Accepted Manuscript

Available online: 31 May, 2017

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain. Articles in Press are accepted, peer reviewed articles that are not yet assigned to volumes/issues, but are citable using DOI.

VNU Journal of Science: Comp. Science & Com. Eng., Vol. 31, No. 3 (2017) 1–13

Dependency-based Pre-ordering For English-Vietnamese Statistical Machine Translation

Tran Hong Viet [1,2], Nguyen Van Vinh[2], Vu Thuong Huyen[3], Nguyen Le Minh[4]

[1]*University of Economic and Technical Industries, Hanoi, Vietnam*
[2]*University of Engineering and Technology, Vietnam National University, Hanoi, Vietnam*
[3]*ThuyLoi University, Hanoi, Vietnam*
[4]*Japan Advanced Institute of Science and Technology*
Email: thviet@uneti.edu.vn, vinhnv@vnu.edu.vn, huyenvt@tlu.edu.vn, nguyenml@jaist.ac.jp

Abstract

Reordering is a major challenge in machine translation (MT) between two languages with significant differences in word order. In this paper, we present an approach as pre-processing step based on a dependency parser in phrase-based statistical machine translation (SMT) to learn automatic and manual reordering rules from English to Vietnamese. The dependency parse trees and transformation rules are used to reorder the source sentences and applied for systems translating from English to Vietnamese. We evaluated our approach on English-Vietnamese machine translation tasks, and showed that it outperforms the baseline phrase-based SMT system.

Keywords: Natural Language Processing, Machine Translation, Phrase-based Statistical Machine Translation.

1. Introduction

Phrase-based statistical machine translation [1] is the state-of-the-art of SMT because of its power in modelling short reordering and local context. However, with phrase-based SMT, long distance reordering is still problematic. The reordering problem (global reordering) is one of the major problems, since different languages have different word order requirements. In recent years, many reordering methods have been proposed to tackle the long distance reordering problem.

Many solutions solving the reordering problem have been proposed, such as syntax-based model [2], lexicalized reordering [3]. Chiang [2] shows significant improvements by keeping the strengths of phrases, while incorporating syntax into SMT. Some approaches were applied at the word level [4]. They are useful for language with rich morphology, for reducing data sparseness. Other kinds of syntax reordering methods require parser trees, such as the work in [4]. The parsed tree is more powerful in capturing the sentence structure. However, it is expensive to create tree structure and build a good quality parser. All the above approaches require much decoding time, which is expensive.

The approach that we are interested in is balancing the quality of translation with decoding time. Reordering approaches as a preprocessing step [5, 6, 7] are very effective (significant improvement over state of-the-art phrase-based and hierarchical machine translation systems and separately quality evaluation of each reordering models).

The end-to-end neural MT (NMT) approach [8] has recently been proposed for MT. However, the NMT method has some limitations that may jeopardize its ability to generate better translation. The NMT system usually causes a serious out-of-vocabulary (OOV) problem, the translation quality would be badly hurt; The NMT de-

* Corresponding author. Email: thviet@uneti.edu.vn

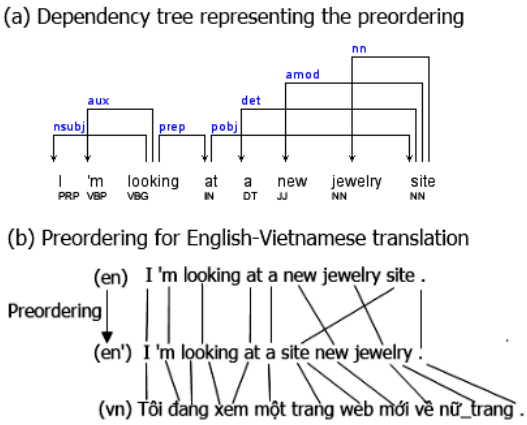

Figure 1: A example of preordering for English-Vietnamese translation.

coder lacks a mechanism to guarantee that all the source words are translated and usually favors short translations. It is difficult for an NMT system to benefit from target language model trained on target monolingual corpus, which is proven to be useful for improving translation quality in statistical machine translation (SMT). NMT need much more training time. In [9], NMT requires longer time to train (18 days) compared to their best SMT system (3 days).

Inspire by this preprocessing approaches, we propose a combined approach which preserves the strength of phrase-based SMT in reordering and decoding time as well as the strength of integrating syntactic information in reordering. Firstly, the proposed method uses a dependency parsing for preprocessing step with training and testing. Secondly, transformation rules are applied to reorder the source sentences. The experimental resulting from English-Vietnamese pair shows that our approach achieved improvements in BLEU scores [10] when translating from English, compared to MOSES [11] which is the state of-the-art phrase-based SMT system.

This paper is structured as follows: Section 1 introduces the reordering problem. Section 2 reviews the related works. Section 3 introduces phrase-based SMT. Section 4 expresses how to apply transformation rules for reordering the source sentences. Section 5 presents a the learning model in order to transform the word order of an input sentence to an order that is natural in the target languages. Section 6 describes experimental results; Section 7 discusses the experimental results. And, conclusions are given in Section 8.

