A Dependency Edge-based Transfer Model for Statistical Machine Translation

Hongshen Chen[†§] **Jun Xie**[†] **Fandong Meng**[†§] **Wenbin Jiang**[†] **Qun Liu**[‡†]
†Key Laboratory of Intelligent Information Processing
Institute of Computing Technology, Chinese Academy of Sciences
§University of Chinese Academy of Sciences
{chenhongshen,xiejun,mengfandong,jiangwenbin}@ict.ac.cn
‡CNGL, School of Computing, Dublin City University
qliu@computing.dcu.ie

Abstract

Previous models in syntax-based statistical machine translation usually resort to some kinds of synchronous procedures, few of these works are based on the analysis-transfer-generation methodology. In this paper, we present a statistical implementation of the analysis-transfer-generation methodology in rule-based translation. The procedures of syntax analysis, syntax transfer and language generation are modeled independently in order to break the synchronous constraint, resorting to dependency structures with dependency edges as atomic manipulating units. Large-scale experiments on Chinese to English translation show that our model exhibits state-of-the-art performance by significantly outperforming the phrase-based model. The statistical transfer-generation method results in significantly better performance with much smaller models.

1 Introduction

Researches in statistical machine translation have been flourishing in recent years. Statistical translation methods can be divided into word-based (Brown et al., 1993), phrase-based (Marcu and Wong, 2002; Koehn et al., 2003) and syntax-based models (Yamada and Knight, 2001; Graehl and Knight, 2004; Chiang, 2005; Liu et al., 2006; Mi et al., 2008; Huang et al., 2006; Lin, 2004; Ding and Palmer, 2004; Quirk et al., 2005; Shen et al., 2008; Xie et al., 2011; Meng et al., 2013). Compared with word-based and phrase-based methods, syntax-based models perform better in long distance reordering and enjoy higher generalization capability by leveraging the hierarchical structures in natural languages, and achieve the state-of-the-art performance in these years.

Most syntax-based models (except for Lin (2004)) utilize some kinds of synchronous generation procedures which directly model the structural correspondence between two languages. In contrast, the analysis-transfer-generation methodology in rule-based translation solves the machine translation problem in a more divided scheme, where the processing procedures of analysis, structural transfer and language generation are modeled separately. The analysis-transfer-generation strategy can tolerate higher non-isomorphism between languages if with a more general transformation unit and it can facilitate elaborating engineering of each processing procedure, however, there isn't a statistical transfer model that shows the comparable performance with the current state-of-the-art SMT model so far.

In this paper, we propose a novel statistical analysis-transfer-generation model for machine translation, to integrate the advantages of the transfer-generation scheme and the statistical modeling. The procedures of transfer and generation are modeled on dependency structures with dependency edges as atomic manipulating units. First, the source sentence is parsed by a dependency parser. Then, the source dependency structure is transferred into a target structure by translation rules, which composed of the source and target edges. Last, the target sentence is finally generated from the target edges which are used as intermediate syntactic structures. By directly modeling the edge, the most basic unit in the dependency tree, which definitely describe the modifying relationship and positional relation between words, our model alleviates the non-isomorphic problem and shows the flexibility of reordering.

Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, pages 1103–1113, Dublin, Ireland, August 23-29 2014.

