Graph-Based Translation Via Graph Segmentation

Liangyou Li and **Andy Way** and **Qun Liu**
ADAPT Centre, School of Computing
Dublin City University, Ireland
{liangyouli,away,qliu}@computing.dcu.ie

Abstract

One major drawback of phrase-based translation is that it segments an input sentence into continuous phrases. To support linguistically informed source discontinuity, in this paper we construct graphs which combine bigram and dependency relations and propose a graph-based translation model. The model segments an input graph into connected subgraphs, each of which may cover a discontinuous phrase. We use beam search to combine translations of each subgraph left-to-right to produce a complete translation. Experiments on Chinese–English and German–English tasks show that our system is significantly better than the phrase-based model by up to +1.5/+0.5 BLEU scores. By explicitly modeling the graph segmentation, our system obtains further improvement, especially on German–English.

1 Introduction

Statistical machine translation (SMT) starts from sequence-based models. The well-known phrase-based (PB) translation model (Koehn et al., 2003) has significantly advanced the progress of SMT by extending translation units from single words to phrases. By using phrases, PB models can capture local phenomena, such as word order, word deletion, and word insertion. However, one of the significant weaknesses in conventional PB models is that only continuous phrases are used, so generalizations such as French *ne . . . pas* to English *not* cannot be learned. To solve this, syntax-based models (Galley et al., 2004; Chiang, 2005; Liu et al., 2006; Marcu et al., 2006) take tree structures into consideration to learn translation patterns by using non-terminals for generalization. However, the expressiveness of these models is confined by hierarchical constraints of the grammars used (Galley and Manning, 2010) since these patterns still cover continuous spans of an input sentence.

Model	C	D	S
(Koehn et al., 2003)	•		sequence
(Galley and Manning, 2010)	•	•	sequence
(Quirk et al., 2005) and (Menezes and Quirk, 2005)		•	tree
This work	•	•	graph

Table 1: Comparison between our work and previous work in terms of three aspects: keeping continuous phrases (C), allowing discontinuous phrases (D), and input structures (S).

By contrast, Quirk et al. (2005), Menezes and Quirk (2005) and Xiong et al. (2007) take treelets from dependency trees as the basic translation units. These treelets are connected and may cover discontinuous phrases. However, their models lack the ability to handle continuous phrases which are not connected in trees but could in fact be extremely important to system performance (Koehn et al., 2003). Galley and Manning (2010) directly extract discontinuous phrases from input sequences. However, without imposing additional restrictions on discontinuity, the amount of extracted rules can be very large and unreliable.

Different from previous work (as shown in Table 1), in this paper we use graphs as input structures and propose a graph-based translation model to translate a graph into a target string. The basic translation unit in this model is a connected subgraph which may cover discontinuous phrases. The main contributions of this work are summarized as follows:

- We propose to use a graph structure to combine a sequence and a tree (Section 3.1). The

Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 97–107,
Berlin, Germany, August 7-12, 2016. ©2016 Association for Computational Linguistics

graph contains both local relations between words from the sequence and long-distance relations from the tree.

- We present a translation model to translate a graph (Section 3). The model segments the graph into subgraphs and uses beam search to generate a complete translation from left to right by combining translation options of each subgraph.
- We present a set of sparse features to explicitly model the graph segmentation (Section 4). These features are based on edges in the input graph, each of which is either inside a subgraph or connects the subgraph with a previous subgraph.
- Experiments (Section 5) on Chinese–English and German–English tasks show that our model is significantly better than the PB model. After incorporating the segmentation model, our system achieves still further improvement.

2 Review: Phrase-based Translation

We first review the basic PB translation approach, which will be extended to our graph-based translation model. Given a pair of sentences $\langle S, T\rangle$, the conventional PB model is defined as Equation (1):

$$p(\bar{t}_1^I \mid \bar{s}_1^I) = \prod_{i=1}^{I} p(\bar{t}_i|\bar{s}_{a_i}) d(\bar{s}_{a_i}, \bar{s}_{a_{i-1}}) \quad (1)$$

The target sentence T is broken into I phrases $\bar{t}_1 \cdots \bar{t}_I$, each of which is a translation of a source phrase $\bar{s}_{a_i}$. d is a distance-based reordering model. Note that in the basic PB model, the phrase segmentation is not explicitly modeled which means that different segmentations are treated equally (Koehn, 2010).

The performance of PB translation relies on the quality of phrase pairs in a translation table. Conventionally, a phrase pair $\langle \bar{s}, \bar{t}\rangle$ has two properties: (i) $\bar{s}$ and $\bar{t}$ are continuous phrases. (ii) $\langle \bar{s}, \bar{t}\rangle$ is consistent with a word alignment A (Och and Ney, 2004): $\forall (i,j) \in A, s_i \in \bar{s} \Leftrightarrow t_j \in \bar{t}$ and $\exists s_i \in \bar{s}, t_j \in \bar{t}, (i,j) \in A$.

PB decoders generate hypotheses (partial translations) from left to right. Each hypothesis maintains a *coverage vector* to indicate which source words have been translated so far. A hypothesis can be extended on the right by translating an uncovered source phrase. The translation process ends when all source words have been translated.

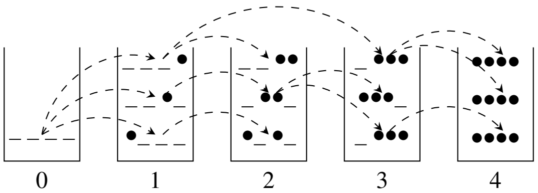


Figure 1: Beam search for phrase-based MT. • denotes a covered source position while _ indicates an uncovered position (Liu and Huang, 2014).

