FREQUENCY-DRIVEN LATE FUSION-BASED WORD DECOMPOSITION APPROACH ON THE PHRASE-BASED STATISTICAL MACHINE TRANSLATION SYSTEMS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

MEHMET TATLICIOĞLU

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

OCTOBER 2013

Approval of the thesis:

FREQUENCY-DRIVEN LATE FUSION-BASED WORD DECOMPOSITION APPROACH ON THE PHRASE-BASED STATISTICAL MACHINE TRANSLATION SYSTEMS

submitted by **MEHMET TATLICIOĞLU** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Adnan Yazıcı
Head of Department, **Computer Engineering**

Prof. Dr. Adnan Yazıcı
Supervisor, **Computer Engineering Department, METU**

Examining Committee Members:

Assoc. Prof. Dr. Pınar Karagöz
Department of Computer Engineering, METU

Prof. Dr. Adnan Yazıcı
Department of Computer Engineering, METU

Assist. Prof. Dr. İsmail Sengör Altıngövde
Department of Computer Engineering, METU

Assist. Prof. Dr. Murat Koyuncu
Department of Information Systems Engineering, Atılım University

Dr. Ruket Çakıcı
Department of Computer Engineering, METU

Date:

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: MEHMET TATLICIOĞLU

Signature :

ABSTRACT

FREQUENCY-DRIVEN LATE FUSION-BASED WORD DECOMPOSITION APPROACH ON THE PHRASE-BASED STATISTICAL MACHINE TRANSLATION SYSTEMS

Tatlıcıoğlu, Mehmet
M.S., Department of Computer Engineering
Supervisor : Prof. Dr. Adnan Yazıcı

October 2013, 73 pages

Machine translation is the process of translating texts from a natural language to another by computers based on linguistic motivations, statistical approaches, or the combination of them. In this study, the frequency-driven late fusion-based word decomposition approach is introduced to improve the translation quality of the phrase-based statistical machine translation system from Turkish to English. This late fusion-based approach is compared with the standalone statistical and rule-based word decomposition approaches when the corpus size changes. This study differs from others by introducing the novel frequency-driven late fusion-based word decomposition method to boost the BLEU score. While the benchmark study in the literature reports a 25.22 BLEU score, the proposed late fusion-based system boosts the accuracy up to a 26.22 BLEU score. This novel approach fuses both of the rule-based and stochastic word decomposition methods. Because of the agglutinative nature of Turkish language, the results can be extended to the other agglutinative languages as well.

Keywords: Machine Learning, Natural Language Processing, Machine Translation, Morphology

ÖZ

TÜMCE TABANLI İSTATİSTİKSEL OTOMATİK ÇEVİRİ SİSTEMLERİNDE FREKANSA DAYALI GEÇ BİRLEŞİM TABANLI KELİME PARÇALAMA YAKLAŞIMI

Tatlıcıoğlu, Mehmet
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi : Prof. Dr. Adnan Yazıcı

Ekim 2013 , 73 sayfa

Makine çeviri sistemleri, belirli bir doğal dilde yazılmış metni bilgisayarlar aracılığıyla otomatik olarak bir başka doğal dile çeviren sistemlerdir. Bu sistemler genellikle dilbilimsel tabanlı, istatistiksel ya da bu iki yaklaşımın birleşimi şeklinde üretilebilmektedirler. Bu çalışmada Türkçeden İngilizceye baraşımı daha yüksek otomatik çeviri sistemleri inşa edebilmek için, frekansa dayalı geç birleşim tabanlı kelime parçalama yaklaşımı tanıtılmıştır. Bu geç birleşim yöntemi farklı boyuttaki eğitim verileri kullanılarak istatistiksel ve kural tabanlı yöntemlerle karşılaştırılmıştır. Bu çalışmanın önceki çalışmalardan en büyük farkı, otomatik çeviri sistemlerinde ilk kez kullanılan geç birleşim tabanlı kelime parçalama yöntemi ile daha yüksek BLEU sonucu elde edebilmesidir. Mevcut çalışmalarda en yüksek 25.22 BLEU sonucunu elde edilmişken, tanıtılan geç birleşim tabanlı sistem 26.22 BLEU sonucunu doğurmaktadır. Tanıtılan sistem kural tabanlı ve istatistiksel yaklaşımları bir araya getirmektedir. Türkçe dilinin sondan eklemeli doğası gereğince, bu çalışmadaki sonuçlar sondan eklemeli diğer diller için de genişletilebilir.

Anahtar Kelimeler: Makine Öğrenmesi, Doğal Dil İşleme, Makine Çevirisi, Ekler

dedicated to my son

ACKNOWLEDGMENTS

Foremost, I would like to mention that this thesis would not have been possible without my advisor Prof. Dr. Adnan Yazıcı and his continuous support, motivation and enthusiasm during my M.Sc. study; therefore, I owe my most sincere gratitude to him.

I would like to thank to my family; my wife, my mother and my brother, for giving their warm support whenever I needed during my study.

Moreover, I have a great honor to express that studying at Department of Computer Engineering in METU was not only an achievement I aimed to accomplish, but also a milestone which enlarged my vision in life.

TABLE OF CONTENTS

CHAPTERS

APPENDICES

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ALGORITHMS

ALGORITHMS

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
BLEU	Bilingual Evaluation Understudy
ML	Machine Learning
MT	Machine Translation
OOV	Out Of Vocabulary
SMT	Statistical Machine Translation
SOV	Subject Object Verb
SVO	Subject Verb Object
TER	Translation Error Rate
WER	Word Error Rate
WSD	Word Sense Disambiguation

CHAPTER 1

INTRODUCTION

Communication is truly the oldest problem that humankind has ever had. With the growing population of the world, various languages spoken by different civilizations have emerged. Today, it is reported that there are more than 4,000 languages spoken by at least a thousand people [34].

Increasing use of textual materials on computers dramatically raised the importance of automated natural language translation tasks, since human aided translations cannot meet the demand at the desired level. Moreover, an increasing number of people speaking different languages through the Internet has attracted attention to automated machine translation systems. With the recent techniques developed in the scope of *Artificial Intelligence* (AI), computers have started to handle the tasks that might be rather time consuming for humans. To solve the communication problem between people speaking different languages, AI proposed various approaches, which are classified under the label of *Machine Translation* (MT), implying that the translation hypotheses of the texts can be generated in an automated way by computers. The main schema of the artificial MT systems are shown as Figure 1.1.

The increasing popularity of MT systems has motivated researchers to utilize artificial MT systems to ease daily life. Today, MT systems are widely used from multilingual web pages to mobile phones. However, the accuracy rates of contemporary MT systems are not at the desired level for humans, and computers are not even close to human translators in terms of translation accuracy rates. Researchers have been working in this area to boost the translation performance of MT systems. Contemporary approaches in MT are far behind the desired level needed to produce an accurate translation which does not need any human post-translation processes [10]. Today, MT systems are heavily used as supplementary translation memories, a sort of extensive look up dictionary for professional translators.

There are many different machine translation systems introduced in the literature. Some of the researchers applied example-based approaches [27], [57], while some of them worked on rule-based approaches [9], [60]. Statistical approaches have also been widely used [4], [11], [39], [43]. In this study, the statistical phrase-based machine translation system paradigm is used for all experiments. To build the translation and reordering models used in this study, an open source phrase-based statistical machine translation toolkit named *Moses* [28] is used during this study.

In the phrase-based statistical machine translation paradigm, the researchers have mostly focused on increasing the translation accuracy by applying various approaches. Most of the successful methods show that for agglutinative languages, exploiting the subword items and

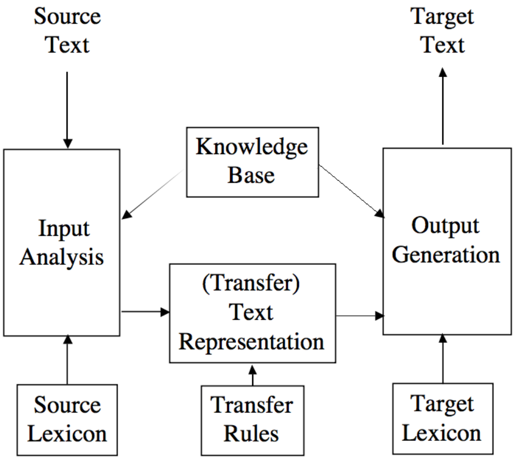

Figure 1.1: General schema of primitive MT systems

exposing the semantic information underlying the subword items boosts the performances of the translation systems [15], [16], [61]. In this thesis, the most effective ways to expose the semantic information underlying the subword items are investigated. The major motivation behind this study is to introduce a method that benefits from both the rule-based and statistical approaches at the same time. The word frequencies are used to determine if a word is decomposed by using rule-based or statistical approach. This determination threshold is learned by a set of experiments. Based on this threshold, frequent words are decomposed by stochastic methods, and the rare words are decomposed by using a rule-based morphological analyzer. The late fusion-based approach combines the outputs of these two separate modules to produce a unique word decomposition hypothesis. The corpus size used during the experiments is adjusted in intervals to measure the reactions of the different approaches using small and large amount of training data as well.

In this study, the frequency-driven late fusion-based word decomposition approach is introduced to improve the translation quality of the phrase-based statistical machine translation system from Turkish to English. This late fusion-based approach is compared with the standalone statistical and rule-based word decomposition approaches with a changing corpus size. This novel approach fuses both the rule-based and stochastic word decomposition methods.

The major objective of this thesis study is to utilize the frequency-driven late fusion-based word decomposition technique to boost the BLEU score of phrase-based statistical machine translation system from Turkish to English and produce more accurate automated translations. Moreover, this novel approach is compared with the pure statistical and rule-based word decomposition techniques with various corpus sizes. These comparison experiments reveal the performances of such approaches with both small and large amount of data. The importance of the training data quantity has already been proven in previous studies [39]. This study aims to measure the importance of the training data quantity for the different approaches including fusion-based approaches.

The major contribution of this study is that the fusion-based utilization of the subword items is built and tested on 16 different sizes of the training data. It is shown that the fusion-based approach outperforms the rule-based and statistical approaches significantly when the corpus size is sufficient. It is measured that the fusion-based approach results in around 10% better accuracy than the pure statistical approach relative to the baseline score. During all the experiments, public SETimes parallel corpora is used for the sake of comparability [58]. The study describing the data reports the baseline BLEU score as 25.22. The baseline accuracy in this study is reported as 25.36. This negligible difference is thought to be caused by the custom pre-processing techniques, which will be explained in Chapter 5 in detail. These scores are listed and shown in Table 6.6.

The list of contributions of this thesis are defined below.

1. *Frequency-driven late fusion-based word decomposition approach* with respect to a threshold value is used to build better phrase-based SMT (Statistical Machine Translation) systems for the first time. This approach increased BLEU score of the baseline SMT system. The methods used in this approach are explained below.
    (a) Frequency threshold is learned by the complete experimental SMT systems
    (b) Based on the learned threshold, either the rule-based or statistical approach is used to decompose the words
2. Character-based combinations of forward and backward n-gram models are applied to the phrase-based statistical machine translation system to increase the BLEU score. The combination operation is accomplished by the multiplication of the probabilities of the character-based forward and backward n-gram language models.
3. Statistical, rule-based, and fusion-based approaches are utilized with the small and large amount of data on the Turkish language. This is the first comprehensive study on the Turkish language evaluating the reactions of rule-based, statistical, and fusion-based approaches with 16 different sizes of training corpus.

After pre-processing the data set, the phrase-based statistical machine translation system consumes the intermediate input. The intermediate input can be a raw text, lowercased, tokenized, split into stems and morphemes, or any combination of them.

Rest of the thesis is organized as follows: Chapter 3 reviews the related studies conducted in this area on the Turkish language, and gives some technical background information regarding MT systems. In Chapter 4, the frequency-driven late fusion-based word decomposition approach is explained in detail. Moreover, the methods to utilize the subword items and exploit them to improve the quality of the translation system are discussed. In Chapter 5, the overall system is explained with its components in detail. This chapter covers the methods used in the pre-processing step, the model generation step, and the post-processing step. Then, how these components are combined is explained. In addition, the data used to train the system is described for the sake of the compatibility with the other studies. In Chapter 6, the experiments conducted in the scope of this thesis are explained. These experiments involve different uses of the components of the system, and testing them with different sizes of parallel training corpus to reveal the reactions. The main aim of the experiments is that introduced frequency-driven late fusion-based word decomposition approach is proven to perform better than the rule-baseand statistical word decomposition-based MT systems from Turkish to English. This late fusion-based approach is compared with the standalone statistical and rule-based word decomposition approaches when the corpus size changes. Comparisons of the experiments and discussions regarding them are mentioned in this section. Lastly, Chapter 7 summarizes the thesis study and puts forward the findings obtained. Moreover, it also denotes how this study can be improved further, and gives the directions for the future researchers focusing on the machine translation area.

CHAPTER 2

BACKGROUND INFORMATION

2.1 What Is Machine Translation

Machine translation is defined as the process of translating texts from a natural language to another natural language by computers based on linguistic motivations, statistical approaches, or combination of them.

An example sentence in Turkish, and it is translation, generated by a human translator is shown below, along with its hypothetical output generated by a MT system.

Example input: Kalıntılar 60 metre derinlikteki bir çukurda bulundu.

Human translated output: The remains were discovered in a pit 60 meters deep.

MT system output: The remains at a depth of 60 meters that they found in a pit.

As shown above, the MT output may not be as accurate as an output of a human translator. To achieve the desired quality of translation, contemporary studies have focused on increasing the translation accuracy of MT systems.

A machine translation system combines the pre-processing machine, the translation machine, and the post-processing machine as a whole. The pre-processing machine reads or listens to the human readable raw text, which can be a word, a phrase, a sentence or a discourse, and converts it to an intermediate form which the translation machine can process. To reduce the complexity of the problem, this step might involve both language dependent or independent normalization tasks. The translation machine is input only by the pre-processing machine. It processes the intermediate form and tries to match the most fluent set of words of the target language with the same meaning of the input text. Finally, the post-processing tool performs the inverse of the task done by the pre-processing machine by reading the intermediate form and producing the human readable output text in the target language.

To achieve the best results of a MT system, human intervention may be used to post-process the output of the system. Mostly, human translators are used to edit the output of this pipeline to produce more accurate and fluent translations in a shorter amount of time.

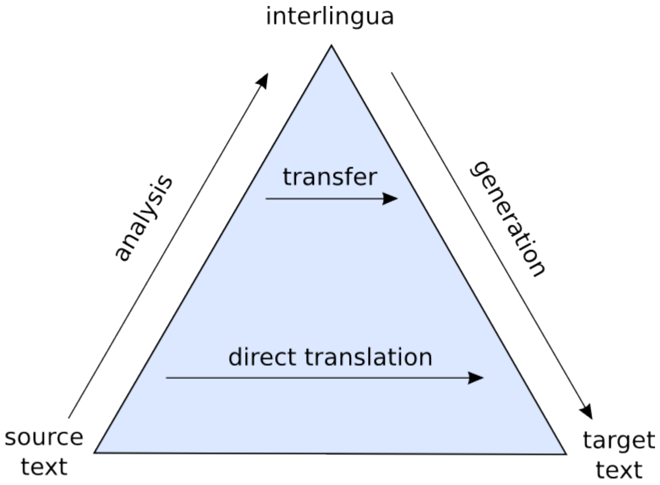

Figure 2.1: Direct translation and transfer translation pyramind

2.2 History of Machine Translation

The idea of the automated translation was thrown out as unfeasible in the 17th century, around the same time the idea of the universal language was born [29]. Until the 1950s, the earlier ideas were based on an artificial language in which feelings and events could be expressed in a more structural way.

In 1954, IBM demonstrated the first applied machine translation system, and 60 Romanized Russian sentences were entered to a computer by a human who did not know Russian and the English translations were printed out [30], [31]. This very promising experiment proved the feasibility of the automated translations without the need of human translators.

After the demonstration of IBM, an increasing number of researchers started focusing on computer-aided translation systems. Until the late 1980s, researchers thought that the best computer-aided machine translation system could be achieved only by the interpretation of the hand written linguistic translation rules designed by the expert linguists. This is called as rule-based MT systems. Interlingual machine translation is one of the first instances of the rule-based machine translation approaches. In this approach, the source language, i.e. the text to be translated, is transformed into an interlingual, i.e. source and target language independent representation. The target language is then generated out of the interlingua. Bernard Vauquois' pyramid, 2.1, shows comparative depths of intermediary representation, interlingual machine translation at the peak, followed by transfer based, then direct translation.

Later on, the statistical and hybrid approaches started to emerge with the help of the increment in computational power. These approaches are based on the *Machine Learning* (ML) techniques which observe the existing translations and try to guess the translation of new unseen sentence instances. Afterwards, the researchers brought the idea of example-based statistical modeling of the translation rules [42]. Example-based statistical modeling of the translation rules means automated generation of these hand-written rules by using artificial intelligence methods.

Today, it is repeatedly stated that the most accurate systems can be obtained by the combination of both structural and statistical approaches, and named as hybrid MT systems [23]. The hybrid approaches aim to benefit from both the rule-based and statistical approaches. During this study, fusion-based utilization of subword items is investigated and their experimental results are compared to the alternative methods. This thesis aims to investigate both of the statistical and the hybrid MT systems, and compare and reveal the evaluation scores of them.

2.3 Challenges in Machine Translation

Machine translation problem is one of the most challenging problems in computer science and it has not been a solved problem yet. In theory, the problem itself can be represented as a search problem, which consists of finding the correct set of target words carrying the same meaning of the input text, and putting them into the correct order to obtain a fluent hypothesis. Both finding the correct set of words and ordering them in a correct way are the problems which have not been solved in polynomial time and space by using the deterministic methods yet. Therefore; some set of heuristics have to be used to overcome the complexity of the problem. These heuristics reduce the set of candidate words in the hypothesis, and result in a smaller set of possible translations to be picked up as the final output. The major reasons behind these difficulties can be grouped as syntax, semantics, lexicon and morphology.

