Augment Dependency-to-String Translation with Fixed and Floating Structures

Jun Xie[†] **Jinan Xu**[‡] **Qun Liu**[†§]
†Key Laboratory of Intelligent Information Processing,
Institute of Computing Technology,Chinese Academy of Sciences
xiejun@ict.ac.cn
‡School of Computer and Information Technology, Beijing Jiaotong University
xja2010@gmail.com
§School of Computing, Dublin City University
qliu@computing.dcu.ie

Abstract

In this paper, we propose an augmented dependency-to-string model to combine the merits of both the head-dependents relations at handling long distance reordering and the fixed and floating structures at handling local reordering. For this purpose, we first compactly represent both the head-dependent relation and the fixed and floating structures into translation rules; second, in decoding we build “on-the-fly” new translation rules from the compact translation rules that can incorporate non-syntactic phrases into translations, thus alleviate the non-syntactic phrase coverage problem of dependency-to-string translation (Xie et al., 2011). Large-scale experiments on Chinese-to-English translation show that our augmented dependency-to-string model gains significant improvement of averaged +0.85 BLEU scores on three test sets over the dependency-to-string model.

1 Introduction

As a representation holding both syntactic and semantic information, dependency grammar has been attracting more and more attention in statistical machine translation. Lin (2004) took paths as the elementary structures and proposed a path-based transfer model. Quirk et al. (2005) extended path to treelets (connected subgraphs of dependency trees) and put forward dependency treelet translation. Ding and Palmer (2005) proposed a model on the basis of dependency insertion grammar. Shen et al. (2008) employed the fixed and floating structures as elementary structures and proposed a string-to-dependency model with state-of-the-art performance. Xie et al. (2011) employs head-dependents relations as elementary structures and proposed a dependency-to-string model with good long distance reordering property. A head-dependents relation (HDR) is composed of a head and all its dependents, which can be viewed as an instance of a sentence pattern or phrase pattern.

However, since dependency trees are much flatter than constituency trees, the dependency-to-string model suffers more severe non-syntactic phrase coverage problem (Meng et al., 2013) than constituency-based models (Galley et al., 2004; Liu et al., 2006; Huang et al., 2006). Non-syntactic phrases are those phrases that can not be covered by whole subtrees. To address this problem, Meng et al. (2013) proposed to translate with both constituency and dependency trees, which can incorporate non-syntactic phrases covered by the constituents of the constituency trees. This model requires both constituency and dependency trees, thus may suffer from both constituency and dependency parse errors. Additionally, there are only few languages that have both constituency and dependency parsers, which limits its practical use.

In this paper, we propose to address non-syntactic phrase coverage problem of the dependency-to-string model without resort to extra resources (Section 3). To this end, we augment the dependency-to-string model at two aspects. First, we combine the merits of both the head-dependent relations and the fixed and floating structures (Shen et al., 2008), and compactly represent these two kinds of knowlege into augmented HDR rules (Section 3.1). We acquire the augmented HDR rules automatically from the

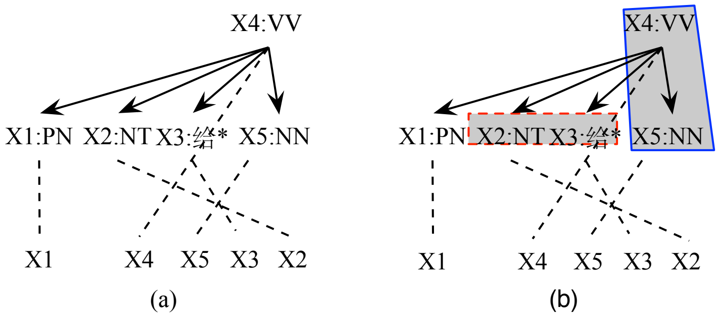

Figure 1: Examples of an HDR rule (a) and an augmented HDR rule (b). Where each "*" denotes a substitute site which is a compact representation of a whole subtree. The shadow with line border indicates a fixed structure and the shadow with dash line border indicates a floating structure.

word-aligned source dependency tree and target string paris (Section 3.2). In decoding we propose an "on-the-fly" rule building strategy, which builds new translation rules from the augmented HDR rules and incorporates non-syntactic phrases into translations (Section 3.4). Large-scale experiments (Section 4) on Chinese-to-English translation show that our augmented model gains significant improvement of averaged +0.85 BLEU points on three test sets over the dependency-to-string model.