2. Related works

The difference of the word order between source and target languages is the major problem in phrase-based statistical machine translation. Fig 1 describes an example that a reordering approach modifies the word order of an input sentence of a source languages (English) in order to generate the word order of a target languages (Vietnamese).

Many preordering methods using syntactic information have been proposed to solve the reordering problem. (Collin 2005; Xu 2009) [4, 5] presented a preordering method which used manually created rules on parse trees. In addition, linguistic knowledge for a language pair is necessary to create such rules. Other preordering methods using automatic created reordering rules or a statistical classifier were studied [12, 7]

Collins [4] developed a clause detection and used some handwritten rules to reorder words in the clause. Partly, (Habash 2007)[13] built an automatic extracted syntactic rules. Xu [5] described a method using a dependency parse tree and a flexible rule to perform the reordering of subject, object, etc... These rules were written by hand, but [5] showed that an automatic rule learner can be used.

Bach [14] propose a novel source-side dependency tree reordering model for statistical machine translation, in which subtree movements and constraints are represented as reordering events associated with the widely used lexicalized reordering models.

(Genzel 2010; Lerner and Petrov 2013) [6, 7] described a method using discriminative classifiers to directly predict the final word order. Cai [15] introduced a novel pre-ordering approach based on dependency parsing for Chinese-English SMT.

Isao Goto [16] described a preordering method using a target-language parser via cross-language

syntactic projection for statistical machine translation.

Joachim Daiber [17] presented a novel examining the relationship between preordering and word order freedom in Machine Translation.

Chenchen Ding, [18] proposed extra-chunk pre-ordering of morphemes which allows Japanese functional morphemes to move across chunk boundaries.

Christian Hadiwinoto presented a novel reordering approach utilizing sparse features based on dependency word pairs [19] and presented a novel reordering approach utilizing a neural network and dependency-based embedding to predict whether the translations of two source words linked by a dependency relation should remain in the same order or should be swapped in the translated sentence [9]. This approach is complex and spend much time to process.

However, there were not definitely many studies on English-Vietnamese to SMT system tasks. To our knowledge, no research address reordering models for English-Vietnamese SMT based on dependency parsing. In comparison with these mentioned approaches, our proposed method has some differences as follows: We investigate to use a reordering models for English-Vietnamese SMT using dependency information. We study SVO language in English-Vietnamese in order to recognize the differences about English-Vietnamese word labels, phrase label as well as dependency labels. We use dependency parser of English sentence for translating from English to Vietnamese. Base on above studies, we utilize the English - Vietnamese transformation rules (manual and automatic rules are extracted from English-Vietnamese parallel corpus) that directly predict target-side word as a preprocessing step in phrase-based machine translation. As the same with [13], we also applied preprocessing in both training and decoding time.

3. Brief Description of the Baseline Phrase-based SMT

In this section, we will describe the phrase-based SMT system which was used for the experiments. Phrase-based SMT, as described by [1] translates a source sentence into a target sentence by decomposing the source sentence into a sequence of source phrases, which can be any contiguous sequences of words (or tokens treated as words) in the source sentence. For each source phrase, a target phrase translation is selected, and the target phrases are arranged in some order to produce the target sentence. A set of possible translation candidates created in this way were scored according to a weighted linear combination of feature values, and the highest scoring translation candidate was selected as the translation of the source sentence. Symbolically,

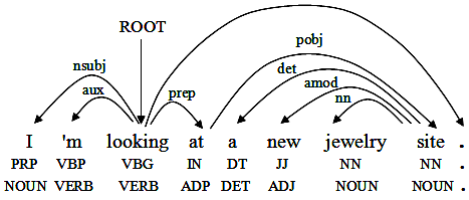

Figure 2: A example with POS tags and dependency parser.

$$\hat{t} = \text{argmax}\ t, a \sum_{i=1}^{n} \lambda_i f_j(s, t, a) \quad (1)$$

when s is the input sentence, t is a possible output sentence, and a is a phrasal alignment that specifies how t is constructed from s, and $\hat{t}$ is the selected output sentence. The weights λ_i associated with each feature f_i are tuned to maximize the quality of the translation hypothesis selected by the decoding procedure that computes the argmax. The log-linear model is a natural framework to integrate many features. The probabilities of source phrase given target phrases, and target phrases given source phrases, are estimated from the bilingual corpus.

Koehn [1] used the following distortion model (reordering model), which simply penalizes non-monotonic phrase alignment based on the word distance of successively translated source phrases with an appropriate value for the parameter α:

$$d(a_i - b_{i-1}) = \alpha^{|a_i - b_{i-1} - 1|} \quad (2)$$

Moses [11] is open source toolkit for statistical machine translation system that allows automatically train translation models for any language pair. When we have a trained model, an efficient search algorithm quickly finds the highest probability translation among the exponential number of choices. In our work, we also used Moses to evaluate on English-Vietnamese machine translation tasks.