2.3.1 Syntax

Syntax designates how sentences are generated in a language. Syntax can also be defined as a set of grammatical rules of a language which determines the order and the behavior of the items which compose a sentence. By definition, every language has its own syntax and generation rules. This uniqueness may require sentence-level word or phrase reordering after producing the translation hypothesis if the syntax structures of two languages are different at some level. This means that after producing the hypothesis, the correct places of subjects, objects or verbs have to be found in the target sentence. This difficulty also emerges in the breadth of this thesis. In essence, the Turkish language mainly uses the *Subject-Object-Verb* (SOV) ordering rule whereas the English language uses the *Subject-Verb-Object* (SVO) ordering rule.

Example 2.3.1

Kalıntılar	*60 metre derinlikteki bir çukurda*	*bulundu*
Direct object	*Indirect object*	*Verb*

The remains	*were discovered*	*in a pit 60 meters deep*
Direct object	*Verb*	*Indirect object*

As it is shown in Example 2.3.1, the *verb* appears right after the *direct object*, or *subject* in English. However in Turkish, the *verb* appears at the end of regular sentences. The longer sentences cause a larger gap between the places of phrases in Turkish and English. This difference between Turkish and English mostly results in wrong placement of the phrases in phrase-based SMT systems. This issue is addressed later in Section 6.3.

2.3.2 Semantics

Semantics loads meanings to the syntactic items, such as words, phrases, signs or symbols. Unlike the syntax, semantics defines the underlying meanings of the character sequences. Discarding semantics, an MT system cannot cope with ambiguous terms in a sentence. The meanings of the terms or phrases are disambiguated by the semantic rules of the language. The very basic contention is that the semantics, somehow, has to be involved in MT systems; otherwise, they are bound by nothing more than a string or forms processing task.

Sometimes, there are many ways in which a word or sentence can be translated into the target language. For example a simple word, *adam*, can be translated in two different ways as shown in Example 2.3.2. This ambiguity is resolved by the *word sense disambiguation* technique [52].

Example 2.3.2 Input: *Adam*

Hypothesis - 1: *The man*

Hypothesis - 2: *My island*

Furthermore, a complete sentence may have more than one correct translation in the target language. These correct translations may carry completely different meanings, as shown in Example 2.3.3.

Example 2.3.3 Input: *Ürdün kamyonlarına Türkiye'ye geçiş izni vermedi.*

Hypothesis - 1: *Jordan didn't give the transition pass to their trucks to Turkey.* (The trucks do not belong to Jordan.)

Hypothesis - 2: *It didn't give the transition pass to Jordan's trucks to Turkey.* (The country not giving the permission is not Jordan.)

Such ambiguous sentences are sometimes not able to be translated even by the professional human translators.

2.3.3 Lexicon and Morphology

In the area of natural language processing, a lexicon is mostly used to define a vocabulary consisting of words or phrases of a language. In most of the traditional MT systems the smallest items in a lexicon are words. Having words as the smallest units of lexicon items is easy to utilize, but brings a considerable problem, which is data sparsity. Especially languages with an agglutinative nature, such as Turkish, require a much more intensive lexicon than other languages.

To decrease the unique word counts and the computational complexity, and to overcome data sparsity problems, the utilization of the subword items is proposed in this thesis. The methods will be explained in Chapter 4.

2.4 Noisy Channel Model

Messages transmitted through a noisy channel, in which the messages may be corrupted, have to be somehow recovered efficiently in many areas. The machine translation problem can be transformed to a problem of message transmission from a source language to the target language. During such a transmission, words might be replaced or scrambled. At the core of the machine translation problem is finding an efficient way to recover the actual messages. By definition, the noisy channel model can be used to model several problems including question answering, speech recognition, machine translation, or spelling correction [3].

The noisy channel model is a framework used in various natural language processing tasks. In this model, the goal is to find the intended word, given a word where the letters have been scrambled (or translated) in some manner. In our case, we aim to translate Turkish sentences into English language. In this context, we assume that we already have correct translations in English. These correct translations then pass through a noisy channel where they are corrupted. The corrupted output of the noisy channel is what we have: Turkish sentences. Therefore; we need to find a way to recover the English sentences, which are inputs to the noisy channel, by using Turkish sentences, which are outputs of the noisy channel. The machine which recovers the actual message given the corrupted one is called a decoder. This flow is represented in Figure 2.2.

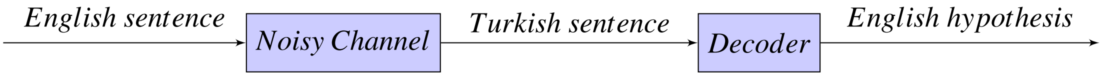

Figure 2.2: Noisy channel model used in the statistical machine translation

To illustrate a translation from Turkish to English, it is assumed that the correct English sentence is transmitted through a noisy channel, shown as Example 2.3. The sentence *The remains were discovered in a pit 60 meters deep.* is fed to the noisy channel in which the English sentence is corrupted (translated) and the Turkish sentence, *Kalıntılar 60 metre derinlikteki bir çukurda bulundu.*, is observed as the output of the channel. Now, the aim is to find out the input sentence in English in which the corruption (translation) results in the observation sentence in Turkish. The decoder part of the pipeline makes mathematical calculations and estimates the most probable input fed to the noisy channel. These calculations are explained in Section 2.5.

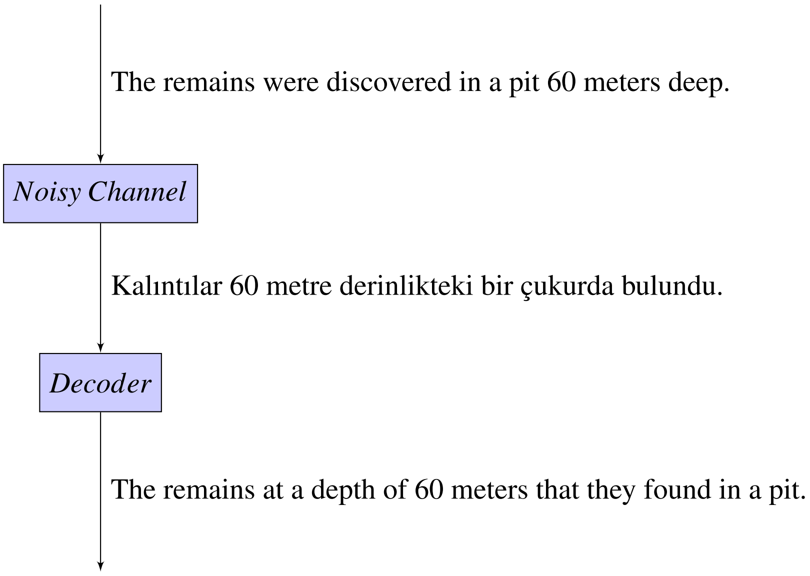
The remains were discovered in a pit 60 meters deep.
Noisy Channel
Kalıntılar 60 metre derinlikteki bir çukurda bulundu.
Decoder
The remains at a depth of 60 meters that they found in a pit.

Figure 2.3: Noisy channel model example in the statistical machine translation paradigm

2.5 Bayes' Theorem

Bayes' Theorem, which is based on the revolutionary studies of Reverend Thomas Bayes during the 18^{th} century, calculates the relations between the marginal probabilities and the conditional probabilities of the observations.

In Section 2.4, it is mentioned that the translation process actually means finding the original message which is corrupted by a noisy channel model. In a noisy channel model, the input is the message received by a noisy channel. The counterpart of this input is the observation in Bayes' Theorem. As it is shown in Equation 2.1 Bayes' Theorem formulates the probability of a hypothesis when an observation, or source sentence, is given.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{2.1}$$

In the equation 2.1, A and B are independent events, and P(A) and P(B) are the observation probabilities of those events.

Bayes' Theorem is adopted to the machine translation problem to find the probability of the hypothesis sentence when the source sentence is observed. For example, while translating a sentence from Turkish to English, Bayes' Theorem tries to find an English sentence which is translated into Turkish. In other words, Bayes' Theorem treats the input sentence as if it is actual output translation, and the aim is to find the input sentence resulting such translation. Therefore; the formula can be rewritten as Equation 2.2, where T is the probability of the observation of Turkish sentences and E is the probability of the observation of English sentences.

$$P(E|T) = \frac{P(T|E)P(E)}{P(T)} \tag{2.2}$$

To illustrate, the translation probability function is shown as Equation 2.3 for a sample input in Turkish to be translated into English.

$$P(E|Harabe\ bulundu.) \propto \frac{P(Harabe\ bulundu.|E)P(E)}{P(Harabe\ bulundu.)} \tag{2.3}$$

The observation probability of the sentence *Harabe bulundu.* is the same for all the translation hypotheses in English. Therefore, this term is reduced from the formula, and the final formula can be written as Equation 2.4.

$$P(E|Harabe\ bulundu.) \propto P(Harabe\ bulundu.|E)P(E) \tag{2.4}$$

The overall translation probability is obtained by the multiplication of two terms, the translation model probability, and the language model probability, shown as Equation 2.4. The term $P(Harabe\ bulundu.|E)$ means the probability of the sentence in English which is translated into Turkish as *Harabe bulundu.*. The second term $P(E)$ is the observation probability of this sentence in English. The fluent and frequent sentences in English have higher observation probabilities.

Since the Turkish sentence is given to be translated, its observation probability cannot change during the arithmetic operations, which means $P(T)$ is equal to the same value for all hypotheses in the formula. Hence, the denominator can be removed from the formula for the statistical machine translation problem as shown in Equation 2.5.

$$P(E|T) = P(T|E)P(E) \tag{2.5}$$

The final formula indicates that the probability of a hypothesis E for a given source sentence T depends on both $P(T|E)$ and $P(E)$. The former is calculated by the translation model and it implies that the translation has to be dependable. The latter is calculated by the language model and it implies that the translation has to be fluent.

2.6 Language Models

Language models, in the textual paradigm, are used to calculate the probability of a phrase, word, or sub-word unit occurrence when the previous context is given. Language models are compiled by using very large-sized monolingual corpora. Language models are generated specifically for a language generally by using a large corpora in a language, or linguistic rules of a language. At query time, statistical language models return the probability of an input sentence and how likely this sentence is to be observed in this language.

Mathematical calculation of the reliability probability of the translation hypotheses requires both the faithfulness and the fluency metric. While the faithfulness is calculated by using the translation table, which is an extensive phrase-based dictionary with probability values, while the fluency is calculated using the language models.

2.6.1 N-Gram Language Models

In computational linguistics, n-gram is defined as a sequence of items with length of n in a given larger context. These substring items can be defined as characters, syllables, words, or longer sequences, such as phrases [21]. The breadth of this thesis defines an item as a word in its surface form.

Similar to other language models, n-gram models help the hypothesis generator machine to boost the probabilities of the fluent sentences in the target language. To compile an n-gram language model, very large sized monolingual corpora are used. Simply by counting the word occurrences and their neighbor items, the probabilities of the occurrences of the words are calculated and stored when the preceding $n-1$ words are given.

By definition, the probability of the occurrence of a word depends on only the preceding $n-1$ words. This implies that the probability of a sentence can be calculated by using a sliding window of size n. For example, a 3-gram language model calculates the observation probability of a sentence with 6 words as Equation 2.7.

$$P(w_1,w_2,w_3,w_4,w_5) = P(w_1,w_2,w_3)xP(w_2,w_3,w_4)xP(w_3,w_4,w_5) \tag{2.6}$$

$$P(tarihi,bir,kalede,bulundu) = P(tarihi,bir,kalede)xP(bir,kalede,bulundu) \tag{2.7}$$

Each window produces a probability of sequences whose length is n. The probability of the whole sentence or text can then be calculated by multiplying all the probability values generated by the sliding window. The higher probability of the sentence satisfies that the more fluent the sentence is generated by the decoder. In theory, the probability of a word in a given context depends on all the preceding words. Hence, the probability of a sentence with k words can be calculated by the formula shown in Equation 2.8.

$$P(w_1,w_2,...,w_k) = \prod_{i=1}^{k} P(w_i|w_1,w_2,...,w_{i-1}) \tag{2.8}$$

As it is observed, the last terms of the production sequence require an intensive amount of information about the observations. This unfortunately results in data sparsity and zero probability for almost all the sentences. The independence assumption resolves such a data sparsity problem [36]. It suggests that the probability of a word occurrence depends on only preceding $n-1$ words. Thus, the previous formula is rewritten as Equation 2.9.

$$P(w_1,w_2,...,w_k) = \prod_{i=1}^{k} P(w_i|w_{i-(n-1)},w_{i-(n-1)+1},...,w_{i-1}) \tag{2.9}$$

After calculating the probabilities $P(w_1, w_2, ..., w_k)$ for each hypothesis $w_1, w_2, ..., w_k$, the most probable sentence, in other words, the most fluent sentence, is likely to be picked as the final translation hypothesis.

2.6.2 Pruning

Pruning is a vital part of the language model generation process. Since language models are compiled by using very large corpora, it is highly possible that the noisy data might be blended into the language model. The pruning process aims to compile robust and reliable language models. In addition, this process helps to reduce the sizes of the language model files by eliminating the noisy n-grams [20].

There are two major approaches to prune the language models once they are created. The first approach scans all of the n-grams compiled from data and discards ones having smaller log-probabilities than a predefined threshold. The threshold value can be adjusted to determine the pruning intensity.

Table 2.1 illustrates a fragmentation of a 5-gram language model. Pruning based on a threshold value of -1.5 discards the n-grams having lower log-probability than 1.5, and keeps the ones having a log-probability higher than 1.5. Table 2.2 shows the resulting n-gram fragmentation after the pruning operation with the threshold value of 1.5.

Table 2.1: A fragmentation of 5-gram language model

Probability formula	**Log-Probability**	**N-Gram**
$\log(P(lessen \mid has, the, potential, to))$	-2.042486	has the potential to lessen
$\log(P(of \mid has, the, largest, growth))$	-1.830927	has the largest growth of
$\log(P(in \mid has, the, largest, expansion))$	-1.928299	has the largest expansion in
$\log(P(of \mid has, the, highest, level))$	-0.740430	has the highest level of
$\log(P(decide \mid has, the, authority, to))$	-0.951597	has the authority to decide
$\log(P(the \mid has, the, backing, of))$	-1.154158	has the backing of the
$\log(P(into \mid has, to, be, taken))$	-0.661742	has to be taken into
$\log(P(by \mid has, to, be, elected))$	-0.721017	has to be elected by

After the pruning operation, the aim is to keep only the n-grams with high probability. Therefore, a good adjustment of the intensity of pruning may increase the fluency in the target language.

The second approach does not deal with the probabilities of the n-grams generated, but the counts of the words. It discards a word if it occurred less than a predefined threshold. Here, the threshold value also can be adjusted to determine the pruning intensity. In this thesis, the second approach is used with the threshold value of 1 for all experiments. The pruning parameters are defined by a set of previous examples which are out of the capacity of this thesis.

Table 2.2: Pruned fragmentation of 5-gram language model

Probability formula	Log-Probability	N-Gram
$\log(P(of\|has,the,highest,level))$	-0.740430	has the highest level of
$\log(P(decide\|has,the,authority,to))$	-0.951597	has the authority to decide
$\log(P(the\|has,the,backing,of))$	-1.154158	has the backing of the
$\log(P(into\|has,to,be,taken))$	-0.661742	has to be taken into
$\log(P(by\|has,to,be,elected))$	-0.721017	has to be elected by

2.6.3 Smoothing

The independence assumption does not guarantee to resolve the data sparsity problem completely, but to reduce it to a considerable level. When the unique word count is k in the training corpus of the language model, and the order of the language model is n, in theory, we would need k^n different n-gram sequence in the model file to resolve the data sparsity. Without having this number of n-gram sequences, we can resolve the data sparsity problem by using smoothing methods.

In practice, smoothed language models never generate zero probability for any n-gram sequences, even for the grammatically incorrect word sequences. This helps not to end up with zero probability while calculating the overall sentence probability.

There are a large number of smoothing methods in the literature, such as the Laplace, Good-Turing, and Kneser-Ney smoothing algorithms [64]. In this study the Witten-Bell [8] smoothing method is used for all the experiments. The smoothing parameters are defined by a set of previous examples which are out of the scope of this thesis.

Laplace smoothing (also known as add-one smoothing) is a smoothing technique to eliminate zero probabilities by adding 1 to counts of every instance. Probability of an instance is calculated by Equation 2.10. Laplace smoothing method adds 1 to counts of every instance, and the new formula to calculate the probabilities is shown as Equation 2.11

$$P_i = \frac{c(i)}{N} \tag{2.10}$$

In the equation 2.10, **N** is the number of total word tokens, and **c(i)** is the count of token-i.

$$P_i^* = \frac{c(i)+1}{N+V} \tag{2.11}$$

In the equation 2.11, **N** is the number of total word tokens, **V** is the number of word types (vocabulary size), **c(i)** is the count of token-i.

On the other hand, the Witten-Bell algorithm is based on the idea that a zero frequency n-gram is an event that hasn't happened yet. The Witten-Bell algorithm suggests to assign the total probability mass to the zero frequency n-grams. The formula to calculate the probability of

2-grams is shown as Equation 2.12.

$$\sum_{i:c_i=0} p^*(w_i|w_x) = \frac{T(w_x)}{N(w_x)+T(w_x)} \tag{2.12}$$

2.6.4 Interpolation

Domain is an important aspect of the language models. Language model domain must be consistent with the aimed translation system domain. If the domains of the parallel sentenced data and the monolingual data are different, then the translation quality will suffer from this difference. In other words, the faithfulness and the fluency reference points must be the same. However, there are techniques which can make use of the sources of different domains [24], [26], [33].

Basically, language model interpolation involves methods which combine the different n-gram probabilities from different language models. Each language model must be assigned a weight, so that the final probability of the n-gram can be calculated as the weighted arithmetic mean. The formula used during this study is shown as Equation 2.13 and Equation 2.14.

$$P(w_1, w_2, ..., w_k) = \sum_{i=1}^{n} \alpha_i P_i(w_1, w_2, ..., w_k) \tag{2.13}$$

$$\sum_{i=1}^{n} \alpha_i = 1 \tag{2.14}$$

The phrase-based statistical machine translation system mentioned in the capacity of this thesis focuses on the political news domain, and only one source is used to compile the language model. Therefore, none of the interpolation methods are used in this thesis study, and all of the a_i values are set to 1, as shown in Equation 2.14.