Beam search (as in Figure 1) is taken as an approximate search strategy to reduce the size of the decoding space. Hypotheses which cover the same number of source words are grouped in a stack. Hypotheses can be pruned according to their partial translation cost and an estimated future cost.

3 Graph-Based Translation

Our graph-based translation model extends PB translation by translating an input graph rather than a sequence to a target string. The graph is segmented into a sequence of connected subgraphs, each of which corresponds to a target phrase, as in Equation (2):

$$\begin{aligned} & p(\bar{t}_1^I \mid G(\tilde{s}_1^I)) \\ & = \prod_{i=1}^{I} p(\bar{t}_i|G(\tilde{s}_{a_i})) d(G(\tilde{s}_{a_i}), G(\tilde{s}_{a_{i-1}})) \\ & \approx \prod_{i=1}^{I} p(\bar{t}_i|G(\tilde{s}_{a_i})) d(\tilde{s}_{a_i}, \tilde{s}_{a_{i-1}}) \end{aligned} \quad (2)$$

where $G(\tilde{s}_i)$ denotes a connected source subgraph which covers a (discontinuous) phrase $\tilde{s}_i$.

3.1 Building Graphs

As a more powerful and natural structure for sentence modeling, a graph can model various kinds of word-relations together in a unified representation. In this paper, we use graphs to combine two commonly used relations: bigram relations and dependency relations. Figure 2 shows an example of a graph. Each edge in the graph denotes either a dependency relation or a bigram relation. Note that the graph we use in this paper is directed, connected, node-labeled and may contain cycles.

Bigram relations are implied in sequences and provide local and sequential information on pairs

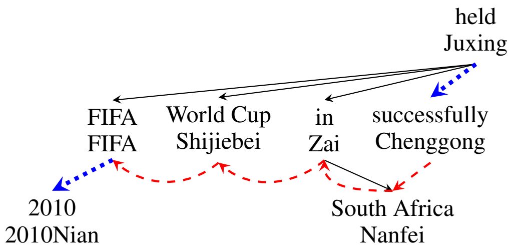


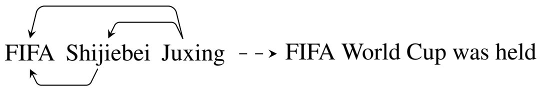
Figure 2: An example graph for a Chinese sentence. Each node includes a Chinese word and its English meaning. Dashed red lines are bigram relations. Solid lines are dependency relations. Dotted blue lines are shared by bigram and dependency relations.

of continuous words. Phrases connected by bigram relations (i.e. continuous phrases) are known to be useful to improve phrase coverage (Hanneman and Lavie, 2009). By contrast, dependency relations come from dependency structures which model syntactic and semantic relations between words. Phrases whose words are connected by dependency relations (also known as treelets) are linguistic-motivated and thus more reliable (Quirk et al., 2005).

By combining these two relations together in graphs, we can make use of both continuous and linguistic-informed discontinuous phrases as long as they are connected subgraphs.

3.2 Training

Different from PB translation, the basic translation units in our model are subgraphs. Thus, during training, we extract subgraph–phrase pairs instead of phrase pairs on parallel graph–string sentences associated with word alignments.[1] An example of a translation rule is as follows:



Note that the source side of a rule in our model is a graph which can be used to cover either a continuous phrase or a discontinuous phrase according to its match in an input graph during decoding.

The algorithm for extracting translation rules is shown in Algorithm 1. This algorithm traverses each phrase pair $\langle \tilde{s}, \bar{t} \rangle$, which is within a length limit and consistent with a given word alignment (lines 1–2), and outputs $\langle G(\tilde{s}), \bar{t} \rangle$ if $\tilde{s}$ is covered by a connected subgraph $G(\tilde{s})$ (lines 6–8). A source phrase can be extended with unaligned source words which are adjacent to the phrase (lines 9–14). We use a queue Q to store all phrases which are consistently aligned to the same target phrase (line 3).

Algorithm 1: Algorithm for extracting translation rules from a graph-string pair.

```
Data: A word-aligned graph–string pair
      $(G(S), T, A)$
Result: A set of translation pairs $R$
1  for each phrase $\bar{t}$ in $T$: $| \bar{t} | \leq L$ do
2      find the minimal (may be discontinuous)
       phrase $\tilde{s}$ in $S$ so that $| \tilde{s} | \leq L$ and $\langle \tilde{s}, \bar{t} \rangle$ is
       consistent with $A$ ;
3      Queue $Q = \{\tilde{s}\}$;
4      while $Q$ is not empty do
5          pop an element $\tilde{s}$ off;
6          if $G(\tilde{s})$ is connected then
7              add $\langle G(\tilde{s}), t \rangle$ to $R$;
8          end
9          if $| \tilde{s} | < L$ then
10             for each unaligned word $s_i$
               adjacent to $\tilde{s}$ do
11                 $\tilde{s}'$ = extend $\tilde{s}$ with $s_i$;
12                 add $\tilde{s}'$ to $Q$;
13             end
14         end
15     end
16 end
```

3.3 Model and Decoding

We define our model in the log-linear framework (Och and Ney, 2002) over a derivation $D = r_1 r_2 \cdots r_N$, as in Equation (3):

$$p(D) \propto \prod_i \phi_i(D)^{\lambda_i} \tag{3}$$

where r_i are translation rules, ϕ_i are features defined on derivations and λ_i are feature weights. In our experiments, we use the standard 9 features: two translation probabilities $p(G(s)|t)$ and $p(t|G(s))$, two lexical translation probabilities $p_{lex}(s|t)$ and $p_{lex}(t|s)$, a language model $lm(t)$ over a translation t, a rule penalty, a word penalty, an unknown word penalty and a distortion feature d for distance-based reordering.