2 Background

For convenience of the description of our augmented dependency-to-string model, we first briefly review the dependency-to-string model and the fixed and floating structures of string-to-dependency model (Shen et al., 2008).

2.1 Dependency-to-String Translation

The dependency-to-string model (Xie et al., 2011) takes head-dependents relations as the elementary structures of dependency trees, and represents the translation rules with the source side as HDRs and the target side as string. Since the HDRs in essence relate to phrase patterns and sentence patterns, the HDR rules specify the reordering of these patterns. For example, Figure 1 (a) is an example HDR rule, which represents a reordering manner of a sentence pattern composed of a proper noun (X1:PN), a temporal noun (X2:NT), an prepositional phrase relate to "给 (give)" (X3:给), a verb (X4:VV) and a noun (X5:NN).

With the HDR rules, the dependency-to-string model gets rid of the extra reordering heuristics and reordering models of the previous models (Lin, 2004; Ding and Palmer, 2005; Quirk et al., 2005). More importantly, the model shows state-of-the-art performance and exhibits good long distance reordering property.

2.2 Fixed and Floating Structures

The fixed structures and floating structures are fundamental structures of the string-to-dependency model (Shen et al., 2008), which are introduced to handle the coverage of non-constituent rules. Given the dependency tree $d_1 d_2 ... d_n$ of a sentence $f_1 f_2 ... f_n$, where d_i indicates the parent word index of word f_i.

Definition 1. *A dependency structure $d_{i...j}$ is* ***fixed*** *on the head h, where $h \in [i, j]$, if and only if it meets the following conditions:*

- $d_h \notin [i, j]$
- $\forall k \in [i, j]$ *and* $k \neq h$, $d_k \in [i, j]$
- $\forall k \notin [i, j]$, $d_k = h$ *or* $d_k \notin [i, j]$

A fixed structure describes a fragment with a sub-root, where all the children of the sub-root are complete.

Definition 2. *A dependency tree $d_{i...j}$ is* ***floating*** *with children C, for a non-empty set $C \subseteq i, ..., j$, if and only if it meets the following conditions:*

- $\exists h \notin [i,j], s.t. \forall k \in C, d_k = h$
- $\forall k \in [i,j]$ *and* $k \notin C$, $d_k \in [i,j]$
- $\forall k \notin [i,j]$, $d_k \notin [i,j]$

A floating structure consists of sibling nodes of a common head, but the head itself is unspecified.

In nature, the fixed and floating structures represent the phrases under the structural constraint of dependency trees, most of them are non-syntactic phrases.

The HDRs are good at handling long distance dependencies, while the fixed and floating structures excels at handling local reordering. This encourages us to address the non-syntactic phrase coverage problem of dependency-to-string model by exploiting these two kinds of structures.

3 Augmented Dependency-to-String Translation

In the following, we will describe our augmented dependency-to-string model in detail, including the augmented HDR rules (Section 3.1), rule acquisition (Section 3.2) and "on-the-fly" rule building in decoding (Section 3.4).

3.1 Augmented HDR rules

Our augmented HDR rules aim at combining the merits of both the HDRs at handling long distance reordering and the fixed and floating structures at handling local reordering. For this purpose, we augment the HDR rules (Xie et al., 2011) by labelling the HDRs with the fixed and floating structures.

Figure 1 (b) shows an example augmented HDR rule. Which is an augmented version of the HDR rule Figure 1 (a) by labelling it with a fixed structure (shadow with line border) and a floating structure (shadow with dash line border). The labeled fixed and floating structures indicate the bilingual phrases that we can incorporate in this sentence pattern.

