UTILIZING OUT-OF-DOMAIN DATA THROUGH LANGUAGE MODELING BASED VOCABULARY SATURATION FOR ENGLISH-TURKISH MACHINE TRANSLATION

by

Burak Aydın

B.S., Computer Engineering, Boğaziçi University, 2011

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering
Boğaziçi University

2014

ACKNOWLEDGEMENTS

I would like to thank my thesis supervisor Assist. Prof. Arzucan Özgür. Her encouragement and continuous guidance made the research easier. I have been really lucky to know her and work with her throughout the preparation of this study.

My deepest gratitude goes to my parents, Haci and Zarife, who supported me for all the decisions I made through my life and gave me strength to finish my research on this study.

I also would like to express my deep and sincere gratitude to İlknur Durgar El-Kahlout and Coşkun Mermer, who had shared their valuable opinions with me on this research.

ABSTRACT

UTILIZING OUT-OF-DOMAIN DATA THROUGH LANGUAGE MODELING BASED VOCABULARY SATURATION FOR ENGLISH-TURKISH MACHINE TRANSLATION

The training data size is of utmost importance for statistical machine translation (SMT), since it affects the training time, model size, decoding speed, as well as the system's overall success. One of the challenges for developing SMT systems for languages with less resources is the limited sizes of the available training data. In this thesis, we propose an approach for expanding the training data by including parallel texts from an out-of-domain corpus. Selecting the best out-of-domain sentences for inclusion in the training set is important for the overall performance of the system. Our method is based on first ranking the out-of-domain sentences using a language modeling approach, and then, including the sentences to the training set by using the vocabulary saturation filter technique. We evaluated our approach for the English-Turkish language pair and obtained promising results. Performance improvements of up to +0.8 BLEU points for the English-Turkish translation is achieved. We compared our results with the translation model combination approaches and the best English-Turkish translation systems as well, then reported the improvements. Moreover, we implemented our system with dependency based language modeling in addition to n-gram based language modeling and reported comparable results.

ÖZET

DİL MODELLEME TEMELLİ KELİME DOYURMA YÖNTEMİYLE ALAN DIŞI DERLEMİN İNGİLİZCE-TÜRKÇE MAKİNE ÇEVİRİSİNDE KULLANILMASI

Eğitim verisi büyüklüğü istatistiksel makine çevirisi (İMÇ) için büyük öneme sahiptir çünkü veri büyüklüğü; eğitim süresi, model büyüklüğü, çözümleme hızı ve sistemin başarım skoru gibi birçok şeyi etkiler. Az kaynaklı diller için İMÇ sistemleri hazırlanırken karşılaşılan en büyük zorluklardan birisi de kullanılabilir eğitim verisi miktarının sınırlı olmasıdır. Bu tezde, alan dışı bir paralel derlem kullanılarak eğitim verisinin genişletildiği bir yaklaşım önerilmiştir. Alan dışı derlemden en iyi cümleleri seçip eğitim verisine eklemek sistemin genel performansı için önemlidir. Önerdiğimiz yöntem ile önce alan dışı derlemdeki cümleler dil modeli kullanılarak sıralanır, daha sonra kelime doyurma süzgeci tekniğiyle içlerinden bazıları seçilerek eğitim verisine eklenir. Önerilen yöntem İngilizce-Türkçe dil çifti için denenmiş ve başarılı sonuçlar elde edilmiştir. İngilizce-Türkçe makine çevirisinde 0.8 BLEU puanına varan skor artışı sağlanmıştır. Sonuçlar öbek tablosu kombinasyonu yöntemleri ve en iyi İngilizce-Türkçe makine çevirisi sistemleri ile de karşılaştırılıp elde edilen gelişmeler raporlanmıştır. Ayrıca cümleler sıralarken n-gram tabanlı dil modellerinin yanı sıra bağımlılık tabanlı dil modellerine göre sıralama da denenmiş ve sonuçlar paylaşılmıştır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS/ABBREVIATIONS

BLEU	Bilingual Evaluation Understudy
DARPA	Defense Advanced Research Projects Agency
DLM	Dependency Language Model
EM	Expectation Maximization
GALE	Global Autonomous Language Exploitation
IWSLT	International Workshop on Spoken Language Translation
LM	Language Modeling
MERT	Minimum Error-Rate Training
NLP	Natural Language Processing
RNN	Recursive Neural Network
SMT	Statistical Machine Translation
SRILM	Stanford Research Institute Language Modeling
TIDES	Translingual Information Detection, Extraction and Summarization
VSF	Vocabulary Saturation Filtering

1. INTRODUCTION

Most of the statistical methods that attempt to solve natural language processing problems achieve better results with increasing training data sizes. In statistical machine translation, the amount of data directly affects system's overall success as well. Increasing the size of the training data by effectively utilizing the data available in other domains (e.g. web, news, medical) using domain adaptation and data selection techniques is a promising research direction for improving the performance of an SMT system. This is especially important for low-resource languages and domains, for which there are only limited amounts of training data available. The English-Turkish language pair is an example low-resource language pair for machine translation. Most of the publicly available corpora for this language pair contain only thousands of sentences, whereas the training sets of language pairs with more resources (e.g. English-French) usually contain millions of sentences. The number of parallel sentences in the training set for English-Turkish does not reach to millions, even when all available corpora from different domains are combined.

In this thesis, we introduce an approach that effectively combines different data selection methods for expanding in-domain training data with the available out-of-domain data in statistical machine translation. The method first scores the sentences in the out-of-domain corpus based on their similarities to the in-domain corpus using a language modeling approach. The ranking process was handled by both n-gram based and dependency based language modeling. Then, it adapts the vocabulary saturation filter technique, which has recently been proposed in [3] for reducing the training data and model sizes, to the domain adaptation problem. The proposed approach is applied to English-Turkish machine translation and improvements in terms of BLEU scores for this language pair are achieved.

1.1. Statistical Machine Translation

The history of machine translation goes back over 70 years [1]. After the usage of computers started to spread, the researchers used them to break message codes in wars. This led to the idea that a sentence in a foreign language is almost like an encrypted version of the very same sentence in the target language. After this change of view, researchers showed great interest on computer systems that translate texts from one language to another. Initially, this translation need was to be solved by human translators and many people thought that it was a futile effort to make computers translate documents instead of humans. However, with the increase in interaction and communication means, the number of documents and languages became more than human translators can handle. Hence, it led to the requirement for automatic translation systems [1].

At first, rule-based systems were proposed. A system named Systran[1] was founded to translate between Russian and English. Some other commercial systems like Logos[2] and METAL[3] were also built for translating between languages. Rule-based systems were thought to be enough for automatic translation. However, it turned out the other way, since natural languages can have an infinite number of rules and it would not be feasible to store all rules, which will cause the systems to be always limited with the rules available. Thus, statistical machine translation methods arose and became very popular [4]. The Defense Advanced Research Projects Agency[4] (DARPA), which is a very famous funding agency in the United States, showed interest in statistical machine translation methods through funding huge programs like TIDES and GALE. The DARPA agency has the initial attempts for most state-of-the-art methods in statistical machine translation as well as other statistical natural language problems.

Today, statistical machine translation systems are heavily developed by top software companies like IBM, Google and Microsoft. Its applications are to be used in

[1] http://www.systransoft.com/
[2] http://www.logos.it/
[3] http://www.utexas.edu/cola/centers/lrc/mt/METAL.html
[4] http://www.darpa.mil/default.aspx

many areas like military, health services and knowledge sharing. Machine translation systems can also be integrated with other useful systems to provide a satisfactory end-to-end communication system, i.e. automatic speech recognition systems [1]. In the following sections of this chapter, preliminary information about the important steps of statistical machine translation will be discussed.

1.2. The Main Steps in Statistical Machine Translation

Statistical machine translation systems include several different components and every component is a whole different natural language problem. Most of these problems are attempted to be solved with statistical machine learning methods. A compact list of the main steps for building a statistical machine translation system is given below:

- Word Alignment
- Building Translation Tables
- Building Language Models
- Decoding

1.2.1. Word Alignment

Statistical machine translation systems are built from parallel texts. A parallel text consists of two documents, each of the documents contain the very same sentence, one of them in the foreign language and the other one in the target language. This parallel text is our data to build machine translation models for the translation system. However, in these parallel texts there is no word alignment between the words of the source and the target sentence, only the sentences that are translations of each other are aligned. Hence, one of the problems is to align the words with each other as shown in Figure 1.1.