4. Dependency Syntactic Preprocessing For SMT

Reordering approaches on English-Vietnamese translation task have limitation. In this paper, we firstly produce a parse tree using dependency parser tools [20]. Figure 3 shows an example of parsed a English sentence.

Source sentence:
that songwriter wrote many romantic songs .
Tagging:
that/DT songwriter/NN wrote/VBD many/JJ romantic/JJ songs/NNS ./.
Parse:
(ROOT
(S
(NP (DT that) (NN songwriter))
(VP (VBD wrote)
(NP (JJ many) (JJ romantic) (NNS songs)))
(. .)))
det(songwriter-2, that-1)
nsubj(wrote-3, songwriter-2)
root(ROOT-0, wrote-3)
amod(songs-6, many-4)
amod(songs-6, romantic-5)
dobj(wrote-3, songs-6)

Figure 3: Example about Dependency Parser of an English sentence using Stanford Parser

Then, we utilize some dependency relations extracted from a statistical dependency parser to create the dependency based on reordering rules. Dependency parsing among words typed with grammatical relations are proven as useful information in some applications relative to syntactic processing.

We use the dependency grammars and the differences of word order between Vietnamese and English to create a set of the reordering rules. There are approximately 50 grammatical relations in English, meanwhile there are 27 ones in Vietnamese based on [21] and the differences of word order between English and Vietnamese to create the set of the reordering rules. Base on these rules, we propose an our method which is capable of applying and combining them simultaneously. We utilize the word labels in [21] to analyze the extract POS tags and head modifier dependencies.

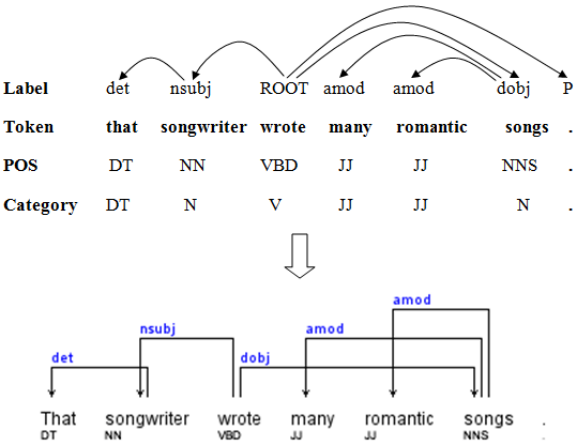

Figure 4: Representation of the Stanford Dependencies for the English source sentence

In addition, we focus on analyzing some popular structures of English language when translating to Vietnamese language. This analysis can achieve remarkable improvements in translation performance. Because English and Vietnamese both are SVO languages, the order of verb rarely change, we focus mainly on some typical relations as noun phrase, adjectival and adverbial phrase, preposition and created manually written reordering rule set for English-Vietnamese language pair. Inspired from [5], our study employ dependency syntax and transyntaxsformation rules to reorder the source sentences and applied to English-Vietnamese translation system.

For example, with noun phrase, there always exists a head noun and the components before and after it. These auxiliary components will move to new positions according to Vietnamese translational order.

Let us consider an example in Figure 6, Fig-

ure 7 to the difference of word order in English and Vietnamese noun phrase and adjectival and adverbial phrase.

4.1. Transformation Rule

This section, we describe a transformation rule.

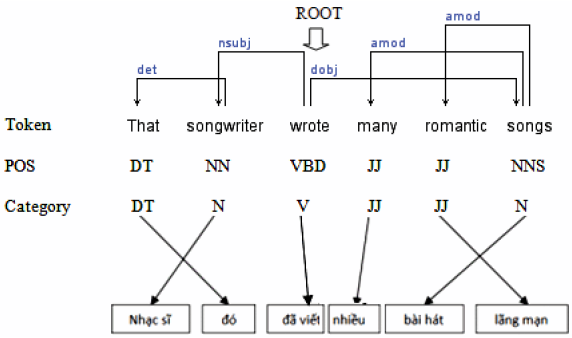

Figure 5: An Example of using Dependency Syntactic before and after our preprocessing

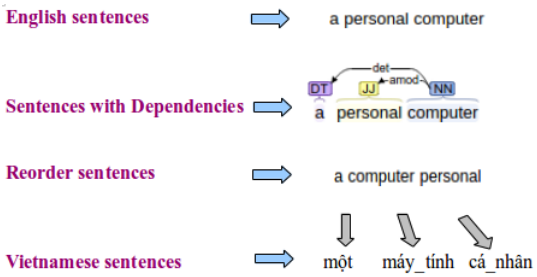

Figure 6: An example of word reordering phenomenon in noun phrase with adjectival modifier (amod) and determiner modifier (det). In this example, the noun "computer" is swapped with the adjectival "personal".

Our rule set is for English-Vietnamese phrase-based SMT. Table 1 shows handwritten rules using dependency syntactic preprocessing to reorder from English to Vietnamese.