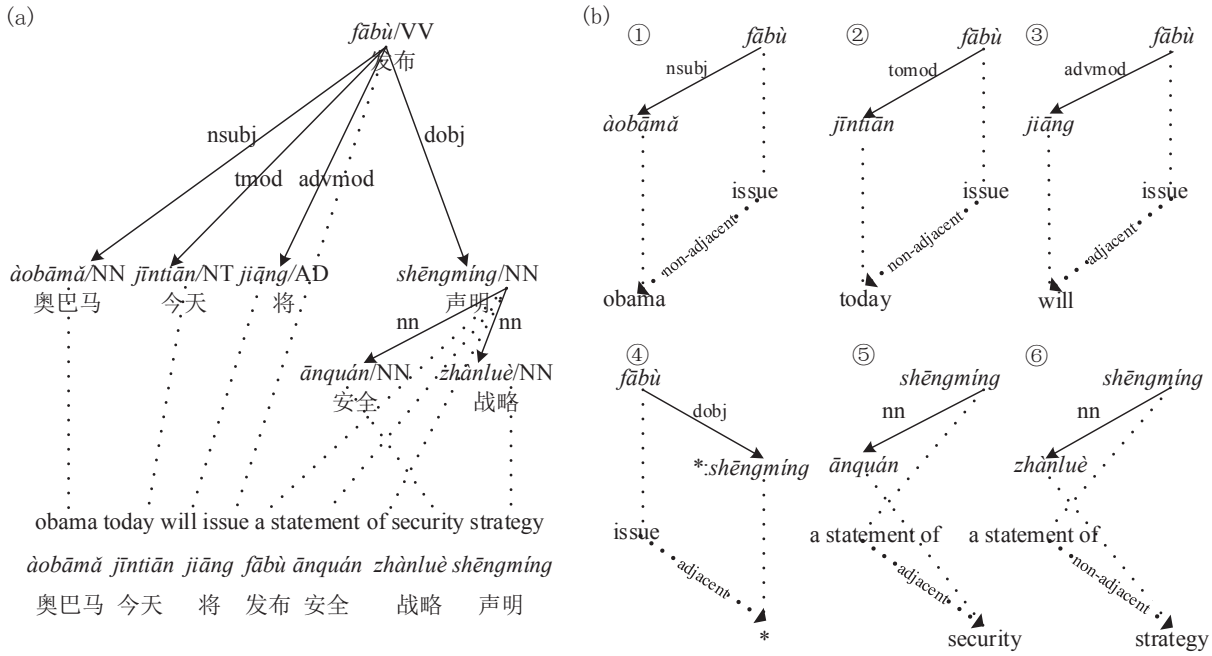

Figure 1: (a)An example of labeled Chinese dependency tree aligned with the corresponding English sentence. (b) Examples of the transfer rules extracted from the tree. "*" denotes a variable. All the inner nodes are treated as variables. The label on the target side of a rule denotes whether the head and the dependent are adjacent or not.

The rest of the paper is organized as follows, we first describe the dependency edge-based transfer model (Section 2). Then, we present our rule acquisition algorithm (Section 3), the decoding and target sentence generation process (Section 4). Finally, large-scale experiments (Section 5) on Chinese-to-English translation show that our edge-based transfer model gains state-of-the-art performance by significantly outperforming the phrase-based model (Koehn et al., 2003) by averaged +1.34 BLEU points on three test sets. To the best of our knowledge, this is the first transfer-generation-based statistical machine translation model that achieves the state-of-the-art performance.

2 Dependency Edge-based Transfer Model

2.1 Edges in Dependency Trees

Given a sentence, its dependency tree is a directed acyclic graph with words in the sentence as nodes. An example dependency tree is shown in Figure 1 (a). An edge in the tree represents a dependency relationship between a pair of words, a head and a dependent. When a nominal dependent acts as a subject and modifies a verbal head, they usually have a fixed relative position. In Figure 1 (a), "*àobāmǎ*" modifies "*fābù*". The grammatical relation label *nsubj* (Chang et al., 2009) between them denotes that a noun phrase acts as the subject of a clause. "*àobāmǎ*" is on the left of "*fābù*".

Based on the above observations, we take the edge as the elementary structure of a dependency tree and regard a dependency tree to be a set of edges.

Definition 1. *An source side* ***edge*** *is a 4-tuple $e = \langle H, D, P, R \rangle$, where H is the head, D is the dependent, P denotes the relative position between H and D, left or right, R is the grammatical relation label*
.

In Figure 1 (b), the upper sides of transfer rules are source side edges extracted from the dependency tree.