2.7 Translation Models

While the language models are used for fluency in the generated hypothesis translation, the translation models are used for obtaining the faithful translation hypotheses. Translation models involve both an extensive phrase level dictionary with translation probabilities and the reordering tables for the scrambled structures after the production of the translation hypotheses. An illustrative phrase table is shown as Table 5.3.

To summarize, a translation model is a probability distribution which returns the probability of a translation for a given source sentence. It assures the protection of the meaning of the source sentence. It should be noted that the translation models do not care about neither the grammar nor the word order of the hypotheses, but rather the protection of the items carrying the semantic information.

2.7.1 Word Alignment

Word alignment is the very first step of the model generation process in a statistical machine translation system. It is also called bi-text word alignment, because it requires the translations of the sentences in two natural languages. When a bunch of parallel sentences are given for the training phase, relations of words are extracted and the correct translations of the words are guessed by the word aligner. It is important to observe that the word alignment phase does not focus on phrases, but words. The word alignment task is done by the GIZA++ [44] toolkit.

An example word alignment diagram is shown in Table 2.3.

Table 2.3: Example word alignment result for a sentence pair

	Scientific	studies	have	proven	a	new	language	,	learning	,	strengthens	overall	brain	function	.
Bilimsel	■														
çalışmalar		■													
,															
yeni						■									
bir					■										
dil							■								
öğrenmenin									■						
genel												■			
beyin													■		
fonksiyonlarını														■	
güçlendirdiğini											■				
kanıtlamıştır			■	■											
.															■

2.7.2 Phrase Extraction

After words are matched with their counterparts in the other language, consecutive words are extracted from each of the sentences. When these consecutive words appear together frequently, they compose the phrases. Since the semantic relation of the words are known already, the phrases can be matched accordingly. By using the example word alignment matrix, shown in Table 2.3, the neighboring black squares can be combined to obtain a larger rectangle. An example of these phrase extraction marks are demonstrated in Table 2.4.

The marks shown in the example result in the phrase pairs which are shown as Table 2.5. The occurrence frequencies of the pairs assesses the translation probabilities of the pairs. This phase is also called phrase scoring. In the thesis scope, it is included in the phrase extraction phase.

Table 2.4: Example phrase extraction result for a sentence pair

	Scientific	studies	have	proven	a	new	language	,	learning	,	strengthens	overall	brain	function	.
Bilimsel	▣	🟩													
çalışmalar	🟩	▣													
,															
yeni					🟩	▣	🟩								
bir					▣	🟩	🟩								
dil					🟩	🟩	▣								
öğrenmenin									■						
genel											🟩	▣	🟩	🟩	
beyin											🟩	🟩	▣	🟩	
fonksiyonlarını											🟩	🟩	🟩	▣	
güçlendirdiğini											▣	🟩	🟩	🟩	
kanıtlamıştır			▣	▣											
.															■

Table 2.5: Extracted phrase pairs from the example parallel sentence

Turkish	**English**
Bilimsel çalışmalar	Scientific studies
yeni bir dil	a new language
genel beyin fonksiyonlarını güçlendirdiğini	strengthens overall brain function
kanıtlamıştır	have proven

2.7.3 Phrase Reordering

In machine translation task, a phrase reordering model is required to obtain a correct sequence of the words and phrases in the target sentence. The initial hypothesis is considered to be a scrambled form of the correct translation. Finding the correct sequence given a scrambled form is a difficult problem to solve, and the time and space complexity of this problem is exponential. Therefore; it is essential to use some heuristics to shrink the actual search space.

The reordering model consists of the phrases and the actions to be taken when these phrases are encountered. The model can be learned by using the phrase marks in the word alignment matrix above. Each extracted phrase translation can follow the previous one monotonically, swap with the previous one, or jump one or more words ahead from the previous one. These behaviors are shown in Figure 2.4.

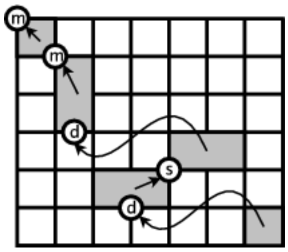

Figure 2.4: Extraction of the order behaviors of the phrases

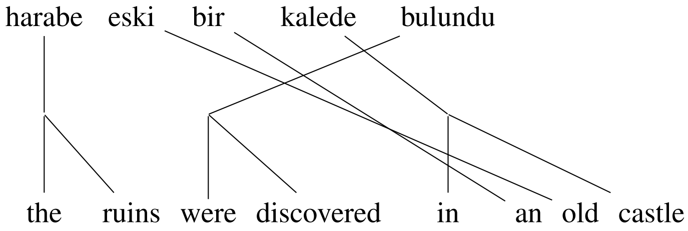

Figure 2.5: Reordering the words in the translated sentences

There are several lexicalized reordering model types introduced to the literature [1], [18]. In this study, all of the monotone, swap, and discontinuous reordering types are taken into account by default.

Reordering Subword Units

As it is explained, finding the correct places of the phrases is a costly task. The Turkish language mainly uses *Subject-Object-Verb* (SOV) ordering rule. On the other hand, the English language uses the *Subject-Verb-Object* (SVO) reordering rule. This difference brings about distant translations of phrases. Moreover, longer sentence pairs require a farther reordering process, i.e. the distance between the words to be reordered is longer when longer sentences occur, which is even more complex problem. Figure 2.5 shows the distant translations of phrase pairs.

The longer sentence pairs require farther reordering process, as stated. Morphological analysis or word decomposition operations split words into stems and morphemes, which result in even longer sentences in Turkish. This increases the difficulty of the reordering problem, which is shown in Figure 2.6. Thus, longer sentences are more likely to harm the translation accuracy due to the reordering drawbacks, as discussed in Section 6.3.

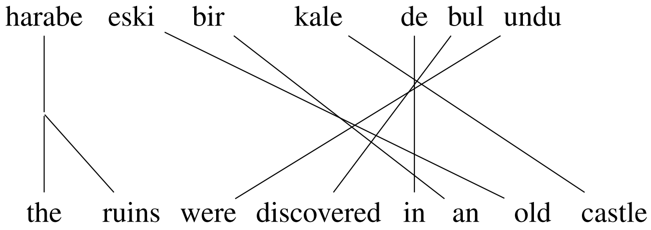

Figure 2.6: Reordering the stems and morphemes in the translated sentences

2.8 Machine Translation Evaluation Metrics

Evaluation has always been a hot topic in the machine translation area. Measuring the quality of a natural language sentence translation is not a straightforward process. Therefore, various matrices are proposed in the literature. Translation error rate (TER) [54], word error rate (WER) [37], METEOR [14], and BLEU [49] are the most well-known and frequently used evaluation metrics.

Translation Error Rate (TER) is an error metric for machine translation that measures the number of edits required to change a system output into one of the references. Each insertion, deletion, update or shift has a penalty score of 1 to produce the output sentence given the input sentence. Thus, a higher TER score means worse translation accuracy in general.

$$TER = \frac{S+D+I+H}{S+D+H+C} \tag{2.15}$$

In the equation 2.15, **S** is the number of substitutions, **D** is the number of deletions, **I** is the number of insertions, **H** is the number of shifts, **C** is the number of corrects.

Word error rate (WER) is a common metric of the performance of a speech recognition or machine translation system. WER can be computed by using the formula, shown as Equation 2.16. The WER is derived from the well-known Levenshtein distance algorithm, working at the word level instead of the phoneme level. The main difference between TER and WER is that WER is a position independent evaluation metric. WER scores translation hypotheses regardless of the positions of the words. Therefore, WER is expected to produce a higher score than TER.

$$WER = \frac{S+D+I}{S+D+C} \tag{2.16}$$

In the equation 2.16, **S** is the number of substitutions, **D** is the number of deletions, **I** is the number of insertions, **C** is the number of corrects.

METEOR score is an evaluation score in MT systems based on the exact, stem, synonym, and paraphrase matches between words and phrases. Segment and system level metric scores are calculated based on the alignments between hypothesis-reference pairs. This evaluation metric is mostly used for the multiple referenced translations, where the translation hypotheses are compared by more than one correct translations. This fact explains why METEOR scores are relatively higher than the TER, WER or BLEU scores.

BLEU is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another. Quality is considered to be the correspondence between a machine's output and that of a human. BLEU was one of the first metrics to achieve a high correlation with human judgements of quality, and remains one of the most popular inexpensive automated metrics.

The translated sentences must be evaluated by a consistent method throughout all the phases including the parameter optimization phase and the evaluation phase. During the parameter optimization phase, a set of values are assigned to the parameters, and then the development corpus is translated. According to the evaluation score on the development set, all parameters should be tuned in order to obtain the highest translation quality. Moreover, the overall quality of the system can be compared among the different systems by using the evaluation metrics on some unseen test set.

In the thesis scope, the BLEU metric is used for both of the optimization and evaluation tasks. The scale of BLEU metric is between 0 and 100, where higher accuracy indicated by a higher BLEU score indicates better translation quality.

CHAPTER 3

RELATED WORK

As mentioned in Section 2.2, after the 1980s, researchers who focused on the machine translation problem had introduced the idea of analogy-based statistical machine translation paradigm [25]. The adaptation of these statistical approaches to the Turkish language dates back to the end of the 1990s [27], [57]. The very first studies on machine translation systems were analogy (example)-based systems. Such methods are often characterized by their use at run time of a bilingual corpus with parallel texts as their main knowledge base. It is essentially a translation by analogy and can be viewed as an implementation of the case-based reasoning approach of machine learning.

It is not difficult to reveal the reason behind the translation qualities of sentences of the agglutinative languages are suffering from the data sparsity. The unique word counts in morphologically rich languages are much higher than morphologically weak languages when they are sampled in the same sizes of sentences. When the sample size changes, the reactions of the unique word counts in Turkish and English are represented in Figure 3.1. It is seen that the unique word counts reflect the agglutinative differences of these two languages. When the statistical translation systems involving Turkish are compared to the others, as expected, lower accuracy and worse translation quality was generally seen. The main reason behind this fact is that the surface forms of the words can be easily inflected and new surface forms are produced by the generative nature of the agglutinative languages, including Turkish.

Given the translated pairs of sentences in Turkish and English, the changes in total words are represented in Figure 3.2. It is observed that the number of total words in Turkish is significantly fewer than English while having the same number of sentences. This implies that the same semantic information is mostly carried by the morphemes rather than the words in Turkish.

Given the translated pairs of sentences in Turkish and English, the changes in unique words are represented in Figure 3.1. It is observed that the number of unique words in Turkish is significantly larger than English while having the same number of sentences. It should be remarked that the Turkish language has a fewer number of total words, but a larger number of unique words than English on average. This implies that Turkish words can be inflected more frequently. In other words, there are much more surface word forms in Turkish than English. The morphological differences between Turkish and English are reflected by Figures 3.1 and 3.2. These figures are obtained by processing the data used for the model training parts of this study.

The generative nature of the Turkish language produces new words by appending the suffixes

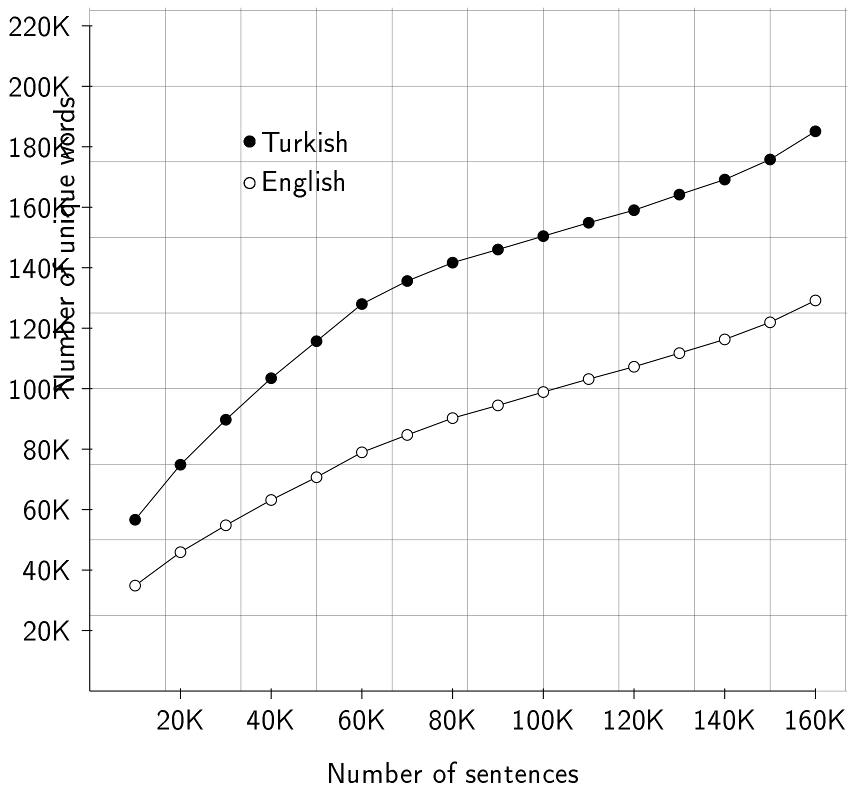

Figure 3.1: Number of unique words versus sentences in Turkish and English

exponentially based on the number of suffixes. Table 3.1 shows the number of words which can be generated by using given number of derivational suffixes from two stems, namely *masa* and *oku* [5], [55], [63]. Interestingly, a single verb stem can produce around 1.5 million different words in theory.

Inspired by the richness of the Turkish language in terms of morphology, the morphemes and sub-word units were treated as if they are actual words themselves. Syntactic morphological analyzers [45] were used in several studies that proved improvements can be achieved by using the underlying morphological information of the word units [15], [16], [62]. In addition, unsupervised word decomposition was also applied before the translation model generation in the Turkish language and a considerable amount of improvement was gained [41].

On the other hand, it has been clearly shown that the success rates of the statistical systems included phrase-based statistical machine translation highly depend on the training data quality and quantity [39].

As mentioned, there have been a substantial amount of studies conducted on the morphological aspect of the Turkish language in the phrase-based statistical machine translation systems. There are also studies to reveal the effects of the corpus quantity on the phrase-based statistical machine translation systems in different languages. This study aims to combine these two types of studies and evaluate the different approaches on the Turkish language while changing the corpus size cumulatively.

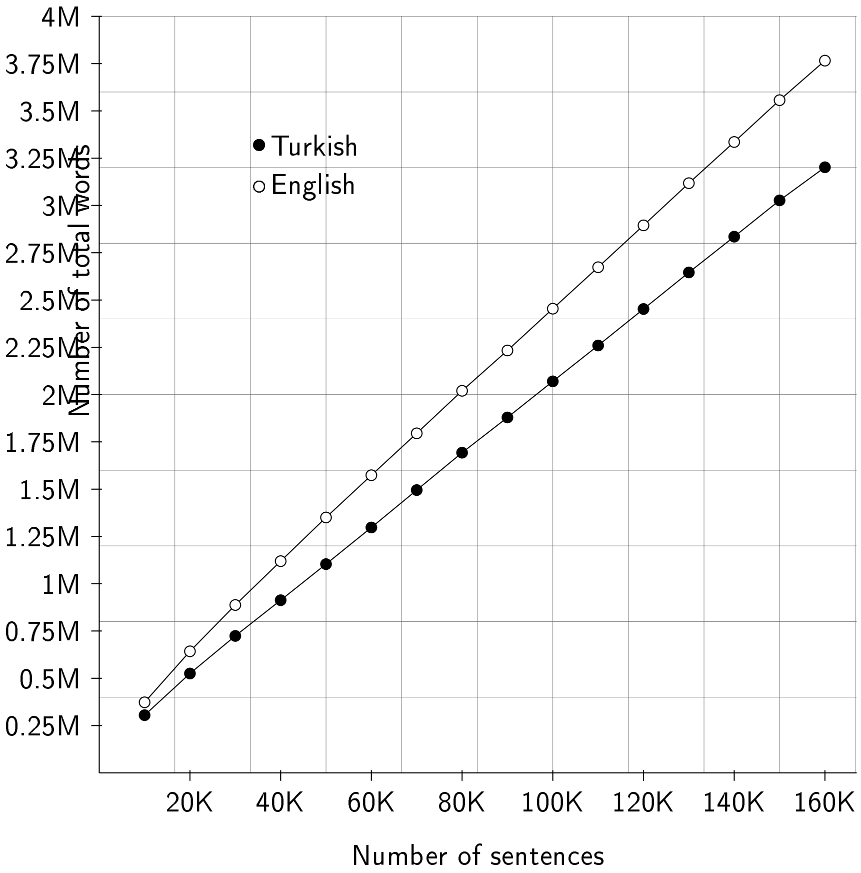

Figure 3.2: Number of total words versus sentences in Turkish and English

The studies which are focused on morphology and subword items have mostly been conducted on agglutinative languages [48], [50]. The languages like Turkish, Finnish and Hungarian have a significantly stronger morphology than the languages like English, French and Spanish [51]. The main reason behind it is that the subword item utilization is not favored for the languages like English and French.

3.1 Exploring different representational units in English-to-Turkish statistical machine translation

Exploring different representational units in English-to-Turkish statistical machine translation [47] is a study which focuses on the utilization of the subword items by performing morphological analysis and various post-processing techniques to increase the translation accuracy of the phrase-based statistical machine translation system. The objectives of this study show similarities with the concept of this thesis, despite different approaches being used.

The study mentioned above is based on the morphological analysis [45], and it includes three different segmentation models which are listed below. Two of these three models have equivalent approaches in this thesis. Therefore, this study is selected to be explained, compared and evaluated.

