A Novel Translation Framework Based on Rhetorical Structure Theory

Mei Tu **Yu Zhou** **Chengqing Zong**
National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences
{mtu,yzhou,cqzong}@nlpr.ia.ac.cn

Abstract

Rhetorical structure theory (RST) is widely used for discourse understanding, which represents a discourse as a hierarchically semantic structure. In this paper, we propose a novel translation framework with the help of RST. In our framework, the translation process mainly includes three steps: 1) **Source RST-tree acquisition**: a source sentence is parsed into an RST tree; 2) **Rule extraction**: translation rules are extracted from the source tree and the target string via bilingual word alignment; 3) **RST-based translation**: the source RST-tree is translated with translation rules. Experiments on Chinese-to-English show that our RST-based approach achieves improvements of 2.3/0.77/1.43 BLEU points on NIST04/NIST05/CWMT2008 respectively.

1 Introduction

For statistical machine translation (SMT), a crucial issue is how to build a translation model to extract as much accurate and generative translation knowledge as possible. The existing SMT models have made much progress. However, they still suffer from the bad performance of unnatural or even unreadable translation, especially when the sentences become complicated. We think the deep reason is that those models only extract translation information on lexical or syntactic level, but fail to give an overall understanding of source sentences on semantic level of discourse. In order to solve such problem, (Gong et al., 2011; Xiao et al., 2011; Wong and Kit, 2012) build discourse-based translation models to ensure the lexical coherence or consistency. Although some lexicons can be translated better by their models, the overall structure still remains unnatural. Marcu et al. (2000) design a discourse structure transferring module, but leave much work to do, especially on how to integrate this module into SMT and how to automatically analyze the structures. Those reasons urge us to seek a new translation framework under the idea of “translation with overall understanding”.

Rhetorical structure theory (RST) (Mann and Thompson, 1988) provides us with a good perspective and inspiration to build such a framework. Generally, an RST tree can explicitly show the minimal spans with semantic functional integrity, which are called elementary discourse units (*edu*s) (Marcu et al., 2000), and it also depicts the hierarchical relations among *edu*s. Furthermore, since different languages’ *edu*s are usually equivalent on semantic level, it is intuitive to create a new framework based on RST by directly mapping the source *edu*s to target ones.

Taking the Chinese-to-English translation as an example, our translation framework works as the following steps:

1) **Source RST-tree acquisition**: a source sentence is parsed into an RST-tree;

2) **Rule extraction**: translation rules are extracted from the source tree and the target string via bilingual word alignment;

3) **RST-based translation**: the source RST-tree is translated into target sentence with extracted translation rules.

Experiments on Chinese-to-English sentence-level discourses demonstrate that this method achieves significant improvements.

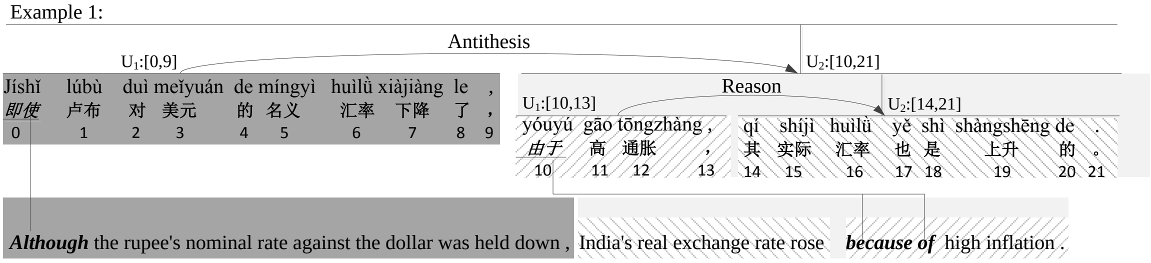
2 Chinese RST Parser

2.1 Annotation of Chinese RST Tree

Similar to (Soricut and Marcu, 2003), a node of RST tree is represented as a tuple *R*-[*s*, *m*, *e*], which means the relation *R* controls two semantic spans U_1 and U_2, U_1 starts from word position *s* and stops at word position *m*. U_2 starts from *m*+1 and ends with *e*. Under the guidance of definition of RST, Yue (2008) defined 12 groups[1] of

[1] They are *Parallel, Alternative, Condition, Reason, Elaboration, Means, Preparation, Enablement, Antithesis, Background, Evidences, Others.*

Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, pages 370–374,
Sofia, Bulgaria, August 4-9 2013. ©2013 Association for Computational Linguistics


Example 1:
Antithesis
U1:[0,9]
U2:[10,21]
Reason
U1:[10,13]
U2:[14,21]
Jíshǐ lúbù duì meǐyuán de míngyì huìlǜ xiàjiàng le ,
即使 卢布 对 美元 的 名义 汇率 下降 了 ，
0 1 2 3 4 5 6 7 8 9
yóuyú gāo tōngzhàng ,
由于 高 通胀 ，
10 11 12 13
qí shíjì huìlǜ yě shì shàngshēng de .
其 实际 汇率 也 是 上升 的 。
14 15 16 17 18 19 20 21
Although the rupee's nominal rate against the dollar was held down ,
India's real exchange rate rose
because of high inflation .

Cue-words pair matching set of cue words for span [0,9] and [10,21]:{即使/由于,即使/NULL,NULL/由于}

Cue-words pair matching set of cue words for span [10,13] and [14,21]:{由于/NULL}

RST-based Rules: *Antithesis*:: 即使[X]/[Y] => Although[X]/[Y] ; *Reason*::由于[X]/[Y] => [Y]/because of[X]

Figure 1: An example of Chinese RST tree and its word alignment of the corresponding English string.

rhetorical relations for Chinese particularly, upon which our Chinese RST parser is developed.

Figure 1 illustrates an example of Chinese RST tree and its alignment to the English string. There are two levels in this tree. The *Antithesis* relation controls U_1 from 0 to 9 and U_2 from 10 to 21. Thus it is written as *Antithesis*-[0,9,21]. Different shadow blocks denote the alignments of different *edu*s. Links between source and target words are alignments of cue words. Cue words are viewed as the strongest clues for rhetorical relation recognition and always found at the beginning of text (Reitter, 2003), such as “即使(although), 由于(because of)”. With the cue words included, the relations are much easier to be analyzed. So we focus on the explicit relations with cue words in this paper as our first try.

2.2 Bayesian Method for Chinese RST Parser

For Chinese RST parser, there are two tasks. One is the segmentation of *edu* and the other is the relation tagging between two semantic spans.

Feature	Meaning
$F_1(F_6)$	left(right) child is a syntactic sub-tree?
$F_2(F_5)$	left(right) child ends with a punctuation?
$F_3(F_4)$	cue words of left (right) child.
F_7	left and right children are sibling nodes?
$F_8(F_9)$	syntactic head symbol of left(right) child.

Table 1: 9 features used in our Bayesian model

Inspired by the features used in English RST parser (Soricut and Marcu, 2003; Reitter, 2003; Duverle and Prendinger, 2009; Hernault et al., 2010a), we design a Bayesian model to build a joint parser for segmentation and tagging simultaneously. In this model, 9 features in Table 1 are used. In the table, punctuations include comma, semicolons, period and question mark. We view explicit connectives as cue words in this paper.

Figure 2 illustrates the conditional independences of 9 features which are denoted with F_1~F_9.

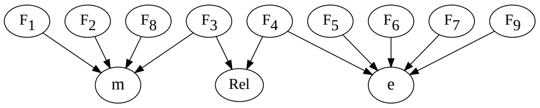


Figure 2: The graph for conditional independences of 9 features.

The segmentation and parsing conditional probabilities are computed as follows:

$$P(m|F_1^9) = P(m|F_1^3, F_8) \quad (1)$$

$$P(e|F_1^9) = P(e|F_4^7, F_9) \quad (2)$$

$$P(Rel|F_1^9) = P(Rel|F_3^4) \quad (3)$$

where F_n represents the n^{th} feature , F_n^l means features from n to l. Rel is short for relation. (1) and (2) describe the conditional probabilities of *m* and *e*. When using Formula (3) to predict the relation, we search all the cue-words pair, as shown in Figure 1, to get the best match. When training, we use maximum likelihood estimation to get all the associated probabilities. For decoding, the pseudo codes are given as below.

```
Nodes={[]}
Parser(0,End)
Parser(s,e): // recursive parser function
    if s > e or e is -1: return -1;
    m = GetMaxM(s,e)  //compute m through Formula(1);if no cue words found, then m=-1;
    e' = GetMaxE(s,m,e)  //compute e' through F (2);
    if m or e' equals to -1: return -1;
    Rel=GetRelation(s,m,e')  //compute relation by F (3)
    push [Rel,s,m,e'] into Nodes
    Parser(s,m)
    Parser(m+1,e')
    Parser(e'+1,e)
    Rel=GetRelation(s,e',e)
    push [Rel,s,e',e] into Nodes
    return e
```