The calculation of the distortion feature d in our

[1] Different from translation rules in conventional syntax-based MT, rules in our model are not learned based on synchronous grammars and so non-terminals are disallowed.

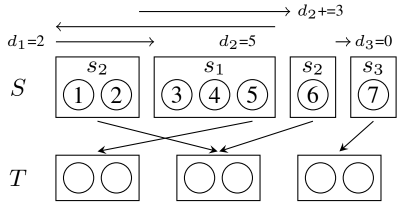


Figure 3: Distortion calculation for both continuous and discontinuous phrases in a derivation.

.

model is different from the one used in conventional PB models, as we need to take discontinuity into consideration. In this paper, we use a distortion function defined in Galley and Manning (2010) to penalize discontinuous phrases that have relatively long gaps. Figure 3 shows an example of calculating distortion for discontinuous phrases.

Our graph-based decoder is very similar to the PB decoder except that, in our decoder, each hypothesis is extended by translating an uncovered subgraph instead of a phrase. Positions covered by the subgraph are then marked as translated.

4 Graph Segmentation Model

Each derivation in our graph-based translation model implies a sequence of subgraphs (also called a segmentation). By default, similar to PB translation, our model treats each segmentation equally as shown in Equation (2). However, previous work on PB translation has suggested that such segmentations provide useful information which can improve translation performance. For example, boundary information in a phrase segmentation can be used for reordering models (Xiong et al., 2006; Cherry, 2013).

In this paper, we are interested in directly modeling the segmentation using information from graphs. By making the assumption that each subgraph is only dependent on previous subgraphs, we define a generative process over a graph segmentation as in Equation (4):

$$\begin{aligned} & p(G(\tilde{s}_1) \cdots G(\tilde{s}_I)) \\ & = \prod_{i=1}^{I} P(G(\tilde{s}_i) | G(\tilde{s}_1) \cdots G(\tilde{s}_{i-1})) \end{aligned} \tag{4}$$

Instead of training a stand-alone discriminative segmentation model to assign each subgraph a probability given previous subgraphs, we implement the model via sparse features, each of which is extracted at run-time during decoding and then directly added to the log-linear framework, so that these features can be tuned jointly with other features (of Section 3.3) to directly maximize the translation quality.

ZH–EN	#Sents	DE–EN	#Sents
Train	1.5M+	Train	2M+
MT02 (Dev)	878	WMT11 (Dev)	3,003
MT04	1,597	WMT12	3,003
MT05	1,082	WMT13	3,000

Table 2: The number of sentences in our corpora.

Since a segmentation is obtained by breaking up the connectivity of an input graph, it is intuitive to use edges to model the segmentation. According to Equation (4), for a current subgraph G_i, we only consider those edges which are either inside G_i or connect G_i with a previous subgraph. Based on these edges, we extract sparse features for each node in the subgraph. The set of sparse features is defined as follows:

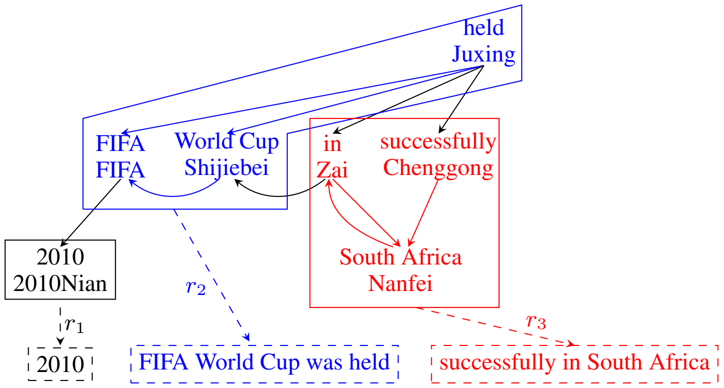
$$\begin{Bmatrix} n.w \\ n.c \end{Bmatrix} \times \begin{Bmatrix} n'.w \\ n'.c \end{Bmatrix} \times \begin{Bmatrix} C \\ P \\ H \end{Bmatrix} \times \begin{Bmatrix} in \\ out \end{Bmatrix}$$

where $n.w$ and $n.c$ are the word and class of the current node n, and $n'.w$ and $n'.c$ are the word and class of a node n' connected to n. C, P, and H denote that the node n' is in the current subgraph G_i or the adjacent previous subgraph G_{i-1} or other previous subgraphs, respectively. Note that we treat the adjacent previous subgraph differently from others since information from the last previous unit is quite useful (Xiong et al., 2006; Cherry, 2013). *in* and *out* denote that the edge is an incoming edge or outgoing edge for the current node n. Figure 4 shows an example of extracting sparse features for a subgraph.

Inspired by success in using sparse features in SMT (Cherry, 2013), in this paper we lexicalize only on the top-100 most frequent words. In addition, we group source words into 50 classes by using *mkcls* which should provide useful generalization (Cherry, 2013) for our model.

