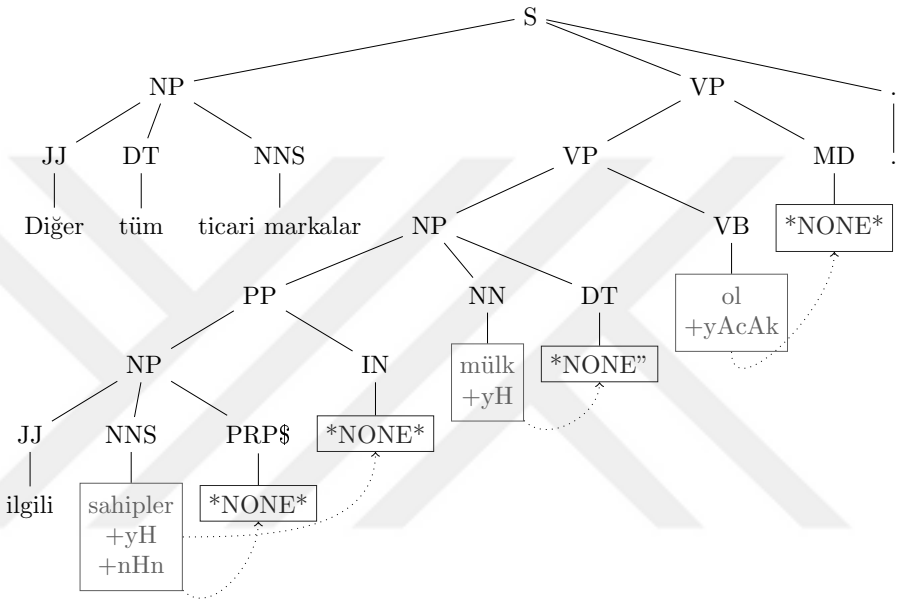

Figure 4.21 Filling the gaps with Turkish suffixes.

Our strategy for transformation is almost applicable to all varieties of sentences, including question sentences with a yes-or-no answer choice. The structure of the Penn Treebank WH questions, on the other hand, represents the one and only exception that prevents us from applying our standard method of transformation. At the beginning of each sentence in a WH question, you will find a question word, such as “where” or “what.” This is the defining characteristic of the WH question format. Figure 4.23(a) shows that the annotator is unable to determine a location that is appropriate for the WH-word WHADVP-1, unless it performs the subtree movement between tree levels, which is something that we do not permit. It would appear that the translation is possible without

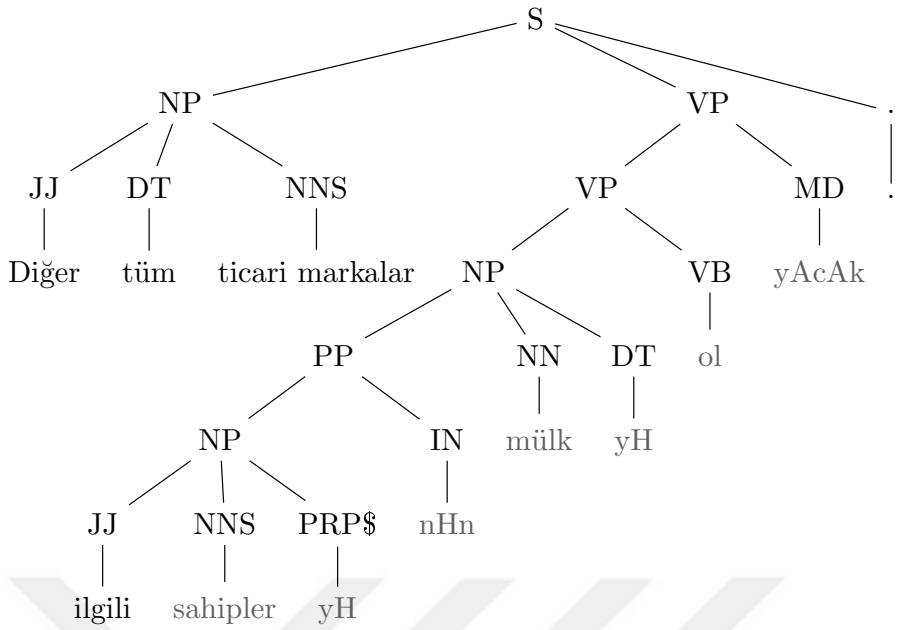

Figure 4.22 The final transformed Turkish tree.

compromising the original word order. However, the syntactic parsing done by Penn Treebank leaves a trace, which is denoted by the notation *T*-1 leave with *NONE* gloss. This notation enables the annotator to put the WH-word in the position so that the word order is maintained (see Figure 4.23(b)).