Table 3.1: Number of words produced by using derivational suffixes

Stem	**Derivational Suffixes**	**New words**	Total words
masa	0	112	112
	1	4,663	4,775
	2	49,640	54,415
	3	493,975	548,390
oku	0	702	702
	1	11,366	12,068
	2	112,877	124,945
	3	1,336,266	1,461,211

- Baseline model: This model is the same as the baseline model in this thesis.
- Fully morphologically segmented model: This model is the same as the rule-based model in this thesis, explained in Section 4.3.
- Selectively Segmented Model: This model statistically tries to find the unaligned subword items, and attach them to the word roots.

The selectively segmented model which does not have any equivalence approach in this study uses GIZA++ toolkit [44] to detect the unaligned tokens. Then, the unaligned tokens are attached to the word roots to undo the segmentation of such tokens. They report that this approach results in more than 9% better accuracy relative to the baseline score. Since the data set that they used is not public, the studies conducted during this thesis are not able to be compared with theirs. However, two of the approaches are the same and the fusion-based approach used in this thesis yields around 10% better accuracy relative to the baseline score.

3.2 The TUBITAK-UEKAE Statistical Machine Translation System for IWSLT 2010

The TUBITAK-UEKAE Statistical Machine Translation System for IWSLT 2010 [41] is a study which introduces unsupervised statistical word decomposition to be used as a preprocessor for the phrase-based statistical machine translation system. This study was conducted and submitted for the participation in the IWSLT 2010 evaluation campaign. The researchers of this study defend that the Turkish language has a strong nature of agglutination, and it highly requires the utilization of subword items, which is quite similar to the approaches explained later in Chapter 4. They extensively investigated using an unsupervised morphological segmentation tool called Morfessor [12], which is publicly available. Morfessor utilizes the minimum description length (MDL) principle to search for the optimal sub-word segmentation of a given corpus. Any word has a single-segmentation hypothesis in this model regardless of the context of the word [41].

The data set used in their study is BTEC (Basic Travel Expression Corpus), which is provided by IWSLT, and it is common for all participants of the campaign. The domain of the data set is quite restricted to the basic travel expressions, as its name implies. Therefore, the participants report quite high BLEU scores. They report the result of 54.05 BLEU score for the bilingual test data set of IWSLT 2010.

CHAPTER 4

UTILIZATION OF SUBWORD ITEMS

4.1 Linguistic Aspects

In the area of machine translation, the complexity of the problem also depends on the total number of unique words in the target domain. The higher number of unique words existing in the domain tends to cause more difficult translation problems. When the aim is to build a general purpose machine translation system without any domain restriction, the problem becomes even more complicated, since it is necessary to involve all the words in a language. Hence, in our case, the difficulty of the problem depends on the fact that different word forms can occur.

In Chapter 3, it is mentioned that the unique word counts differ considerably in the Turkish and English languages for a set of translations of sentences. In Turkish, the unique word count is larger than the one for English. This is caused by the agglutinative nature of Turkish. In this language, the word forms can be easily changed by adding suffixes. This brief information explains why we need to decompose the words in Turkish to decrease the unique word counts.

In other words, an English phrase containing multiple words can be translated into a single Turkish word. For example, the phrase *in my car* in English can be translated to Turkish as *arabamda*. Decomposing Turkish words into stem and morphemes, like *araba +m +da*, helps to increase the similarity between the language pair as well.

In this study, the aim is to validate the success of the frequency-driven late fusion-based word decomposition approach by comparing it with the rule-based and stochastic approaches.

4.2 N-Gram-Based Decomposition

N-gram-based word decomposition aims to split stems and morphemes from each other by using unsupervised learning methods. This task requires only a raw corpus in Turkish to learn the probabilities of the letter sequences in the language.

In Section 2.6.1, the n-gram approach is explained in detail. For the word decomposition, this approach slightly differs from the previous one. In this phase, the items used for n-gram modeling are not words, but letters. Thus, this approach is also called the letter or character-based n-gram modeling in the literature.

Character-based n-gram approaches have been used for many different tasks so far, such as information retrieval [38], text categorization [6], named entity recognition [53], language identification [7], and morphological analysis [32].

This approach suggests calculating the probabilities of the occurrences of each letter after $n-1$ preceding letters. To illustrate, guessing the upcoming letter for the sequence *sandaly_* is easier than guessing the upcoming letter for the sequence *sandal_*. For the sequence *sandaly_*, the upcoming letter can only be *e* to compose a valid sequence; however, for the sequence *sandal_*, the upcoming letter can be *ı*, *a*, *d*, *l*, *y*, *s* to compose another valid Turkish sequence. By using raw text, the probability distributions for all the letter sequences are calculated. By using these probabilities, it is easy to measure the possible letter variety for a given sequence. If the letter variety is large, then it is quite possible to have a morpheme boundary right after the given sequence. Otherwise, the upcoming letter can be guessed easily, then the given sequence does not usually indicate a morpheme boundary.

4.2.1 Forward N-Gram Modeling

Forward n-gram modeling is the standard way for character-based n-gram modeling. In this approach, the words are split into letters, so each letter appears as if it is a separate word. It should be noted that the word boundaries have to be marked so they will not be lost.

$$mevsim\, normalleri \longrightarrow < w > m\ e\ v\ s\ i\ m < /w > < w > n\ o\ r\ m\ a\ l\ l\ e\ r\ i < /w > \tag{4.1}$$

After the pre-processing step, the IRSTLM toolkit [17] creates the character-based n-gram model.

While decomposing the unseen words, the interpolation technique, mentioned in Section 2.6.4, is used for the purpose of not generating the zero probabilities. Thereby, the most probable sequence is marked as a morpheme boundary.

4.2.2 Backward N-Gram Modeling

While forward n-gram modeling aims to learn the valid letter sequences in the language, backward n-gram modeling aims to learn the most probable endings and morphemes in the language. The modeling formalism in backward n-gram modeling is the same as the forward n-gram modeling; however, the pre-processing step differs from it. This method requires the reversed forms of the words, and n-grams are generated by starting at the end of the word and traversing to the front.

The aim of the backward n-gram modeling approach is to guess weather an ending is a valid morpheme or morpheme group by traversing the letters from the end to the beginning.

To illustrate, the pre-processing phase of the approach can be shown as follows.

$$mevsim\, normalleri \longrightarrow < w > m\ i\ s\ v\ e\ m < /w > < w > i\ r\ e\ l\ l\ a\ m\ r\ o\ n < /w > \tag{4.2}$$

By traversing the letters, the probabilities of being a valid morpheme can be calculated. The valid morphemes are demonstrated below.

$$< w > m i s v e m < /w >< w > i r e l l a m r o n < /w > \tag{4.3}$$

As shown for the word *mevsim*, if the approach is only based on rules to match the valid morphemes, it is quite possible to overstem the words. For this reason, the same n-gram language modeling toolkit, IRSTLM [17] is used to calculate the probabilities on each morpheme boundary candidate.

4.2.3 Combination of Forward and Backward N-Gram Modelling

This approach combines the forward n-gram modeling and backward n-gram modeling approach for each morpheme boundary candidate. Each point between two letters of a word is considered as a morpheme boundary candidate. A word with n letter has $n-1$ morpheme boundary candidates. For each morpheme boundary candidate, forward and backward n-gram models are used to calculate these two probabilities. Then, for each candidate, the multiplication of these two probabilities is considered as the final probability for being an actual morpheme boundary. At this step, $n-1$ morpheme boundary probabilities are calculated for a word with n letters. By simply traversing the morpheme boundary probabilties, the ones at the peak are selected as the actual boundaries. In other words, the probabilities larger than the previous one and the latter one is considered to be a morpheme boundary.

The overall approach helps to decompose Turkish words without having any linguistic resources or gramatical rules. Since the approach is consistent for the training and the testing phases of the machine translation system, it is expected to gain improvement while having a limited amount of parallel data resource.

4.3 Rule-based Morphological Analysis

The most accurate approaches in the literature for the morphological segmentation of the words are the rule-based approaches. The rules written by linguists usually outperforms the ones based on supervised, semi-supervised or unsupervised statistical approaches; although, the rule-based ones require more effort and time to be implemented. Due to the lack of human resource to design the language-dependent morphological rule set, this approach may be replaced by the statistical ones.

In this study, for the rule-based experiments, one of the very rare rule-based morphological analyzers for Turkish among the literature is used [45]. This analyzer uses a finite state machine designed to consider the morphotactic rules in Turkish.

4.4 Fusion-Based Word Decomposition

It has been mentioned that a complete machine translation system consists of three major modules, which can be listed as the pre-processing machine, the translation machine and the post-processing machine. Since the translation machine consumes only the intermediate output generated by the pre-processing machine, the output of the pre-processing machine may not necessarily be grammatical in terms of morphotactics, but deterministic and consistent.

Fusion-based word decomposition, as the name implies, proposes to make use of more than one approach to split the inflected words into smaller items. In the scope of this thesis, the statistical approaches are used along with the rule-based ones for the fusion-based word decomposition experiments.

Initially, the raw input data is analyzed by the rule-based and statistical word decomposition approaches separately. These individual approaches are explained in detail above. The raw input corpus is processed by using these two different methods and two different output hypothesis sets are obtained. Then, the fusion decomposition module picks a hypothesis either from the first set or the second set for a given input word as shown in Figure 4.1.

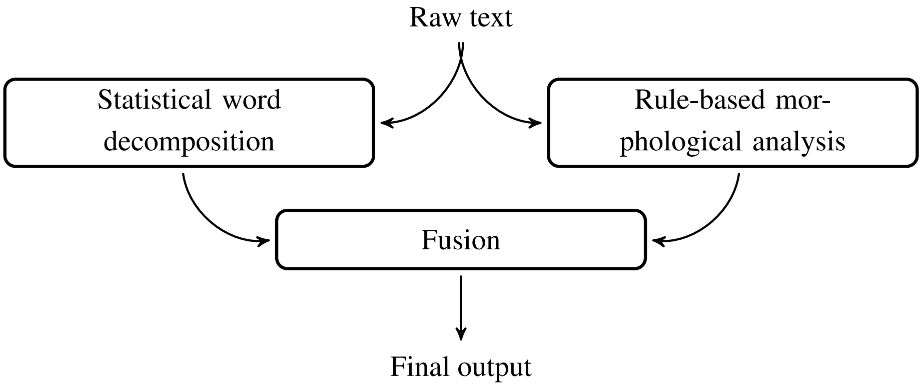
Raw text
Statistical word decomposition
Rule-based morphological analysis
Fusion
Final output

Figure 4.1: Fusion-based word decomposition schema

Fragmentation of the Data Set

The fusion-based approach fragments the training data into two subsets. The rule-based morphological analyzer is used to find out the subword items of the first subset, whereas the unsupervised word decomposition approach performs the same task for the second subset. This fragmentation operation is performed according to the word frequencies in the training data. Intuitively, the statistical word decomposition approach might find out the subword items more accurately if the amount of the training instances is sufficient. However, the rule-based morphological analysis approach does not benefit from the statistical characteristics of the training data. Therefore, the statistical word decomposition approach is used for frequent words, and the rule-based morphological analysis approach is used for the rare words in the training data.

The classification algorithm of the words as frequent and rare ones prior to the late fusion-based word decomposition is shown as Algorithm 1. The time complexity of this algorithm can be denoted as $O(n)$ with respect to the input size.

Algorithm 1: Fragmentation of the data set for fusion-based experiments

```
Input: Set of words in training data
Output: Two subset of words
foreach Word w do
    CountOccurences(w);
foreach Threshold t do
    foreach Word w do
        if Occurence(w) $<$ t then
            MarkAsRare(w);
        else
            MarkAsFrequent(w);
```

4.5 Data Description

In this section, all the linguistic resources used during this study are explained in detail. For the sake of compatibility, a public parallel corpus of Balkan languages is used for all linguistic data in this study [58]. This data is derived from the SETimes news portal at `http://www.setimes.com`, which publishes news in 10 different languages.

The SETimes corpus is used for both monolingual and bilingual data. For the monolingual part, only the related language is extracted from the corpus and used. For language modeling, it is recommended to use much larger corpora instead of a limited one; however, in this study no data except the SETimes resource is used. The main reason behind this choice is to enable easier comparisons of this study with others.

4.6 Monolingual Data

In this study, monolingual data is only used for the English language model generation step. The generated language model is common for all experiments conducted during the study. Only the English part of the training corpus is used for the language model generation purpose, and neither the development data nor the evaluation data is used at this step to build a fair model.

The detailed description of the monolingual data is shown in Table 4.1.

The reason why the number of sentences used for the language model generation is less than 160K is that the language model generator discards sentences having more than 40 words to generate a set of robust n-grams.

Since the source of the corpus used in this study is a universal news agency, the data mostly involves sentences in the news domain. This situation makes the overall system slightly domain dependent for the final usage. This slight dependency explains why the number of unique words in the corpus are lower than expected. However, it is important that the system is trained, optimized and evaluated using data from the same domain which satisfies the consistency condition of the system.

Table 4.1: Details of pre-processed monolingual data used for the language model generation

Metric	Value
Number of sentences	145,749
Number of total words	4,470,083
Number of unique words	43.631
Number of total characters	25,348,017
Number of words per sentence	30.67
Number of characters per word	5.67

4.7 Parallel Data

A parallel data, or parallel text, can be defined as a text which are aligned pair of translations in the machine translation area. In the case of the thesis, a parallel data is a pair of files, each of which contains a sentence on each line. For example, the i^{th} sentence (line) of the first file and the i^{th} sentence (line) of the second file must be the translations of each other.

The statistics regarding the parallel text depends on the experiment type which can be listed as word-based, rule-based decomposition, and statistical decomposition experiments. It is important to express that only the Turkish side of the parallel text changes depending on the experiment type. This is because all sub-word experiments are conducted for Turkish language.

The fragmentation of the parallel data as the training, development and evaluation sets is shown in Table 4.2. The training, development and evaluation sets are fragmented by using random sampling of the whole corpus.

Table 4.2: Fragmentation of the parallel text

	Number of sentences
Training set	160,000
Development set	500
Evaluation set	500

Since the training and the optimization phases require powerful computational units for the statistical machine translation tasks, cross validation methods may not be feasible for such tasks. Therefore, the development and evaluation sets are built by using random sampling and the same data sets are used for the same type of experiments in this thesis.

4.7.1 Word-Based Parallel Data

Word-based parallel data is obtained by pre-processing the raw data obtained from the SE-Times source. In this experiment type, all words in the Turkish corpus are represented as they are without changing any property of the surface form.

The detailed description of the parallel data for the word-based experiments is shown in Table 4.3.

Table 4.3: Details of pre-processed parallel data used for the translation model generation

Metric	Turkish	English
Number of sentences	160,000	160,000
Number of total words	4,145,871	4,593,299
Number of unique words	112,676	44,957
Number of total characters	29,919,406	26,421,412
Number of words per sentence	25.91	28.71
Number of characters per word	7.22	5.75

By observing the statistics above, it is clear that, on the average, the words in Turkish are longer than the words in English. Moreover, the number of words in a sentence for Turkish is larger than the number of words in a sentence for English. These two important details also explain the necessity of the word decomposition in Turkish, to some extent.

The algorithm run for the word-based experiments to produce the intermediate input prior to the SMT system is shown as Algorithm 2. The time complexity of this algorithm can be denoted as $O(n)$ with respect to the input size.

Algorithm 2: Processes prior to SMT for word-based experiments

```
Input: Pairs of sentences in Turkish and English
Output: Pre-processed pairs of sentences in Turkish and English
foreach SentencePair $(t,e)$ do
    Tokenize($t,e$);
    Lowercase($t,e$);
    NormalizeNumbers($t,e$);
    MarkSentenceBoundary($t,e$);
```

Case Study

Input: Kalıntılar 60 metre derinlikteki bir çukurda bulundu.

Regardless of the experiment type, every input text is pre-processed prior to the machine translation operations. This pre-processing step involves case normalization, number normalization, punctuation normalization and sentence boundary marking, as explained in Section 5.1.

Pre-processed intermediate input: <s> kalıntılar $number$ metre derinlikteki bir çukurda bulundu . </s>

After pre-processing the input text, the translation model queries the phrase table, which is an extensive phrase-based dictionary with forward and backward translation probabilities. Since the word-based experiments are the baseline model for this study, no operation, other than the common pre-processing steps, is performed. The intermediate input, produced by the pre-processing operations and shown above, is directly fed to the translation machine.

The phrase-based statistical machine translation system trained by word-based data set with 160K pairs of sentences produced the following intermediate output:

Intermediate output: <s> the lost of the pit were found at a depth of $number$ meters . </s>

After obtaining the intermediate output from the translation machine, the intermediate output is processed to be transformed to human readable form, which is called post-processing. This operation is explained in Section 5.5. The post-processed output of the system is shown below.

Output: The lost of the pit were found at a depth of 60 meters.

Expected output: The remains were discovered in a pit 60 meters deep.

Serious semantic and grammatical errors can be seen resulting in incorrect translation. The output sentence implies that the remains of a pit were found; however, the input sentence expresses that the remains were found in a pit. The intuitive justification behind this result is that the word-based data set may still involve data sparsity even with 160K pairs of sentences. This is due to the words being used in their surface form without any sort of utilization of subword level items. Therefore, the suffixes may result in completely different words, and they may potentially increase the unique word counts. Finally, the data sparsity may result in inaccurate translations as shown above.

4.7.2 Rule-Based Word Decomposition of Parallel Data

In the literature, rule based approaches are the most popular approaches for word decomposition as a pre-processing step of machine translation systems for Turkish [56]. A linguistically motivated rule-based morphological analyzer [45] is used for the morphological decomposition of Turkish words. This analyzer produced a set of analysis candidates for each word in Turkish. In other words, a Turkish word can be morphologically analyzed in many different ways. To disambiguate the ambiguous decomposition analysis and produce a single hypothesis for each word, a perceptron-based morphological disambiguation for Turkish is used [52]. By using the combination of the morphological parser and the disambiguator, Turkish words are analyzed and the stems and morphemes are separated from each other. Each stem and morpheme is treated as if they are separate words.

The detailed description of the parallel data for the rule-based word decomposition experiments is shown in Table 4.4.