For example in Figure 1, for the first iteration, *s*=0 and *m* will be chosen from {1-20}. We get *m*=9 through Formula (1). Then, similar with *m*, we get *e*=21 through Formula (2). Finally, the relation is figured out by Formula (3). Thus, a node is generated. A complete RST tree constructs until the end of the iterative process for this sentence. This method can run fast due to the simple greedy algorithm. It is plausible in our cases, because we only have a small scale of manually-annotated Chinese RST corpus, which prefers simple rather than complicated models.

3 Translation Model

3.1 Rule Extraction

As shown in Figure 1, the RST tree-to-string alignment provides us with two types of translation rules. One is common phrase-based rules, which are just like those in phrase-based model (Koehn et al., 2003). The other is RST tree-to-string rule, and it's defined as,

$$relation :: U_1(\alpha, X)/U_2(\gamma, Y) \\ \Rightarrow U_1(tr(\alpha), tr(X)) \sim U_2(tr(\gamma), tr(Y))$$

where the terminal characters α and γ represent the cue words which are optimum match for maximizing Formula (3). While the non-terminals *X* and *Y* represent the rest of the sequence. Function *tr*(•) means the translation of • . The operator ~ is an operator to indicate that the order of *tr(U_1)* and *tr*(U_2) is monotone or reverse. During rules' extraction, if the mean position of all the words in *tr*(U_1) precedes that in *tr*(U_2), ~ is monotone. Otherwise, ~ is reverse.

For example in Figure 1, the *Reason* relation controls U_1:[10,13] and U_2:[14,21]. Because the mean position of *tr*(U_2) is before that of *tr*(U_1), the reverse order is selected. We list the RST-based rules for Example 1 in Figure 1.

3.2 Probabilities Estimation

For the phrase-based translation rules, we use four common probabilities and the probabilities' estimation is the same with those in (Koehn et al., 2003). While the probabilities of RST-based translation rules are given as follows,

(1) $P(r_e|r_f, Rel) = \frac{Count(r_e, r_f, relation)}{Count(r_f, relation)}$: where r_e is the target side of the rule, ignorance of the order, i.e. $U_1(tr(\alpha), tr(X)) \sim U_2(tr(\gamma), tr(Y))$ with two directions, r_f is the source side, i.e. $U_1(\alpha, X)/U_2(\gamma, Y)$, and Rel means the relation type.

(2) $P(\tau|r_e, r_f, Rel) = \frac{Count(\tau, r_e, r_f, relation)}{Count(r_e, r_f, relation)}$: $\tau \in \{monotone, reverse\}$. It is the conditional probability of re-ordering.

4 Decoding

The decoding procedure of a discourse can be derived from the original decoding formula $e_1^I = argmax_{e_1^I} P(e_1^I|f_1^J)$. Given the rhetorical structure of a source sentence and the corresponding rule-table, the translating process is to find an optimal path to get the highest score under structure constrains, which is,

$$\text{argmax}_{e_s}\{P(e_s|, f_t)\} \\ = \text{argmax}_{e_s}\{\prod_{f_n \in f_t} P(e_{u1}, e_{u2}, \tau|f_n)\}$$

where f_t is a source RST tree combined by a set of node f_n. e_s is the target string combined by series of e_n (translations of f_n). f_n consists of U_1 and U_2. e_{u1} and e_{u2} are translations of U_1 and U_2 respectively. This global optimization problem is approximately simplified to local optimization to reduce the complexity,

$$\prod_{f_n \in f_t} \text{argmax}_{e_n}\{P(e_{u1}, e_{u2}, \tau|f_n)\}$$

In our paper, we have the following two ways to factorize the above formula,

Decoder 1:

$$P(e_{u1}, e_{u2}, \tau|f_n) \\ = P(e_{cp}, e_X, e_Y, \tau|f_{cp}, f_X, f_Y) \\ = P(e_{cp}|f_{cp})P(\tau|e_{cp}, f_{cp})P(e_X|f_X)P(e_Y|f_Y) \\ = P(r_e|r_f, Rel)P(\tau|r_e, r_f, Rel)P(e_X|f_X)P(e_Y|f_Y)$$