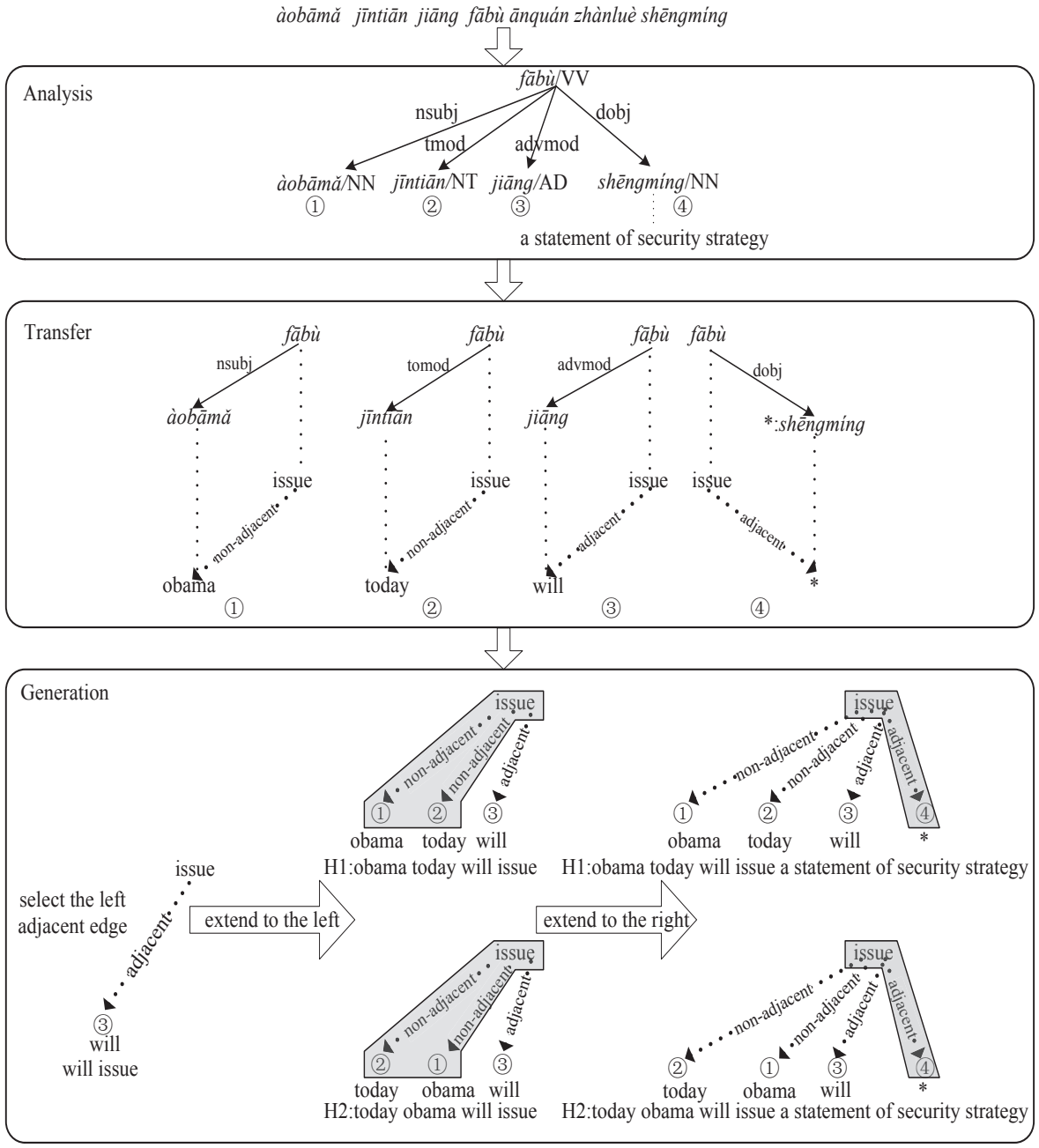

Figure 2: An example partial generation of translation. The same set of rules generate two target hypotheses with the same words and different word order. Assume the sub-tree rooted at "*shēngmíng*" has been translated to the corresponding target sentence fragment.

2.2 Transfer Rules

A transfer rule of our model represents the reordering and relative positions of edges between language pairs. For example, in Figure 1 (b), the first rule shows that when a nominal subject modifies a verb, the target side keeps the same position relations. "obama" is also on the left of "issue", the same with the source side relative position. The 5-th and 6-th rules show the inversion relations between the source and the target. Formally, a transfer rule can be defined as a triple $\langle e, f, \sim \rangle$, where e is an edge extracted from the source dependency tree, f is a target edge. $\sim$ denotes one-to-one correspondence between variables in e and f.

Figure 1 (b) are part of transfer rules extracted from the word aligned sentence in Figure 1 (a). The target edge denotes whether the target dependent is on the left or the right side of the target head, the

label on the edge indicates whether the target head and the target dependent are adjacent or not. If the dependent is an internal node(contrast with the leaf nodes in the dependency tree), then it will be regarded as a substitution node. The dependent in the 4-th transfer rule is an internal node and the its corresponding target side is a substitution variable.

Figure 2 shows a partial transfer-generation of our model which involves three phases. First, *analysis*. Given a source language sentence, we obtain its dependency tree using a dependency parser. We assume that the sub-tree of the substitution node has been translated. Second, *transfer*. For each internal node, we transfer the source side edges between the head and all its dependents into the target sides. In the second block of Figure 2, we transfer four edges into the target sides. Third, *generation*, corresponding to the third block of Figure 2. We generate the target sentence with the target side edges starting from the target head, "issue". We first try to concatenate the edges to the left. First, we select a target side edge that is on the left side of "issue" and adjacent to it to form a consecutive phrase. Edge 3 is selected and "to issue" is generated. Then, we enumerate all possible left concatenations of the other edges that are not adjacent to "issue". The two sequences(1,2,3 and 2,1,3) of the edges are generated, corresponding to the two hypotheses. After that, we extend the two hypotheses to the right. The internal node "*shēngmíng*" is a substitution node, so the candidate translation of the sub-tree rooted at "*shēngmíng*" is concatenated to the two hypotheses. Finally, we generate the two candidate translations of the input sentence.