Since the inflectional morphological analysis is performed on the Turkish side of the parallel text, a single word is fragmented into several sub items. The average number of the token per word metric denotes that a single Turkish word produces 3.16 tokens on average. These tokens are the stem of the word and the morphemes which follow the stem.

It is shown that there is a considerable amount of increase in the number of total words in Turkish text after the morphological decomposition is performed. The new number of to-

Table 4.4: Details of pre-processed parallel data used for rule-based word decomposition experiments

Metric	Turkish	English
Number of sentences	160,000	160,000
Number of total words	13,084,518	4,593,299
Number of unique words	39,861	44,957
Average number of tokens per word	3.16	1.00

tal words in Turkish is almost twice of its English counterpart. Therefore, it is suggested to reduce the density of the morphological fragmentation to some extent to ease the word alignment task. In a pair of sentences, the closer the number of words results in a better word alignment performance in general. Since the scope of the thesis does not aim to reach the best accuracy in the translation systems from Turkish to English, this task is left as a possible future work.

Another important result of the statistics above is the decrease in the unique word counts on the Turkish side. The unique word count is decreased to $1/3$ of the original value after the morphological analysis. This is another factor which decreases the computational complexity of the problem.

The algorithm run for the rule-based morphological analysis experiments to produce the intermediate input prior to the SMT system is shown as Algorithm 3. The time complexity of this algorithm can be denoted as $O(n)$ with respect to the input size.

Algorithm 3: Processes prior to SMT for rule-based morphological analysis experiments

```
Input: Pairs of sentences in Turkish and English
Output: Pre-processed pairs of sentences in Turkish and English
foreach SentencePair $(t,e)$ do
    Tokenize$(t,e)$;
    Lowercase$(t,e)$;
    NormalizeNumbers$(t,e)$;
    MarkSentenceBoundary$(t,e)$;
foreach Sentence $t$ do
    foreach Word $w$ in $t$ do
        AnalysisHypotheses $h$ = PerformMorphologicalAnalysis$(w)$;
        $w$ := DisambiguateHypotheses$(t, h)$;
```

Case Study

Input: Kalıntılar 60 metre derinlikteki bir çukurda bulundu.

Prior to the machine translation operations, every input text is pre-processed regardless of the experiment type. This pre-processing step involves case normalization, number normalization, punctuation normalization and sentence boundary marking, as expressed in Section 5.1.

Pre-processed intermediate input: <s> kalıntılar $number$ metre derinlikteki bir çukurda bulundu . </s>

Between the pre-processing and decoding steps, the rule-based morphological analyzer finds the stem and morpheme hypotheses. In other words, the rule-based morphological analysis approach splits the agglutinated words into stems and morphemes by using linguistic rules.

The ambiguous analysis results can be shown in Table 4.5.

Table 4.5: Ambiguous analyses resulted by the rule-based morphological analyzer

Input word	**Ambiguous hypotheses**
kalıntılar	kalıntı _a3pl kalıntı _p3pl
metre	metre
derinlikte	derin _ness _loc _rel
bir	bir
çukurda	çukur _loc
bulundu	bulun _pos _past bulun _pass _pos _past

The morphological analysis does not assign any probability or weight to the ambiguous analyses, which are shown in Table 4.5. Every hypothesis given the input word has an equal probability to be selected as the final hypothesis.

Having the ambiguous results, the perception-based morphological disambiguation approach is used to reveal the correct hypotheses among the ambiguous ones. It is reported that this approach results in 98% accuracy [52]. This approach is widely called word sense disambiguation (WSD).

Given all of the ambiguous hypotheses, WSD picks only one hypothesis, and discards the rest of them. This decision is made according to the neighborhood of the input word. The unambiguous results for the given input sentence are shown in Table 4.6.

Unambiguous hypotheses give the following sentence.

Intermediate input: <s> kalıntı _a3pl $number$ metre derin _ness _loc _rel bir çukur _loc bulun _pos _past . </s>

Only stems and morphemes are shown in the intermediate input above. All the allomorphs, which can be defined as the variant forms of a morpheme, are replaced by the common name morpheme. This replacement significantly reduces the OOV rate.

After performing the rule-based morphological analysis, the translation model queries the phrase table, which is an extensive phrase-based dictionary with forward and backward translation probabilities. The intermediate input shown above is fed to the translation machine.

Table 4.6: Unambiguous analyses produced by the perception-based WSD

Input word	Unambiguous hypotheses
kalıntılar	kalıntı _a3pl
metre	metre
derinlikte	derin _ness _loc _rel
bir	bir
çukurda	çukur _loc
bulundu	bulun _pos _past

The phrase-based statistical machine translation system trained by rule-based data set with 160K pairs of sentences resulted the following intermediate output:

Intermediate output: <s> ruins around $number$ meters at the deep , explains they found in soon . </s>

After being obtained from the translation machine, the intermediate output is processed to transform it to the human readable form, which is called post-processing. This operation is explained in Section 5.5. The post-processed output of the system is shown below.

Output: Ruins around 60 meters at the deep, explains they found in soon.

Expected output: The remains were discovered in a pit 60 meters deep.

Like the word-based case study, the output still has grammatical and semantic errors which result in incorrect translation. In addition, the output hypothesis does not mention enough information about if the place where the remains were found is a pit or not. The rule-based morphological analysis result in much longer set of tokens than any other approach introduced in this study. Such a long sequence of tokens might be translated to irrelevant words, prepositions, or punctuation marks. Therefore, this thesis proposes to use the morphological analysis approach only when the training data is not sufficient. In Section 6.2, the minimum sufficient training data to overcome the need for the morphological analysis is shown as 30K pairs of sentences.

4.7.3 Statistical Word Decomposition of Parallel Data

In machine translation, the major aim in the morphological analysis is to shrink the number of unique words in a language and make the sentences of the language pair as similar to each other as possible. Because of the hardness of the morpheme generation, all morphological operations are performed only on the source language.

The density level of the morphological analysis, or how aggressively the words are split into morphemes, does not affect the difficulty of the sentence generation in English, since none of the morphological operations is performed on the English side of the corpus. This flexibility also does not require a linguistically correct analysis of the morphemes. It is quite possible that some statistical approaches to split the words may still result good translation performance.

In the literature, unsupervised word decomposition has been recently used for the machine translation task for various inflectional languages including Turkish [40], [59]. These studies mostly use the same unsupervised word decomposition tool Morfessor [13] to fragment the stems and morphemes. In the thesis, the approach explained in Section 4.2 is used instead of the Morfessor tool. According to the quick evaluations manually, the explained approach and the toolkit performs very similarly.

The detailed description of the parallel data for the statistical word decomposition experiments is shown in Table 4.4.

Table 4.7: Details of pre-processed parallel data used for statistical word decomposition experiments

Metric	Turkish	English
Number of sentences	160,000	160,000
Number of total words	6,248,370	4,593,299
Number of unique words	46,802	44,957
Average number of tokens per word	1.51	1.00

As shown by the statistics above, the number of sub-word units of the statistical word decomposition is smaller than the number of sub-word units of the rule-based word decomposition. This implies that the strength of the decomposition decreases when the machine learning techniques are used.

This unsupervised approach for word decomposition processes the words without considering their neighbors. Moreover, it processes the character sequences without using any linguistic resource. Each word is decomposed in a single way which does not produce ambiguous decomposition. Hence, unlikely unsupervised word decomposition of words does not require an disambiguation phase.

The algorithm run for the statistical word decomposition experiments to produce the intermediate input prior to the SMT system is shown as Algorithm 4. The time complexity of this algorithm can be denoted as $O(n)$ with respect to the input size.

Case Study

Input: Kalıntılar 60 metre derinlikteki bir çukurda bulundu.

Prior to the machine translation operations, every input text is pre-processed regardless of the experiment type. This pre-processing step involves case normalization, number normalization, punctuation normalization and sentence boundary marking, as expressed in Section 5.1.

Pre-processed intermediate input: <s> kalıntılar $number$ metre derinlikteki bir çukurda bulundu . </s>

Algorithm 4: Processes prior to SMT for statistical word decomposition experiments

```
Input: Pairs of sentences in Turkish and English
Output: Pre-processed pairs of sentences in Turkish and English
foreach SentencePair $(t, e)$ do
    Tokenize($t, e$);
    Lowercase($t, e$);
    NormalizeNumbers($t, e$);
    MarkSentenceBoundary($t, e$);
foreach Sentence $t$ do
    foreach Word $w$ in $t$ do
        for $i := 0$ to Length(w) $-1$ do
            2GramCount($w_i, w_{i+1}$)++;
$sum := 0$;
$count := 0$;
foreach LetterPair $(m, n)$ do
    $sum$ += 2GramCount($m$,$n$);
    $count$ ++;
$average2GramCount := sum / count$;
foreach Sentence $t$ do
    foreach Word $w$ in $t$ do
        for $i := 0$ to Length(w) $-1$ do
            if 2GramCount($w_i, w_{i+1}$) $<$ $average2GramCount$ then
                DecomposeWordAtIndex($w$,$i$);
```

Between the steps pre-processing and decoding, the stochastic word decomposition approach finds the stem and morpheme hypotheses. In other words, the statistical word decomposition approach splits the agglutinated words into stems and morphemes.

The output of the stochastic word decomposition method is shown in Table 4.8.

Table 4.8: Hypotheses produced by the statistical word decomposition approach

Input word	**Unambiguous hypotheses**
kalıntılar	kalın _tı _lar
metre	metre
derinlikte	derin _lik _te _ki
bir	bir
çukurda	çukur _da
bulundu	bul _undu

The hypotheses listed above are the actual outputs of the unsupervised word decomposition module. This module is explained in Section 4.2.3 in detail. The combination of forward and backward character-based n-gram language models are used to find the morpheme boundary hypotheses. It should be said that this approach does not require any linguistic rule set, or any kind of data annotation. The only requirement is a large amount of raw text in a language. Once the word decomposition module is trained by such a raw text, it is able to guess the stem and morphemes.

This stochastic method does not necessarily find the grammatically correct stem and morphemes, but the most likely character sequences given a raw corpus. This explains that word *kalıntılar* is decomposed as *kalın _tı _lar*, which is grammatically incorrect; although *kalın* is a valid stem in Turkish. Similarly the word *bulundu* is decomposed as *bul _undu* which is not correct. The correct analysis of these words can be obtained by the rule-based morphological analyzers for some other specific applications.

The stochastic decomposition may result in ungrammatical hypotheses, which has been expressed before. However, this is not the final output of the system and this intermediate output is consumed by the translation machine. Therefore, a deterministic approach resulting in ungrammatical hypotheses can still boost the overall translation accuracy of the MT system.

Hypotheses generated by the statistical word decomposition approach produce the following sentence.

Intermediate input: <s> kalın _tı _lar $number$ metre derin _lik _te _ki bir çukur _da bul _undu . </s>

The intermediate input above shows that it has only stems and morphemes. All allomorphs, which can be defined as the variant forms of a morpheme, are replaced by the common label of the morpheme. This replacement significantly reduces the OOV rate.

After performing the statistical word decomposition process, the translation model queries the phrase table, which is an extensive phrase-based dictionary with forward and backward translation probabilities. The intermediate output, produced by the pre-processing operations and shown above, is fed directly to the translation machine. This process is the same as the one used for the previous experiments.

The phrase-based statistical machine translation system trained by statistically decomposed data set with 160K pairs of sentences resulted the following intermediate output:

Intermediate output: <s> at a depth of $number$ meters , they found . </s>

After obtaining the intermediate output from the translation machine, the intermediate output is processed to transform it to the human readable form, which is called post-processing. This operation is explained in Section 5.5. The post-processed output of the system is shown below.

Output: At a depth of 60 meters, they found.

Expected output: The remains were discovered in a pit 60 meters deep.

The output seems to be grammatically correct; although, there is still considerable semantic loss. The output hypothesis does not mention enough information regarding what they found and the place where they found the remains. The main reason behind this loss is that some of the word decomposition hypotheses, including *kalın _tı _lar*, are incorrect.

Furthermore, the total token count after the stochastic word decomposition is significantly lower than the rule-based morphological analysis data set, which may result semantic information loss.

4.7.4 Fusion-Based Parallel Data

In the machine translation area, contemporary approaches have increasingly focused on the hybrid or fusion-based methods which congregate multiple techniques. Such methods make use of the information derived by the statistical properties of the data along with the linguistic rules designed by expert linguists. The major motivation behind the fusion-based studies is to benefit from the statistical characteristics of the large amount of data and the linguistic rules for the small amount of data.

The fusion-based approach fragments the training data into two subsets. The rule-based morphological analyzer is used to find out the subword items of the first subset, whereas the unsupervised word decomposition approach performs the same task for the second subset. This fragmentation operation is performed according to the word frequencies in the training data. Intuitively, the statistical word decomposition approach might find the subword items more accurately if the training instances are dense. However, the rule-based morphological analysis approach does not benefit from the statistical characteristics of the training data. Therefore, the statistical word decomposition approach is used for the frequent words, and the rule-based morphological analysis approach is used for the rare words in the training data.

The detailed description of the parallel data for the fusion-based word decomposition experiments is shown in Table 4.9.

It has been determined that the average number of tokens per word (the number of subword items derived by splitting the actual word) for the fusion-based experiments is between the

Table 4.9: Details of pre-processed parallel data used for fusion-based word decomposition experiments

Metric	Turkish	English
Number of sentences	160,000	160,000
Number of total words	11,021,317	4,593,299
Number of unique words	60,025	44,957
Average number of tokens per word	2.49	1.00

average number of tokens per word for the rule-based morphological analysis and the average number of tokens per word for statistical word decomposition experiments. This implies that the intensity of the word decomposition operation for the fusion-based experiments is between the intensities of the rule-based and statistical approaches, which yields better accuracy, as shown and discussed in Section 6.2.

The algorithm run for the fusion-based word decomposition experiments to produce the intermediate input prior to the SMT system is shown as Algorithm 5. The time complexity of this algorithm can be denoted as $O(n)$ with respect to the input size.

Case Study

Input: Kalıntılar 60 metre derinlikteki bir çukurda bulundu.

Prior to the machine translation operations, every input text is pre-processed regardless of the experiment type. This pre-processing step involves case normalization, number normalization, punctuation normalization and sentence boundary marking, as expressed in Section 5.1.

Pre-processed intermediate input: <s> kalıntılar $number$ metre derinlikteki bir çukurda bulundu . </s>

Between the pre-processing and decoding steps, the fusion-based word decomposition module finds the stem and morpheme hypotheses. In other words, the fusion-based word decomposition approach splits the agglutinated words into stems and morphemes. The frequency threshold value 10 is used for this case study since it results the highest accuracy.

To categorize the words as frequent or rare, their occurrences are counted, as shown in Table 4.10.

According to the categories of the input words the decomposition method is decided. The statistical approach is used for the frequent words, since the stochastic approaches benefit from the dense data. The rule-based approach is used for the rare words according to the frequency threshold value. The rare words are also disambiguated after the rule-based morphological analysis by using the same approach in [52].

Each word is decomposed by using an appropriate method according to the frequencies of the words. This late fusion approach results in the word decomposition hypotheses, as shown in

Algorithm 5: Processes prior to SMT for fusion-based word decomposition experiments

```
foreach Sentence t do
    foreach Word w in t do
        for $i := 0$ to Length(w) $-1$ do
            2GramCount($w_i, w_{i+1}$)++;
sum := 0;
count := 0;
foreach LetterPair $(m, n)$ do
    sum $+=$ 2GramCount(m,n);
    count $++$;
average2GramCount := sum / count;
Input: Set of words in training data
Output: Two subset of words
foreach Word w do
    CountOccurences(w);
foreach Threshold t do
    foreach Word w do
        if Occurence(w) $< t$ then
            AnalysisHypotheses h = PerformMorphologicalAnalysis(w);
            w := DisambiguateHypotheses(t, h);
        else
            for $i := 0$ to Length(w) $-1$ do
                if 2GramCount($w_i, w_{i+1}$) $<$ average2GramCount then
                    DecomposeWordAtIndex(w,i);
```

Table 4.10: Frequent and rare word categorization at the pre-fusion step

Input word	**Occurrence**	**Category**	**Decomposition method**
kalıntılar	9	Rare	Rule-based morphological analysis
metre	138	Frequent	Statistical word decomposition
derinlikteki	3	Rare	Rule-based morphological analysis
bir	57,464	Frequent	Statistical word decomposition
çukurda	2	Rare	Rule-based morphological analysis
bulundu	3356	Frequent	Statistical word decomposition

Table 4.11.

Table 4.11: Hypotheses resulted by the fusion-based word decomposition approach

Input word	**Fusion hypotheses**
kalıntılar	kalıntı _a3pl
metre	metre
derinlikte	derin _ness _loc _rel
bir	bir
çukurda	çukur _loc
bulundu	bulun _undu

As it is shown in Table 4.11, both the allomorphs (_undu) and names of morphemes (_a3pl, _ness, _loc, _rel) are used together. This fact slightly increases the unique word counts in the resulting fusion-based data set. The unique word counts in fusion-based data set is higher than both the statistical and rule-based data set, but lower than the baseline data set. This implies that the fusion-based approach has the lowest word decomposition intensity.

Fusion-based word decomposition approach produces in the following sentence.

Intermediate input: <s> kalıntı _a3pl $number$ metre derin _ness _loc _rel bir çukur _loc bul _undu . </s>

The intermediate input above shows that it has stems, allomorphs, and the names of the morphemes all together.

After performing the fusion-based word decomposition operation, the translation model queries the phrase table, which is an extensive phrase-based dictionary with forward and backward translation probabilities. The intermediate output, produced by the pre-processing operations and shown above, is fed directly to the translation machine.

The phrase-based statistical machine translation system trained by fusion-based data set with 160K pairs of sentences resulted in the following intermediate output:

Intermediate output: <s> the remains at a depth of *number* meters that they found in a pit .

</s>

After obtaining the intermediate output from the translation machine, the intermediate output is processed to transform it to the human readable form, which is called as post-processing. This operation is explained in Section 5.5. The post-processed output of the system is shown below.