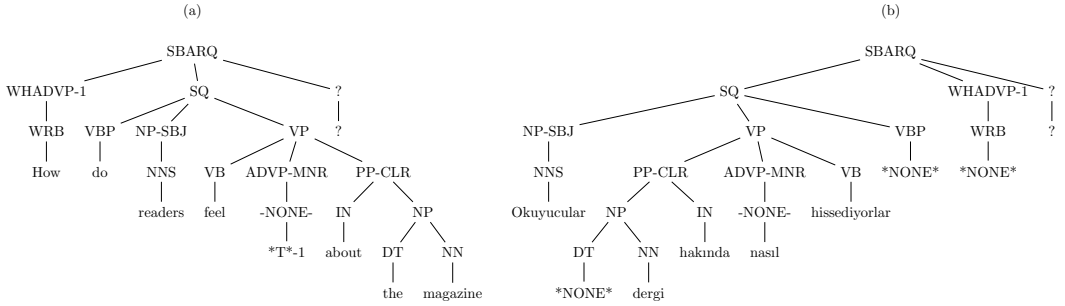

Figure 4.23 A sample WH-question sentence and translation in Turkish.

Besides the WH-question transformation approach, we also treat the compund words different. The annotator is able to translate any English word to multiple Turkish words, if it is required. For example, English VB “regain” is translated to a single word “yeniden-kazanmak” in Turkish.

Table 4.4 Statistics from Turkish corpus: *NONE* leaves and POS tags replaced with top-3 morphemes.

POS Tag	Morpheme	Total
DT	+sH (1)	4783
EX	N/A	110
IN	+DA (878), +nHn (735), +DAn (420)	4736
JJ	+yAbil (1), +yAmA (1), +yAmA+Hyor (1)	58
MD	+yAcAk (317), +yAbil+Hr (192), +yAcAk+DH (39)	1040
POS	+nHn (469), +Hn (12), +DAn (6)	701
PRP	+lAr (116), +yHm (80), +yHz (65)	1598
PRP$	+sH (100), +sH+nH (81), +lArH+nH (46)	562
RB	+mA (389), +yAmA (52), +yAmA+DH (15)	871
TO	+yA (715), +nA (109), +mAk (75)	1578
VB	+Hl (57), +n (10), +DH (4)	233
VBD	+yDH (539), +DH (270), +mHs+yDH (27)	1060
VBG	+yAcAk (13), +Hyor (7), +Hl (4)	55
VBN	+mHs (18), +Hyor (13), +Hl (12)	148
VBP	+DHr (179), +Hyor (169),+DH (105)	1027
VBZ	+DHr (643), +Hyor (198), +DH (156)	1701
WDT	+yA (22), +yHncA (1)	77
WP	+yAn (13), +SH (3), +nA (1)	112

4.3 Closed-domain Treebank

This thesis also investigates the applicability of the proposed transformation approach in the closed-domain. We selected the telecommunication domain and initiated the treebank creation process by selecting user manuals and product documentation from domain experts. In contrast to the open-domain annotation process, closed-domain annotation is different in terms of syntactic complexity and vocabulary variety:

- Open-domain sentences are complex in terms of syntax and morphology, but closed-domain sentences is simple and mostly in simple present tense.
- Verbal structures are complex in open-domain, while the closed-domain verbals are mostly in VBG, VBP or VBZ form.
- Vocabulary size in open-domain is diversified, however closed-domain vocabulary is limited to the domain specific terms.

- Syntactic and morphological expertise is extremely needed during the open-domain annotation process, but closed-domain annotation process mostly relies on the domain knowledge in terms of vocabulary.
- Sentences in open-domain are mostly full sentences, but closed-domain has subsentences and phrases in considerable amount.

We have selected 8.3K sentences out of pre-selected documentation by domain experts in diffrent lenghts (see Table 4.5). We take the flat sentences and convert them to the tree structure in square bracketed format by utilizing a syntactic parser. We use Stanford parser (Klein and Manning, 2003) for syntactic parsing. The annotation process is performed using the NlpToolkit which we built for the open-domain treebank.

Table 4.5 Number of sentences by number of tokens in closed-domain.

	1 to 15	16 to 25	26 to 40	41 to 50	over 50
# of samples	4928	2357	947	79	16

In the scope of closed-domain treebank translation efforts, an application is also developed that is capable of creating a translation model by making use of both existing and candidate collections, making use of the existing translation model as input and performing technical translation with this translation model. However, the tool is not publicly available and is used internally. The translation tool contains two modules:

- **Translation Module**: The module has been developed in order to translate the English technical document set into Turkish. The English technical document set is provided as a plain text file (see Figure 4.24). In addition, the translation module accepts a translation model that has been supplied by the user as an input. The English syntactic parsing relies on the integrated parser for processing.
- **Modelling Module**: The module that has been syntactically analyzed, and it has been developed to create a translation model. The module takes