3.2 Rule Acquisition

Given a word-aligned parallel corpus defined as a set of triples $\langle T, e, A\rangle$, where T is a dependency tree of source sentence f_1^J, e_1^I is the target sentence and A is an alignment relation between f_1^J and e_1^I, we acquire the augmented HDR rules by three steps: tree annotation, acceptable HDR identification and rule induction. The process is similar with that of Xie et al. (2011). However, we make some extensions so that we can take the fixed and floating structures into account.

3.2.1 Tree Annotation

Besides annotating each node of T with *head span* and *dependency span* as Xie et al. (2011), we also label the tree with consistent fixed and floating structures.

Definition 3. *The* ***head span*** *hsp(n) of a node n is the closure of the set taking the index of the target words aligned to n as its elements.*

The *closure* of a set contains all the elements between the minimum and the maximum of the set and each element has only one copy. For example, the closure of set $\{1, 3\}$ is $\{1, 2, 3\}$.

We say a head span is *consistent* with alignment if the bilingual phrase it covers is consistent with the alignment (Koehn et al., 2003).

Definition 4. *Given a subtree T' rooted at n, the* ***dependency span*** *dsp(n) of n is the closure of the union of the consistent head spans of all the nodes of T'.*

$$dsp(n) = closure(\bigcup_{\substack{n' \in T' \\ hsp(n') \ is \ consistent}} hsp(n'))$$

If no head spans of all the nodes of T' are consistent, $dsp(n) = \varnothing$.

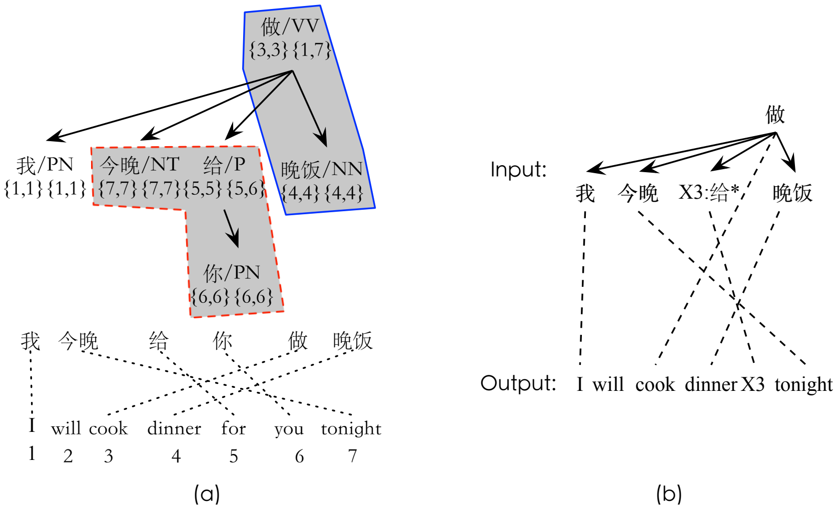

Figure 2: An example annotated dependency tree (a) and an example lexicalized augmented HDR rule (b) induced from the top-level HDR of (a). Each node of the dependency tree is annotated with two spans: head span (the former) and dependency span (the latter). The shadows denote a consistent fixed structure (shadow with line border) and a floating structure (shadow with dash line border). The "*" denotes a substitute site.

Definition 5. *A fixed or floating structure is* ***consistent*** *with alignment if the phrase it covers is consistent with alignment.*

Tree annotation can be readily accomplished by a single post-order traversal of dependency tree T. For each accessed node n, annotate it with head span and dependency span according to A. If n is an internal node, enumerate all the fixed and floating structures relate to n, and label those consistent ones on T. Repeat the above process till the root is accessed.