Notice that we have an incomplete data problem here. Our texts are only sentence-wise aligned. We do not have the word alignments. If we had word alignments, we would have lexical translations for each word, yet we do not have the lexical

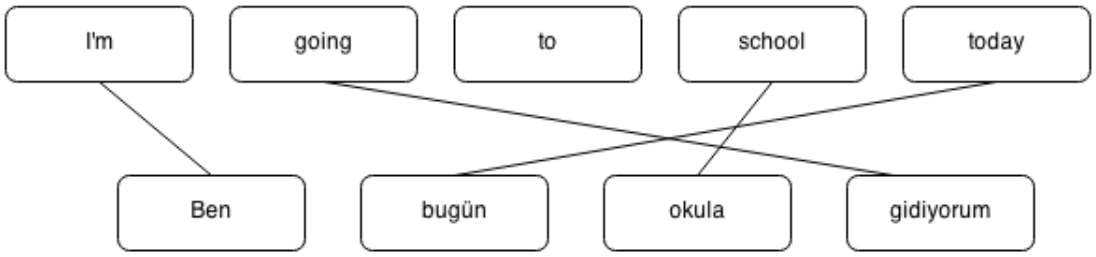

Figure 1.1. A possible word alignment between source and target sentence.

translations either [1]. Since we have neither alignments nor the lexical translations, we should use an algorithm to solve the incomplete data problem. The Expectation-Maximization(EM) algorithm [5] is heavily used for finding word alignments. We are trying to estimate a model here, namely lexical translations, but our data is incomplete due to the lack of word alignments. EM work as follows:

(i) Build an initial model with a distribution, generally uniform.
(ii) Apply lexical model to our incomplete data.
(iii) Then learn a new model from the data.
(iv) Iterate steps (ii) and (iii) until the algorithm converges.

After convergence, we have an estimate of what word in the source sentence corresponds to which word in the target sentence.

1.2.2. Building Translation Tables

After determining the word alignments, the next step is to extract translation rules from them. These translation rules will be used in the decoding step to translate a foreign sentence into the target langauge. In earlier studies, word-based models were popular. In word-based models, the extracted translation rule table contains entries showing the translation of one foreign word into one target word. Examples of word-based models are shown in Table 1.1. At the start, this seemed to be a sufficient approach. However, its accuracy was limited since the lexical translation of a word may differ from one sentence to other, depending on the context of the word.

Table 1.1. Sample entries for word-based models.

Source Word	Target Word	Probability Score
house	ev	0.8
house	konut	0.2
going	gidiyorum	0.7
going	giriyorum	0.05
going	geçiyorum	0.15
going	geliyorum	0.1
today	bugün	0.95
today	günümüzde	0.05

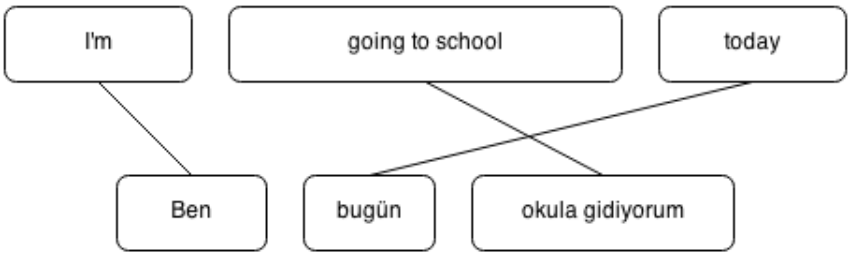

Figure 1.2. A possible phrase alignment between source and target sentence.

Phrase-based models are proposed in order to handle the cases where word-based models fail. Most lexical word translations earn meaning with the word around it and also vary from usage in one sentence to another. From the alignment points we found before, we extract phrases. The extraction is to be done with minimal matching concept. If a word in source is added to the source phrase, its aligned word in the target should be added to the target phrase. The words between start and end of the phrases should not be excluded even if they are not aligned. If the foreign phrase borders unaligned words, then it is extended to these words, and the extended phrase is also added as a translation of the foreign phrase [1]. A sample phrase alignment is shown in Figure 1.2.

Notice that there are multiple phrases available to be extracted and all of them should be available in the phrase translation table with their statistics. The possible phrases extracted from the word alignment in Figure 1.1 are shown in Table 1.2.

Table 1.2. Sample entries for phrase-based models.

Source Phrase	Target Phrase
I'm	ben
going	gidiyorum
going to	gidiyorum
school	okula
school today	bugün okula
to school	okula
to school today	bugün okula
today	bugün
I'm going	Ben bugün okula gidiyorum
I'm going to	Ben bugün okula gidiyorum
I'm going to school	Ben bugün okula gidiyorum
I'm going to school today	Ben bugün okula gidiyorum
going to school	okula gidiyorum
going to school today	bugün okula gidiyorum

Phrase-based models now give better and more accurate translations than word-based models and many systems use this approach while building statistical translation models.

1.2.3. Building Language Models

Language modeling is used in many natural language processing problems. In SMT, language models are used in the decoding step and generally n-gram language modeling is preferred. In the phrase translation table, there are many possible translations for a specific source phrase. So the suitable target phrase is selected after it is scored with respect to a previously built language model. The language model may be built by using the target side of the parallel text or it may be built from an outside and probably a huge target language corpus. In SMT, phrase translation models pro-

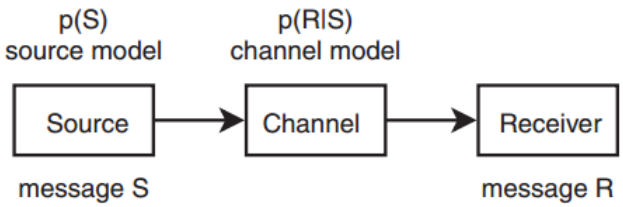

Figure 1.3. Noisy-channel model [1].

vide accuracy, whereas language models provide fluency in the produced translation. Combining the translation model with a language model is called as the noisy-channel model which is depicted in Figure 1.3.

1.2.4. Decoding

Decoding in statistical machine translation refers to finding the translation of a sentence which gives the best score in the search path. There are exponential number of choices with respect to the sentence length while translating hence decoding is a hard problem [1]. Heuristic search methods are generally used since scoring all choices is not feasible due to time and memory constraints. The scores for a specific sentence come from the phrase translation table and the language model used. The decoder tries to find the best scoring sentence. So if smaller and more compact models are available, then the translation speed increases. The methods proposed in this thesis also diminish models' size with increased success rate and this helps for faster decoding.

One of the first attempts to reduce search space is pruning. Pruning is like ignoring some of the translation options at the very first steps of the decoding. At the start of the search path, if a phrase or a phrase combination has score below a predefined threshold, then that phrase and all the possible search paths that this phrase can expand are removed from the path.

Another widely used method is A* search, which is described in many artificial intelligience books e.g. [6]. It again tries to trace over possible search paths with pruning but also uses an additional parameter, future cost to prune out more. Future

cost is the cost that will add up when a possible phrase translation option is added to the search path [1].

1.3. Evaluating Statistical Machine Translation

Evaluating a translation system is a hard problem to handle. In contrast to many other natural language processing problems such as speech recognition, there is not a single correct answer for a specific input to a machine translation system. Even different human translators can translate one sentence into different target sentences conveying the same meaning. So, here the problem to evaluate machine translation systems arises.

Human evaluation by the native speakers of the target language is the very first attempt that comes to mind, to evaluate results of a machine translation system. It is good for judging translation systems according to usability. However, this metric is not efficient since it requires a lot of human power when we take the number of languages and the sizes of the test sets into account.