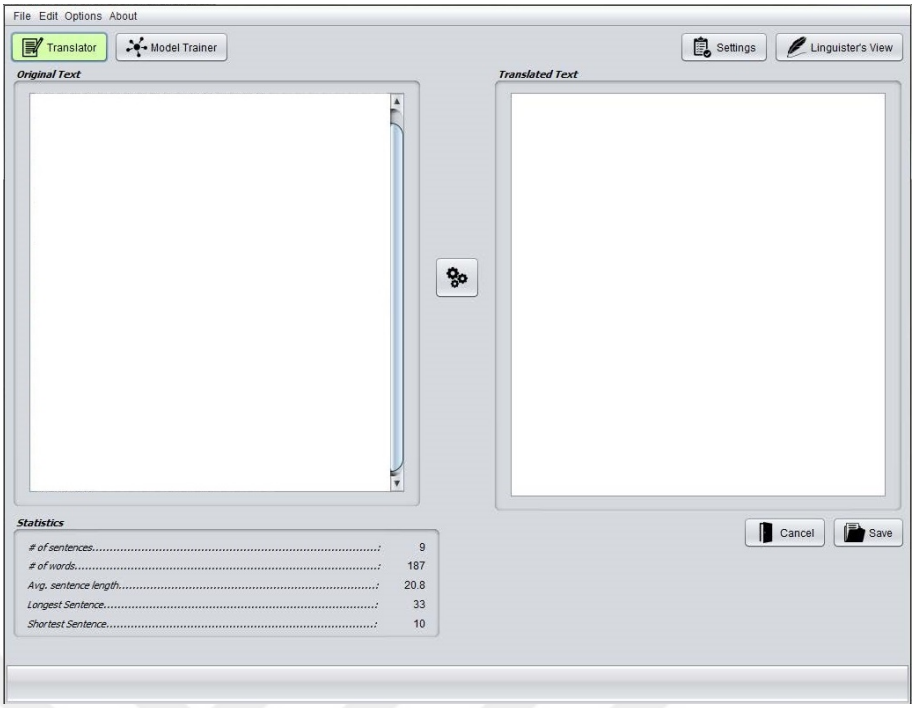

Figure 4.24 Screenshot of translation module.

the morphologically enriched collection as its input, and it has been developed in order to create the model. The user can, in essence, also view the collection's sentences, each of which is composed of syntactic and morphological language components that are denoted by parentheses and are spaced apart from one another.

CHAPTER 5

5. TREEBANK EVALUATION

Within the scope of this thesis, we aim to build an English-Turkish parallel treebank by applying a 3-step transformation process. Moreover, we extend our study to the closed-domain and build another treebank in the telecommunications domain. We also aim to demonstrate that the transformation process is applicable to different domains, but most importantly, that both treebanks are adequate for machine translation task. For this purpose, we perform two types of experiments:

- Treebank fluency check through perplexity analysis for both open-domain and closed-domain treebanks.
- Tree-based statistical machine translation experiments on both open-domain and closed-domain treebanks with different setups.

5.1 Perplexity Analysis

Language models are statistical methods for determining the fluency of language resources such as treebanks. In fact, language model is defined as the power of guessing the next word when it is given the previous words. One can build different language models based on the different number of given previous words (n-gram). The metric shows us how our data resource performs in building syntactically correct and meaningful sentences. Perplexity is an n-gram based (Chen et al., 1998) measure, and it is one of the most widely used evaluation metrics to measure the quality of the language model when unseen data is provided to the model. Perplexity combines the probabilities assigned to n-grams (Jurafsky

and Martin, 2009) and tries to minimize the inverse probability value to zero. Perplexity is formally defined as follows:

- Given the test set $T = \{w_1, ..., w_n\}$, inverse probability of T (5.1) normalized by counts:

$$PP_{p_M} = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P_M(w_i|w_1, ..., w_{i-1})}} \tag{5.1}$$

By applying perplexity analysis, we aim to evaluate the usefulness of both Turkish treebanks without human-judgement (Chang et al., 2009). We follow the given steps below through our evaluation:

1. Use the entire treebanks: 17K sentences from the open-domain treebank and 8.3K sentences from the closed-domain treebank.
2. Flatten the Turkish sentences.
3. Apply k-fold ($k = 10$) cross validation to the open-domain and closed-domain treebanks, and obtain train and test data sets.
4. Build n-gram based language models for $n = 2, 3, 4, 5$.
5. Avoid zero probabilities when an out-of-vocabulary word is given by applying a smoothing technique (Kneser-Ney smoothing (Kneser and Ney, 1995)).

The evaluation results for both open-domain and closed-domain trebanks are given in Table 5.1 and Figure 5.1. Results show that as n increases, both treebanks yield better results. The open-domain treebank performs the best in the 5-gram language model, its performance increases slightly as the n increases. For closed-domain treebank, we observe the same pattern as we see in open-domain, however closed-domain performance increases relatively by 46% between $n = 4$ and $n = 5$. A close-domain treebank is a domain-specific treebank with a large number of long-distance proper noun structures. Therefore, 5-gram results for closed-domain treebank are reasonable. Open-domain treebank, on the other hand, suffers from complex morphotactics at the word level.

Table 5.1 The results of fluency analysis for open-domain and closed-domain perplexity scores.