Output: The remains at a depth of 60 meters that they found in a pit.

Expected output: The remains were discovered in a pit 60 meters deep.

The output is not a grammatically complete sentence, but a phrase; however, it includes all of the necessary semantic information carried by the input sentence. The grammatical structure and the semantic information captured by the output is the strongest candidate for the best translation among the outputs of other approaches. The BLEU scores of the fusion-based approach also clearly support this evaluation, as shown in Section 6.2.

CHAPTER 5

BUILDING PHRASE-BASED STATISTICAL MACHINE TRANSLATION SYSTEM

5.1 Pre-processing

Pre-processing is a vital part for almost all natural language processing tasks, including the machine translation problem. In this study, the statistical machine translation task aims to translate an infinite set of sentences in the source language to the target language. To decrease complexity, some words or letters should be marked, inserted, or removed in advance. Converting the natural language sentences into the input form of the decoder, or the translation machine, is named as the pre-processing in general.

Pre-processing the input sentence requires post-processing the output sentence to convert it from the decoder output form to the natural language sentence. This can also be considered as the restoration of the hypotheses. Thus, the density or the weight of the pre-processing is the same as the density or the weight of the post-processing. Therefore, the pre-processing density should not be at the intensive level in order for the difficulty of the output restoration process not to increase.

In this study, the pre-processing step involves the punctuation normalization, case normalization, numerical normalization and sentence boundary marking operations in the both source language and the target language, Turkish and English. These four normalization operations allow deterministic restoration of the hypotheses at the post-processing step.

5.1.1 Punctuation Normalization

The punctuation marks in natural languages are usually appended to the preceding word, but separated from the next word. Even if the punctuation marks are appended to the preceding word, they mostly affect the sentence level meaning instead of the word level meaning. Therefore, it is important that the appended punctuation should be separated from the preceding word as well.

In addition, some punctuation marks like the quotation mark or the apostrophe are attached to the next word as well. For the same reason above, these punctuation marks should be separated from the words that they are attached.

An example for the punctuation normalization process is shown below.

$$\mathit{Tuzluluk}, \mathit{Akdeniz}'\mathit{de}\ \%3.8'\mathit{dir.} \longrightarrow \mathit{Tuzluluk}\ ,\ \mathit{Akdeniz}\ '\ \mathit{de}\ \%\ 3.8\ '\ \mathit{dir}\ . \tag{5.1}$$

5.1.2 Case Normalization

Like many languages, the first letter of each sentence is written with a capital letter. In the string processing level, the machine recognizes the words starting with a lowercase and the ones starting with an uppercase differently, despite the fact that they carry the same semantic information. To remove such difference in the syntactic level, all letters are lowercased at the very first step.

An example for the case normalization process is shown below.

$$\mathit{Tuzluluk}\ ,\ \mathit{Akdeniz}\ '\ \mathit{de}\ \%\ 3.8\ '\ \mathit{dir}\ . \longrightarrow \mathit{tuzluluk}\ ,\ \mathit{akdeniz}\ '\ \mathit{de}\ \%\ 3.8\ '\ \mathit{dir}\ . \tag{5.2}$$

5.1.3 Numerical Normalization

Numerical normalization is a highly important pre-processing operation for the machine translation task to reduce the out-of-vocabulary rate and the complexity of the problem. Similar to the words, there are infinite numbers in every language. It should be realized that the translations of the numerical expressions are same in both the source language and the target language. Therefore, including numerical expressions in the translation table or the language model seems useless.

In the pre-processing step, all the decimal or fractional numbers, and the numerical date and time expressions are replaced by a common label to shrink the search space as much as possible. The original values of the numerical expressions must be saved before the replacement process to be able to recover them in the hypothesis translation sentence.

An example for the numerical normalization process is shown below.

$$\mathit{tuzluluk}\ ,\ \mathit{akdeniz}\ '\ \mathit{de}\ \%\ 3.8\ '\ \mathit{dir}\ . \longrightarrow \mathit{tuzluluk}\ ,\ \mathit{akdeniz}\ '\ \mathit{de}\ \%\ \$frac\$\ '\ \mathit{dir}\ . \tag{5.3}$$

5.1.4 Sentence Boundary Marking

Sentence boundary marking is a minor but important part of the pre-processing phase. The beginning and the end of each sentence is marked accordingly. The reason behind this marking process is to help the language modeling toolkit to take the places of the words into account correctly. The underlying semantics of the words at the beginning of the sentence or at the end of the sentence may be quite different from each other.

An example for the sentence boundary marking process is shown below.

$$tuzluluk\,,\,akdeniz'\,de\,\%\,\$frac\$'\,dir\,.\longrightarrow < s > tuzluluk\,,\,akdeniz'\,de\,\%\,\$frac\$'\,dir\,.\,< /s > \tag{5.4}$$

5.2 Language Model

Since the thesis scope aims to evaluate the various machine translation systems from the Turkish language to the English language, the language model for only the target language, English, is generated during the whole study. To be consistent for the all the experiments and conduct controlled experiments, the same language model for English is used for all the experimental setups.

5.2.1 Methods Used

For the sake of the replicability of the study, the English side of the parallel training data is used for the language model generation task. The detailed data description used for the language model is explained in Section 4.6. IRSTLM toolkit [17] is used for the English language model generation in the standard ARPA format with the order of 5. During the generation of the model, the singleton words, which are the words which appear only once, are removed from the corpus. The reason behind this removal is to clean the possible noisy data from the corpus. Furthermore, the Witten-Bell [8] smoothing method is applied to abstain possible zero probabilities for some unseen n-grams. Then, by using the same toolkit, the language model is binarized to shrink the size and increase the querying efficiency of the model.

In addition, the same pre-processing operations used for the translation model generation of the system, explained in Section 5.1, are used for the pre-processing of the monolingual corpus to build the language model. For the consistency of the components, it is essential to perform the same pre-processing tasks for any text or data in the statistical machine translation systems.

In the resulting language model, the n-gram counts are mentioned in Table 5.1.

Table 5.1: N-gram counts in the English language model used in the study

N-Gram Order	N-Gram Count
1	47,699
2	649,534
3	381,345
4	327,260
5	239,588

An important remark regarding the scope of this study is that the aim of this thesis is to compare and evaluate the reactions of the different phrase-based machine translation approaches for the Turkish language when the corpus size changes. Therefore, finding the best parameters for the language model generation is out of the breadth of this thesis. Only the parameters which have been observed with relatively good performance and accuracy are used during this study.

5.2.2 Evaluation of Language Models

After the language model generation process as explained in Section 5.2.1, the generated model is evaluated using the same unseen test data used for the evaluation of the machine translation system. The English part of the test data is extracted and used for the evaluation. For the comparability of the results and the components used in the study, the perplexity and the out-of-vocabulary rates are measured by using the evaluation data which is described in Section 4.6.

The evaluation scores on the unseen test data of the English language model used during this study are shown in Table 5.2.

Table 5.2: English language mode evaluation scores on unseen test data

Metric	Value
Number of total words	12,783
Number of out-of-vocabulary words	651
Out-of-vocabulary rate	5.09%
Perplexity ignoring out-of-vocabulary words	125.92
Overall perplexity	224.93

5.3 Translation Model

The translation models used during this study are generated by processing the parallel sentence data in both Turkish and English. Since the translation model generation process involves the pre-processing of Turkish sentences which absolutely depend on the experimental setup, a unique translation model is generated for each of the experiments conducted in the scope of the study.

After the pre-processing of the parallel corpus phase, Moses toolkit [35] is used for the translation model generation. The sentence level parallel corpus is processed and the words are aligned to the corresponding translations [44]. As explained in Section 2.7.2, the phrases are derived from the parallel corpus. Each phrase pair is assigned a probability value which indicates how likely this translation is an accurate one. The translation model is the collection of all the phrases extracted from the parallel data. It involves even the least likely phrases with very low probability values. The largest translation table generated during this study is the one for the word-based experimental setup with 160,000 parallel sentences. This translation table involves more than 5 million phrase pairs with their probabilities.

An illustrative small part of the translation table is shown in Table 5.3.

Table 5.3: An illustrative phrase table segment

Turkish phrase	English phrase	Forward probability
altyapının	infrastructure	0.33
altyapının yeniden inşa edilmesinde	rebuild infrastructure	0.33
altyapının yeniden inşa edilmesinde	helped rebuild infrastructure	0.21
altyapının yeniden inşa	rebuild infrastructure	0.47
altyapının yeniden inşa	helped rebuild infrastructure	0.12

Forward translation probability means the probability of the translation from Turkish to English, and vice versa for the backward translation probability. During the experiments, both of the forward and backward translation probabilities are calculated and used for the decoding phase.

It is also shown that the phrase table may involve semantically incorrect but statistically possible phrase pairs as well.

Parameter Optimization

The parameter optimization phase is another cardinal part of the phrase-based statistical machine translation system. This parameter set includes the number of n-best hypotheses generated, distortion (reordering) weight, language model weight, translation model weight, penalty for the sentence length difference, and the maximum allowed reordering of the words in the hypothesis. For the optimization of these parameters, a development set is reserved. The translation system is run on this development set iteratively. After each iteration, the translation quality is measured by using the BLEU metric [49]. According to the changes in the score, the iteration either continues or stops and returns the best set of the parameters.

In the decoding phase, the parameters obtained by the parameter optimization phase are used.

5.4 Decoding

Decoding, reverse of encoding, is basically the process of transforming information from one format into another. In the thesis enclosure, it can be defined as the revealing the actual message, which is the translation hypothesis, by using the encrypted text, which is the input text in the source language.

The decoder engine of Moses toolkit [35] is used to generate of the translation hypotheses. This decoder makes use of both the language model and the translation model. The combination of models and their weights are defined by the optimized parameter set stored in a configuration file.

An important point is that the sentences have to be pre-processed before being fed to the decoder machine. This pre-processing phase must be essentially the same as the one in the training phase. The only difference between the pre-processing part of the training phase and the decoding phase is the numerical normalization task. In the training or parameter optimization step, the numerical values are replaced by some predefined labels as explained in Section 5.1.3. Since the reference translation is also pre-processed and the numerical expressions in it are also replaced by the same tag, the recovery of these numbers are not necessary. However, in the decoding phase, the restoration of the numbers and dates are essential. When a sentence involving numbers and dates is entered as an input to the system, it is expected to produce the correct numbers and dates in the translation rather than some strange labels. For this reason, the numerical expressions are wrapped by a special XML expression, so that both the label and the actual value is fed to the decoder. Then, the decoder treats this expression as if it is nothing more than one of the predefined labels while querying the language model and the translation table. However, at the time of the hypothesis generation, the actual value is outputted instead of the predefined label.

An example XML markup is shown below.

$$akdeniz'\,de\,\%\,3.8'\,dir\,. \longrightarrow; akdeniz'\,de\,\%\; < n\,translation = \text{"3.8"} > \$frac\$ < /n > \;'\,dir\,. \tag{5.5}$$

The semantic of this notation tells the correct translation of an expression to the decoder, so that it does not try to translate it. This XML markup method can also be used for different purposes, such as fixing the translation of some words. During this study, none of the translations of the words are fixed.

5.5 Post-processing

Post-processing can be defined as the modifying the intermediate output to improve its quality and produce the final output of the system in general. The post-processing phase cannot be considered apart from the pre-processing phase, because it aims to convert the decoder output to the punctuated human readable sentence. It is mostly used to revert the changes made during the pre-processing phase.

All operations performed before decoding like punctuation normalization, case normalization, numerical normalization and sentence boundary marking, have to be undone after the decoder runs. The order of this undoing process should be the reverse of their application order. This implies the first post-processing operation is the sentence boundary mark removal. After the sentence boundary mark removal, numerical normalization, case normalization and punctuation normalization are performed respectively.

$$< s > the\ salinity\ was\ \$frac\$\ percent. < /s > \longrightarrow the\ salinity\ was\ \$frac\$\ percent. \tag{5.6}$$

$$the\ salinity\ was\ \$frac\$\ percent. \longrightarrow the\ salinity\ was\ 3.8\ percent. \tag{5.7}$$

$$the\ salinity\ was\ 3.8\ percent. \longrightarrow The\ salinity\ was\ 3.8\ percent. \tag{5.8}$$

$$The\ salinity\ was\ 3.8\ percent. \longrightarrow The\ salinity\ was\ 3.8\ percent. \tag{5.9}$$

As it is shown above, after the post-processing step, the decoder output is transformed into the grammatically correct format in the target language.

This post-processing step is used for the conversion of the hypothetical output to the human readable format.

CHAPTER 6

EXPERIMENTS

6.1 Determination of the Threshold Value for Fusion-Based Approach

In order to make a deterministic fragmentation of the training data, a threshold value has to be defined. If a word appears more frequently (or equally) than the predefined threshold value, then it would be tagged as a frequent word. Otherwise, the word would be sent to the rare word subset. A set of experiments are performed to find out the threshold value resulting in the highest accuracy for the overall phrase-based statistical machine translation system.

Table 6.1: Fragmentation of the training data as frequent and rare words

Threshold value	Frequent words (%)	Rare words (%)
2	55	45
3	40	60
4	33	67
5	28	72
10	17	83
20	10	90
50	6	94
100	3	97
500	1	99

In order to learn the most accurate threshold value, 27 different complete machine translation systems are built using all the threshold values listed in Table 6.1. Three different threshold determination experiments are conducted. First of them is conducted by using a small amount of training corpus with 50K sentences. The second experiment is conducted by using a middle-sized training corpus with 100K sentences. Then, the last experiment is conducted by using the complete set of training data, 160K sentences. For the experiments with training data having less than 50K sentences, the first threshold value is used. For the experiments with training data having between 50K and 100K sentences, the second threshold value is used. For the experiments with training data having more than 100K sentences, the third threshold value is used.

A relatively small amount of training data, having 50K parallel sentences, is used to obtain

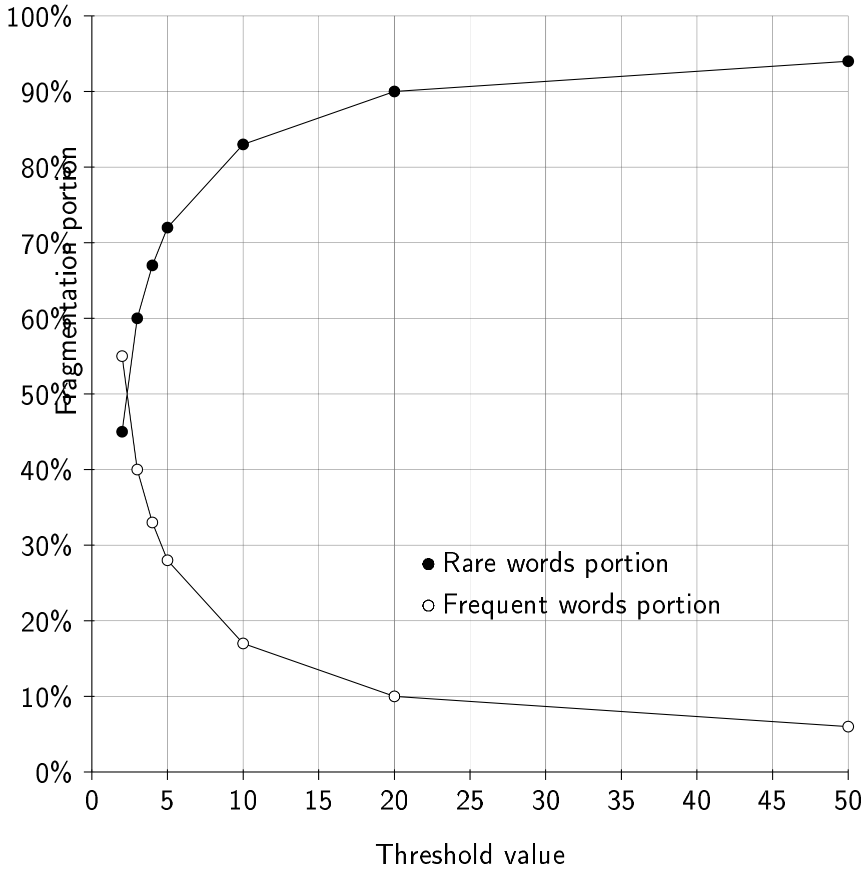

Figure 6.1: Fragmentation of the training data as frequent and rare words

the threshold values to be used for the experiments with training data having less than 50K sentences. The resulting BLEU scores are stated in Table 6.2.

A relatively middle-sized training data, having 100K parallel sentences, is used to obtain the threshold values to be used for the experiments with training data having between 50K and 100K sentences. The resulting BLEU scores are stated in Table 6.3.

A relatively large amount of training data, having 160K parallel sentences, is used to obtain the threshold values to be used for the experiments with training data having more than 100K sentences. The resulting BLEU scores are stated in Table 6.4.

As a result, the experiments having training data with less than 50K sentences use the threshold value of 50, the experiments having training data with between 50K and 100K sentences use the threshold value of 20, while the others use the threshold value of 10.

As a result, the parameters of the fusion approach are determined by conducting a set of controlled experiments. It should be acknowledged that the average length of tokens in Turkish is measured as 7.22 as shown in Table 4.3. This number increases to 8.12 if the single letter tokens are discarded. This measurement reflects the intuitive fact that the frequent words are the words appearing more frequently than the average word frequency in the language.

Table 6.2: BLEU score versus fragmentation threshold with 50K sentences

Threshold value	BLEU Score
2	19.20
3	19.23
4	19.34
5	19.69
10	19.82
20	19.98
50	**20.32**
100	20.26
500	20.26

Table 6.3: BLEU score versus fragmentation threshol with 100K sentencesd

Threshold value	BLEU Score
2	23.16
3	23.41
4	23.77
5	24.01
10	24.19
20	**24.28**
50	24.11
100	24.08
500	24.06

6.2 Corpus Size Experiments

In order to verify the success of the introduced frequency-driven late fusion-based word decomposition approach, a set of comparison experiments are conducted. The late fusion-based approach is compared with the pure rule-based and pure statistical approach by conducting numerous experiments when the training corpus size changes in a large range. The only changing parameter in the controlled experiments is the size of the training corpus. All the remaining parameters, such as pre-processing operations or the development and evaluation sets are fixed for the same type of the experiments.