Due to the feasibility problems of human evaluation, automatic evaluation metrics took the scene in order to evaluate SMT systems. Precision and recall are the metrics that are heavily used in evaluation of most NLP problems and they are also good candidates for machine translation evaluation. In the following formulas, *correct* corresponds to the number of word matches between the output sentence and the reference sentence. The *output-length* and *reference-length* mean the number of words in the output sentence and the reference sentence, respectively.

$$precision = \frac{correct}{output\text{-}length} \tag{1.1}$$

$$recall = \frac{correct}{reference\text{-}length} \tag{1.2}$$

Notice that both of these metrics can be easily tricked. The system may output only certain high scoring words, which will end up with short translations. This will result in high precision but low recall. In another case, the system may produce all possible words so that the likelihood to match with the reference words increases. This will result in long translations, high recall, but low precision. In SMT, both of these metrics are important hence a common combination of them, namely f-measure, is more suitable.

$$f\text{-}measure = \frac{precision * recall}{(precision + recall)/2} \tag{1.3}$$

After reformulation:

$$f\text{-}measure = \frac{correct}{(output\text{-}length + reference\text{-}length)/2} \tag{1.4}$$

After some research though, the Bilingual Evaluation Understudy (BLEU) metric has been proposed and is now heavily used for evaluating machine translation systems [7]. BLEU aims to evaluate matches of n-grams between the output sentence and the reference sentence. BLEU-n is defined as follows:

$$BLEU\text{-}n = brevity\text{-}penalty \prod_{n} precision_n \tag{1.5}$$

$$brevity\text{-}penalty = min\left(1, \frac{output\text{-}length}{reference\text{-}length}\right) \tag{1.6}$$

The precision-based metrics in SMT do not take word drops in the output into account. However, in BLEU, it is addressed by the brevity-penalty. Hence, we end up with the formula where n is the order of n-gram and $precision_n$ corresponds to the

precision calculated for that n-gram:

$$BLEU\text{-}n = min\left(1, \frac{output\text{-}length}{reference\text{-}length}\right) \prod_n precision_n \tag{1.7}$$

One can notice that the BLEU metric may fail in some cases since the same meaning can be conveyed by different words and since BLEU looks for exact matches, this may result in poor evaluation. Using multiple references can overcome this problem if it is very likely to occur. What is more, the BLEU metric has been shown to be correlated with human judgement.

2. RELATED WORK

In statistical machine translation, several approaches have been proposed for data selection, domain adaptation, and data preprocessing and cleaning. In [8], it is proposed to select a subset of a monolingual corpus and human-translate it to use for a low-resource language pair. They also ranked the corpus to improve token coverage by taking the sentence length into account. In addition to this method, they also ranked with respect to the similarities based on TF-IDF (Term Frequency-Inverse Document Frequency) yet it did not improve much over the first approach.

The aim for selecting a suitable subset is not only for decreasing the model size, but also for improving the translation quality as in the work in [9]. They proposed a pre-processing method on the training data to detect favourable items such as paraphrases and multiword expressions, before using them in word alignment. Their cleaning technique is specified for the word alignment step of a statistical machine translation system. By improving the aligment score, the SMT system also has been improved and the BLEU scores of their system increased with this pre-processing method.

Data selection and preprocessing methods generally aim to be successful in reducing data and model size significantly with a minimum score loss [3]. Domain adaptation techniques on the other hand, target not only to optimize the data and model size, but also to improve system score. A number of different approaches including language modeling [10] and source-sentence classification [11] have been proposed for domain adaptation. Machine translation systems mostly work well only in one domain and domain adaptation techniques usually improve the score of a system in that domain. In [12], the authors attempted to build a system that works well in multiple-domains simultaneously. Their method tries to use models of different domains in a combined system and automatically detects the domain and its parameters at runtime.

In some circumstances, there may be lack of in-domain bilingual data. In one work [13], the researchers used out-of-domain corpora to train a baseline system and

then used in-domain translation dictionaries and in-domain monolingual corpora to improve the in-domain performance. Their method unifies old and newly produced resources in a combined framework. The contribution of their work was to investigate ways of using in-domain monolingual corpora with the in-domain translation dictionary to improve the scores of the baseline translation system.

In [14], the authors tried to solve the efficient data selection problem in the language model training step, which is an essential feature of statistical machine translation systems. In this work, they compared the cross-entropy according to domain-specific and non-domain-specific language models for each sentence that is used to produce the non-domain-specific model. They calculated the cross-entropy difference to select the data. Using this approach, they produced better language models to use in their machine translation system. The difference of our method is that we use perplexity based ranking for the training of the whole system, not for building efficient language models only.

In another work, the authors attempted to significantly improve the performance of machine translation by exploiting large monolingual in-domain data [15]. They synthesized a bilingual corpus by translating the monolingual adaptation data. Their work is based on adapting an already developed translation system into another domain in which there is no enough parallel data available. The authors built their translation and language models after they translated their monolingual corpus into the target langauge with the baseline machine translation system. Their best improvement is achieved for the case when in-domain monolingual data are available for the target language.

Dependency based language modeling was also investigated in many studies. Shen et al. built a framework to employ a target dependency language model (DLM) for machine translation [16]. It predicts the next child based on the previous children of the current head. DLM was used in many tasks such as sentence realisation [17], speech recognition [18] and sentence completion [19]. In our dependency-based language modeling approach we represent the sentences with trigrams (i.e. dependent, head, and

dependency type) extracted from their dependency parse trees. The out-of-domain sentences are ranked based on the dependency relation language models learned from the in-domain corpus.

For English-Turkish statistical machine translation, several methods are applied in order to improve system scores. In [20], both Turkish and English were represented at the morpheme level. The sub-word representations and building models from these subwords provided improvements over the baseline system. In [21], several different state-of-the-art methods were experimented for English-Turkish machine translation. Morphological representation and system combination were the ones that improved the baseline systems significiantly. The authors also investigated incorporating an out-of-domain corpus to the training set. However, this resulted in decrease in BLEU scores. Using the same training and test sets as well as the same out-of-domain corpus, we report significantly higher BLEU scores than their best scores [21], even though we didn't incorporate all the useful features that they proposed to our system.

Another relevant approach is the phrase table combination method proposed in [22]. They tried to expand the in-domain phrase table with out-of-domain phrase table in an efficient manner. In the fill-up method, they used the phrases from the out-of-domain table only if they are not available in the in-domain table. We compared our results with their method and also with the linear interpolation technique's results. Our systems obtained better BLEU scores in overall for the English-Turkish language pair. Additionally, their system uses all of the available corpora for phrase table building, whereas ours use a proportion of the out-of-domain data in addition to the in-domain data.

Recently, in [3] an algorithm named Vocabulary Saturation Filter (VSF) is proposed. The algorithm tries to significantly reduce the training data size. It starts by counting the n-grams from the beginning of the corpus and when all n-grams of a specific sentence reach to a previously defined threshold frequency t then this sentence is excluded from the resulting subset that is to be used for machine translation system training. They showed that unigrams are sufficient to choose a subset, since higher

orders resulted in selecting the majority of the original corpus.

Our approach is an effective combination of language modeling and vocabulary saturation filtering (VSF) for expanding training data using an out-of-domain corpus. VSF has originally been used for data size reduction [3], but in this study we adapt it to use in expanding in-domain data with an out-of-domain corpus for machine translation. Before applying the VSF technique, we pre-rank the out-of-domain parallel corpus based on the sentence perplexities calculated using an in-domain language model. We apply the approach for the English-Turkish language pair, and report promising results in various settings.

The main contributions of this thesis can be summarized as follows:

- To the best of our knowledge, this is the first study that investigates applying dependency constrained models and the VSF algorithm for Turkish as a target language.
- An approach based on sorting sentences with respect to n-gram language modeling and dependency based language modeling before applying VSF is proposed.
- The language modeling based VSF algorithm is applied to out-of-domain data for selecting sentences to expand the training set and increase in BLEU scores are achieved with respect to strong baselines.

3. DEPENDENCY CONSTRAINT MODELS

In SMT, linguistic information may sometimes be useful. Tree-based approaches show promising results such as hierarchical phrase based modeling [23]. This approach tries to extract phrases containing words that are far away from each other in the sentence. The algorithm tries to find every possible phrase with gaps and unknown words, which are marked with a common non-terminal X. Sample hierarchical rules from our previous alignment points may be as in Table 3.1.

Table 3.1. Sample entries for hierarchical phrase-based models.