	Open-domain	**Closed-domain**
2-gram	570.12	585.39
3-gram	539.28	524.53
4-gram	509.64	507.01
5-gram	508.74	273.74

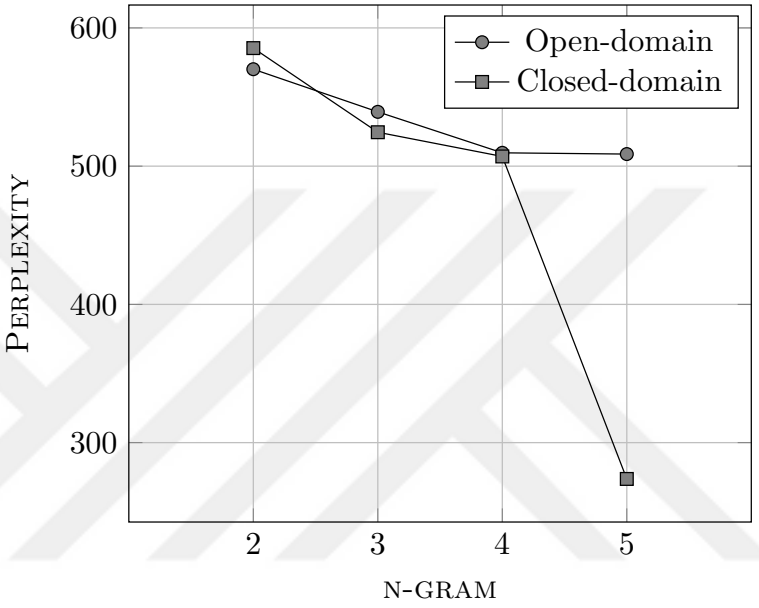

Figure 5.1 Perplexity score graph for open-domain and closed-domain.

5.2 Tree-based Statistical Machine Translation

In this set of experiments, we employ both treebanks in machine translation task and measure the performance of both treebanks. Machine translation task performance also constitutes the basic motivation of the thesis. We set up different experiments on the open-domain and closed-domain treebanks.

We propose a straightforward statistical machine translation schema based on the permutation and gloss replacement probabilities. Next, we report our findings in terms of BLEU scores for each setup.

5.2.1 Translation Approach

We propose a three-phased machine translation schema (Görgün et al., 2016): (i) compute the tree permutation probability; (ii) compute the gloss replacement probability; and (iii) maximize the combination of tree permutation and gloss replacement probabilities.

We have different trees in English and Turkish treebanks due to the treebank transformation step. We aim to extract the statistics on how the tree order for subtrees changes after the tree transformation. As the tree transformation phase is limited to reordering and gloss replacement, we have the same tree in Turkish with a different gloss order. Therefore, we iterate all the trees in both treebanks and count the permutations of trees in both English and Turkish treebanks. For example, we present a snapshot of permutations for the tree in Figure 4.4 in Table 5.2: the permutation in the English tree and its Turkish equivalent in the given order.

Table 5.2 Tree permutation rules extracted from tree in Figure 4.4.

rule	**permutation**
S → NP VP	(0,1)
NP → DT JJ NNS	(1,0,2)
VP → MD VP	(1,0)
VP → VB NP	(1,0)
NP → DT NN PP	(2,1,0)
PP → IN NP	(1,0)
NP → PRP$ JJ NNS	(2,0,1)

We count the permutation rules and assign probabilities to each permutation rule by Equation 5.2.

$$p_{\Pi_i} = \frac{c_{\pi_i}}{\sum_i c_{\pi_i}} \tag{5.2}$$

Next, we follow a very similar schema to extract statistics for gloss replacements in both treebanks. We iterate through the entire parallel treebank and count the number of times any word in English and its Turkish translation have

the same POS tag. We calculate the probabilities for each gloss replacement by Equation 5.3.

We use a simple relative count of occurrences to assign probabilities to possible replacements of leaves in an English tree. So, for a Turkish word t and an English word e, the probability of t replacing e is calculated as follows:

$$p_e(t) = \frac{c_e(t)}{c_e} \tag{5.3}$$

where $c_e(t)$ denotes the number of time e is translated as t and c_e presents the number of occurrences e in the treebank. While English gloss are words or functional words, Turkish gloss can be words or morphemes in lexical form. So, in the gloss replacement step, we calculate the probabilities for the combination of surface-level words and lexical-level morphemes in the Turkish side.

According to our translation schema, the total probability score for the translation candidate is calculated by the multiplication of two components; tree permutation and gloss replacement probabilities, respectively. For any given tree, we calculate both probabilities and multiply them. For the sample sentence in Figure 4.4, we calculate the tree permutation probability by multiplying the probabilities of the grammar rules given in Table 5.2.

Once, we selected the best tree permutation, we iterate the tree by leaves from the left to the right, and start replacing the gloss based on the probability scores. The gloss replacement step is computationally intensive phase and we try to optimize it by keeping the N-best list. Assuming that we are given the best tree permutation for the translation, we follow the logic for gloss replacement:

- Iterate through the leaves from left-to-right,
    - If the current gloss is the first word in the sentence:
        - Calculate the N-best gloss replacement for the current English gloss.
        - Enqueue them to the N-best list.