Figure 2 (a) shows an example annotated dependency tree. Where each node is annotated with two spans: head span (the former) and dependency span (the latter). Moreover, the dependency tree is also labeled with two consistent fixed and floating structures that cover phrases "做 饭" and "今晚 给 你" respectively.

3.2.2 Acceptable HDR Identification

From the annotated dependency tree, we identify the HDRs that are suitable for rule induction. These HDRs are called as acceptable HDRs. To this end, we traverse the annotated dependency tree in post-order and identify the HDRs with the following properties:

- for the head, its head span is consistent;
- for the dependents, the dependency span of each dependent should not be $\varnothing$ unless the dependent is a leaf node;
- the intersection of the head span of the head and the dependency spans of the dependents is $\varnothing$ (or do not overlap).

Different from those acceptable HDRs of Xie et al. (2011), the acceptable HDRs here may be labeled with fixed and floating structures. For example, the top level of Figure 2 (a) is an acceptable HDR, which is labeled with a fixed structure and a floating structures. Typically an acceptable HDR has three types of nodes: leaf node (of the dependency tree), internal node (of the dependency tree) and head node (an internal node function as the head of the HDR).

3.2.3 Rule Induction

From each acceptable HDR, we induce a set of lexicalized and unlexicalized augmented HDR rules. This process is similar with that of Xie et al. (2011) except that here we have to consider the consistent fixed

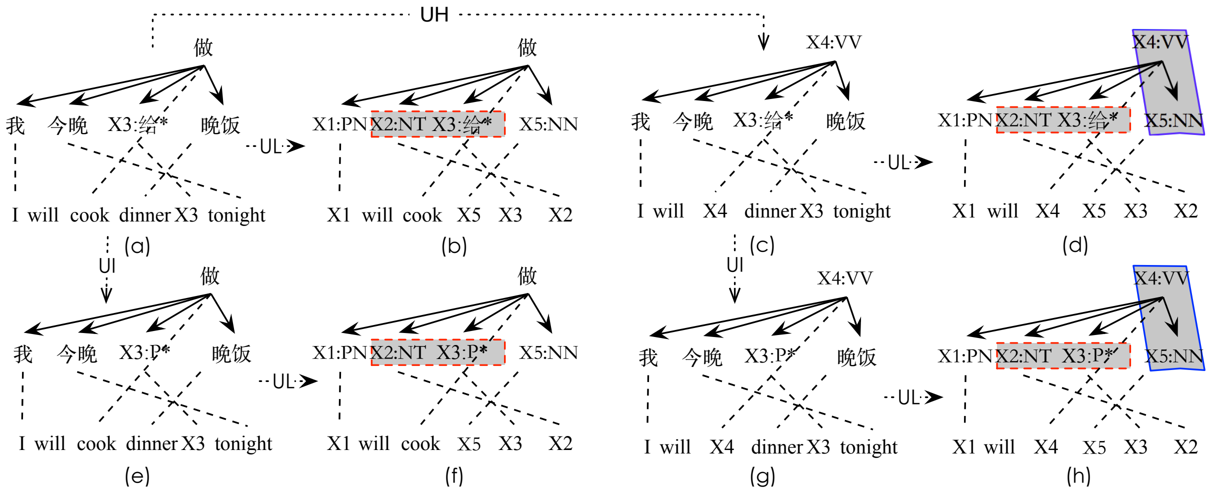

Figure 3: Lexicalized augmented HDR rule (a) and unlexicalized augmented HDR rules (b)∼(h) induced from the top level HDR of the annotated dependency tree in Figure 2. Where "UH", "UI" and "UL" denotes "unlexicalize head", "unlexicalize internal" and "unlexicalize leaf" , respectively. The shadows with line border denote fixed structures and the shadows with dash line border denotes floating structures.

and floating structures.

First, we induce a lexicalized augmented HDR rule with the following principles:

1. extract the HDR, mark each internal node as a variable, and label the HDR with the floating structures that cover only variables. This forms the input of a lexicalized rule.