Source Phrase	Target Phrase
I'm going to school X	Ben X okula gidiyorum
I'm going to X today	Ben bugün X gidiyorum
going to X	X gidiyorum
X to school	okula X

This non-terminal X is fixed but it can be enhanced as well, i.e. with non-terminals of a syntactic grammar as in Table 3.2. This enhancement results in more constrained models with less rules than the original hierarchical phrase based approach.

Table 3.2. Sample entries for syntax-based phrase-based models.

Source Phrase	Target Phrase
I'm going to school NP	Ben NP okula gidiyorum
I'm going to NP today	Ben bugün NP gidiyorum
going to NP	NP gidiyorum
VP to school	okula VP

Since the constraints get tighter, the success of the decoder decreases due to the fact that some of the phrase options are excluded, however, it works faster due to the very same reason. We can not use syntactic based models for translating into Turkish,

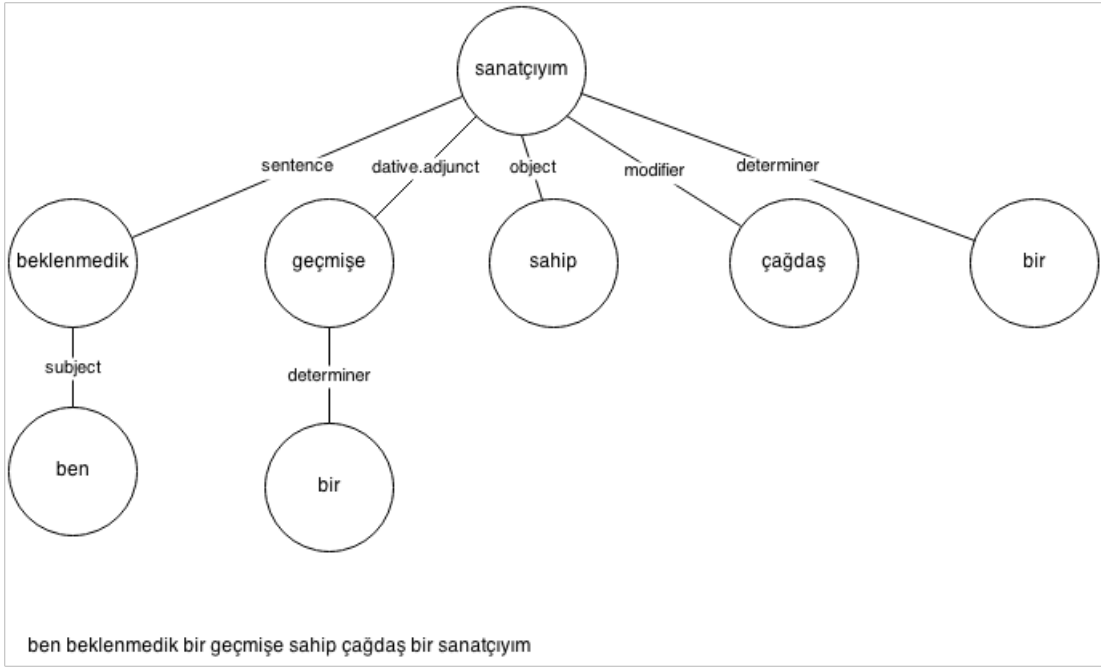

Figure 3.1. A sample dependency tree for the sentence "Ben beklenmedik geçmişe sahip çağdaş bir sanatçıyım." obtained by using the dependency parser in [2] (I am a comtemporary artist with an unexpected background.).

since there does not exist a syntactic constituency parser for Turkish yet. On the other hand, there is a dependency parser available for Turkish [2]. We used dependency labels to define constraints in our model and compared the results with the standard phrase-based baselines. An example dependency tree for a sentence in the WIT corpus is illustrated in Figure 3.1.

Instead of non-terminal X, here we use dependency tree labels as non-terminals and translation rules are extracted according to these constraints. Possible phrase translation rules are shown in Table 3.3.

Table 3.3. Sample entries for Turkish side dependency-based phrase-based models.

Source Phrase	Target Phrase
contemporary artist	MODIFIER sanatçıyım
unexpected background	beklenmedik DATIVE.ADJUNCT
an unexpected background	beklenmedik DETERMINER geçmişe

The results of the dependency-based machine translation system exhibited worse scores than the standard phrase-based baselines. Linguistic information does not necessarily improve systems but sometimes constraints the models and decrease the score. The only gain may be in model size and decoding speed. The results of the dependency constrained system are shown in Table 3.4.

There are several reasons why the dependency-based system did not improve the translation system. These reasons can be stated as follows:

- Syntax-based systems mostly aim at decreasing model size and improving decoding speed with a minimum score loss.
- For Turkish, there is no syntactic parser and the information from dependency parser is probably not as helpful as syntactic trees while extracting translation rules.
- Since the success ratio of Turkish dependency parser is not that high (around 73%, probably lower for our data), it leads to extraction of malformed labels and problematic dependency-based phrases that ends up in poor translation options.

Table 3.4. BLEU scores for dependency-based English-Turkish translation.

System	Test2010	Test2011	Test2012	Test2013
Phrase-based	7.94	7.94	8.02	7.09
Dependency-based	6.33	6.42	6.34	5.35

4. LANGUAGE MODELING BASED VOCABULARY SATURATION FILTER

The effects of more data on improving BLEU scores is clearly observed through experiments: as more data is added, BLEU scores increase [24]. However, the relationship between quantity of data and BLEU is not linear, such that addition of new data does not increase much after some point. There is a saturation point of data and the VSF algorithm attempts to find it [3]. It selects the data within a threshold and uses it for machine translation. The algorithm is very successful in reducing training data size [3]. In this thesis, we use VSF to improve translation results by expanding the training data with out-of-domain data adaptation. Before modifying the algorithm to use for out-of-domain data selection, the algorithm is also applied on the in-domain corpus only for Turkish-English translation and data reduction with a little score loss is achieved as well.

The VSF algorithm starts reading from the beginning of the corpus. It divides the current sentence into n-grams and counts them in a list. If the counts of all the n-grams in the current sentence exceed the previously defined threshold t, then that sentence is excluded from our resulting subset and the algorithm continues from the next sentence. The heuristic behind the algorithm is that every corpus has a threshold point at which the usefullness of data stops and no gain is retrieved from the subset beyond this point. Moreover, it is observed that this point is found using vocabulary saturation, that is to count words of each sentence up to some fixed parameter to select data. The size of the subsets which are selected from the training data, varies with respect to the value of the threshold parameter t, hence different experimental values should be applied in the experiments to see the negative or positive effects.

The pseudo-code for the VSF algorithm is given in Figure 4. The explanation for each variable in the code is as follows:

- t: Frequency threshold for the n-gram counts.

```
Input: ParallelCorpus, t, n
Output: SelectedCorpus
foreach sp in ParallelCorpus do
    S ⇐ EnumNgrams(sp.src, n);
    T ⇐ EnumNgrams(sp.tgt, n);
    selected ⇐ false;
    foreach (s,t) in (S,T) do
        if SrcCnt[s] < t or TgtCnt[t] < t then
            selected ⇐ true;
        end if
    end foreach
    if selected then
        SelectedCorpus.Add(sp);
        foreach (s,t) in (S,T) do
            SrcCnt[s] ⇐ SrcCnt[s] + 1;
            SrcCnt[s] ⇐ SrcCnt[s] + 1;
        end foreach
    end if
end foreach
```

Figure 4.1. Pseudo-code for VSF.

- n: Order of the n-gram.
- *sp*: A sentence pair in the parallel corpus.
- *sp.src*: Source sentence of the pair.
- *sp.tgt*: Target sentence of the pair.
- S: N-grams of the source sentence.
- T: N-grams of the target sentence.
- *SrcCnt*: Count map for the overall source n-grams.
- *TgtCnt*: Count map for the overall target n-grams.

4.1. Language Modeling based VSF on In-domain Data

The VSF algorithm has originally been proposed to reduce training data size with a possible loss in BLEU scores and its power has been shown for French-English translation by applying the algorithm on the sentences in the given order [3]. Here, before proceeding with the utilization of out-of-domain data, we applied it for English-Turkish translation with a language modeling based pre-sorting approach. The data selection with VSF starts from the beginning of the corpus, hence the sentence order makes difference for the algorithm's choice. The procedure for the described method can be formalized in the following steps:

(i) Build a language model using the target side of the parallel corpus.
(ii) Score the target side of the parallel corpus based on the previously build langauge model.
(iii) Rank the parallel corpus with respect to the sentence perplexity scores.
(iv) Apply the VSF algorithm on the ordered corpus to select a subset and use it as training data of the SMT system.