- Otherwise, calculate the N-best gloss replacement for the current English gloss. Iterate through the N-best translation list and gloss replacement list:
  - Attach the gloss candidate to the candidate translation.
  - Prune the N-best translation list by the N-best partial translations.
  - If the gloss candidate is morpheme, apply the Turkish morphotactics to eliminate impossible transitions.

The translation logic yields the best translation out of the N-best list based on the probability score.

5.2.2 Translation Results

We have prepared and run different test setups for machine translation task in open-domain. We can divide the machine translation experiments into 3-phases: (i) initial experiments with open-domain treebank of 5K sentences; (ii) experiments with open-domain treebank in publicly available translation systems; (ii) experiments with the closed domain treebank. In contrast, we have conducted limited experiments with closed-domain in translation task using all data.

In the first phase (OD-Baseline-BestTree), we executed the experiments in open-domain with limited data of 5K sentences. We apply k-fold ($k = 10$) cross validation and created our train and test data sets. In decoding step, we have used n-best lists in different sizes (1,5, 10, and 50). As result, we report the mean scores and standard deviations (see Table 5.3). The best result we have obtained is 12.8 where the n-best list size is 50.

Table 5.3 The best tree BLEU scores using initial data set for different n-best list size.

1	5	10	50
9.5 ∓ 0.4	11.5 ∓ 0.2	12.1 ∓ 0.3	12.8 ∓ 0.5

We apply a translation approach which contains to components, and report the product of these components as our result. As the next step, we aimed to measure the effect of each individual component (OD-Baseline-OptimalTree). Therefore, we assume that we have the correct tree (optimal tree) permutation and try to measure our performance on gloss replacement. Table 5.4 shows that we obtained BLEU score of 16.3 where $n = 50$, as well. Even though we gained 27.3% relative performance improvement, this reveals the first evidence that we lack insufficient treebank size and dictionary in the gloss replacement step.

Table 5.4 The optimal tree BLEU scores using initial data set for different n-best list size.

1	5	10	50
10.8 ∓ 0.1	13.7 ∓ 0.3	14.5 ∓ 0.2	16.3 ∓ 0.5

In the second phase (OD-PublicMT), we executed experiments on our treebank with publicly available machine translation service, Google Translate, and presented our translation results in terms of BLEU score. According to the results, we achieved 11.6 ∓ 1.0 BLEU score (Görgün et al., 2016). The Google Translate results show similar patterns as we did in Phase-1, despite having large variance. We did not compare our results with the most recent results from Google Translate. Google Translate moved from phrase-based translation systems to Neural Machine Translation. Therefore, for integrity, we decided to compare the results from a phrase-based system.

For the last set of experiments, we employed our closed-domain corpus in machine translation task (Görgün and Yildiz, 2022). We followed the same schema as we did for the open-domain. In contrast, we also extracted the translation dictionary as a domain dictionary. The domain dictionary contains automatically extracted words based on the transformation results. We also corrected the translated trees again manually by correcting the domain-specific terms and proper nouns. We got a BLEU score of 26.8 (optimal tree) so far in closed-domain, because the corpus is not very complex in terms of morphology and

is dictionary-dependent. Table 5.5 summarizes all experimental results for the machine translation task.

Table 5.5 Summary of the machine translation results.

Experiment	**BLEU**
OD-Baseline-BestTree	12.8
OD-Baseline-OptimalTree	16.3
OD-PublicMT	11.6
Closed Domain	26.8

CHAPTER 6

6. CONCLUSION

This thesis discussed the creation of an English-Turkish parallel treebank as well as its application in statistical machine translation task. We created a treebank of 17K sentences in open-domain by transforming the Penn Treebank sentences based on a 3-step annotation process. We applied the same tree transformation approach by building an English-Turkish closed-domain parallel treebank of 8.3K in the telecommunication domain. Besides tree transformation, we also formed a domain dictionary for the selected sub-domain, especially for technical gloss substitution. Unlike the open-domain treebank, the closed-domain treebank has very limited syntactic variety and a limited vocabulary. We also evaluated the closed-domain both intrinsically in terms of perplexity and extrinsically in machine translation tasks as well.

It should not come as a surprise that the BLEU scores for initial attempts are quite low. However, the results still seems to be quite promising even with limited data. The best BLEU score of 12.8 in open-domain that we obtained on unseen data is encouraging. We observed that our translation model is successful to capture the correct permutation in leaf order, but it suffers from the gloss replacement. It is clear that an additional dictionary is needed in order to translate the unseen words. For closed-domain translation, we measured the BLEU score of 26.8 which is the best score that we have obtained so far in limited domain.