5 Experiment

We conduct experiments on Chinese–English (ZH–EN) and German–English (DE–EN) translation tasks. Table 2 provides a summary of our corpora. Our ZH–EN training corpus contains 1.5M+ sentences from LDC. NIST 2002 (MT02) is taken as a development set to tune weights, and NIST



Sparse features for r_3:

W:Zai_W:Nanfei_C_in	C:4_W:Nanfei_C_in	W:Zai_C:5_C_in	C:4_C:5_C_in
W:Zai_W:Nanfei_C_out	C:4_W:Nanfei_C_out	W:Zai_C:5_C_out	C:4_C:5_C_out
W:Zai_W:Shijiebei_P_out	C:4_W:Shijiebei_P_out	W:Zai_C:3_P_out	C:4_C:3_P_out
W:Zai_W:Juxing_P_in	C:4_W:Juxing_P_in	W:Zai_C:7_P_in	C:4_C:7_P_in
W:Nanfei_W:Zai_C_in	C:5_W:Zai_C_in	W:Nanfei_C:4_C_in	C:5_C:4_C_in
W:Nanfei_W:Zai_C_out	C:5_W:Zai_C_out	W:Nanfei_C:4_C_out	C:5_C:4_C_out
W:Nanfei_W:Chenggong_C_in	C:5_W:Chenggong_C_in	W:Nanfei_C:6_C_in	C:5_C:6_C_in
W:Chenggong_W:Nanfei_C_out	C:6_W:Nanfei_C_out	W:Chenggong_C:5_C_out	C:6_C:5_C_out
W:Chenggong_W:Juxing_P_in	C:6_W:Juxing_P_in	W:Chenggong_C:7_P_in	C:6_C:7_P_in

Figure 4: An illustration of extracting sparse features for each node in a subgraph during decoding. The decoder segments the graph in Figure 2 into three subgraphs (solid rectangles) and produces a complete translation by combining translations of each subgraph (dashed rectangles). In this figure, the class of a word is randomly assigned.

2004 (MT04) and NIST 2005 (MT05) are two test sets used to evaluate the systems. The Stanford Chinese word segmenter (Chang et al., 2008) is used to segment Chinese sentences. The Stanford dependency parser (Chang et al., 2009) parses a Chinese sentence into a projective dependency tree which is then converted to a graph by adding bigram relations.

The DE–EN training corpus is from WMT 2014, including Europarl V7 and News Commentary. News-Test 2011 (WMT11) is taken as a development set while News-Test 2012 (WMT12) and News-Test 2013 (WMT13) are test sets. We use mate-tools[2] to perform morphological analysis and parse German sentences (Bohnet, 2010). Then, MaltParser[3] converts a parse result into a projective dependency tree (Nivre and Nilsson, 2005).

5.1 Settings

In this paper, we mainly report results from five systems under the same configuration. **PBMT** is built by the PB model in Moses (Koehn et al., 2007). **Treelet** extends PBMT by taking treelets as the basic translation units (Quirk et al., 2005; Menezes and Quirk, 2005). We implement a Treelet model in Moses which produces translations from left to right and uses beam search for decoding. **DTU** extends the PB model by allowing discontinuous phrases (Galley and Manning, 2010). We implement DTU with source discontinuity in Moses.[4] **GBMT** is our basic graph-based translation system while **GSM** adds the graph segmentation model into GBMT. Both systems are implemented in Moses.

Word alignment is performed by GIZA++ (Och and Ney, 2003) with the heuristic function *grow-diag-final-and*. We use SRILM (Stolcke, 2002) to train a 5-gram language model on the Xinhua portion of the English Gigaword corpus 5th edition with modified Kneser-Ney discounting (Chen and Goodman, 1996). Batch MIRA (Cherry and Foster, 2012) is used to tune weights. BLEU (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2011), and TER (Snover et al., 2006) are used for evaluation.

[2] http://code.google.com/p/mate-tools/

[3] http://www.maltparser.org/

[4] The re-implementation of DTU in Moses makes it easier to meaningfully compare systems under the same settings.

Metric	System	ZH–EN		DE–EN	
		MT04	MT05	WMT12	WMT13
BLEU ↑	PBMT	33.2	31.8	19.5	21.9
	Treelet	33.8*	31.7	19.6	22.1*
	DTU	**34.5***	32.3*	**19.8***	**22.3***
	GBMT	**34.7***	**32.4***	**19.8***	**22.4***
	GSM	**34.9**$^{*+}$	**32.7**$^{*+}$	**20.3**$^{*+}$	**22.9**$^{*+}$
METEOR ↑	PBMT	32.1	32.3	28.0	29.2
	Treelet	31.9	31.8	28.0	29.1
	DTU	**32.3***	**32.4**	**28.2***	**29.5***
	GBMT	**32.4**$^{*+}$	**32.5***	**28.2***	**29.4***
	GSM	**32.7**$^{*+}$	**32.6**$^{*+}$	**28.5**$^{*+}$	**29.8**$^{*+}$
TER ↓	PBMT	60.6	61.6	63.7	60.2
	Treelet	60.1*	61.4	63.2*	59.6*
	DTU	60.0*	61.5	63.5*	59.8*
	GBMT	59.8^{*+}	61.3*	63.5*	59.8*
	GSM	60.5	62.1	63.1^{*+}	**59.3**$^{*+}$

Table 3: Metric scores for all systems on Chinese–English (ZH–EN) and German–English (DE–EN). Each score is an average over three MIRA runs (Clark et al., 2011). $*$ means a system is significantly better than PBMT at $p \leq 0.01$. Bold figures mean a system is significantly better than Treelet at $p \leq 0.01$. $+$ means a system is significantly better than DTU at $p \leq 0.01$. In this table, we mark a system by comparing it with previous ones.