2. generate the target string according to head span of the head and the dependency spans of the related dependents, and turn the word sequences covered by the dependency spans of the internal nodes into variables. This forms the output of a lexicalized rule.

Figure 2 (b) illustrates a lexicalized augmented HDR rule induced from the top-level HDR of the annotated dependency tree Figure 2 (a).

From each lexicalized augmented HDR rule (along with the acceptable HDR), we then induce a set of unlexicalized augmented HDR rules with the following principles:

1. turn each type (leaf, internal or head) of nodes simultaneously into variables;
2. when turning a head or leaf node into a variable, change the counterpart of the target side into the variable; label the unlexicalized HDR with the fixed and floating structures that cover only variables.
3. when turing an internal node into a variable, keep the counterpart of the target side unchanged.

Totally, we will obtain eight types of augmented HDR rules from an acceptable HDR. In this paper, we call the lexicalized and unlexicalized HDRs generated by the above process as instances of the HDR.

Figure 3 illustrates the rule induction of seven unlexicalized augmented HDR rules (b)-(h) from lexicalized augmented HDR rule (a). Where "UH", "UI" and "UL" on the dash arrows indicate "unlexicalize head", "unlexicalize internal" and "unlexicalized leaf", respectively.

3.2.4 Probability Estimation

We take the augmented HDR rules acquired from word-aligned parallel corpus as the observed data, and employ relative frequency estimation to calculate the translation probabilities of the rules. Note that, here we take the labeled fixed and floating structures of the augmented HDR rules as indicators of bilingual phrases that can be incorporated in the sentence patterns and phrases patterns represented by the HDRs. So we consider only the HDRs when counting the augmented HDR rules.

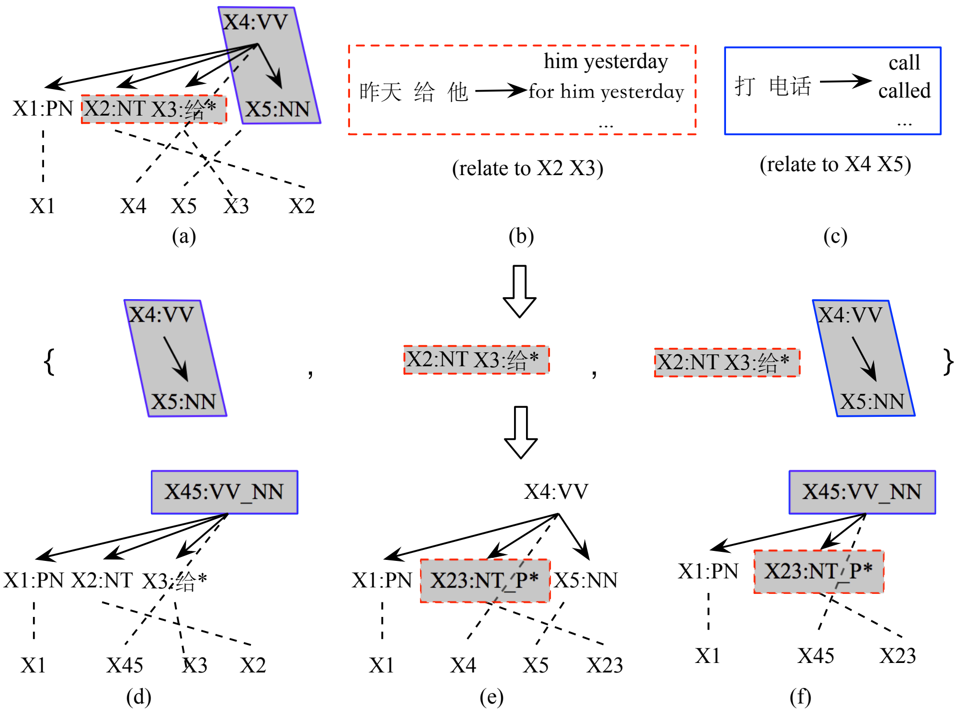

Figure 4: Illustration of "on-the-fly" translation rule building.