Obviously, the annotation schema that we proposed is the only option in order to obtain a Turkish constituency treebank, since there is no constituency parser implemented for Turkish. This limitation also leads us to manual work to

annotate the corpus. As a result, we are also limited to one way translation from English to Turkish. In addition to working with limited data, the constraints, such as making structural changes in constituency tree, we put forward to facilitate the annotation step also indirectly affected the translation task performance. In addition, the structural constraints we propose in the tree transformation process make it difficult to translate specific language structures such idiomatic expressions, multi-word expressions.

Although we have created various software to facilitate the annotation process, we have observed that even experienced annotators make mistakes in this error-prone process. Another finding is that the errors occur especially in the morphological disambiguation step. All these errors, which were noticed in the next steps, required going back to the translation step according to the severity of the error. On the bright side, these feedbacks not only slowed down the process, but also increased the translation quality and enabled us to create a valuable treebank for further Turkish NLP studies.

As a future work, we plan to improve the shortcomings we have found. The improvement points are categorized into three: (i) improve the translation results in open-domain by expanding the translation dictionary for gloss replacement; (ii) focus on other intrinsic evaluation methods; (iii) invest on the tool development to improve the annotation and to extend the corpus. We plan to invest more on the constituency parser development, and believe that will be a great contribution to Turkish NLP studies.

REFERENCES

Abeillé, A., Clément, L., and Toussenel, F. (2003). *Building a Treebank for French*, pages 165–187. Springer Netherlands, Dordrecht.

Ahrenberg, L. (2007). LinES: An English-Swedish parallel treebank. In *Proceedings of the 16th Nordic Conference of Computational Linguistics (NODALIDA 2007)*, pages 270–273, Tartu, Estonia. University of Tartu, Estonia.

Atalay, N. B., Oflazer, K., and Say, B. (2003). The annotation process in the Turkish treebank. In *Proceedings of 4th International Workshop on Linguistically Interpreted Corpora (LINC-03) at EACL 2003*, pages 33–38, Budapest, Hungary.

Brants, S., Dipper, S., Hansen, S., Lezius, W., and Smith, G. (2002). The TIGER treebank. In *Proceedings of the workshop on treebanks and linguistic theories*, pages 24–41.

Brown, P. F., Lai, J. C., and Mercer, R. L. (1991). Aligning sentences in parallel corpora. In *29th Annual Meeting of the Association for Computational Linguistics*, pages 169–176, Berkeley, California, USA. Association for Computational Linguistics.

Çakıcı, R. and Baldridge, J. (2006). Projective and non-projective Turkish parsing. In *In Proceedings of the 5th International Treebanks and Linguistic Theories Conference*, pages 43–54.

Chang, J., Boyd-Graber, J. L., Gerrish, S., Wang, C., and Blei, D. M. (2009). Reading tea leaves: How humans interpret topic models. In Bengio, Y.,

Schuurmans, D., Lafferty, J. D., Williams, C. K. I., and Culotta, A., editors, *Neural Information Processing Systems.*, pages 288–296. Curran Associates, Inc.

Chen, S. F., Beeferman, D., and Rosenfeld, R. (1998). Evaluation metrics for language models. In *Proceedings of DARPA Broadcast News Transcription and Understanding Workshop*, pages 275–280, Lansdowne, VA, USA.

Chiang, D. (2005). A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd annual meeting of the association for computational linguistics (acl'05)*, pages 263–270.

Chomsky, N. (1957). *Syntactic Structures*. Mouton and Co., The Hague.

Church, K. W. (1993). Char_align: A program for aligning parallel texts at the character level. In *31st Annual Meeting of the Association for Computational Linguistics*, pages 1–8, Columbus, Ohio, USA. Association for Computational Linguistics.

Čmejrek, M., Cuřín, J., Hajič, J., and Havelka, J. (2005). Prague Czech-English dependency treebank: resource for structure-based mt. In *Proceedings of the 10th EAMT Conference: Practical applications of machine translation*, pages 73–78, Budapest, Hungary. European Association for Machine Translation.

Cyrus, L., Feddes, H., and Schumacher, F. (2003). Fuse – a multi-layered parallel treebank. In Nivre, J. and Hinrichs, E., editors, *Proceedings of the Second Workshop on Treebanks and Linguistic Theories*, pages 213–216, Växjö, Sweden. Växjö University Press.

DellaPietra, S. and DellaPietra, V. (1994). Candide: A statistical machine translation system. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994.*

Durgar El-Kahlout, İ. (2009). *A prototype English Turkish statistical machine translation system*. PhD thesis, Sabancı University.

Enright, J. A. and Kondrak, G. (2007). A fast method for parallel document identification. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pages 29–32. The Association for Computational Linguistics.

Erjavec, T. (2002). The IJS-ELAN Slovene-English parallel corpus. *International Journal of Corpus Linguistics*, 7:1–20.

Eryiğit, G., Nivre, J., and Oflazer, K. (2008). Dependency parsing of Turkish. *Computational Linguistics*, 34(3):357–389.

Eryiğit, G. and Oflazer, K. (2006). Statistical dependency parsing for Turkish. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 89–96, Trento, Italy. Association for Computational Linguistics.