5.2 Results and Discussion

Table 3 shows our evaluation results. We find that our GBMT system is significantly better than PBMT as measured by all three metrics across all test sets. Specifically, the improvements are up to +1.5/+0.5 BLEU, +0.3/+0.2 METEOR, and -0.8/-0.4 TER on ZH–EN and DE–EN, respectively. This improvement is reasonable as our system allows discontinuous phrases which can reduce data sparsity and handle long-distance relations (Galley and Manning, 2010). Another argument for discontinuous phrases is that they allow the decoder to use larger translation units which tend to produce better translations (Galley and Manning, 2010). However, this argument was only verified on ZH–EN. Therefore, we are interested in seeing whether we have the same observation in our experiments on both language pairs.

We count the used translation rules in MT02 and WMT11 based on different target lengths. The results are shown in Figure 5. We find that both DTU and GBMT indeed tend to use larger translation units on ZH–EN. However, more smaller translation units are used on DE–EN.[5] We presume this is because long-distance reordering is performed more often on ZH–EN than on DE–EN. Based on the fact that the distortion function d measures the reordering distance, we find that the average distortion value in PB on ZH–EN MT02 is 18.4 and 3.5 on DE–EN WMT11. Our observations suggest that the argument that discontinuous phrases allow decoders to use larger translation units should be considered with caution when we explain the benefit of discontinuity on different language pairs.

System	# Rules	
	ZH–EN	DE–EN
DTU	224M+	352M+
GBMT	99M+	153M+

Table 4: The number of rules in DTU and GBMT.

Compared to PBMT, the Treelet system does not show consistent improvements. Our system achieves significantly better BLEU and METEOR scores than Treelet on both ZH–EN and DE–EN, and a better TER score on DE–EN. This suggests that continuous phrases are essential for system robustness since it helps to improve phrase coverage (Hanneman and Lavie, 2009). Lower phrase coverage in Treelet results in more short phrases being used, as shown in Figure 5. In addition, we find that both DTU and our systems do not achieve consistent improvements over Treelet in terms of TER. We observed that both DTU and our systems tend to produce longer translations than Treelet, which might cause unreliable TER evaluation in our experiments as TER favours shorter sentences (He and Way, 2010).

Since discontinuous phrases produced by using syntactic information are fewer in number but

[5] We have the same finding on all test sets.

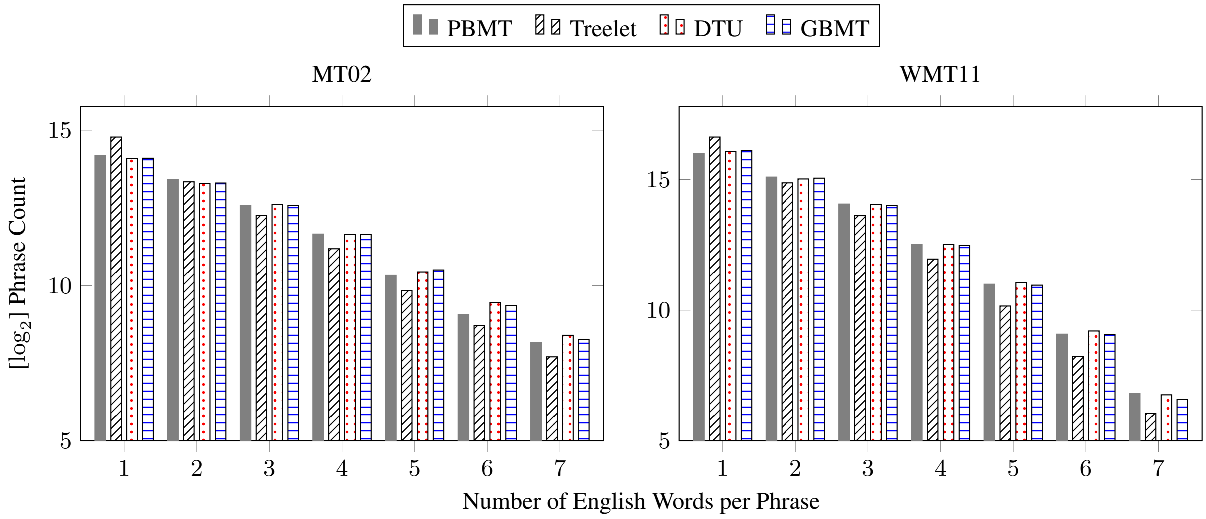


Figure 5: Phrase Length Histogram for MT02 and WMT11.

more reliable (Koehn et al., 2003), our GBMT system achieves comparable performance with DTU but uses significantly fewer rules, as shown in Table 4. After integrating the graph segmentation model to help subgraph selection, GBMT is further improved and the resulted system G2S has significantly better evaluation scores than DTU on both language pairs. However, our segmentation model is more helpful on DE–EN than ZH–EN. We find that the number of features learned on ZH–EN (25K+) is much less than on DE–EN (49K+). This may result in a lower feature coverage during decoding. The lower number of features in ZH–EN could be caused by the fact that the development set MT02 has many fewer sentences than WMT11. Accordingly, we suggest to use a larger development set during tuning to achieve better translation performance when the segmentation model is integrated.

Our current model is more akin to addressing problems in phrase-based and treelet-based models by segmenting graphs into pieces rather than extracting a recursive grammar. Therefore, similar to those models, our model is weak at phrase reordering as well. However, we are interesting in the potential power of our model by incorporating lexical reordering (LR) models and comparing it with syntax-based models.