3.3 The model

Following Och and Ney (2002), we adopt a general log-linear model for our augmented dependency-to-string model. Let d be a derivation that converts a source dependency tree T into a target string e. The probability of derivation d is defined as:

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

where ϕ_i are features defined on derivation and λ_i are feature weights.

In our implementation, we make use of eleven features, including seven features inherited from the dependency-to-string model:

- translation probabilities $P(f|e)$ and $P(e|f)$ and lexical translation probabilities $P_{lex}(f|e)$ and $P_{lex}(e|f)$ of augmented HDR rules
- rule penalty $exp(-1)$
- language model $P_{lm}(e)$
- word penalty $exp(|e|)$, where $|e|$ is the length of the generated target string

and four extra features for bilingual phrases relate to fixed and floating structures:

- translation probabilities $P_{bp}(f|e)$ and $P_{bp}(e|f)$ and lexical translation probabilities $P_{bp\_lex}(f|e)$ and $P_{bp\_lex}(e|f)$ of bilingual phrases

3.4 "On-the-Fly" Decoding

The task of the decoder is to find the best derivation from all possible derivations. Our decoder is based on bottom-up chart parsing, which characterizes at "on-the-fly" translation rule building.

Given an input dependency tree T, the decoder traverses it in post-order. For each accessed node n, the decoder first enumerates all instances of the HDR rooted at n as we do in rule induction, and checks for matched augmented HDR rules. If a matched rule is labeled with fixed and floating structures, the decoder builds new translation rules "on the fly" with the following principles:

1. check the phrases covered by the labeled fixed and floating structures for matched bilingual phrases;
2. if there are no matched bilingual phrases for all labeled fixed and floating structures, take the augmented HDR rule as a HDR rule of dependency-to-string model; otherwise,
    - enumerate all combinations of the fixed and floating structures with matched bilingual phrases;
    - for each combination, build a new translation rule by turning the variable sequences covered by the fixed and floating structures into new variables;
    - the new-built rule inherits the translation probabilities of the deriving augmented HDR rule, and the new variables take the matched bilingual phrases as their translation hypothesis.

Figure 4 illustrates the "on-the-fly" rule building process. Suppose augmented HDR rule (a) is the matched rule, and bilingual phrases (b) and (c) match the phrases covered by the labeled fixed and floating structures of (a). There will be three combinations of the labeled fixed and floating structures as shown in the middle of Figure 4. For each combination, the decoder builds a new translation rule by turning variable sequences "X2:NT X3:给*" and/or "X4:VV X5:NN" into new variables "X23:NT_P*" and/or "X45:VV_NN". And we will obtain three new translation rules (d)-(f) that can incorporate non-syntactic phrases into translations.

If there are no matched rules, the decoder builds a pseudo translation rule with monotonic reordering.

The decoder then employs cube pruning (Chiang, 2007; Huang and Chiang, 2007) to generate k-best hypothesis with integrated language model for node n.

Repeat the above process till the root of T is accessed. The hypothesis with the highest score is output as translation.

4 Experiments

We evaluated our augmented model by comparison with dependency-to-string model and hierarchical phrase-based model on Chinese-to-English translation in terms of BLEU (Papineni et al., 2002).

4.1 Experimental Setup

The parallel training corpus include 1.25M Chinese-English sentence pairs.[1] We parse the Chinese sentences with Stanford Parser (Klein and Manning, 2003) into projective dependency trees, obtain word alignment by running GIZA++ (Och and Ney, 2003) in both directions and applying "grow-diag-final" refinement (Koehn et al., 2003), and train a 4-gram language model by SRI Language Modeling Toolkit (Stolcke, 2002) with Kneser-Ney smoothing on the Xinhua portion of the Gigaword corpus.