Eryiğit, G., Adalı, E., and Oflazer, K. (2006). Türkçe cümlelerin kural tabanlı bağlılık analizi. In *15th Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN 2006)*, pages 17–24, Muğla, Turkey.

Gale, W. A. and Church, K. W. (1991). A program for aligning sentences in bilingual corpora. In *29th Annual Meeting of the Association for Computational Linguistics*, pages 177–184, Berkeley, California, USA. Association for Computational Linguistics.

Gale, W. A. and Church, K. W. (1993). A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–102.

Görgün, O. and Yildiz, O. T. (2012). A novel approach to morphological disambiguation for Turkish. In Gelenbe, E., Lent, R., and Sakellari, G., editors,

Computer and Information Sciences II, pages 77–83, London. Springer London.

Gustafson-Čapková, S., Samuelsson, Y., and Volk, M. (2007). Smultron (version 1.0) - the Stockholm MULtilingual parallel TReebank. an English-German-Swedish parallel treebank with subsentential alignment.

Görgün, O. and Yildiz, O. T. (2022). Evaluating the English-Turkish parallel treebank for machine translation. *Turkish Journal of Electrical Engineering and Computer Sciences*, 30:184.

Görgün, O., Yildiz, O. T., Solak, E., and Ehsani, R. (2016). English-Turkish parallel treebank with morphological annotations and its use in tree-based SMT. In Marsico, M. D., di Baja, G. S., and Fred, A. L. N., editors, *Proceedings of the 5th International Conference on Pattern Recognition Applications and Methods, ICPRAM 2016, Rome, Italy, February 24-26, 2016*, pages 510–516. SciTePress.

Hajič, J., Hajičová, E., Panevová, J., Sgall, P., Bojar, O., Cinková, S., Fučíková, E., Mikulová, M., Pajas, P., Popelka, J., Semecký, J., Šindlerová, J., Štěpánek, J., Toman, J., Urešová, Z., and Žabokrtský, Z. (2012). Announcing Prague Czech-English dependency treebank 2.0. In *Proceedings of the 8th International Conference on Language Resources and Evaluation (LREC 2012)*, pages 3153–3160, İstanbul, Turkey. ELRA, European Language Resources Association.

Hakkani, D. Z., Tür, G., Oflazer, K., Mitamura, T., and Nyberg, 3rd, E. H. (1998). An English-to-Turkish interlingual MT system. In *Proceedings of the Third Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 83–94, Langhorne, PA, USA. Springer.

Hakkani-Tür, D. Z., Oflazer, K., and Tür, G. (2002). Statistical morphological disambiguation for agglutinative languages. *Computers and the Humanities*,

36(4):381–410.

Haverinen, K., Nyblom, J., Viljanen, T., Laippala, V., Kohonen, S., Missilä, A., Ojala, S., Salakoski, T., and Ginter, F. (2014). Building the essential resources for Finnish: the Turku dependency treebank. *Language Resources and Evaluation*, 48:493–531.

Jurafsky, D. and Martin, J. H. (2009). *Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition*. 2nd Edition, Pearson Prentice Hall, Upper Saddle River, N.J.

Klein, D. and Manning, C. D. (2003). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 423–430, Sapporo, Japan. Association for Computational Linguistics.

Kneser, R. and Ney, H. (1995). Improved backing-off for m-gram language modeling. In *1995 International Conference on Acoustics, Speech, and Signal Processing*, volume 1, pages 181–184.

Koehn, P. (2002). Europarl: A multilingual corpus for evaluation of machine translation. Draft.

Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Koehn, P. (2010). *Statistical machine translation*. Cambridge University Press40 W. 20 St. New York, NYUnited States.

Kornfilt, J. (1997). *Turkish*. Descriptive grammars. Routledge.

Maamouri, M., Bies, A., Buckwalter, T., and Mekki, W. (2004). The penn arabic treebank: Building a large-scale annotated arabic corpus. In *NEMLAR*

conference on Arabic language resources and tools, volume 27, pages 466–467. Cairo.

Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Megyesi, B., Dahlqvist, B., Csató, É. Á., and Nivre, J. (2010). The English-Swedish-Turkish parallel treebank. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 3393–3397, Valletta, Malta. European Language Resources Association (ELRA).

Megyesi, B., Dahlqvist, B., Pettersson, E., and Nivre, J. (2008). Swedish-Turkish parallel treebank. In Chair), N. C. C., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., and Tapias, D., editors, *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, pages 470–473, Marrakech, Morocco. European Language Resources Association (ELRA).

Och, F. J. (2002). *Statistical machine translation : from single-word models to alignment templates*. PhD thesis, RWTH Aachen University, Aachen.

Och, F. J. and Ney, H. (2003). A systematic comparison of various statistical alignment models. *Computational linguistics*, 29(1):19–51.

Och, F. J. and Ney, H. (2004). The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449.

Och, F. J., Tillmann, C., and Ney, H. (1999). Improved alignment models for statistical machine translation. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28.