Table 5 shows BLEU scores of the hierarchical phrase-based (HPB) system (Chiang, 2005) in Moses[6] and GBMT combined with a word-based LR model (Koehn et al., 2005). We find that the LR model significantly improves our system. GBMT+LR is comparable with the Moses HPB model on Chinese–English and better than HPB on German–English.

System	ZH–EN		DE–EN	
	MT04	MT05	WMT12	WMT13
GBMT+LR	36.0	33.9	20.6	23.6
HPB	36.1	34.1	20.3	22.8

Table 5: BLEU scores of a Moses hierarchical phrase-based system (HPB) and our system (GBMT) with a word-based lexical reordering model (LR).

5.3 Examples

Figure 6 shows three examples from MT04 to better explain the differences of each system. Example 1 shows that systems which allow discontinuous phrases (namely Treelet, DTU, GBMT, and GSM) successfully translate a Chinese collocation "*Yu ... Wuguan*" to "*have nothing to do with*" while PBMT fails to catch the generalization since it only allows continuous phrases.

In Example 2, Treelet translates a discontinuous phrase "*Dui ... Zuofa*" (to ... practice) only as "*to*" where an important target word "*practice*" is dropped. By contrast, bigram relations allow our systems (GBMT and GSM) to find a better phrase to translate: "*De Zuofa*" to "*of practice*". In addition, DTU translates a discontinuous phrase "*De Zuofa ... Buman*" to "*dissatisfaction with the approach of*". However, the phrase is actually not

[6] For a fairer comparison, we disallow target discontinuity in HPB rules. This means that a non-terminal on the target side is either the first symbol or the last symbol.

Example 1

PBMT: the united states has indicated that the united states and north korea delegation has visited

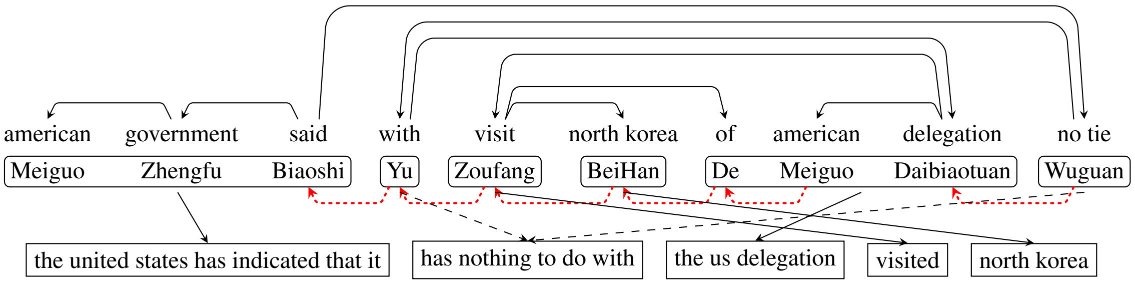
Treelet: the united states has indicated that it has nothing to do with the us delegation visited the north korea

DTU: the united states has indicated that it has nothing to do with the us delegation visited north korea

GBMT: the united states has indicated that it has nothing to do with the us delegation visited north korea

GSM: the united states has indicated that it has nothing to do with the us delegation visited north korea

REF: the american government said that it has nothing to do with the american delegation to visit north korea



Example 2

PBMT: the united states government to brazil has repeatedly expressed its dissatisfaction .

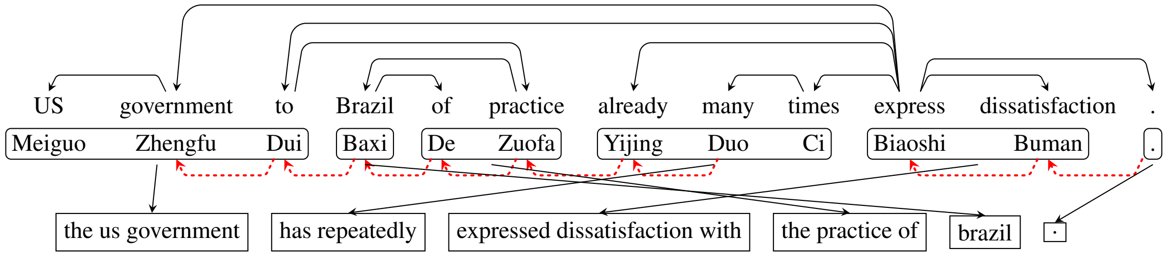
Treelet: the government of brazil to the united states has on many occasions expressed their discontent .

DTU: the united states has repeatedly expressed its dissatisfaction with the approach of the government to brazil .

GBMT: the us government has repeatedly expressed dissatisfaction with the practice of brazil .

GSM: the us government has repeatedly expressed dissatisfaction with the practice of brazil .

REF: the us government has expressed their resentment against this practice of brazil on many occasions .



Example 3

PBMT: the government and all sectors of society should continue to explore in depth and draw on collective wisdom .

Treelet: the government must continue to make in-depth discussions with various sectors of the community and the collective wisdom .

DTU: the government must continue to work together with various sectors of the community to make an in-depth study and draw on collective wisdom .

GBMT: the government must continue to work together with various sectors of the community in-depth study and draw on collective wisdom .

GSM: the government must continue to make in-depth discussions with various sectors of the community and draw on collective wisdom .

REF: the government must continue to hold thorough discussions with all walks of life to pool the wisdom of the masses .