We take NIST MT Evaluation test set 2002 as our development set, 2003 (MT03), 2004 (MT04) and 2005 (MT05) as our test sets, evaluate the quality of translations by *case insensitive* NIST BLEU-4 metric[2], tune the feature weights by Max-BLEU strategy with MERT (Och, 2003), and check the statistical difference between the systems with significance test (Collins et al., 2005).

4.2 Systems

We take "Moses-Chart" of Moses[3] (Koehn et al., 2007) as hierarchical phrase-based model baseline. In our experiments, we use the default settings.

Both the dependency-to-string baseline and our augmented model employ the same settings as those of Xie et al. (2011), with the beam threshold, beam size and rule size are set to 10^{-3}, 200 and 100 respectively. And both systems employ bilingual phrases with length $\leq$ 7 extracted by Moses.

4.3 Experiment results

Table 1 shows the results of the BLEU scores of the three systems. Where "dep2str" and "dep2str-aug" denote dependency-to-string model baseline and our augmented dependency-to-string model, respectively. As we can see, "dep2str" shows better performance (+0.31 BLEU on average) than "Moses-Chart"

[1] From LDC2002E18, LDC2003E07, LDC2003E14, Hansards portion of LDC2004T07, LDC2004T08 and LDC2005T06.

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

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

System	Rule#	MT03	MT04	MT05	Average
Moses-Chart	116.4M	34.65	36.47	34.39	35.17
dep2str	37M+32.5M	34.92	36.82	34.71	35.48
dep2str-aug	37M+32.5M	**35.66*** (**+0.74**)	**37.61*** (**+0.79**)	**35.74*** (**+1.03**)	**36.33** (**+0.85**)

Table 1: Statistics of the extracted rules and BLEU scores (%) on the test sets of the three systems. Where "37M+32.5M" denotes 37M rules and 32.5M bilingual phrases. And "*" indicates *dep2str-aug* are statistically better than *dep2str* with $p < 0.01$.

Source: 桑帕约 对 葡 中 两 国 在 世博会 事务 方面 的 合作 寄予 厚望 。

Reference 1: *Sampaio has placed high hopes on the Portuguese-Sino cooperation in the World Expo.*

Reference 2: *Sampaio expressed his high expectations on the Sino-Portuguese cooperation in the work of the world exposition.*

Moses-Chart: Sampaio on cooperation between the two countries in the world expo affairs Portugal and China places great .

Dep2Str: President placed great cooperation between Portugal and China , the two countries in the World Expo affairs .

Dep2Str-aug: Sampaio placed high expectations of the Portuguese - Chinese cooperation in World Expo affairs .

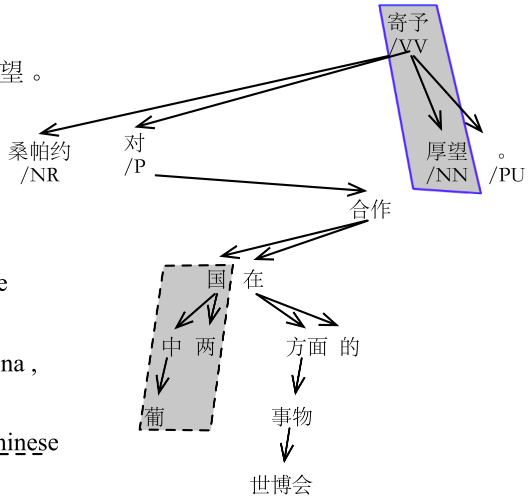

Figure 5: Translation examples of "Moses-Chart", "dep2str" and "dep2str-aug". The line border shadow denotes the phrases successfully captured by "dep2str-aug".

and is a strong baseline."Dep2str-aug" gains significant improvements of +0.74, +0.79 and +1.03 BLEU points over "dep2str" on the test sets, respectively.