The results of the original approach and language modeling based approach are shown in Table 7.1 in the results section.

4.2. Dependency Language Modeling based VSF on In-domain Data

In the previous section, the pre-ranking process on the training set is applied based on n-gram language modeling. The corpus is ranked with respect to sentence perplexity scores. We proposed other sorting mechanisms than n-grams based modeling and experimented in end-to-end machine translation systems to see the effect. The main idea is to use the dependency relations between the words in a sentence and to identify whether these relations will provide a better ranking for the sentences in the training corpus. Notice the data selection procedure here is almost the same as the method in the previous section, the only difference is the sorting mechanism. We tried several combinations of these relation features and reported the results. These

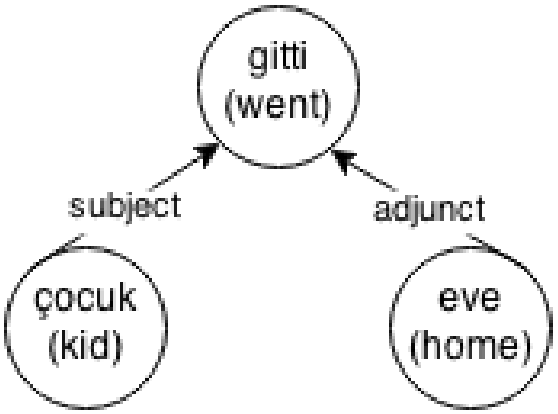

Figure 4.2. Dependency tree for sentence “Çocuk eve gitti.” (The kid went home.).

representations extracted from the tree in Figure 4.2 are shown in Table 4.1. The English version of the very same sentence was also depicted as in Figure 4.3 using Stanford dependency parser [25].

Table 4.1. Dependency-based representation of the sentence in Figure 4.2.

System	Representation	Example
No dependency	-	çocuk eve gitti
System 1	word_label_word	çocuk_subject_gitti eve_adjunct_gitti
System 2	pos_label_pos	noun_subject_verb noun_adjunct_verb
System 3	word_label_pos	çocuk_subject_verb eve_adjunct_verb
System 4	All	Representations from system 1-3 are combined

After we represented our training corpus with the settings exemplified in Table 4.1, we collected the statistics based on these modified corpus’ dependency-based unigrams and sorted the corpus accordingly. Afterwards, VSF is applied through the corpus to select a more efficient subset from the beginning. The selected corpus is used in training statistical machine translation systems and results are compared with the baseline and n-gram based ranked VSF systems on Table 7.2.

We decided to focus on the setting which gives the best BLEU score and applied it for the ranking when we are trying to utilize out-of-domain data, which will be investigated deeply in the next section.

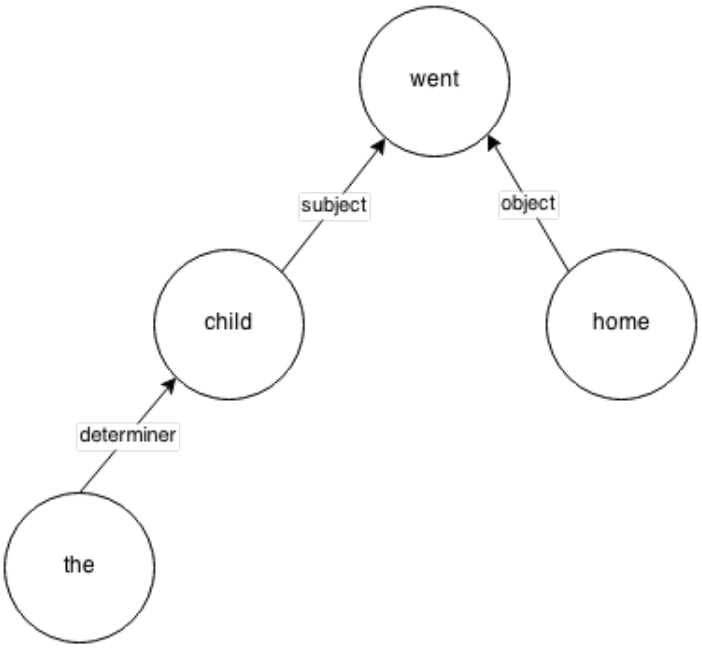

Figure 4.3. Dependency tree for sentence “The kid went home.”.

4.3. Language Modeling based VSF to Utilize Out-of-domain Data

As briefly stated before, the best scoring method proposed in this thesis is a combination of different data selection algorithms, namely Language Modeling (LM) and Vocabulary Saturation Filter (VSF). VSF has originally been proposed to reduce training data and model size with a minimum score loss. However, in this thesis, we adapt the VSF approach to increase the training data size using out-of-domain data with the goal of improving the performance of an SMT system.

The VSF algorithm selects the training subset from the corpus by counting the seen n-grams. The algorithm starts to read the corpus from the beginning, so it is more likely that the sentences at the beginning of the corpus will be chosen by the algorithm. This affects the sentence choice. What we propose is to order the out-of-domain data with respect to a language model built from the in-domain data and select a subset from it through VSF in order to add into the in-domain data for training. The approach, whose workflow is shown in Figure 4.4, may be summarized as in the following steps.

- Build a language model with the target side of the in-domain parallel corpus.

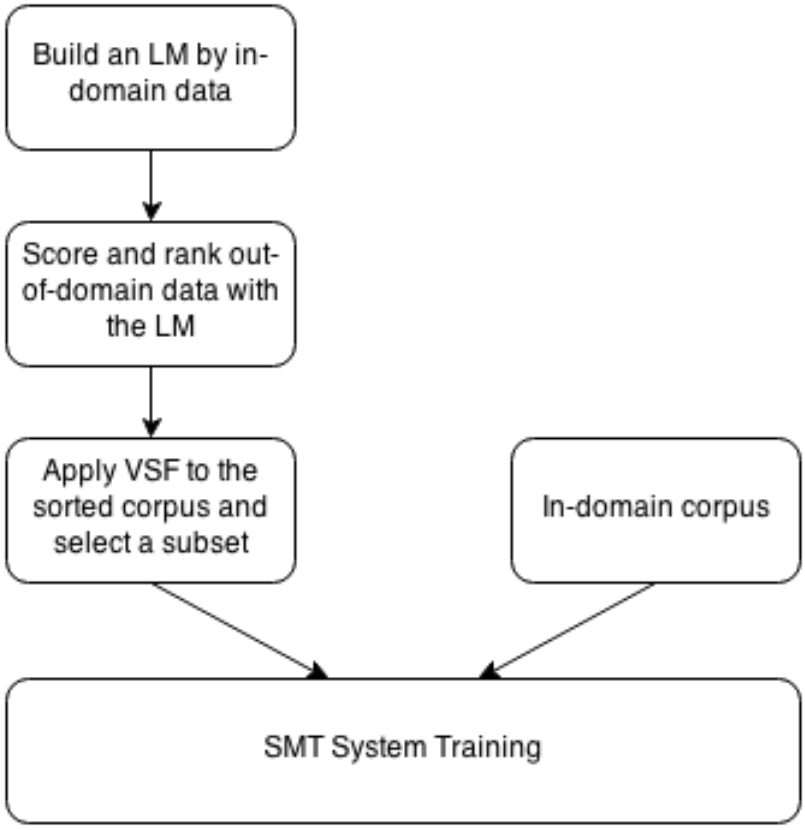

Figure 4.4. Workflow for the proposed approach to utilize out-of-domain data.

- Score the sentences in the target side of the out-of-domain parallel corpus by the language model produced in the previous step.
- Rank the sentences in the out-of-domain corpus based on sentence perplexity scores.
- Apply the VSF algorithm on the sorted corpus.
- Use the selected out-of-domain corpus sentences together with the in-domain corpus sentences for training a machine translation system.