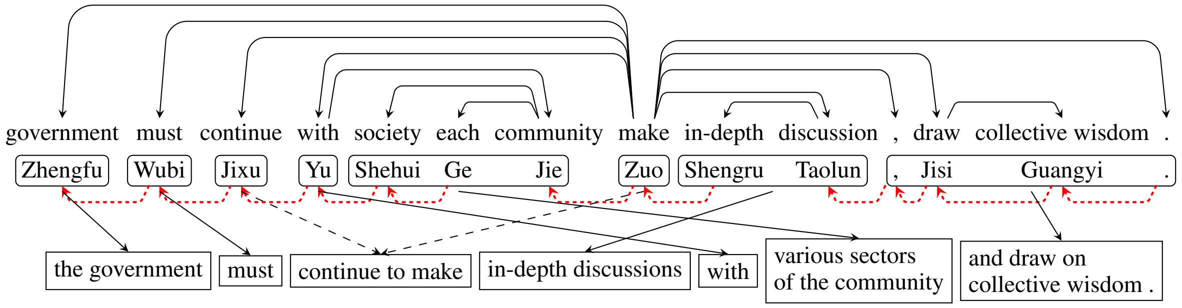


Figure 6: Translation examples from MT04 produced by different systems. Each source sentence is annotated by dependency relations and additional bigram relations (dotted red edges). We also annotate phrase alignments produced by our system GSM.

linguistically motivated and could be unreliable. By disallowing phrases which are not connected in the input graph, GBMT and GSM produce better translations.

Example 3 illustrates that our graph segmentation model helps to select better subgraphs. After obtaining a partial translation "*the government must*", GSM chooses to translate a subgraph which covers a discontinuous phrase "*Jixu* ... *Zuo*" to "*continue to make*" while GBMT translates "*Jixu Yu*" (continue ... with) to "*continue to work together with*". By selecting the proper subgraph to translate, GSM performs a better reordering on the translation.

6 Related Work

Starting from sequence-based models, SMT has been benefiting increasingly from complex structures.

Sequence-based MT: Since the breakthrough made by IBM on word-based models in the 1990s (Brown et al., 1993), SMT has developed rapidly. The PB model (Koehn et al., 2003) advanced the state-of-the-art by translating multi-word units, which makes it better able to capture local phenomena. However, a major drawback in PBMT is that only continuous phrases are considered. Galley and Manning (2010) extend PBMT by allowing discontinuity. However, without linguistic structure information such as syntax trees, sequence-based models can learn a large amount of phrases which may be unreliable.

Tree-based MT: Compared to sequences, trees provide recursive structures over sentences and can handle long-distance relations. Typically, trees used in SMT are either phrasal structures (Galley et al., 2004; Liu et al., 2006; Marcu et al., 2006) or dependency structures (Menezes and Quirk, 2005; Xiong et al., 2007; Xie et al., 2011; Li et al., 2014). However, conventional tree-based models only use linguistically well-formed phrases. Although they are more reliable in theory, discarding all phrase pairs which are not linguistically motivated is an overly harsh decision. Therefore, exploring more translation rules usually can significantly improve translation performance (Marcu et al., 2006; DeNeefe et al., 2007; Wang et al., 2007; Mi et al., 2008).

Graph-based MT: Compared to sequences and trees, graphs are more general and can represent more relations between words. In recent years, graphs have been drawing quite a lot of attention from researchers. Jones et al. (2012) propose a hypergraph-based translation model where hypergraphs are taken as a meaning representation of sentences. However, large corpora with annotated hypergraphs are not readily available for MT. Li et al. (2015) use an edge replacement grammar to translate dependency graphs which are converted from dependency trees by labeling edges. However, their model only focuses on subgraphs which cover continuous phrases.

7 Conclusion

In this paper, we extend the conventional phrase-based translation model by allowing discontinuous phrases. We use graphs which combine bigram and dependency relations together as inputs and present a graph-based translation model. Experiments on Chinese–English and German–English show our model to be significantly better than the phrase-based model as well as other more sophisticated models. In addition, we present a graph segmentation model to explicitly guide the selection of subgraphs. In experiments, this model further improves our system.

In the future, we will extend this model to allow discontinuity on target sides and explore the possibility of directly encoding reordering information in translation rules. We are also interested in using graphs for neural machine translation to see how it can translate and benefit from graphs.

Acknowledgments

This research has received funding from the People Programme (Marie Curie Actions) of the European Union's Framework Programme (FP7/2007-2013) under REA grant agreement n^{o} 317471 and the European Union's Horizon 2020 research and innovation programme under grant agreement n^{o} 645452 (QT21). The ADAPT Centre for Digital Content Technology is funded under the SFI Research Centres Programme (Grant 13/RC/2106) and is co-funded under the European Regional Development Fund. The authors thank all anonymous reviewers for their insightful comments and suggestions.

References

Bernd Bohnet. 2010. Very High Accuracy and Fast Dependency Parsing is Not a Contradiction. In *Proceedings of the 23rd International Conference on*

Computational Linguistics, pages 89–97, Beijing, China, August.

Peter F Brown, Vincent J Della Pietra, Stephen A Della Pietra, and Robert L Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311.

Pi-Chuan Chang, Michel Galley, and Christopher D. Manning. 2008. Optimizing Chinese Word Segmentation for Machine Translation Performance. In *Proceedings of the Third Workshop on Statistical Machine Translation*, pages 224–232, Columbus, Ohio, June.

Pi-Chuan Chang, Huihsin Tseng, Dan Jurafsky, and Christopher D. Manning. 2009. Discriminative Reordering with Chinese Grammatical Relations Features. In *Proceedings of the Third Workshop on Syntax and Structure in Statistical Translation*, pages 51–59, Boulder, Colorado, June.