3 Acquisition of Transfer Rules

Transfer rules can be extracted automatically from a word-aligned corpus, which is a set of triples $\langle T, S, A\rangle$, where T is a source dependency tree, S is a target side sentence and A is an alignment relation between T and S. Following the dependency-to-string model (Xie et al., 2011), we extract transfer rules from each triple $\langle T, S, A\rangle$ by three steps:

1. Tree Annotation: Label each node in the dependency tree with the alignment information
2. Edges Identification: Identify acceptable edges from the annotated dependency tree
3. Rule induction: Induce a set of lexicalized and un-lexicalized transfer rules from the acceptable edges.

3.1 Tree Annotation

Given a triple $\langle T, S, A\rangle$ as Figure 3 shows, we define two attributes for every node in T: node span and sub-tree span:

Definition 2. *Given a node n, its* ***node span*** $nsp(n)$ *is a set of consecutive indexes of the target words aligned with the node n.*

For example, nsp(*ānquán*)={7-8}, which corresponds to the target word "of" and "security".

Definition 3. *A node span $nsp(n)$ is* ***consistent*** *if for any other node n' in the dependency tree, $nsp(n)$ and $nsp(n')$ are not overlapping.*

For example, nsp(*zhànluè*) is consistent, while nsp(*ānquán*) is not consistent for it corresponds to the same word "of" with nsp(*shēngmíng*).

Definition 4. *Given a sub-tree T' rooted at n, the* ***sub-tree span*** $tsp(n)$ *of n is a consecutive target word indexes from the lower bound of the nsp of all the nodes in T' to the upper bound of those spans.*

For example, tsp(*shēngmíng*)={5-9},which corresponds to the target phrase "a statement of security strategy".

Definition 5. *A sub-tree span $tsp(n)$ is* ***consistent*** *if for any other node n' that is not in the sub-tree rooted at n in the dependency tree, $tsp(n)$ and $nsp(n')$ are not overlapping.*

For example, tsp(*shēngmíng*) is consistent, even though nsp(*shēngmíng*) is not consistent, while tsp(*ānquán*) is not consistent for "*shēngmíng*" is not a node in sub-tree rooted at "*ānquán*" and "*ānquán*" corresponds to the same word "*of*" with nsp(*shēngmíng*) .

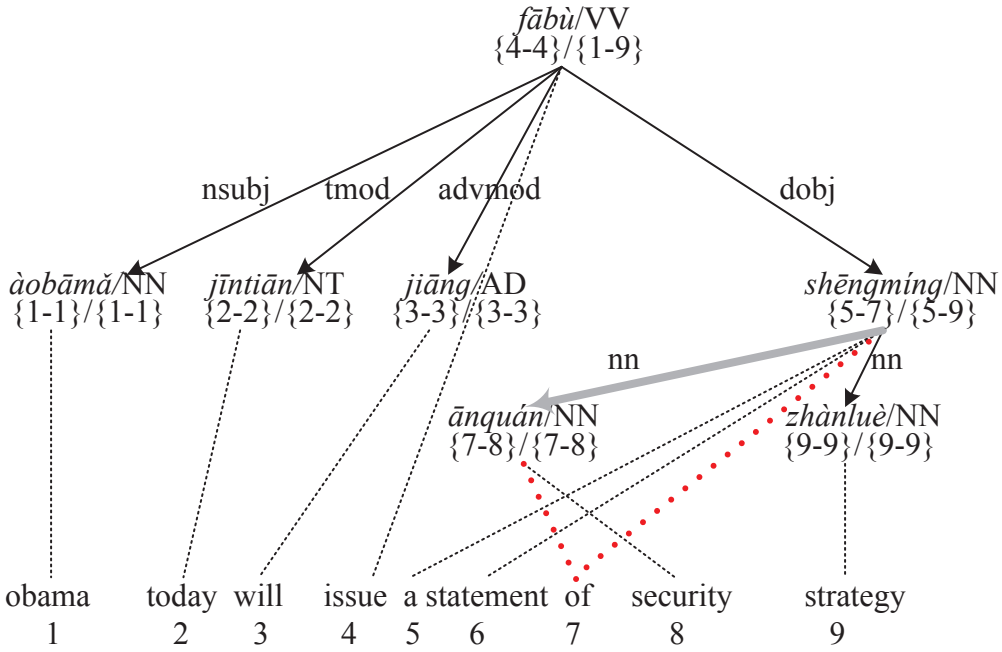

Figure 3: An example of annotated dependency tree. Each node is annotated with two spans, the former is node span and the latter is sub-tree span. The gray edge is not acceptable. It is different from Figure 1, because "ānquán" aligned with two words in Figure 3. "of" in the target side is aligned with both "ānquán" and "shēngmíng" which makes the gray edge un-acceptable.

3.2 Acceptable Edges Identification

We identify the edges from the annotated dependency tree that are **acceptable** for rule induction. For an acceptable edge, its *node span* of the head nsp(head) and the *sub-tree span* of the dependent tsp(dependent) satisfy the following properties:

1. nsp(head) and tsp(dependent) are consistent.
2. nsp(head) and tsp(dependent) are non-overlapping.