5-gram language models are built from the in-domain data after tokenization using the SRILM toolkit [26]. After scoring the out-of-domain sentences with the language model learned from the in-domain data, the sentences are ranked according to their perplexity scores. Since lower perplexity scores correspond to better fitting to the applied language model, sentences with lower perplexity score appeared at the top of the corpus. Hence, their chances to be selected by the VSF algorithm increased. We applied the VSF technique on the sorted out-of-domain sentences by using the

unigrams. In other words, we counted the seen unigrams while selecting the subset from the ranked corpus, since higher order n-grams lead to the selection of almost the entire corpus.

We also ranked our out-of-domain corpus with respect to dependency-based relations investigated in the previous section. Only the best scoring system is applied for the utilization of out-of-domain data and its results are also reported. The results of all systems that utilize out-of-domain data are shown in Table 7.3.

The proposed approach aims at increasing the score through data adaptation between domains, rather than solely reducing training data size as in the original VSF method. Using VSF with perplexity-based pre-ranking of out-of-domain sentences for expanding the in-domain training data for statistical machine translation is the main contribution of this thesis.

5. EXPERIMENTAL SETUP

We used string-to-string baseline machine translation systems for the English-Turkish language pair, in order to investigate the effects of the proposed method. We implemented the SMT systems with a phrase-based approach [27]. We generated word alignments using MGIZA [28] and used Moses Open Source toolkit [29] for decoding. The parameters of the system are tuned and optimized with the minimum error rate training (MERT) algorithm [30]. We tuned the system with 3 different seeds and reported the best result obtained for each setting. We trained conventional 5-gram language models (LMs) from the available parallel corpora. All language models were trained with the SRILM toolkit using the modified Kneser-Ney smoothing technique [31] and then, binarized using KenLM [32]. We used the sentence perplexity scores produced by SRILM in order to pre-rank the parallel corpora. Moreover, in our implementation of the VSF algorithm, the threshold t values were chosen in the range $[1, 10]$ for English-Turkish. The details of the corpora used for training are provided in the data section. For n-gram choice in VSF, unigrams are used as in the original study, since higher order n-grams select more than half of the original data. Choosing the major proportion of the data diminishes the system score as shown with the experiments, where all out-of-domain data is added to the training phase.

6. DATA

For the experiments, we used WIT [33] data as in-domain and SETIMES [34] data as out-of-domain data. The WIT[5] corpus contains a collection of transcribed and translated talks and the core is the TED talks. On the other hand, SETIMES[6] corpus is in the news domain, collected from a website covering events in the Balkans. The statistics for the English-Turkish parallel corpora are given in Table 6.1 and in Table 6.2.

Table 6.1. WIT training data statistics.

Data Set	Sentences	Unique Words	Total Words
Turkish	131K	158K	1.8M
English	131K	45K	2.5M

Table 6.2. SETIMES training data statistics.

Data Set	Sentences	Unique Words	Total Words
Turkish	165K	143K	3.9M
English	165K	60K	4.6M

As test sets, we used test2010, test2011, test2012 and test2013 sets. The system is tuned with the dev2010 data set. These test sets were used in the IWSLT[7] competitions in the respective years. These test and development sets also contain collections of talks retrieved from TED talks. The sentence counts for the test and development sets are given in Table 6.3.

As the statistics on Table 6.1, 6.2 and 6.3 show, the Turkish side of the sets contain more words than the English side due to the fact that Turkish is a more aggluginative and richer language in terms of morphology. This makes English to

[5]https://wit3.fbk.eu/
[6]http://opus.lingfil.uu.se/SETIMES.php
[7]http://www.iwslt2013.org/

Table 6.3. English-Turkish test data statistics.

Data Set	Sentences	Unique Words	Total Words
dev2010 (En)	887	3341	20256
dev2010 (Tr)	887	3701	15302
test2010 (En)	1568	3897	32367
test2010 (Tr)	1568	4454	24560
test2011 (En)	1433	3672	27224
test2011 (Tr)	1433	4124	20787
test2012 (En)	1698	4056	30975
test2012 (Tr)	1698	4654	23748
test2013 (En)	1022	3572	22833
test2013 (Tr)	1022	3981	17660

Turkish translation harder, since the problem is to generate morphologically richer sentences from a more compact and limited source side, English. On the other hand, Turkish to English translation systems achieve better scores since the target side is morphologically less complex than the source and it is easy to produce better sentences from a richer context. All in all, this difference in the morphological structures of the two languages still makes it a challenging task to translate between them whatever the target is.

7. RESULTS

The models are trained with the corpora described in the previous section. The BLEU-4 score [7] is used as an evaluation metric on the test sets. First, we implemented the baseline systems trained with the original in-domain data. Afterwards, we used the proposed method on in-domain data only with n-gram based ranking and dependency based ranking. Then we experimented the utilization of out-of-domain data by the very same approach. We added all of the out-of-domain data to the baseline system and retrieved the results. Using the whole out-of-domain data did not increase the BLEU score as much as expected. Afterwards, we ranked the out-of-domain data with respect to the sentence-perplexity scores based on a language model built with in-domain data. Then, we selected subsets of the sorted corpus with various VSF frequency threshold settings and used them in the machine translation systems.

7.1. Results with only in-domain data

According to the results in Table 7.1, the data reduction also worked for the English-Turkish system. Using the 58% of the total data, we recovered 93%, 94%, 94% and 95% of the BLEU scores for test sets test2010, test2011, test2012 and test2013, respectively. On the other hand, the language modeling based approach did not help to improve much for this case. There are several reasons why it did not improve the original approach much, but the major one is the fact that language modeling based sorting for an only in-domain parallel corpus may not be a good metric. If we have only one parallel corpus available and we are to sort it, then metrics such as translation quality of sentence pairs can be provide more promising results in only in-domain data case.

After using the n-gram language modeling based ranking approach, we experimented with various dependency based ranking approaches to select data. It can be seen in Table 7.2 that selection with dependency-based rankings did not overperform the n-gram based one. The representation *System1* was the best scoring representa-

Table 7.1. BLEU scores for the system on in-domain data only: t is the frequency threshold for the VSF algorithm.

System	Sentence Count	test2010	test2011	test2012	test2013
PB baseline	131K	7.94	7.94	8.02	7.09
VSF(t=1)	77K	7.34	7.46	7.48	6.69
VSF(t=2)	92K	7.56	7.51	7.57	7.06
VSF(t=5)	108K	7.60	7.71	7.56	6.83
sorted data + VSF(t=1)	85K	7.32	7.42	7.59	6.81
sorted data + VSF(t=2)	100K	7.29	7.60	7.70	6.88
sorted data + VSF(t=5)	115K	7.54	7.70	7.83	6.94

tion, hence we experimented only this setting in the machine translation systems that include the utilization of out-of-domain data.

Table 7.2. BLEU scores for the dependency-based ranked systems stated in Table 4.1: t is the frequency threshold for the VSF algorithm.

System	Sentence Count	test2010	test2011	test2012	test2013
PB baseline	131K	7.94	7.94	8.02	7.09
ngram based + VSF (t=1)	85K	7.32	7.42	7.59	6.81
System 1 + VSF (t=1)	89K	7.42	7.37	7.59	6.32
System 2 + VSF (t=1)	74K	6.92	6.86	7.34	6.48
System 3 + VSF (t=1)	91K	6.78	7.06	7.32	6.12
System 4 + VSF (t=1)	92K	7.09	6.75	7.14	6.22

7.2. Results with out-of-domain data

In English to Turkish translation systems that use out-of-domain data additionally, the best scoring system for test2012 set uses only 33% of the SETIMES data, that is our out-of-domain data. For test2011, again the same system achieves the best score. The improvement in the BLEU score is around 0.3 points. The improvements over individual systems for test2011 and test2012 were computed to be statistically

Table 7.3. BLEU scores for the systems utilizing out-of-domain data as well: t is the frequency threshold for the VSF algorithm (Sentence count starting with "+" indicates the additional amount of sentences included to the data of the baseline system shown in the first row).