Och, F. J. and Weber, H. (1998). Improving statistical natural language translation with categories and rules. In *Proceedings of the 36th Annual Meeting of the Association of Computational Linguistics*, pages 985–989. FAU Erlangen - Computer Science Institute.

Oflazer, K. (1994). Two-level description of Turkish morphology. *Literary and linguistic computing*, 9(2):137–148.

Oflazer, K. and Durgar El-Kahlout, İ. (2007). Exploring different representational units in English-to-Turkish statistical machine translation. In *Proceedings of the 2nd Workshop on Statistical Machine Translation*, pages 25–32, Prague, Czech Republic.

Oksefjell, S. (1999). A description of the English-norwegian parallel corpus : Compilation and further developments. *International Journal of Corpus Linguistics*, 4:197–219.

Riedel, S., Çakıcı, R., and Meza-Ruiz, I. (2006). Multi-lingual dependency parsing with incremental integer linear programming. In *Proceedings of the 10th Conference on Computational Natural Language Learning (CoNLL-X)*, pages 226–230, New York City.

Sagay, Z. (1981). *A computer translation of English to Turkish.* PhD thesis, Middle East Technical University, Ankara, Turkey.

Sak, H., Güngör, T., and Saraçlar, M. (2007). Morphological disambiguation of Turkish text with perceptron algorithm. In Gelbukh, A., editor, *Computational Linguistics and Intelligent Text Processing*, pages 107–118, Berlin, Heidelberg. Springer Berlin Heidelberg.

Sulger, S., Butt, M., King, T. H., Meurer, P., Laczkó, T., Rákosi, G., Dione, C. B., Dyvik, H., Rosén, V., De Smedt, K., Patejuk, A., Çetinoğlu, Ö., Arka, I. W., and Mistica, M. (2013). ParGramBank: The ParGram parallel treebank. In *Proceedings of the 51st Annual Meeting of the Association for*

Computational Linguistics (Volume 1: Long Papers), pages 550–560, Sofia, Bulgaria. Association for Computational Linguistics.

Thouin, B. (1981). The meteo system. In *Translating and the Computer: Practical experience of machine translation*, London, UK. Aslib.

Tillmann, C. (2003). A projection extension algorithm for statistical machine translation. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, pages 1–8.

Turhan, C. K. (1997). An English to Turkish machine translation system using structural mapping. In *Fifth Conference on Applied Natural Language Processing*, pages 320–323, Washington, DC, USA. Association for Computational Linguistics.

Täger, W. (2011). The sentence-aligned European patent corpus. In *Proceedings of the 15th Conference of the European Association for Machine Translation*, pages 177–184. European Association for Machine Translation.

Vauquois, B. (1968). A survey of formal grammars and algorithms for recognition and transformation in mechanical translation. In Morrel, A. J. H., editor, *IFIP Congress (2)*, pages 1114–1122.

Vogel, S., Ney, H., and Tillmann, C. (1996). Hmm-based word alignment in statistical translation. In *COLING 1996 Volume 2: The 16th International Conference on Computational Linguistics*, pages 836–841.

Weaver, W. (1947). Letter to Norbert Wiener.

Weaver, W. (1955). Translation. *Machine translation of languages*, 14:15–23.

Xue, N., Xia, F., Chiou, F.-D., and Palmer, M. (2005). The Penn Chinese treebank: Phrase structure annotation of a large corpus. *Natural Language Engineering*, 11(2):207–238.

Yeniterzi, R. and Oflazer, K. (2010). Syntax-to-morphology mapping in factored phrase-based statistical machine translation from English to Turkish. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 454–464, Uppsala, Sweden. Association for Computational Linguistics.

Yıldız, O. T., Avar, B., and Ercan, G. (2019). An open, extendible, and fast Turkish morphological analyzer. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1364–1372, Varna, Bulgaria. INCOMA Ltd.

Yıldız, O. T., Solak, E., Görgün, O., and Ehsani, R. (2014). Constructing a Turkish-English parallel TreeBank. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 112–117, Baltimore, Maryland. Association for Computational Linguistics.

Yüret, D. (2006). Dependency parsing as a classification problem. In *Proceedings of the Tenth Conference on Computational Natural Language Learning (CoNLL-X)*, CoNLL-X '06, page 246–250, USA. Association for Computational Linguistics.

Yuret, D. and Türe, F. (2006). Learning morphological disambiguation rules for Turkish. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 328–334, New York City, USA. Association for Computational Linguistics.

Zhang, D., Li, M., Li, C.-H., and Zhou, M. (2007). Phrase reordering model integrating syntactic knowledge for smt. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 533–540.

CURRICULUM VITAE

He received his B.S. degree in Information Technologies from Işık University and M.Sc. degree in Information Technologies from Işık University. He worked as Research and Teaching Assistant in Information Technologies Department of Işık University and Computer Engineering Department of Işık University. He is currently working as a Solutions Architect for a private company. His research interests are natural language processing, machine learning, and software engineering.