For example, tsp(*ānquán*) and nsp(*shēngmíng*) are neither consistent nor non-overlapping. So the gray edge between head "*shēngmíng*" and dependent "*ānquán*" is not an acceptable edge. nsp(*fābù*) and tsp(*shēngmíng*) are consistent and the two spans are non-overlapping. Thus, the edge between head "*fābù*" and dependent "*shēngmíng*" is an acceptable edge.

3.3 Transfer Rule Induction

From each acceptable source side edge, we induce a set of lexicalized and un-lexicalized transfer rules. We induce a lexicalized transfer rule from an acceptable edge by the following procedures:

1. extract the source side edge and mark the internal nodes as substitution sites. This form the input of a transfer rule.

2. extract the position information according to nsp(head) and tsp(dependent), whether they are adjacent or not and whether tsp(dependent) is on the left side or the right side of nsp(head).

In Figure 4, the first transfer rule is lexicalized rule, it is induced from the edge between "*fābù*" and "*àobāmǎ*".

In addition to the lexicalized rules described above, we also generalized the rules by replacing the word in an source side edge with a wild card and the part of speech of the word. For example, the rule in Figure 4 can be generalized in two ways. The generalized versions of the rule apply to "*àobāmǎ*" modifying any verb and "*fābù*" modifying any noun, respectively. The generalized rules are also called

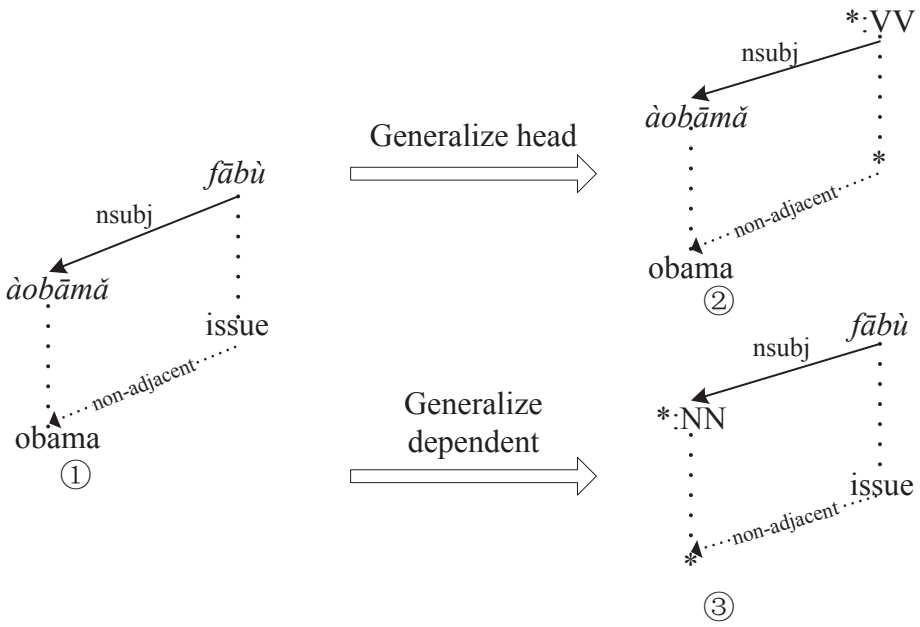

Figure 4: Generalization of transfer rule.

un-lexicalized rules for the loss of word information. The single node translations of the generalized words are also extracted.

The unaligned words of the target side is handled by extending nsp(head) and tsp(dependent) on both left and right directions. We do this process similar with the method of Och and Ney (2004). We might obtain $m(m \geq 1)$ extended rules from an acceptable edge. The frequency of each rule is divided by m. We take the extracted rule set as observed data and make use of relative frequency estimator to obtain the translation probabilities $P(t|s)$ and $P(s|t)$.

4 Decoding and Generation

We follow Och and Ney (2002), using a general log-linear model to score the sentence generated by each concatenation of the target edges. Let c be concatenations that concatenate the target edges to generate the target sentence e. The probability of e is defined as:

$$P(c) \propto \prod_i \phi_i(c)^{\lambda_i} \tag{1}$$

where $\phi_i(c)$ are features defined on concatenations and λ_i are feature weights. In our experiments of this paper, thirteen features are used as follows:

- Transfer rules translation probabilities $P(t|s)$ and $P(s|t)$, and lexical translation probabilities $P_{lex}(t|s)$ and $P_{lex}(s|t)$;
- Bilingual phrases probabilities $P_{bp}(t|s)$ and $P_{bp}(s|t)$, and bilingual phrases lexical translation probabilities $P_{bplex}(t|s)$ and $P_{bplex}(s|t)$;
- Transfer rule penalty $exp(-1)$;
- Bilingual phrase penalty $exp(-1)$;
- Pseudo translation rule penalty $exp(-1)$;
- Target word penalty $exp(|e|)$;
- Language model $P_{lm}(e)$.