Stanley F. Chen and Joshua Goodman. 1996. An Empirical Study of Smoothing Techniques for Language Modeling. In *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics*, ACL '96, pages 310–318, Santa Cruz, California, June.

Colin Cherry and George Foster. 2012. Batch Tuning Strategies for Statistical Machine Translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 427–436, Montreal, Canada, June.

Colin Cherry. 2013. Improved Reordering for Phrase-Based Translation using Sparse Features. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 22–31, Atlanta, Georgia, June.

David Chiang. 2005. A Hierarchical Phrase-based Model for Statistical Machine Translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 263–270, Ann Arbor, Michigan, June.

Jonathan H. Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better Hypothesis Testing for Statistical Machine Translation: Controlling for Optimizer Instability. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, pages 176–181, Portland, Oregon, June.

Steve DeNeefe, Kevin Knight, Wei Wang, and Daniel Marcu. 2007. What Can Syntax-Based MT Learn from Phrase-Based MT? In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 755–763, Prague, Czech Republic, June.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 85–91, Edinburgh, Scotland, July.

Michel Galley and Christopher D. Manning. 2010. Accurate Non-hierarchical Phrase-Based Translation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 966–974, Los Angeles, California, June.

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What's in a Translation Rule? In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, page 273280, Boston, Massachusetts, USA, May.

Greg Hanneman and Alon Lavie. 2009. Decoding with Syntactic and Non-syntactic Phrases in a Syntax-based Machine Translation System. In *Proceedings of the Third Workshop on Syntax and Structure in Statistical Translation*, pages 1–9, Boulder, Colorado, June.

Yifan He and Andy Way. 2010. Metric and reference factors in minimum error rate training. *Machine Translation*, 24(1):27–38.

Bevan Jones, Jacob Andreas, Daniel Bauer, Karl Moritz Hermann, and Kevin Knight. 2012. Semantics-Based Machine Translation with Hyperedge Replacement Grammars. In *COLING 2012, 24th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers*, pages 1359–1376, Mumbai, India, December.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, pages 48–54, Edmonton, Canada, July.

Philipp Koehn, Amittai Axelrod, Alexandra Birch, Chris Callison-Burch, Miles Osborne, and David Talbot. 2005. Edinburgh System Description for the 2005 IWSLT Speech Translation Evaluation. In *Proceedings of the International Workshop on Spoken Language Translation 2005*, pages 68–75, Pittsburgh, PA, USA, October.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration*

Sessions, pages 177–180, Prague, Czech Republic, June.

Philipp Koehn. 2010. *Statistical Machine Translation*. Cambridge University Press, New York, NY, USA, 1st edition.

Liangyou Li, Jun Xie, Andy Way, and Qun Liu. 2014. Transformation and Decomposition for Efficiently Implementing and Improving Dependency-to-String Model In Moses. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, October.

Liangyou Li, Andy Way, and Qun Liu. 2015. Dependency Graph-to-String Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, September.

Lemao Liu and Liang Huang. 2014. Search-Aware Tuning for Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1942–1952, Doha, Qatar, October.

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string Alignment Template for Statistical Machine Translation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 609–616, Sydney, Australia, July.

Daniel Marcu, Wei Wang, Abdessamad Echihabi, and Kevin Knight. 2006. SPMT: Statistical Machine Translation with Syntactified Target Language Phrases. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 44–52, Sydney, Australia.

Arul Menezes and Chris Quirk. 2005. Dependency Treelet Translation: The Convergence of Statistical and Example-Based Machine-translation? In *Proceedings of the Workshop on Example-based Machine Translation at MT Summit X*, September.

Haitao Mi, Liang Huang, and Qun Liu. 2008. Forest-Based Translation. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 192–199, Columbus, Ohio, USA, June.

Joakim Nivre and Jens Nilsson. 2005. Pseudo-Projective Dependency Parsing. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 99–106, Ann Arbor, Michigan, June.

Franz Josef Och and Hermann Ney. 2002. Discriminative Training and Maximum Entropy Models for Statistical Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 295–302, Philadelphia, PA, USA, July.

Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51, March.

Franz Josef Och and Hermann Ney. 2004. The Alignment Template Approach to Statistical Machine Translation. *Computational Linguistics*, 30(4):417–449, December.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, July.

Chris Quirk, Arul Menezes, and Colin Cherry. 2005. Dependency Treelet Translation: Syntactically Informed Phrasal SMT. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 271–279, Ann Arbor, Michigan, June.

M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of Association for Machine Translation in the Americas*, pages 223–231, Cambridge, Massachusetts, USA, August.

Andreas Stolcke. 2002. SRILM An Extensible Language Modeling Toolkit. In *Proceedings of the International Conference Spoken Language Processing*, pages 901–904, Denver, CO.

Wei Wang, Kevin Knight, and Daniel Marcu. 2007. Binarizing Syntax Trees to Improve Syntax-Based Machine Translation Accuracy. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 746–754, Prague, Czech Republic, June.

Jun Xie, Haitao Mi, and Qun Liu. 2011. A Novel Dependency-to-string Model for Statistical Machine Translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 216–226, Edinburgh, United Kingdom, July.

Deyi Xiong, Qun Liu, and Shouxun Lin. 2006. Maximum Entropy Based Phrase Reordering Model for Statistical Machine Translation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th Annual Meeting of the Association for Computational Linguistics*, pages 521–528, Sydney, Australia, July.

Deyi Xiong, Qun Liu, and Shouxun Lin. 2007. A Dependency Treelet String Correspondence Model for Statistical Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 40–47, Prague, June.