System	Sentences	test2011	test2012	test2013
1. Only WIT	131K	7.94	8.02	7.09
2. (1) + SETIMES	+165K	8.00	8.1	7.16
3. (1) + ngram sorted-SETIMES + vsf(t=1)	+53K	8.24	**8.38**	7.27
4. (1) + ngram sorted-SETIMES + vsf(t=2)	+66K	8.12	8.2	7.15
5. (1) + ngram sorted-SETIMES + vsf(t=5)	+80K	8.05	8.15	6.88
6. (1) + dep. sorted-SETIMES + vsf(t=1)	+72K	**8.27**	**8.38**	7.19
7. (1) + dep. sorted-SETIMES + vsf(t=2)	+93K	7.89	8.11	**7.42**
8. (1) + dep. sorted-SETIMES + vsf(t=5)	+118K	8.18	8.32	7.2
9. linear(WIT + SETIMES)	+165K	7.64	7.81	7.16
10. fillup(WIT + SETIMES)	+165K	7.46	7.84	6.87

significiant with a 95% confidence interval ($p<0.05$) [35]. Note that the BLEU scores for this language pair are generally low due the differences between the Turkish and English languages. Turkish is morphologically more complex and the word orders between this language pair differ as well. Additionally, the size of the data for this pair does not reach to million sentences, which is generally a case for language pairs like French-English. In [21], it is discussed that SETIMES data was not helpful to increase the BLEU scores for English-Turkish translation in the IWSLT test sets. However, our results show that this data may be utilized carefully to improve translation quality of the corresponding sets.

The other proposed sorting metrics related dependency relations and part-of-speech tags have also shown minor improvements in some of the test sets. For test set test2013, the best scoring system is the one trained with dependency-based ranked out-of-domain corpus. What is more, the phrase table combination methods between different domains proposed in [22], did not bring any improvement for these data and

test sets for the English-Turkish language pair, hence our method outperforms them as well.

Table 7.4. Sample translation output from *test2013*.

Input	I had my first apartment, my first green little American Express card, and I had a very big secret.
Reference	İlk apartman daireme, ilk küçük, yeşil American Express kartıma sahip olmuştum ve çok büyük bir sırrım vardı.
Baseline	İlk küçük yeşil ilk dairem, American Express kartı, ve çok büyük bir sırrım vardı.
Our system	İlk küçük yeşil American Express kartım ilk dairemi aldım, ve çok büyük bir sırrım vardı.

Additionally, we compared our system with TUBITAK's best system for English-Turkish translation in the IWSLT 2013 evaluation campaign by implementing their work. In [21], it is stated that adding all of the SETIMES data did not improve the performance of their system, it even decreased it. Their best system was trained with hierarchical phrase-based translation [23] and made use of morphological and lexical features specific to the Turkish language. We adopted their work and also reported that the addition of the entire out-of-domain data to the baseline system decreases the BLEU score. Next, we integrated our proposed data selection methods to the system. The results that we obtained are shown in Table 7.5. It can be seen that the BLEU score has increased by 0.8 points and this score is higher than the best score in the corresponding IWSLT evaluation campaign for English-Turkish translation. The increase was tested using [35] and computed to be statistically significant with a 95% confidence interval ($p<0.05$). Although our replication of the system by [21] did not include some of the features that they have used and shown to improve performance, our system is still able to outperform their reported best system for the test2013 data set. In this system setting, we also experimented to apply VSF on non-sorted (original) SETIMES data and expand the in-domain corpus to see whether sorting makes a difference or not. The results in Table 7.5 showed that the n-gram sorted selection approach performs better than the non-sorted VSF based selection as well, even though it did not help

much in the experiments where the models are only trained with in-domain data.

Table 7.5. BLEU scores of the IWSLT 2013's best system and the proposed approaches: t is the frequency threshold for the VSF algorithm (Sentence count starting with "+" indicates the additional amount of sentences included to the data of the baseline system shown in the first row).

System	Sentences	test2013
1. TUBITAK IWSLT Best	131K	8.41
2. (1) + SETIMES	+165K	8.37
3. (1) + ngram sorted-SETIMES + vsf(t=1)	+53K	9.14
4. (1) + ngram sorted-SETIMES + vsf(t=2)	+66K	**9.20**
5. (1) + ngram sorted-SETIMES + vsf(t=5)	+80K	8.58
6. (1) + dep. sorted-SETIMES + vsf(t=1)	+72K	8.66
7. (1) + dep. sorted-SETIMES + vsf(t=2)	+93K	8.61
8. (1) + dep. sorted-SETIMES + vsf(t=5)	+118K	8.75
9. (1) + no-sort-SETIMES + vsf(t=1)	+74K	7.85
10. (1) + no-sort-SETIMES + vsf(t=2)	+94K	8.65
11. (1) + no-sort-SETIMES + vsf(t=5)	+121K	8.91

The results show that the proposed technique successfully utilizes the available out-of-domain data in such a way that there will be improvements in BLEU for the specified domain. Addition of all out-of-domain data to the translation systems leads to minor differences in the performances over the test sets when we take the amount of data into account. In Table 7.4, the improvement with respect to the baseline system can be observed on the translation output.

The approach is more a data adaptation technique within domains, rather than domain adaptation in which a pre-built system in a specific domain is being adapted to a completely different domain. It is useful for building better and more successful systems for a domain, where there is not much data in that domain, but there is a lot of data in different domains.

8. CONCLUSIONS

8.1. Discussion

The work presented in this thesis is a new approach for English-Turkish translation. This language pair does not have sufficient amount of parallel texts, hence it is important to fully utilize and use all the available texts from different domains. Due to the morphological and word-sequential differences between the English and Turkish languages, most translation systems produce low BLEU scores. Turkish is an agglutinative language and is subject-object-verb oriented, whereas English is more compact and subject-verb-object oriented. This is the reason why improvement in BLEU with this new approach seems limited yet the improvements are statistically significiant. Especially, languages like Turkish that exhibit free word order may require recursive-neural-network (RNN) language model scores for ranking since RNN based LMs handle long-distance dependencies better than conventional n-gram based LMs.

The proposed approach may easily be integrated to state-of-the-art machine translation systems and applied to other language pairs. Since additional and valuable out-of-domain data is selected through this method, we believe it will lead to improvement of machine translation systems' overall successes for other languages as well. The VSF and language modeling perplexity-based rankings are algorithms that have already been proposed for machine translation. However, the combination of these approaches for expanding in-domain training data with out-of-domain data is a new approach for statistical machine translation.

8.2. Future Work

In the proposed methodology, there are several future directions to investigate. Ranking the corpus with an external language model is a possible direction to follow. Other potential avenues for research are using different sentence sorting metrics and features. Instead of the sentence perplexity based scoring method, other features such

as sentence length or the feature functions introduced in [36] can be integrated into the system. Language dependent features can also be adapted, if a translation system for a specific pair is to be built.

Moreover, we plan to examine the effects of the method specifically on the translation model. That is, we will create phrase-tables from different domains and sort the phrases in the out-of-domain phrase table with respect to an in-domain language model. Then, we will apply VSF on the sorted out-of-domain phrase table to select a subset of phrases to concatenate with the in-domain phrase table in the machine translation system.

As mentioned up to now, we had data from two different domains in our experiments. We are planning to investigate the effect of adding more out-of-domain data from more different domains and to see if the system score will continue to increase or not.

8.3. Conclusion

In this thesis, we proposed several data selection methods for improving English-Turkish statistical machine translation systems. The improvements are in many perspectives such as increase in translation success and decrease in model size.

Model sizes are decreased via language modeling or depedendency modeling based VSF algorithm on in-domain training data. This decrease can be perceived from the sentence counts reported in each result table. The lower the sentence count, the smaller phrase options and language models will be produced after training. Using VSF for Turkish language and using language modeling based ranking before VSF were the brand new approaches investigated in this thesis.

Improvements in BLEU, on the other hand, were achieved through the utilization of the out-of-domain data combined with the in-domain data. To the best of our knowledge, this is the first study that uses VSF as a domain adaptation technique. Its

combination with n-gram and dependency language modeling are the novel contributions of this thesis as well. The proposed systems exhibited higher BLEU scores than the fill-up phrase table combination technique [22], which is one of the state-of-the-art domain adaptation methods based on expanding in-domain data with out-of-domain data. In addition, the proposed approach outperformed the state-of-the-art English-Turkish statistical translation system that has been developed by TUBITAK in the context of the IWSLT 2013 challenge [21].