Our decoder is based on a bottom-up chart-based beam-search algorithm. We regard the decoding process as the composition of the target side edges. For a given source language sentence, we obtain its

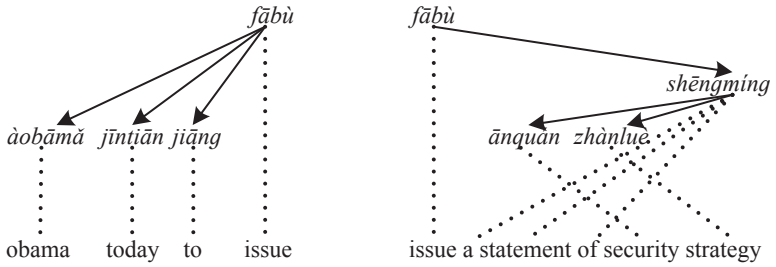

Figure 5: Two examples of the phrases incorporated in our model.

dependency tree T with an external dependency parser. Each node in T is traversed in post-order. For each internal node and root node n, we do the transfer-generation translation as the following procedures:

1. Extract all the source side edges including the lexicalized and generalized edges between n and all its dependents using the same way we extract the source side edges of the transfer rules.
2. *Transfer* the source side edges into target side edges. For a generalized rule, we restore it to a lexicalized rule by combining it with the single word translation. For no matched edges, we construct the pseudo translation rule according to the word order of the source head-dependent relation.
3. *Generate* the target sentence by bi-directional extension from an adjacent target edge. We first group all the target edges by their heads. For each group, we generate translation hypotheses with the following procedures:

   (a) Select an adjacent target edge as the starting position;

   (b) Extend to the left side and enumerate all possible permutations of the target edges directing left;

   (c) Extend to the right side and enumerate all possible permutations of the target edges directing right.

   Considering that in dependency trees, a head may relate to more than 4 edges which results in massive search space. We reduce the time complexity by using the maximum distortion limit. The distortion is defined as $(a_i - b_{i-1} - 1)$, where a_i denotes the start position of the source side edge that is translated into the ith target side edge and b_{i-1} denotes the end position of the source side edge translated into the $(i-1)$th target side edge.

When we reach the root node, the candidate translations of the input sentence are generated.

In our model, only the adjacent target edge of a transfer rule can be regarded as a consecutive phrase and its corresponding source side length is only 2. As we start extending the target sentence from the target head, it is quite natural to incorporate the bilingual phrases to make the target sentences be extended from the phrases as well as the single target head word. Due to the flexibility of our model, we can incorporate not only the syntactic phrases which are phrases covering a whole sub-tree, but also the non-syntactic phrases as the fixed dependency structures in Shen et al. (2008) which are consecutive phrases covering the head. Figure 5 shows two examples of the phrases incorporated in our model.

We prune the search space in several ways. First, beam threshold β, items with a score worse than β times of the best score in the same span will be discarded; second, beam size b, items with a score worse than the bth best item will be discarded. For our experiments, we set $\beta = 10^{-3}$ and $b = 300$; Third, we also prune rules for the same edge with a fixed rule limit ($r = 200$), which denotes the maximum number of rules we keep.

5 Experiments

In this section, the performance of our model is evaluated by comparing with phrase-based model (Koehn et al., 2003), on the NIST Chinese-to-English translation tasks. We also present the influence of the

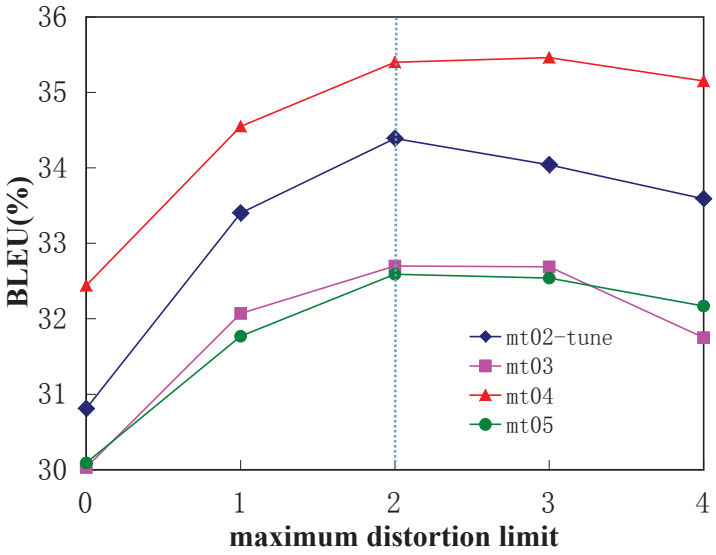

Figure 6: Effect of different maximum distortion limits on development set (mt02) and three tests(mt03,04,05). The performance of all the sets are consistent.

maximum distortion limit to our model. We take open source phrase-based system *Moses* (with default configuration)[1] as our baseline system.