In the proposed approach, a transform rule is a mapping from T to a set of tuples (L, W, O)

- T is the part-of-speech (POS) tag of the head in a dependency parse tree node.

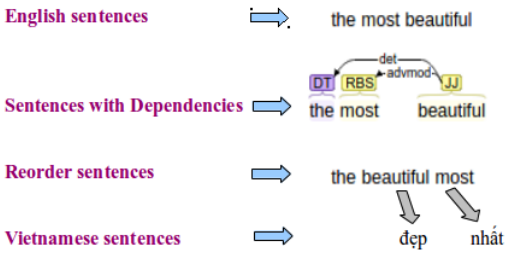

Figure 7: An example of word reordering phenomenon in adjectival phrase with adverbial modifier (advmod) and determiner modifier (det).

- L is a dependency label for a child node.
- W is a weight indicating the order of that child node.
- O is the type of order (either NORMAL or REVERSE).

Our rule set provides a valuable resource for preordering in English-Vietnamese phrase-based SMT.

4.2. Dependency Syntactic Processing

We aim to reorder an English sentence to get a new English, and some words in this sentence are arranged as Vietnamese words order. The type of order is only used when we have multiple children with the same weight, while the weight is used to determine the relative order of the children, going from the largest to the smallest. The weight can be any real valued number. The order type NORMAL means we preserve the original order of the children, while REVERSE means we flip the order. We reserve a special label self to refer to the head node itself so that we can apply a weight to the head, too. We will call this tuple a precedence tuple in later discussions. In this study, we use manually created rules only.

Suppose we have a reordering rule: NNS → (prep, 0, NORMAL), (rcmod, 1, NORMAL), (self, 0, NORMAL), (poss, -1, NORMAL), (admod,-2, REVERSE). For the example shown in Figure 4, we would apply it to the ROOT node and result in "songwriter that wrote many songs romantic."

We apply them in a dependency tree recursively starting from the root node. If the POS tag

T	(L, W, O)
JJ or JJS or JJR	(advcl,1,NORMAL)
	(self,-1,NORMAL)
	(aux,-2,REVERSE)
	(auxpass,-2,REVERSE)
	(neg,-2,REVERSE)
	(cop,0,REVERSE)
NN or NNS	(prep,0,NORMAL)
	(rcmod,1,NORMAL)
	(self,0,NORMAL)
	(poss,-1, NORMAL)
	(admod,-2,REVERSE)
IN or TO	(pobj,1,NORMAL)
	(self,2,NORMAL)

Table 1: Handwritten rules For Reordering English to Vietnamese using Dependency syntactic preprocessing

of a node matches the left-hand-side of a rule, the rule is applied and the order of the sentence is changed. We go through all the children of the node and get the precedence weights for them from the set of precedence tuples. If we encounter a child node that has a dependency label not listed in the set of tuples, we give it a default weight of 0 and default order type of NORMAL. The children nodes are sorted according to their weights from highest to lowest, and nodes with the same weights are ordered according to the type of order defined in the rule.

Figure 5 gives examples of original and preprocessed phrase in English. The first line is the original English sentences: "that songwriter wrote many songs romantic.", and the fourth line is the target Vietnamese reordering "Nhạc sĩ đó đã viết nhiều bài hát lãng mạn.". This sentences is arranged as the Vietnamese order. We aim to preprocess as in Figure 5. Vietnamese sentences is the output of our method. As you can see, after reordering, original English line has the same word order.

5. Classifier-based Preordering for Phrase-based SMT

Current time, state-of-the-art phrase-based SMT system using the lexicalized reordering model in Moses toolkit. In our work, we also used Moses to evaluate on English-Vietnamese machine translation tasks.

5.1. Classifier-based Preordering

In this section, we describe a the learning model that can transform the word order of an input sentence to an order that is natural in the target language. English is used as source language, while Vietnamese is used as target language in our discussion about the word orders.

For example, when translating the English sentence:

I 'm looking at a new jewelry site.

to Vietnamese, we would like to reorder it as:

I 'm looking at a site new jewelry.

And then, this model will be used in combination with translation model.

The feature is built for "site, a, new, jewelry" family in Figure 2:

NN, DT, det, JJ, amod, NN, nn, 1230, 1023

We use the dependency grammars and the differences of word order between English and Vietnamese to create a set of the reordering rules. From part-of-speech (POS) tag and parse the input sentence, producing the POS tags and head-modifier dependencies shown in Figure 2. Traversing the dependency tree starting at the

Corpus	Sentence pairs	Training Set	Development Set	Test Set
General	132636	131236	400	1000
			Vietnamese	English
Training	Sentences		131236	
	Average Length		18.91	17.98
	Word		2481762	2360727
	Vocabulary		39071	54086
Development	Sentences		400	
	Average Length		22.73	21.41
	Word		9092	8567
	Vocabulary		1537	1920
Test	Sentences		1000	
	Average Length		22.70	21.42
	Word		22707	21428
	Vocabulary		2882	3816

Table 2: Corpus Statistical

Feature	**Description**
T	The head's POS tag
1T	The first child's POS tag
1L	The first child's syntactic label
2T	The second child's POS tag
2L	The second child's syntactic label
3T	The third child's POS tag
3L	The third child's syntactic label
4T	The fourth child's POS tag
4L	The fourth child's syntactic label
O1	The sequence of head and its children in source alignment
O2	The sequence of head and its children in target alignment.