Since the frequency-driven late fusion-based word decomposition approach is compared with the rule-based and statistical approaches, there are three different experiment types in this study, which can be listed in the word-based experiments, the rule-based word decomposition experiments, and the statistical word decomposition experiments. In the word-based experiments, all the words are used as their surface forms, and they are not decomposed into smaller units. Word-based experiments can also be named as the baseline experiments, because the

Table 6.4: BLEU score versus fragmentation threshold with 160K sentences

Threshold value	BLEU Score
2	24.29
3	24.96
4	26.02
5	25.09
10	**26.22**
20	25.96
50	25.52
100	25.26
500	25.12

remaining two types are obtained by applying some operations on this type of experiment. The second type is the rule-based word decomposition experiments. In this experiment, the stems and the morphemes of the words are found by using linguistic morphology rules [45]. Then, the morphology analyses are disambiguated by the help of the word context [52]. This operation is also called word sense disambiguation and it is used for many natural language processing problems. The last type of experiment is the statistical decomposition experiment. In these experiments, the words are split into the smaller units without any linguistic or morphological concerns. The smaller units which are obtained by the statistical word decomposition may be grammatically incorrect units, and there can be both the over stemming or under stemming as well. This is because this approach is based on unsupervised learning algorithms of artificial intelligence area, and it requires neither the linguistic resources nor the annotated corpora.

For each type of experiment, 16 different experiments are completed, and 16 different machine translation systems are built. Firstly, 10,000 parallel sentences are randomly extracted from the training corpus, and the first experiment is conducted by using only this 10,000 sentence part of the training data. After this experiment, another random sample of 10,000 sentences are appended to the training data. For example, a 20,000 sentence parallel text is used for the second experiment, a 30,000 sentence parallel text is used for the third experiment, and so on. The complete set of corpus, which is a 160,000 sentence parallel text, is used only for the last experiment of each type. The overall system description in this study is shown in Figure 6.2. The most accurate systems are colored as green.

By changing the corpus size by 10,000 at each experiment, the translation accuracy is evaluated using the BLEU scoring tool [49]. The results of the experiment types with different corpus sizes in the BLEU metric are shown in Figure 6.3.

The demonstrated results above reflect a number of interesting facts revealed in this thesis. In the literature, it is repeatedly stated that the morphological features can improve the baseline accuracy for the morphologically rich languages in machine translation tasks [16], [22], [46]. The motivation behind the morphological analysis or the word decomposition for the machine translation system is to reduce the number of out-of-vocabulary (unseen) words and split the items carrying semantic data themselves. By doing so, it is aimed to resolve the

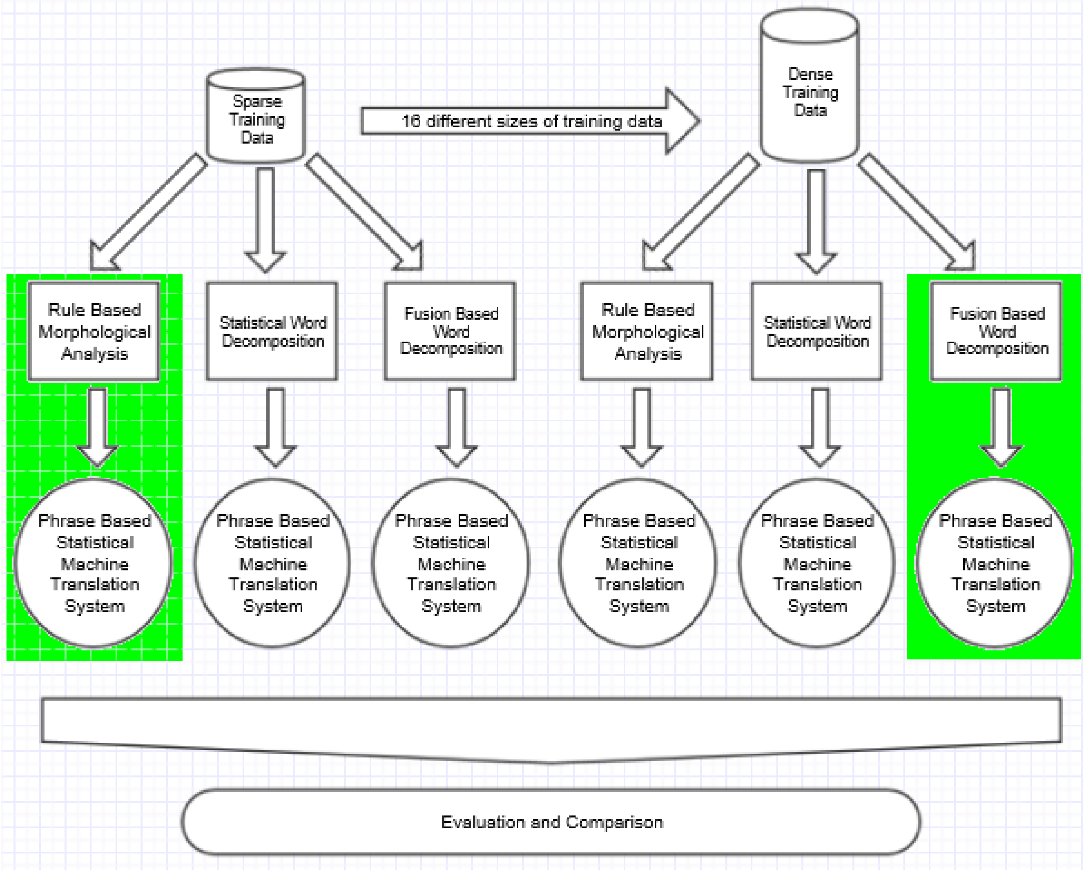
Sparse Training Data
16 different sizes of training data
Dense Training Data
Rule Based Morphological Analysis
Statistical Word Decomposition
Fusion Based Word Decomposition
Rule Based Morphological Analysis
Statistical Word Decomposition
Fusion Based Word Decomposition
Phrase Based Statistical Machine Translation System
Phrase Based Statistical Machine Translation System
Phrase Based Statistical Machine Translation System
Phrase Based Statistical Machine Translation System
Phrase Based Statistical Machine Translation System
Phrase Based Statistical Machine Translation System
Evaluation and Comparison

Figure 6.2: Overall system description

Table 6.5: BLEU scores of different complete machine translation systems

Number of training instances	Word-based	Rule-based	Statistical	Fusion-based
10K	10.11	12.40	13.14	**13.22**
20K	12.74	**14.84**	14.66	14.97
30K	14.80	**16.02**	14.91	15.41
40K	16.49	**17.40**	15.80	16.67
50K	19.57	20.26	19.15	**20.32**
60K	20.69	21.04	20.22	**21.39**
70K	21.81	22.13	21.98	**23.21**
80K	22.53	23.11	21.79	**23.30**
90K	23.31	23.78	22.54	**23.96**
100K	24.19	24.06	22.88	**24.28**
110K	24.61	24.40	23.09	**24.87**
120K	24.61	24.24	23.41	**25.06**
130K	24.66	24.26	23.11	**24.99**
140K	25.09	24.71	24.14	**25.69**
150K	25.28	24.99	24.04	**26.01**
160K	25.36	25.12	23.87	**26.22**

complexity of the problem as far as possible. This study reveals that the alternative way of the morphological analysis is to increase the training text size, which can result in even better results continuously. For the domain and the data set used in this thesis, it is proposed to have at least 50,000 sentence pairs to eliminate the necessity of the morphological analyzers. This number can be clearly shown by Figure 6.3. In addition, Figure 6.4 shows the detailed BLEU scores of the approaches with the dense training data. Moreover, this figure shows the state of the art BLEU score with the dashed line, 25.22, as stated in [58]. The word-based experimental setup results in very similar results compared to those reported in the literature; however, the fusion-based approach helps to gain a slight improvement, which is around a 1.00 BLEU score. Thereby, this study reveals the fact that making use of the statistical approaches along with the rule-based ones may yield better results. Instead of using pure rule-based or statistical approaches, it is proposed to combine them in such a way that the accuracy of the translation hypotheses increases. The method and parameters of the fusion experiments may be quite diverse which can be a good area of research and investigation for the future works.

Moreover, making use of the fusion-based approaches may yield better results than the rule-based ones and statistical ones since they benefit from the statistical properties of the dense portions of data. If the training data is large enough then hybrid-based approaches help to gain improvements in the accuracy. However, when the training data set is relatively small, namely less than 50,000 pairs of sentences, then the rule-based approaches still outperform the others. The reason behind this fact is that the probability distribution does not converge for the training instances due to the data sparsity.

In addition, the most widely used public machine translation systems, *Google Translate* and

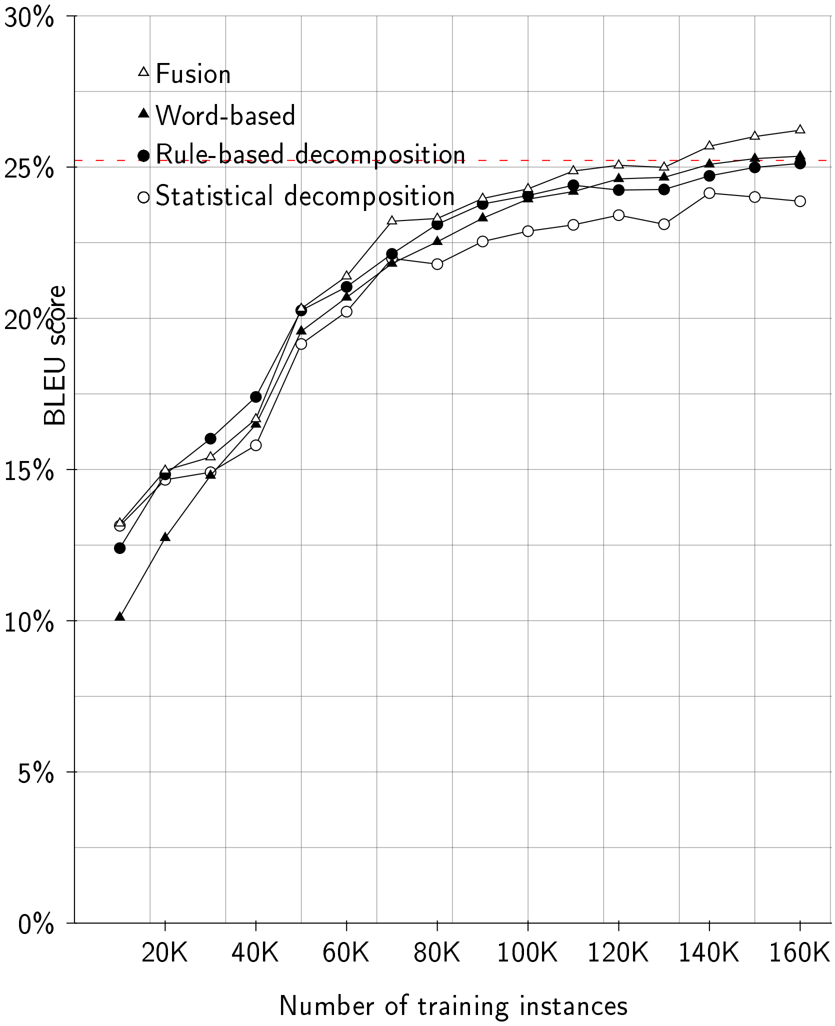

Figure 6.3: Number of parallel sentences used for the training versus BLEU score

Bing Translator, are evaluated using the same test data used during this study. The results were 23.05 and 23.79 BLEU score for *Google Translate* and *Bing Translator* respectively. Even the baseline scores reported in this thesis are higher than these two enterprise MT systems. The major reason behind this difference is that *Google Translate* and *Bing Translator* are machine translation systems which are completely domain independent. They can translate the sentences in very different domains with a consistent accuracy. However, the data set used in this study is in political news domain, and the models are evaluated on the same domain, which results in better BLEU scores than expected.

6.3 Error Analysis

The evaluation of the machine translation systems has always been a challenging problem in the literature. A sentence in a language can naturally have numerous different valid translation in another language. There is not an easy algorithm to score these valid translations and the

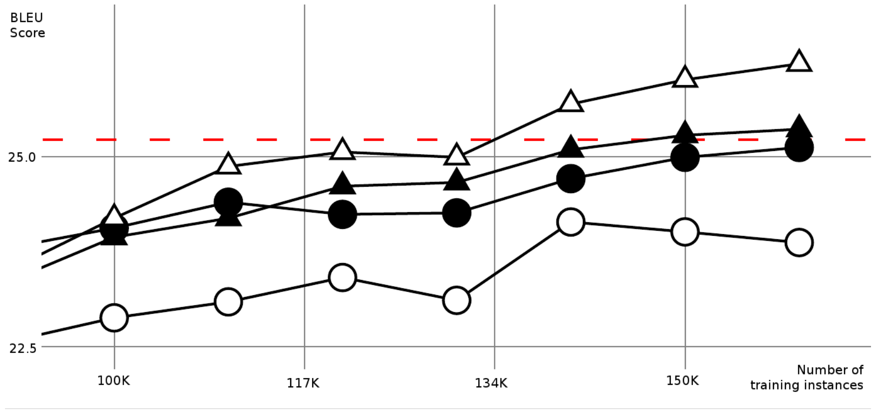

Figure 6.4: Number of parallel sentences used for the training versus BLEU score for the dense training data

Table 6.6: Reported BLEU scores of complete SMT systems

SMT System	BLEU Score
Baseline system	25.36
Fusion-based system	26.22
Benchmark study [58]	25.22
Google Translate	23.05
Bing Translator	23.79

others.

In this study, the translation outputs of almost every experiment are checked manually to ensure the correctness of the experimental setups. If any of the hypothesis translations is not as expected, the overall system is debugged and retrained if necessary.

By checking the translation hypotheses of the systems, it is observed that there exists a phrase reordering problem among all of the experiments. In the hypotheses, the words are translated with a good accuracy; however, the order of phrases is mostly incorrect. The major reason behind it can be explained as the difference in sentence structures of Turkish and English. The English sentence structure has the order of *Subject* - *Verb* - *Object*. However, in Turkish, this order changes to *Subject* - *Object* - *Verb*. This means the verb at nearly the beginning of the English sentence must be aligned to the word at the end of Turkish sentence. This alignment type is called discontinuous word alignment as explained in Section 2.7.3. For the long sentence pairs, such as sentences containing around 40 words, this discontinuous word alignment results in large penalty value. Because of this large penalty, such hypotheses are discarded to shrink the search space of the word alignment matrices. Thus, the distant word pairs are rarely aligned to each other correctly. This problem can be solved by the increasing or disabling the distortion limit defined in the configuration file. However, disabling or increasing this limit can result in a large amount of time and space consumption of the translation process. This is a kind of trade-off between the translation quality and the efficiency in general.

Using 160K sentences, 4 different SMT systems are built: word-based-SMT (baseline system), rule-based morphological analysis-based SMT, statistical word decomposition-based SMT, and fusion-based word decomposition-based SMT. The time requirements of all four systems are evaluated by the example input, analyzed as case studies.

Input: Input: Kalıntılar 60 metre derinlikteki bir çukurda bulundu.

Expected output: The remains were discovered in a pit 60 meters deep.

The outputs of different SMT systems described above are listed below.

Word-based experiment output: The lost of the pit were found at a depth of 60 meters.

Rule-based experiment output: Ruins around 60 meters at the deep, explains they found in soon.

Statistical experiment output: At a depth of 60 meters, they found.

Fusion-based experiment output: The remains at a depth of 60 meters that they found in a pit.

The time complexity of the phrase based statistical machine translation system is linear, since a distortion limit of 6 is used during all of the experiments. The distortion limit value restricts the number of shifts of a phrase in the sentence generation phase. If the distortion limit was not used, the time complexity of the system would be exponential [19] since all of the permutations of the target phrases need to be generated. Unlimited distortion increases both the translation quality and time requirement of the SMT system. The time requirements of four experiment types and two enterprise applications for the given input above are shown in Table 6.7. The same hardware configuration is used for the time measurements to observe the relative differences.

Table 6.7: Time requirements of SMT systems for sample input

SMT system	Translation time (sec)
Word-based (baseline)	3.177
Rule-based morphological analysis	3.901
Statistical word decomposition	4.470
Fusion-based word decomposition	4.854
Google Translate	Less than 1.0
Bing Translator	Less than 1.0

It is important that the fusion-based word decomposition-based SMT system requires a longer time to perform a translation. This requirement is caused by the prior processes, such as classification of the input words as frequent and rare ones. All of the training operations, including frequency calculation for the fusion-based word decomposition or model generation for the perception-based word sense disambiguation are excluded during the time requirement tests. These tests only include the decoding processes and translation operations.

The outputs of the rule-based morphological analysis can also be transformed into such a form that the similarity between the language pair is increased. It is shown that such experiments on the post processing of the rule-based morphological analysis output increase the translation accuracy as well [2], [47].

Furthermore, the unsupervised machine learning approach for the statistical word decomposition mostly incorrectly fragments the words linguistically. Therefore, this approach cannot be used as a *morphological analyzer* itself, but as a pre-processor which generates intermediate output which is consumed by another component. Since the same errant decomposition is always repeated for the same word, this technique can be considered as a consistent and deterministic approach.

Another important point is to perform some local modifications on the rule-based morphological analysis output to improve the overall statistical machine translation quality. The techniques, such as morpheme removal or local word reordering can definitely improve the translation quality [15]. Moreover, these techniques can also be applied to the statistical word decomposition-based experiments to improve the overall accuracy. However, they only shift the minimum amount of the required parallel text for the baseline experiments to outperform the others. Ultimately, the word-based translation system will perform better than the sub-word unit-based approaches, because the main motivation behind the sub-word unit-based approaches is to solve the problems caused by the high out-of-vocabulary rate. When this rate is decreased to the regular level by compiling additional training data, it is needless to say that the problem does not emerge at all.