5.1 Experimental Setting

Our training corpus consists of 1.25M sentence pairs from LDC data, including LDC2002E18, LDC2003E07, LDC2003E14, Hansards portion of LDC2004T07, LDC2004T08 and LDC2005T06.

To obtain the dependency trees of the source side, we parse the source sentences with Stanford Parser (Klein and Manning, 2003) into projective dependency structures with nodes annotated by POS tags and edges by dependency labels.

To obtain the word alignments, we run GIZA++ (Och and Ney, 2003) on the corpus in both directions and apply "grow-diag-and" refinement (Koehn et al., 2003). We extract the phrases covering no more than 10 nodes of the fixed structures.

We use SRILM (Stolcke, 2002) to train a 4-gram language model with modified Kneser-Ney smoothing on the Xinhua portion of the Gigaword corpus.

We use NIST MT Evaluation test set 2002 as our development set, 2003-2005 NIST datasets as testsets. The quality of translations is evaluated by the case insensitive NIST BLEU-4 metric[2].

We make use of the minimum error rate training algorithm (Och, 2003) in order to maximize the BLEU score of the development set.

The statistical significance test is performed by *sign-test* (Collins et al., 2005).

5.2 Influence of Maximum Distortion Limit

Figure 6 gives the performance of our system with different maximum distortion limits in terms of uncased BLEU of three NIST test sets. The performance of different distortion limit are consistent on both development set and three test sets. Maximum distortion limit 2 gets the best performances. A low distortion limit may cause the target sentence been translated more close to the sequence of the source, especially when the distortion limit equals to 0, none of the reordering is allowed, while a high distortion limit may lead the good translations be flooded by too many ambiguities when enumerating the possible sequences of the target non-adjacent dependents. We choose 2 as the maximum distortion limit in the next experiments.

[1] http://www.statmt.org/moses/

[2] ftp://jaguar.ncsl.nist.glv/mt/resources/mteval-v11b.pl.

System	Rule #	MT03	MT04	MT05	Average
Moses	44.49M	32.03	32.83	31.81	32.22
DEBT	30.7M	**32.7***	**35.4***	**32.59***	**33.56**

Table 1: Statistics of the extracted rules on training data and the BLEU scores (%) on the test sets. "DEBT" denotes our edge-based transfer model. The "*" denotes that the results are significantly better than the baseline system ($p<0.01$).

5.3 Performance of Our Model

Tabel 1 illustrates the translation results of our experiments. We (*DEBT*) surpass the baseline over +1.34 BLEU points on average. Our model significant outperforms the baseline phrase-based model, with $p < 0.01$ on statistical significance test *sign-test* (Collins et al., 2005).

We also list the statistical number of rules extracted from the training corpus. The number of our transfer rules is only 69.0% of the rules extracted by *Moses*, thus, the total rules in our model is 31% smaller than *Moses*.

6 Related Work

Transfer-based MT systems usually take a parse tree in the source language and translate it into a parse tree in the target language with transfer rules. Both our model and some of those previous works acquired transfer rules automatically from word-aligned corpus (Richardson et al., 2001; Carbonell et al., 2002; Lavoie et al., 2002; Lin, 2004). Gimpel and Smith (2009) and Gimpel and Smith (2014) used quasi-synchronous dependency grammar for MT and they are similar to our idea of doing transfer of dependency syntax in a non-synchronous setting. They do the translation as monolingual lattice parsing.

As dependency-based system, Lin (2004) used path as the transfer unit and regarded the translation problem with minimal path covering. Quirk et al. (2005) and Xiong et al. (2007) used treelets to model the source dependency tree using synchronous grammars. Quirk et al. (2005) projected the source dependency structure into target side by word alignment and faced the problem of non-isomorphism between languages. Xiong et al. (2007) directly modeled the treelet to the corresponding target string to alleviate the problem. Xie et al. (2011) directly specified the ordering information in head-dependents rules that represent the source side as head-dependents relations and the target side as string.

Differently, our model uses a much simpler elementary structure, edge, which consist of only a head and a dependent. As a transfer-generation model, we transfer an edge in the source dependency tree into target side and incorporate the position information on the target edge , which alleviate non-isomorphism problem and incorporate ordering among different target edges simultaneously. Moreover, our decoding method is quite different from previous dependency tree-based works. After parsing a given source language sentence, we transfer and generate the target sentence fragments recursively on each internal node of the dependency tree bottom-up.