Table 3: Set of features used in training data from corpus English-Vietnamese

root to reordering. We determine the order of the head and its children (independently of other decisions) for each head word and continue the traversal recursively in that order. In the above example, we need to decide the order of the head "looking" and the children "I", "'m", and "site.".

The words in sentence are reordered by a new sequence learned from training data using multi-classifier model. We use SVM classification model [22] that supports multi-class prediction. The class labels are corresponding to reordering sequence, so it is enable to select the best one from many possible sequences.

5.2. Features

The features extracted based on dependency tree includes POS tag and alignment information. We traverse the tree from the top, in each family we create features with the following information:

- The head's POS tag.
- The first child's POS tag, the first child's syntactic label.
- The second child's POS tag, the second child's syntactic label.
- The third child's POS tag, the third child's syntactic label.
- The fourth child's POS tag, the fourth child's syntactic label.
- The sequence of head and its children in source alignment.
- The sequence of head and its children in target alignment. It is class label for SVM classifier model.

We limited our self by processing families that have less than five children based on counting total families in each group: 1 head and 1 child, 1 head and 2 children, 1 head and 3 children, 1 head

Pattern	Order	Example
NN, DT, det, JJ, amod, NN, nn	1,0,2,3	I 'm looking at a new jewelry site . →I 'm looking at a site new jewelry .
NNS, JJ, amod, CC, cc, NNS, con	2,1,0,3	it faced a blank wall . → it faced a wall blank .
NNP, NNP, nn, NNP, nn	2,1,0	it 's a social phenomenon . → it 's a phenomenon social .

Table 4: Examples of rules and reorder source sentences

Algorithm 1 Extract rules

```
input: dependency trees of source sentences
       and alignment pairs;
output: set of automatic rules;
for each family in dependency trees of subset
and alignment pairs of sentences do
   generate feature (pattern + order) ;
end for
Build model from set of features;
for each family in dependency trees in the rest
of the sentences do
   generate pattern for prediction;
   get predicted order from model;
   add (pattern, order) as new rule in set of rules;
end for
```

Algorithm 2 Apply rule

```
input: source-side dependency trees , set of rules;
output: set of new sentences;
for each dependency tree do
   for each family in tree do
      generate pattern
      get order from set of rules based on pattern
      apply transform
   end for
   Build new sentence;
end for
```

and 4 children ... We found out that the most common families appear (80%) in our training sentences is less than and equal four children.

We trained a separate classifier for each number of possible children. In hence, the classifiers learn to trade off between a rich set of overlapping features. List of features are given in table 3.

We use SVM classification model in the WEKA tools [23] that supports multi-class prediction. Since it naturally supports multi-class prediction and can therefore be used to select one out of many possible permutations. The learning algorithm produces a sparse set of features. In our experiments, the models were based on features that generated from 100k English - Vietnamese sentence pairs.

When extracting the features, every word can be represented by its word identity, its POS-tags from the treebank, syntactic label. We also include pairs of these features, resulting in potentially bilexical features.

5.3. Training Data for Preordering

In this section, we describe a method to build training data for a pair English to Vietnamese. Our purpose is to reconstruct the word order of input sentence to an order that is arranged as Vietnamese words order.

For example with the English sentence in Figure 2:

I 'm looking at a new jewelry site.

is transformed into Vietnamese order:

I 'm looking at a site new jewelry.

For this approach, we first do preprocessing to encode some special words and parser the sentences to dependency tree using Stanford Parser [24]. Then, we use target to source alignment and dependency tree to generate features. We add source, target alignment, POS tag, syntactic label of word to each node in the dependency tree. For each family in the tree, we generate a training instance if it has less than and equal four children. In

case, a family has more than and equal five children, we discard this family but still keep traversing at each child.

Each rule consists of: pattern and order. For every node in the dependency tree, from the top-down, we find the node matching against the pattern, and if a match is found, the associated order applies. We arrange the words in the English sentence, which is covered by the matching node, like Vietnamese words order. And then, we do the same for each children of this node. If any rule is applied, we use the order of original sentence. These rules are learnt automatically from bilingual corpora. The our algorithm's outline is given as Alg. 1 and Alg. 2

Algorithm 1 extracts automatically the rules with input including dependency trees of source sentences and alignment pairs.

Algorithm 2 proceeds by considering all rules after finish Algorithm 1 and source-side dependency trees to build new sentence.