Another observation for error analysis in this study is that as expected, the translation quality suffers from the out-of-vocabulary rate in the systems with small amount of training data. Unseen words are left as they are during the translation, and they do not contribute the accuracy at all.

6.4 Discussion

This thesis aims to get benefit from both the statistical and rule-based word decomposition approaches at a time to bring about a more accurate word decomposition module for the phrase-based statistical machine translation systems. Among the literature, there are various studies which prove the improvement when the subword units are used instead of the word-based approaches for the statistical machine translation tasks. This founding is approved by some experiments in this thesis as well. When the circumstances of the experiment changes, the hypotheses may possibly become antiquated.

To compare the rule-based morphological analysis and the machine learning-based word decomposition in terms of the level of fragmentation (i.e. average number of tokens per word), it is observed that the machine learning approaches split the words less aggressively. After the application of the rule-based morphological analysis, the number of unique words in Turkish becomes smaller than those in English. On the other hand, after the application of machine learning-based word decomposition, the number of unique words is still larger than those in English. To get closer to the number of the unique words in English, a hybrid approach between the rule-based morphological analysis and the machine learning-based word decomposition may result in sentences which are more similar to their English translations.

CHAPTER 7

CONCLUSIONS

In this study, the frequency-driven late fusion-based word decomposition approach is introduced to improve the translation quality of the phrase-based statistical machine translation system from Turkish to English. This late fusion-based approach is compared with the standalone statistical and rule-based word decomposition approaches when the corpus size changes. This study differs from others by introducing the novel frequency-driven late fusion-based word decomposition method to boost the BLEU score. While the benchmark study in the literature reports a 25.22 BLEU score with the same data set, the proposed late fusion-based system boosts the accuracy up to a 26.22 BLEU score. Thus, the need for the rule-based morphological analysis can be overcome by the fusion-based approaches, given a sufficient amount of pairs of sentences. However, the rule-based approaches may still be necessary when the quantity of parallel data is not sufficient, or application specific requirements emerge.

Another observation made during this study is that the unsupervised machine learning techniques perform similar to the rule-based morphological analysis based on linguistic motivations when the training corpus size is sufficient. For the unsupervised machine learning techniques, the only necessary resource is a raw corpus in the language. On the contrary, the morphological analysis approach requires the hand written linguistic rules for the decomposition and annotated corpora for the disambiguation operation. However, it should be remarked that the machine learning approaches cannot be used only for the morphological analysis of the words themselves. Instead, they are used as a pre-processor which generates intermediate output which is consumed by other natural language processing applications, such as MT systems.

To summarize, in this study, a novel frequency-driven late fusion-based word decomposition technique is introduced to build more accurate phrase-based statistical machine translation systems. This proposed method is also compared with the well known rule-based morphological analysis and character-based n-gram modeling word decomposition approaches.

In addition, the corpus size experiments are conducted in the scale from 10,000 parallel sentences to 160,000. The rule-based, statistical, and the fusion-based approaches are tested and compared among each other. When the size of the training corpus is relatively small, the rule-based approaches perform much better, as it is stated repeatedly in the literature. However, this study clearly shows that the fusion-based approach outperforms when the training corpus size is sufficient and the training instances are dense. 160,000 parallel sentences are used for the dense training data set experiments, and it is the only publicly available corpus for the Turkish language. Moreover, it was possible to compare the results with the benchmark scores, and the baseline setup performs very similar to those in the previous studies. However, the fusion-based approach results in around a 10% better BLEU score than the baseline setup. The larger corpora may result in more interesting or adventitious results. Because of the inadequate amount of public parallel corpora, these further experiments are left as the future work.

Moreover, the machine learning techniques can also be improved by using continuously introduced techniques in the literature for the unsupervised word decomposition task. The better linguistic decomposition of the stems and morphemes may achieve higher BLEU scores in the phrase-based statistical machine translation systems.

REFERENCES

[1] A. Axelrod, R. B. Mayne, C. Callison-burch, M. Osborne, and D. Talbot. Edinburgh system description for the 2005 iwslt speech translation evaluation. In *In Proc. International Workshop on Spoken Language Translation (IWSLT*, 2005.

[2] A. Bisazza and M. Federico. Morphological pre-processing for Turkish to English statistical machine translation. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, pages 129–135, December 2009.

[3] E. Brill and R. C. Moore. An improved error model for noisy channel spelling correction. In *Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, ACL '00, pages 286–293. Association for Computational Linguistics, 2000.

[4] P. F. Brown, V. J. D. Pietra, S. A. D. Pietra, and R. L. Mercer. The mathematics of statistical machine translation: parameter estimation. *Comput. Linguist.*, 19(2):263–311, June 1993.

[5] M. Butt, H. Dyvik, T. H. King, H. Masuichi, and C. Rohrer. The parallel grammar project. In *Proceedings of the 2002 workshop on Grammar engineering and evaluation - Volume 15*, COLING-GEE '02, pages 1–7. Association for Computational Linguistics, 2002.

[6] W. B. Cavnar and J. M. Trenkle. N-gram-based text categorization. In *In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval*, pages 161–175, 1994.

[7] H. Ceylan and Y. Kim. Language identification of search engine queries. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 1066–1074. Association for Computational Linguistics, 2009.

[8] S. F. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, ACL '96, pages 310–318. Association for Computational Linguistics, 1996.

[9] D. Chiang. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 263–270. Association for Computational Linguistics, 2005.

[10] R. A. Cole, I. Chief, J. Mariani, H. Uszkoreit, A. Zaenen, G. Varile, A. Z. (eds.), A. Zampolli, R. Cole, V. Zue, and V. Zue. Survey of the state of the art in human language technology, 1995.

[11] M. Collins, P. Koehn, and I. Kučerová. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 531–540. Association for Computational Linguistics, 2005.

[12] M. Creutz and K. Lagus. Morfessor in the morpho challenge. In *Proceedings of the PASCAL Challenge Workshop on Unsupervised Segmentation of Words into Morphemes*, 2006.

[13] M. Creutz and K. Lagus. Unsupervised models for morpheme segmentation and morphology learning. *ACM Trans. Speech Lang. Process.*, 4(1):3:1–3:34, Feb. 2007.

[14] M. Denkowski and A. Lavie. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. In *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation*, 2011.

[15] I. D. EI-Kahlout and K. Oflazer. Exploiting morphology and local word reordering in english to turkish phrase-based statistical machine translation. *Trans. Audio, Speech and Lang. Proc.*, 18(6):1313–1322, Aug. 2010.

[16] I. D. El-Kahlout and K. Oflazer. Initial explorations in english to turkish statistical machine translation. In *Proceedings of the Workshop on Statistical Machine Translation*, WMT '06, pages 7–14. Association for Computational Linguistics, 2006.

[17] M. Federico, N. Bertoldi, and M. Cettolo. Irstlm: an open source toolkit for handling large scale language models. In *INTERSPEECH*, pages 1618–1621. ISCA, 2008.

[18] M. Galley and C. D. Manning. A simple and effective hierarchical phrase reordering model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 848–856. Association for Computational Linguistics, 2008.

[19] M. Galley and C. D. Manning. A simple and effective hierarchical phrase reordering model. In *In Proceedings of EMNLP 2008*, 2008.

[20] J. Gao and M. Zhang. Improving language model size reduction using better pruning criteria. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 176–182. Association for Computational Linguistics, 2002.

[21] G. Giannakopoulos, V. Karkaletsis, G. Vouros, and P. Stamatopoulos. Summarization system evaluation revisited: N-gram graphs. *ACM Trans. Speech Lang. Process.*, 5(3):5:1–5:39, Oct. 2008.

[22] S. Goldwater and D. McClosky. Improving statistical mt through morphological analysis. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 676–683. Association for Computational Linguistics, 2005.

[23] D. Groves and A. Way. Hybrid data-driven models of machine translation. *Machine Translation*, 19:301–323, 2005. 10.1007/s10590-006-9015-5.

[24] A. Gutkin. Log-linear interpolation of language models, 2000.

[25] H. A. Güvenir and I. Cicekli. Learning translation templates from examples. *Information Systems*, 23(6):353 – 363, 1998. <ce:title>6th annual workshop on information technologies and systems</ce:title>.

[26] K. Hacioglu and W. Ward. On combining language models: oracle approach. In *Proceedings of the first international conference on Human language technology research*, HLT '01, pages 1–4. Association for Computational Linguistics, 2001.

[27] D. Z. Hakkani, G. Tür, K. Oflazer, T. Mitamura, and E. Nyberg. An english to turkish interlingual mt system. In *Proceedings of the Third Conference of the Association for Machine Translation in the Americas on Machine Translation and the Information Soup*, AMTA '98, pages 83–94, London, UK, 1998. Springer-Verlag.

[28] H. Hoang, A. Birch, C. Callison-burch, R. Zens, R. Aachen, A. Constantin, M. Federico, N. Bertoldi, C. Dyer, B. Cowan, W. Shen, C. Moran, and O. Bojar. Moses: Open source toolkit for statistical machine translation. pages 177–180, 2007.

[29] J. Hutchins. First steps in mechanical translation. In *Association for Machine Translation in the Americas*, pages 14–23, San Diego, California, 1997. ISCA.

[30] W. J. Hutchins. The georgetown-ibm experiment demonstrated in january 1954. In R. E. Frederking and K. Taylor, editors, *AMTA*, volume 3265 of *Lecture Notes in Computer Science*, pages 102–114. Springer, 2004.

[31] W. J. Hutchins, L. Dostert, and P. Garvin. The georgetown-i.b.m. experiment. In *In*, pages 124–135. John Wiley & Sons, 1955.

[32] C. Jordan, J. Healy, and V. Keselj. Swordfish: an unsupervised ngram based approach to morphological analysis. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 657–658, New York, NY, USA, 2006. ACM.

[33] B. june (paul Hsu. Generalized linear interpolation of language models. In *IEEE Workshop on ASRU*, pages 136–140, 2007.

[34] K. Katzner. *The Languages of the World.* Routledge, 3rd edition edition, 2002.

[35] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. Moses: open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180. Association for Computational Linguistics, 2007.

[36] D. Lewis. Naive (bayes) at forty: The independence assumption in information retrieval. In C. Nédellec and C. Rouveirol, editors, *Machine Learning: ECML-98*, volume 1398 of *Lecture Notes in Computer Science*, pages 4–15. Springer Berlin / Heidelberg, 1998. 10.1007/BFb0026666.

[37] I. A. McCowan, D. Moore, J. Dines, D. Gatica-Perez, M. Flynn, P. Wellner, and H. Bourlard. On the use of information retrieval measures for speech recognition evaluation. Idiap-RR Idiap-RR-73-2004, IDIAP, Martigny, Switzerland, 0 2004.

[38] P. McNamee and J. Mayfield. Character n-gram tokenization for european language text retrieval. *Information Retrieval*, 7:73–97, 2004. 10.1023/B:INRT.0000009441.78971.be.

[39] P. McNamee, J. Mayfield, and C. Nicholas. Translation corpus source and size in bilingual retrieval. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, NAACL-Short '09, pages 25–28. Association for Computational Linguistics, 2009.

[40] C. Mermer and A. A. Akin. Unsupervised search for the optimal segmentation for statistical machine translation. In *Proceedings of the ACL 2010 Student Research Workshop*, ACLstudent '10, pages 31–36. Association for Computational Linguistics, 2010.

[41] C. Mermer, H. Kaya, and M. U. Doğan. The TÜBİTAK-UEKAE Statistical Machine Translation System for IWSLT 2010. In M. Federico, I. Lane, M. Paul, and F. Yvon, editors, *Proceedings of the seventh International Workshop on Spoken Language Translation (IWSLT)*, pages 183–188, 2010.

[42] M. Nagao. A framework of a mechanical translation between japanese and english by analogy principle. In *Proc. of the international NATO symposium on Artificial and human intelligence*, pages 173–180, New York, NY, USA, 1984. Elsevier North-Holland, Inc.

[43] F. J. Och and H. Ney. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 295–302. Association for Computational Linguistics, 2002.

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

[45] K. Oflazer. Two-level description of turkish morphology. In *Proceedings of the sixth conference on European chapter of the Association for Computational Linguistics*, EACL '93, pages 472–472. Association for Computational Linguistics, 1993.

[46] K. Oflazer. Statistical machine translation into a morphologically complex language. In *Proceedings of the 9th international conference on Computational linguistics and intelligent text processing*, CICLing'08, pages 376–387, Berlin, Heidelberg, 2008. Springer-Verlag.

[47] K. Oflazer and I. D. El-Kahlout. Exploring different representational units in english to turkish statistical machine translation. In *Proceedings of the Second Workshop on Sta-*

tistical Machine Translation, StatMT '07, pages 25–32. Association for Computational Linguistics, 2007.

[48] M. Orhun. Computational detection of uyghur multiword expressions. In *Communication Software and Networks (ICCSN), 2011 IEEE 3rd International Conference on*, pages 501–505. IEEE, 2011.

[49] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318. Association for Computational Linguistics, 2002.

[50] J. M. Patton and F. Can. Determining translation invariant characteristics of james joyce's dubliners. *Quantitative Methods in Corpus-based Translation Studies: A Practical Guide to Descriptive Translation Research*, 51:209, 2012.

[51] S. RENALS and T. HAIN. 12 speech recognition. *The Handbook of Computational Linguistics and Natural Language Processing*, 57:299, 2010.

[52] H. Sak, T. Güngör, and M. Saraçlar. Morphological disambiguation of Turkish text with perceptron algorithm. In *CICLing 2007*, volume LNCS 4394, pages 107–118, 2007.

[53] P. M. Shishtla, P. Pingali, and V. Varma. A character n-gram based approach for improved recall in indian language ner. In *Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages*, pages 67–74, Hyderabad, India, January 2008. Asian Federation of Natural Language Processing.

[54] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. A study of translation edit rate with targeted human annotation. In *In Proceedings of Association for Machine Translation in the Americas*, pages 223–231, 2006.

[55] A. Solak and K. Oflazer. Parsing agglutinative word structures and its application to spelling checking for turkish. In *Proceedings of the 14th conference on Computational linguistics - Volume 1*, COLING '92, pages 39–45. Association for Computational Linguistics, 1992.

[56] A. C. Tantuğ, E. Adali, and K. Oflazer. Machine translation between turkic languages. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 189–192. Association for Computational Linguistics, 2007.

[57] C. K. Turhan. An english to turkish machine translation system using structural mapping. In *Proceedings of the fifth conference on Applied natural language processing*, ANLC '97, pages 320–323. Association for Computational Linguistics, 1997.

[58] F. M. Tyers and M. S. Alperen. South-east european times: A parallel corpus of the balkan languages. In *Proceedings of the Workshop on Exploitation of multilingual resources and tools for Central and (South) Eastern European Languages*, LREC '10, 2010.

[59] S. Virpioja, J. Väyrynen, A. Mansikkaniemi, and M. Kurimo. Applying morphological decomposition to statistical machine translation. In *Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR*, WMT '10, pages 195–200. Association for Computational Linguistics, 2010.

[60] H. Watanabe and K. Takeda. A pattern-based machine translation system extended by example-based processing. In *Proceedings of the 17th international conference on Computational linguistics - Volume 2*, COLING '98, pages 1369–1373. Association for Computational Linguistics, 1998.

[61] R. Yeniterzi and K. Oflazer. 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*, ACL '10, pages 454–464. Association for Computational Linguistics, 2010.

[62] R. Yeniterzi and K. Oflazer. 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*, ACL '10, pages 454–464. Association for Computational Linguistics, 2010.

[63] D. Yuret and F. Türe. Learning morphological disambiguation rules for turkish. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, pages 328–334. Association for Computational Linguistics, 2006.

[64] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Trans. Inf. Syst.*, 22(2):179–214, Apr. 2004.

APPENDIX A

SAMPLE PAIRS OF SENTENCES

Table A.1: Sample pairs of sentences used for the training corpus

Turkish	**English**
Örneğin Priştine Hastanesi'nde her hafta aşağı yukarı bir aşırı doz vak'ası ile karşılaşılıyor.	The Pristina Hospital, for example, records approximately one case of drug overdose per week.
Ülke bu sektörde köklü bir geleneğe sahip.	The country has a long tradition in the industry.
Ancak Yunanistan, yabancı gemilere kendi gemicilik kütüğünde yer alma izni veren tek AB üye ülkesi olacak.	Greece, however, will be the only EU member state that allows foreign ships in its shipping register.
Bunların yüzde 10 ila 15'ini ise 18 yaş altı kız çocukları oluşturuyor.	About 10 to 15 per cent of them are girls under the age of 18.
Gazetenin haberinde, Bulgaristan'daki fabrikanın Türkiye'ye yakınlığı nedeniyle teröristler için avantajlı olduğu belirtiliyor.	The Bulgarian one would have been advantageous to the terrorists because of its proximity to Turkey, the paper suggests.
Dünya Savaşı'nda kazanılan zaferin ardından 1945 yılında monarşiyi devirdiler.	But the Communists abolished the monarchy in 1945, after victory in World War II.
Sırp Ortodoks Kilisesi başkanı, Veliaht Aleksandar Karacorceviç'e de bu meselede daha etkin rol alması çağrısında bulundu.	The head of the Serbian Orthodox Church called on Crown Prince Aleksandar Karadjordjevic to take a more active part in the matter.
Engel, hasarlı evler hakkındaki ilk tahmini rakamın 6.643 olduğunu söyledi.	He said the first estimate of damaged homes was 6,643.
Paranın 30 milyon euroluk büyük kısmı çoktan harcanmış durumda.	Most of the money, 30m euros, has already been spent.
AB, Türkiye'nin bu alanda kaydettiği ilerlemeleri 2004 yılı sonlarında gözden geçireceğini söyledi.	The EU has said it will review Turkey's progress in that area in late 2004.
Farklı ideolojik fraksiyonlarla bunların anlaşmazlıkları devrimci harekette parçalanmalara yol açtı.	The different ideological factions and their disagreements led to the fragmentation of the revolutionary movement.