Additionally, we compare the actual translations generated by "Moses-Chart", "dep2str" and "dep2str-aug". Figure 5 shows the translations of these three systems on a sentence of MT05. The source sentence holds a common sentence pattern in Chinese, which is composed of a proper noun, a verb, a noun and a prepositional phrases (corresponding to the top level of the dependency tree on the right). However, the preposition phrase related to "对" holds nine words, thus the simple pattern becomes a long distance dependency that challenges SMT systems. Limited by the phrase-based rules, "Moses-Chart" fails to capture the sentence pattern and outputs a messy translation with little sense. "Dep2str", resorting to HDR rules, successfully captures the pattern and outputs a translation with correct reordering, but it is still hard to understand. With the help of augmented HDR rules, "dep2str-aug" captures both the sentence pattern and non-syntactic phrase "寄予 厚望" and gives an translation with good adequacy and fluency.

These results reveal the merits of our augmented dependency-to-string model at handling both long distance reordering (with HDR) and local reordering (with fixed and floating structures), which is promising for translating language pairs that are syntactically divergent.

5 Conclusion and Future Work

In this paper, we propose an augmented dependency-to-string model to address the non-syntactic phrase coverage problem for dependency-to-string model. To this purpose, we make two important augmentations to the dependency-to-string model. First, we propose an compact representation to combine both head-dependent relation and the fixed and floating structures into translation rules. Second, in decoding we build "on the fly" new translation rules from the compact translation rules and incorporate non-syntactic phrases into translations. By this way, we can combine the merits of both head-dependents relation at handling long distance reordering and bilingual phrases at handling local reordering. Large-scale experiments show that our augmented dependency-to-string model gains significant improvements over the dependency-to-string model.

In the future work, we would like to incorporate semantic knowledge such as typed dependencies and WordNet[4] (Miller, 1995) so as to better direct the process of translation.

Acknowledgments

The authors were supported by National Nature Science Foundation of China (Contract 61370130 and 61379086). Liu was partially supported by the Science Foundation Ireland (Grant No. 07/CE/I1142) as part of the CNGL at Dublin City University. We sincerely thank the anonymous reviewers for their careful review and insightful suggestions.

References

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33.

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the ACL 2005*, pages 531–540, Ann Arbor, Michigan, June.

Yuan Ding and Martha Palmer. 2005. Machine translation using probabilistic synchronous dependency insertion grammars. In *Proceedings of ACL 2005*, pages 271–279.

Michel Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2004. What's in a translation rule? In *HLT-NAACL 2004: Main Proceedings*, pages 273–280, Boston, Massachusetts, USA, May 2 - May 7.

Liang Huang and David Chiang. 2007. Forest rescoring: Faster decoding with integrated language models. In *Proceedings of ACL 2007*, pages 144–151, Prague, Czech Republic, June.

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

Dan Klein and Christopher D. Manning. 2003. Fast exact inference with a factored model for natural language parsing. In *In Advances in Neural Information Processing Systems 15 (NIPS)*, pages 3–10. MIT Press.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, Edmonton, Canada, July.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL 2005: Interactive Poster and Demonstration Sessions*, pages 177–180.

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.

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment template for statistical machine translation. In *Proceedings of ACL 2006*, pages 609–616, Sydney, Australia, July.

Fandong Meng, Jun Xie, Linfeng Song, Yajuan Lü, and Qun Liu. 2013. Translation with source constituency and dependency trees. In *Proceedings of EMNLP 2013*, pages 1066–1076, Seattle, Washington, USA, October.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of ACL 2002*, pages 295–302, Philadelphia, Pennsylvania, USA, July.

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

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of ACL-2003*, pages 160–167, Sapporo, Japan, July.

[4] http://wordnet.princeton.edu

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of ACL 2002*, pages 311–318, Philadelphia, Pennsylvania, USA, July.

Chris Quirk, Arul Menezes, and Colin Cherry. 2005. Dependency treelet translation: Syntactically informed phrasal smt. In *Proceedings of ACL 2005*, pages 271–279.

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 2008: HLT*, pages 577–585, Columbus, Ohio, June.

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 EMNLP 2011*, pages 216–226, Edinburgh, Scotland, UK., July.