5.4. Classification Model

The reordering decisions are made by multi-class classifiers (correspond with number of permutation: 2, 6, 24, 120) where class labels correspond to permutation sequences. We train a separate classifier for each number of possible children. Crucially, we do not learn explicit tree transformations rules, but let the classifiers learn to trade off between a rich set of overlapping features. To build a classification model, we use SVM classification model in the WEKA tools. The following result are obtained using 10 folds-cross validation.

We apply them in a dependency tree recursively starting from the root node. If the POS-tags of a node matches the left-hand-side of the rule, the rule is applied and the order of the sentence is changed. We go through all the children of the node and matching rules for them from the set of automatically rules.

Table 4 gives examples of original and preprocessed phrase in English. The first line is the original English: " I'm looking at a new jewelry site .", and the target Vietnamese reordering " Tôi đang xem một trang web mới về nữ_trang .". This sentences is arranged as the Vietnamese order. Vietnamese sentences are the output of our method. As you can see, after reordering, the original English line has the same word order: " I 'm looking at a site new jewelry ." in Figure 1.

6. Experimental Results

6.1. Data set and Experimental Setup

For evaluation, we used an Vietnamese-English corpus [25], including about 131236 pairs for training, 1000 pairs for testing and 400 pairs for development test set. Table 2 gives more statistical information about our corpora. We conducted some experiments with SMT Moses Decoder [11] and SRILM [26]. We trained a trigram language model using interpolate and kndiscount smoothing with Vietnamese mono corpus. Before extracting phrase table, we use GIZA++ [3] to build word alignment with grow-diag-final-and algorithm. Besides using preprocessing, we also used default reordering model in Moses Decoder: using word-based extraction (wbe), splitting type of reordering orientation to three classes (monotone, swap and discontinuous – msd), combining backward and forward direction (bidirectional) and modeling base on both source and target language (fe) [11]. To contrast, we tried preprocessing the source sentence with manual rules and automatic rules.

We implemented as follows:

- We used Stanford Parser [24] to parse source sentence and apply to preprocessing source sentences (English sentences).
- We used classifier-based preordering by using SVM classification model [22] in Weka tools [23] for training the features-rich discriminative classifiers to extract automatic rules and apply them for reordering words in English sentences according to Vietnamese word order.
- We implemented preprocessing step during both training and decoding time.
- Using the SMT Moses decoder [11] for decoding.

We give some definitions for our experiments:

- **Baseline**: use the baseline phrase-based SMT system using the lexicalized reordering model in Moses toolkit.
- **Manual Rules**: the phrase-based SMT systems applying manual rules [27].
- **Auto Rules** : the phrase-based SMT systems applying automatic rules [28].
- **Auto Rules + Manual Rules**: the phrase-based SMT systems applying automatic rules, then applying manual rules.

6.2. Using Manual Rules

In this section, we present our experiments to translate from English to Vietnamese in a statistical machine translation system. We used Stanford Parser [24] to parse source sentence and apply to preprocessing source sentences (English sentences). According to typical differences of word order between English and Vietnamese, we have created a set of dependency-based rules for reordering words in English sentence according to Vietnamese word order and types of rules including noun phrase, adjectival and adverbial phrase, preposition which is described in table 1.

6.3. Using Automatic Rules

We present our experiments to translate from English to Vietnamese in a statistical machine translation system. In hence, the language pair chosen is English-Vietnamese. We used Stanford Parser [24] to parse source sentence (English sentences).

We used dependency parsing and rules extracted from training the features-rich discriminative classifiers for reordering source-side sentences. The rules are automatically extracted from English-Vietnamese parallel corpus and the dependency parser of English examples. Finally, they used these rules to reorder source sentences. We evaluated our approach on English-Vietnamese machine translation tasks with systems in table 5 which shows that it can outperform the baseline phrase-based SMT system.

6.4. BLEU score

The result of our experiments in table 6 showed size of phrase tables built from translation model base on our method. In this method, we can find out various phrases in the translation model. So that, they enable us to have more options for decoder to generate the best translation.

Table 7 describes the BLEU score of our experiments. As we can see, by applying preprocessing in both training and decoding, the BLEU score of "Auto Rules" system is lower by 0.49 point than "Manual Rules" system. This result is due to the fact that manual rules have better quality than automatic rules. However, "Auto Rules + Manual Rules" system is the best system because applying the combination rules can cover much linguistic phenomena.

The above result proved that the effect of applying transformation rule base on the dependency parse tree.