7 Conclusions and Future Work

In this paper, we present a novel dependency edge-based transfer model using dependency trees on the source side for machine translation. We directly *transfer* the edges in source dependency tree into the target sides and then *generate* the target sentences by beam-search. With the concise transfer rules, our model is compatible with both the syntactic and non-syntactic phrases. Although the generation process of our model seems relatively simple, it still exhibits a good performance and outperforms the phrase-based model on large scale experiments. For the first time, a statistical transfer model shows a comparable performance with the state-of-the-art translation models.

Since the translation procedure is divided into three phases and each phase can be modeled independently, we would like to take further steps focusing on modeling the target language generation process specifically to ensure a better grammatical translation with the help of natural language generation methods.

Acknowledgments

The authors were supported by National Key Technology R&D Program (No. 2012BAH39B03), CAS Action Plan for the Development of Western China (No. KGZD-EW-501), and Sino-Thai Scientific and Technical Cooperation (No. 60-625J). Sincere thanks to the anonymous reviewers for their thorough reviewing and valuable suggestions.

References

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.

Jaime Carbonell, Katharina Probst, Erik Peterson, Christian Monson, Alon Lavie, Ralf Brown, and Lori Levin. 2002. Automatic rule learning for resource-limited mt. In *Machine Translation: From Research to Real Users*, pages 1–10. Springer.

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. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Michael Collins, Philipp Koehn, and Ivona Kučerová. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, pages 531–540. Association for Computational Linguistics.

Yuan Ding and Martha Palmer. 2004. Synchronous dependency insertion grammars: A grammar formalism for syntax based statistical mt. In *Workshop on Recent Advances in Dependency Grammars (COLING)*, pages 90–97.

Kevin Gimpel and Noah A Smith. 2009. Feature-rich translation by quasi-synchronous lattice parsing. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 219–228. Association for Computational Linguistics.

Kevin Gimpel and Noah A Smith. 2014. Phrase dependency machine translation with quasi-synchronous tree-to-tree features. *Computational Linguistics*.

Jonathan Graehl and Kevin Knight. 2004. Training tree transducers. In Daniel Marcu Susan Dumais and Salim Roukos, editors, *HLT-NAACL 2004: Main Proceedings*, pages 105–112, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Liang Huang, Kevin Knight, and Aravind Joshi. 2006. Statistical syntax-directed translation with extended domain of locality. In *Proceedings of AMTA*, pages 66–73.

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*, NAACL '03, pages 48–54, Stroudsburg, PA, USA. Association for Computational Linguistics.

Benoit Lavoie, Michael White, and Tanya Korelsky. 2002. Learning domain-specific transfer rules: an experiment with korean to english translation. In *Proceedings of the 2002 COLING workshop on Machine translation in Asia-Volume 16*, pages 1–7. Association for Computational Linguistics.

Dekang Lin. 2004. A path-based transfer model for machine translation. In *Proceedings of Coling 2004*, pages 625–630, Geneva, Switzerland, Aug 23–Aug 27. COLING.

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. Association for Computational Linguistics.

Daniel Marcu and William Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 133–139. Association for Computational Linguistics.

Fandong Meng, Jun Xie, Linfeng Song, Yajuan Lü, and Qun Liu. 2013. Translation with source constituency and dependency trees. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1066–1076, Seattle, Washington, USA, October. Association for Computational Linguistics.

Haitao Mi, Liang Huang, and Qun Liu. 2008. Forest-based translation. In *Proceedings of ACL-08: HLT*, pages 192–199, Columbus, Ohio, June. Association for Computational Linguistics.

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*, ACL '02, pages 295–302, Stroudsburg, PA, USA. Association for Computational Linguistics.

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

Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational linguistics*, 30(4):417–449.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Stephen Richardson, William Dolan, Arul Menezes, and Jessie Pinkham. 2001. Achieving commercial-quality translation with example-based methods. In *Proceedings of MT Summit VIII*, pages 293–298. Santiago De Compostela, Spain.

Libin Shen, Jinxi Xu, and Ralph Weischedel. 2008. A new string-to-dependency machine translation algorithm with a target dependency language model. In *Proceedings of ACL-08: HLT*, pages 577–585, Columbus, Ohio, June. Association for Computational Linguistics.

Andreas Stolcke. 2002. Srilm-an extensible language modeling toolkit. In *Proceedings of ICSLP*, volume 30, pages 901–904.

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*, EMNLP '11, pages 216–226, Stroudsburg, PA, USA. Association for Computational Linguistics.

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*, StatMT '07, pages 40–47, Stroudsburg, PA, USA. Association for Computational Linguistics.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, pages 523–530. Association for Computational Linguistics.