REFERENCES

1. Koehn, P., *Statistical Machine Translation*, Cambridge University Press, 2010.

2. Eryigit, G., J. Nivre and K. Oflazer, "Dependency Parsing of Turkish.", *Computational Linguistics*, Vol. 34, No. 3, pp. 357–389, 2008.

3. Lewis, W. and S. Eetemadi, "Dramatically Reducing Training Data Size Through Vocabulary Saturation", *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pp. 281–291, Association for Computational Linguistics, Sofia, Bulgaria, August 2013.

4. Brown, P. F., J. Cocke, S. A. Della-Pietra, V. J. Della-Pietra, F. Jelinek, J. D. Lafferty, R. L. Mercer and P. Rossin, "A Statistical Approach to Machine Translation", *Computational Linguistics*, Vol. 16, No. 2, pp. 76–85, 1990.

5. Brown, P. F., S. A. Della-Pietra, V. J. Della-Pietra and R. L. Mercer, "The Mathematics of Statistical Machine Translation", *Computational Linguistics*, Vol. 19, No. 2, pp. 263–313, 1993.

6. Russell, S. J. and P. Norvig, *Artificial Intelligence - A Modern Approach (3. internat. ed.)*, Pearson Education, 2010.

7. Papineni, K., S. Roukos, T. Ward and W.-J. Zhu, *BLEU: A Method for Automatic Evaluation of Machine Translation*, Tech. Rep. RC22176(W0109-022), IBM Research Report, September 17 2001.

8. Eck, M., S. Vogel and A. Waibel, "Low Cost Portability for Statistical Machine Translation based on N-gram Frequency and TF-IDF", *Proceedings of the International Workshop on Spoken Language Translation*, October 2005.

9. Okita, T., "Data Cleaning for Word Alignment.", *In Proceedings of Joint Confer-*

ence of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP 2009) Student Research Workshop, pp. 72–80, 2009.

10. Bulyko, I., S. Matsoukas, R. M. Schwartz, L. Nguyen and J. Makhoul, "Language Model Adaptation in Machine Translation from Speech", *International Conference on Acoustics, Speech, and Signal Processing*, pp. 117–120, 2007.

11. Banerjee, P., J. Du, B. Li, S. Naskar, A. Way and J. van Genabith, "Combining Multi-Domain Statistical Machine Translation Models using Automatic Classifiers", *Proceedings of the Ninth Conference of the Association for Machine Translation in the Americas*, 2010.

12. Wang, W., K. Macherey, W. Macherey, F. Och and P. Xu, "Improved Domain Adaptation for Statistical Machine Translation", *Proceedings of the Tenth Conference of the Association for Machine Translation in the Americas (AMTA)*, 2012.

13. Wu, H., H. Wang and C. Zong, "Domain Adaptation for Statistical Machine Translation with Domain Dictionary and Monolingual Corpora", *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pp. 993–1000, Coling 2008 Organizing Committee, Manchester, UK, August 2008.

14. Moore, R. C. and W. D. Lewis, "Intelligent Selection of Language Model Training Data", *Association for Computational Linguistics (Short Papers)*, pp. 220–224, 2010.

15. Bertoldi, N. and M. Federico, "Domain Adaptation for Statistical Machine Translation with Monolingual Resources", *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pp. 182–189, Association for Computational Linguistics, Athens, Greece, March 2009.

16. Shen, L., J. Xu and R. Weischedel, "A New String-to-Dependency Machine Trans-

lation Algorithm with a Target Dependency Language Model", *Proceedings of ACL-08: HLT*, pp. 577–585, Association for Computational Linguistics, Columbus, Ohio, June 2008.

17. Guo, Y., J. van Genabith and H. Wang, "Dependency-Based N-Gram Models for General Purpose Sentence Realisation", *International Conference on Computational Linguistics (COLING)*, pp. 297–304, 2008.

18. Lambert, B., B. Raj and R. Singh, "Discriminatively Trained Dependency Language Modeling for Conversational Speech Recognition", *INTERSPEECH*, pp. 3414–3418, 2013.

19. Gubbins, J. and A. Vlachos, "Dependency Language Models for Sentence Completion", *Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pp. 1405–1410, 2013.

20. Oflazer, K. and I. Durgar El-Kahlout, "Exploring Different Representational Units in English-to-Turkish Statistical Machine Translation", *Proceedings of the Second Workshop on Statistical Machine Translation*, pp. 25–32, Association for Computational Linguistics, Prague, Czech Republic, June 2007.

21. Yılmaz, E., İlknur Durgar El-Kahlout, B. Aydın, Z. S. Özil and C. Mermer, "TÜBİTAK Turkish-English Submissions for IWSLT 2013", *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*, 2013.

22. Bisazza, A., N. Ruiz and M. Federico, "Fill-up versus Interpolation Methods for Phrase-based SMT Adaptation", M. Federico, M.-Y. Hwang, M. Rödder and S. Stüker (Editors), *Proceedings of the seventh International Workshop on Spoken Language Translation (IWSLT)*, pp. 136–143, 2011.

23. Chiang, D., "Hierarchical Phrase-Based Translation", *Computational Linguistics*, Vol. 33, No. 2, 2007.

24. Gascó, G., M.-A. Rocha, G. Sanchis-Trilles, J. Andrés-Ferrer and F. Casacuberta, "Does More Data Always Yield Better Translations?", *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 152–161, Association for Computational Linguistics, Avignon, France, April 2012.

25. de Marneffe, M.-C., B. MacCartney and C. D. Manning, "Generating Typed Dependency Parses from Phrase Structure Parses", *In Proceedings of International Conference on Language Resources and Evaluation (LREC)*, pp. 449–454, 2006.

26. Stolcke, A., "SRILM-An Extensible Language Modeling Toolkit", *Proceedings International Conference on Spoken Language Processing*, pp. 257–286, November 2002.

27. Koehn, P., F. J. Och and D. Marcu, "Statistical Phrase Based Translation", *Proceedings of the Joint Conference on Human Language Technologies and the Annual Meeting of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL)*, 2003.

28. Gao, Q. and S. Vogel, "Parallel Implementations of Word Alignment Tool", *Software Engineering, Testing, and Quality Assurance for Natural Language Processing*, pp. 49–57, Association for Computational Linguistics, Columbus, Ohio, June 2008.

29. Koehn, P., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. J. Dyer, O. Bojar, A. Constantin and E. Herbst, "Moses: Open Source Toolkit for Statistical Machine Translation", *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pp. 177–180, Association for Computational Linguistics, Prague, Czech Republic, June 2007.

30. Och, F. J., "Minimum Error Rate Training in Statistical Machine Translation",

E. Hinrichs and D. Roth (Editors), *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pp. 160–167, 2003.

31. Kneser, R. and H. Ney, "Improved Backing-Off for M-Gram Language Modeling", *Proceedings of the IEEE International Conference on Accoustics, Speech and Signal Processing*, Vol. 1, 1995.

32. Heafield, K., "KenLM: Faster and Smaller Language Model Queries", *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pp. 187–197, Association for Computational Linguistics, Edinburgh, Scotland, July 2011.

33. Cettolo, M., C. Girardi and M. Federico, "WIT3: Web Inventory of Transcribed and Translated Talks", M. Cettolo, M. Federico, L. Specia and A. Way (Editors), *Proceedings of th 16th International Conference of the European Association for Machine Translation (EAMT)*, pp. 261–268, 2012.

34. Tyers, F. M. and M. S. Alperen, "South-East European Times: A Parallel Corpus of the Balkan Languages", *Proceedings of the LREC Workshop on Exploitation of Multilingual Resources and Tools for Central and (South) Eastern European Languages*, 2010.

35. Koehn, P., "Statistical Significance Tests for Machine Translation Evaluation", D. Lin and D. Wu (Editors), *Proceedings of the Conference on Empirical Methods on Natural Language Processing (EMNLP) 2004*, pp. 388–395, Association for Computational Linguistics, Barcelona, Spain, July 2004.

36. Taghipour, K., S. Khadivi and J. Xu, "Parallel Corpus Refinement as an Outlier Detection Algorithm", *Proceedings of the 13th Machine Translation Summit (MT Summit XIII)*, pp. 414–421, International Association for Machine Translation, 2011.