7. Analysis and Discussion

We have found that in our experiments work is sufficiently correlated to the translation quality done manually. Besides, we also have found some errors cause such as parse tree source sentence quality, word alignment quality and quality of corpus. All the above errors can effect automatic reordering rules. Table 9 showed the translation output examples are better than baseline system produced by our system for the input sentences from English-Vietnamese test set. Go here for more examples of translations for input sentences sampled randomly from our corpus. Some phrases in English source sentence were reordered corresponding to Vietnamese target sentence order. We focus mainly on some typical relations as noun phrase, adjectival and adverbial phrase, preposition and created manually written reordering rule set for English-Vietnamese

Name	Description
Baseline	Phrase-based system
Manual Rules	Phrase-based system with corpus which preprocessed using manual rules
Auto Rules	Phrase-based system with corpus which preprocessed using automatic learning rules
Auto Rules + Manual Rules	Phrase-based system with corpus which preprocessed using automatic learning rules and manual rules

Table 5: Our experimental systems on English-Vietnamese parallel corpus

Name	Size of phrase-table
Baseline	1152216
Manual Rules	1231365
Auto Rules	1213401
Auto Rules + Manual Rules	1253401

Table 6: Size of phrase tables

System	BLEU (%)
Baseline	36.89
Manual Rules	37.71
Auto Rules	37.12
Auto Rules + Manual Rules	37.85

Table 7: Translation performance for the English-Vietnamese task

Number children of head	**Number**	**Description**
1	79142	Family has 1 children
2	40822	Family has 2 children
3	26008	Family has 3 children
4	15990	Family has 4 children
5	7442	Family has 5 children
6	2728	Family has 6 children
7	942	Family has 7 children
8	307	Family has 8 children
9	83	Family has 9 children

Table 8: Statistical number of family on corpus English-Vietnamese

language pair. Our study employed dependency syntactic and transformation rules to reorder the source sentence and applied to English to Vietnamese translation systems.

For example, with noun phrase, there always exists a head noun and the components before and after it. These auxiliary components will move to new positions according to Vietnamese translational order. These rules can popular source linguistic phenomena equivalent to target language ones as follows:

- The phrase-based systems applying rules with category JJ or JJS
- The phrase-based systems applying rules with category NN or NNS
- The phrase-based systems applying rules with category IN or TO

Based on these phenomena, translation quality has significantly improved. We carried out error analysis sentences and compared to the golden reordering. Our analysis has also the benefits of automatic reordering rules on translation quality. In combination with machine learning method in related work [7], it is shown that applying classifier method to solve reordering problems automatically.

According to typical differences of word order between English and Vietnamese, we have created a set of automatic rules for reordering words in English sentence according to Vietnamese word order and types of rules including noun phrase, adjectival and adverbial phrase, as well as preposition phrase. Table 8 gives statistical families which have larger or equal 4 children in our corpus. The number of children in each family has limited 4 children in our approach. So in target language (Vietnamese), the number of children in each family is the same.

The manual rules have good quality [5, 13], the phrase-based SMT systems applying manual rules is better than the phrase-based SMT sys-

Input sentence:	Translation (Baseline):	Translation (Auto):	Translation (human):
The coat was far too big - it completely enveloped him .	Chiếc áo khoác là quá lớn - nó hoàn toàn phủ anh ta .	Chiếc áo khoác là quá lớn - nó phủ hoàn toàn anh ta .	Chiếc áo khoác quá lớn - nó hoàn toàn phủ anh ta .
Manh Cuong is a young football player with potential great .	Manh Cuong là một cầu thủ bóng đá với nhiều tiềm năng .	Manh Cuong là một cầu thủ bóng đá trẻ có tiềm năng lớn .	Mạnh Cường là cầu thủ bóng đá trẻ rất nhiều triển vọng .

Table 9: An example of a translation produced by our system for an input sentence sampled from English-Vietnamese corpus.

tems applying automatic rules. We believe that the quality of the phrase-based SMT systems applying automatic rules will be better when we have a better corpus.

8. Conclusion

In this paper, we present a preprocessing approach based on the dependency parser. The proposed approach is applying for English - Vietnamese translation system. The experimental results show that our approach achieved statistical improvements in BLEU scores over a state-of-the-art phrase-based baseline system. By applying manual rules and automatic rules, the quality of English-Vietnamese translation system is improving. In our study, our rules cover some linguistic reordering phenomena. These reordering rules benefit English-Vietnamese languages pair.

We will focus on word order problems much more with linguistic reordering phenomena on English-Vietnamese to learn better the dependency-based reordering rules (manual rules and automatic rules). This is necessary in improving SMT systems and that might lead to its a wider adoption.

Acknowledgment

This work described in this paper has been partially funded by Hanoi National University (QG.15.23 project)

References

[1] P. Koehn, F. J. Och, D. Marcu, Statistical phrase-based translation, in: Proceedings of HLT-NAACL 2003, Edmonton, Canada, 2003, pp. 127–133.

[2] D. Chiang, A hierarchical phrase-based model for statistical machine translation, in: Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05), Ann Arbor, Michigan, 2005, pp. 263–270.

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

[4] M. Collins, P. Koehn, I. Kucerová, Clause restructuring for statistical machine translation, in: Proc. ACL 2005, Ann Arbor, USA, 2005, pp. 531–540.

[5] P. Xu, J. Kang, M. Ringgaard, F. Och, Using a dependency parser to improve smt for subject-object-verb languages, in: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Association for Computational Linguistics, Boulder, Colorado, 2009, pp. 245–253.

[6] D. Genzel, Automatically learning source-side reordering rules for large scale machine translation, in: Proceedings of the 23rd International Conference on Computational Linguistics, COLING '10, 2010, pp. 376–384.

[7] U. Lerner, S. Petrov, Source-side classifier preordering for machine translation., in: EMNLP, 2013, pp. 513–523.

[8] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, et al., Googles neural machine translation system: Bridging the gap between human and machine translation, arXiv preprint arXiv:1609.08144, 2016.

[9] C. Hadiwinoto, H. T. Ng, A dependency-based neural reordering model for statistical machine translation, arXiv preprint arXiv:1702.04510, 2017.

[10] S. R. T. W. Papineni, Kishore, W. Zhu, Bleu: A method for automatic evaluation of machine translation., in: ACL, 2002.

[11] 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, E. Herbst, Moses: Open source toolkit for statistical machine translation, in: Proceedings of ACL, Demonstration Session, 2007.

[12] N. Yang, M. Li, D. Zhang, N. Yu, A ranking-based approach to word reordering for statistical machine translation, in: Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1, Association for Computational Linguistics, 2012, pp. 912–920.

[13] N. Habash, Syntactic preprocessing for statistical machine translation, Proceedings of the 11th MT Summit, 2007.

[14] N. Bach, Q. Gao, S. Vogel, Source-side dependency tree reordering models with subtree movements and constraints, in: Proceedings of the Twelfth Machine Translation Summit (MTSummit-XII), International Association for Machine Translation, Ottawa, Canada, 2009.

[15] E. S. Y. Z. Jingsheng Cai, Masao Utiyama, Dependency-based pre-ordering for chinese-english machine translation, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, 2014.

[16] I. Goto, M. Utiyama, E. Sumita, S. Kurohashi, Preordering using a target-language parser via cross-language syntactic projection for statistical machine translation, ACM Transactions on Asian and Low-Resource Language Information Processing 14 (3) (2015) 13.

[17] J. Daiber, M. Stanojevic, W. Aziz, K. Sima'an, Examining the relationship between preordering and word order freedom in machine translation, in: Proceedings of the First Conference on Machine Translation (WMT16), Berlin, Germany, August. Association for Computational Linguistics, 2016.

[18] C. Ding, K. Sakanushi, H. Touji, M. Yamamoto, Inter-, intra-, and extra-chunk pre-ordering for statistical japanese-to-english machine translation, ACM Trans. Asian Low-Resour. Lang. Inf. Process. 15 (3) (2016) 20:1–20:28. doi:10.1145/2818381.
URL http://doi.acm.org/10.1145/2818381

[19] C. Hadiwinoto, Y. Liu, H. T. Ng, To swap or not to swap? exploiting dependency word pairs for reordering in statistical machine translation, in: Thirtieth AAAI Conference on Artificial Intelligence, 2016.

[20] B. M. de Marneffe, C. D.Manning, Generating typed dependency parses from phrase structure parses, in: In the Proceeding of the 5th International Conference on Language Resources and Evaluation, 2006.

[21] T. L. Nguyen, M. L. Ha, V. H. Nguyen, T. M. H. Nguyen, P. Le-Hong, Building a treebank for vietnamese dependency parsing, in: 2013 IEEE RIVF International Conference on Computing and Communication Technologies, Research, Innovation, and Vision for the Future, RIVF 2013, Hanoi, Vietnam, November 10-13, 2013, 2013, pp. 147–151.

[22] L. Wang, Support Vector Machines: theory and applications, Vol. 177, Springer Science & Business Media, 2005.

[23] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, The weka data mining software: An update, SIGKDD Explor. Newsl. 11 (1) (2009) 10–18.

[24] D. Cer, M.-C. de Marneffe, D. Jurafsky, C. D. Manning, Parsing to stanford dependencies: Trade-offs between speed and accuracy, in: 7th International Conference on Language Resources and Evaluation (LREC 2010), 2010.

[25] H. V. Huy, T.-L. N. Phuong-Thai Nguyen, M. Nguyen, Boostrapping phrase – based statistical machine translation via wsd integration, in: In Proceeding of the Sixth International Joint Conference on Natural Language Processing (IJCNLP 2013), 2013, pp. 1042–1046.

[26] A. Stolcke, Srilm - an extensible language modeling toolkit, in: Proceedings of International Conference on Spoken Language Processing, Vol. 29, 2002, pp. 901–904.

[27] V. H. Tran, V. V. Nguyen, M. L. Nguyen, Improving english-vietnamese statistical machine translation using preprocessing dependency syntactic, In Proceedings of the 2015 Conference of the Pacific Association for Computational Linguistics (Pacling 2015) 115–121.

[28] V. H. Tran, H. T. Vu, V. V. Nguyen, M. L. Nguyen, A classifier-based preordering approach for english-vietnamese statistical machine translation, 17th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2016).