where e_X, e_Y are the translation of non-terminal parts. f_{cp} and e_{cp} are cue-words pair of source and target sides. The first and second factors are just the probabilities introduced in Section 3.2. After approximately simplified to local optimization, the final formulae are re-written as,

$$\text{argmax}_r\{P(r_e|r_f, Rel)P(\tau|r_e, r_f, Rel)\} \quad (4)$$

$$\text{argmax}_{e_X}\{P(e_X|f_X)\} \quad (5)$$

$$\text{argmax}_{e_Y}\{P(e_Y|f_Y)\} \quad (6)$$

Taking the source sentence with its RST tree in Figure 1 for instance, we adopt a bottom-up manner to do translation recursively. Suppose the best rules selected by (4) are just those written in the figure, Then span [11,13] and [14,21] are firstly translated by (5) and (6). Their translations are then re-packaged by the rule of *Reason*-[10,13,21]. Iteratively, the translations of span [1,9] and [10,21] are re-packaged by the rule of *Antithesis*-[0,9,21] to form the final translation.

Decoder 2 : Suppose that the translating process of two spans *U₁* and *U₂* are independent of each other, we rewrite $P(e_{u1}, e_{u2}, \tau | f_n)$ as follows,

$$\begin{aligned} & P(e_{u1}, e_{u2}, \tau | f_n) \\ & = P(e_{u1}, e_{u2}, \tau | f_{u1}, f_{u2}) \\ & = P(e_{u1}|f_{u1}) P(e_{u2}|f_{u2}) P(\tau | r_f, Rel) \\ & = P(e_{u1}|f_{u1}) P(e_{u2}|f_{u2}) \sum_{r_e} P(\tau | r_e, r_f, Rel) P(r_e | r_f, Rel) \end{aligned}$$

after approximately simplified to local optimization, the final formulae are re-written as below,

$$\operatorname{argmax}_{e_{u1}} \{Pr(e_{u1}|f_{u1})\} \quad (7)$$

$$\operatorname{argmax}_{e_{u2}} \{Pr(e_{u2}|f_{u2})\} \quad (8)$$

$$\operatorname{argmax}_{r} \{\sum_{e} Pr(\tau | r_e, r_f, Rel) Pr(r_e | r_f, Rel)\} \quad (9)$$

We also adopt the bottom-up manner similar to Decoder 1. In Figure 1, *U₁* and *U₂* of *Reason* node are firstly translated. Their translations are then re-ordered. Then the translations of two spans of *Antithesis* node are re-ordered and constructed into the final translation. In Decoder 2, the minimal translation-unit is *edu*. While in Decoder 1, an *edu* is further split into cue-word part and the rest part to obtain the respective translation.

In our decoders, language model(LM) is used for translating *edu*s in Formula(5),(6),(7),(8), but not for reordering the upper spans because with the bottom-to-up combination, the spans become longer and harder to be judged by a traditional language model. So we only use RST rules to guide the reordering. But LM will be properly considered in our future work.

5 Experiment

5.1 Setup

In order to do Chinese RST parser, we annotated over 1,000 complicated sentences on CTB (Xue et al., 2005), among which 1,107 sentences are used for training, and 500 sentences are used for testing. Berkeley parser[2] is used for getting the syntactic trees.

The translation experiment is conducted on Chinese-to-English direction. The bilingual training data is from the LDC corpus[3]. The training corpus contains 2.1M sentence pairs. We obtain the word alignment with the grow-diag-final-and strategy by GIZA++[4]. A 5-gram language model is trained on the Xinhua portion of the English Gigaword corpus. For tuning and testing, we use NIST03 evaluation data as the development set, and extract the relatively long and complicated sentences from NIST04, NIST05 and CWMT08[5] evaluation data as the test set. The number and average word-length of sentences are 511/36, 320/34, 590/38 respectively. We use case-insensitive BLEU-4 with the shortest length penalty for evaluation.

To create the baseline system, we use the toolkit Moses[6] to build a phrase-based translation system. Meanwhile, considering that Xiong et al. (2009) have presented good results by dividing long and complicated sentences into sub-sentences only by punctuations during decoding, we re-implement their method for comparison.

5.2 Results of Chinese RST Parser

Table 2 shows the results of RST parsing. On average, our RS trees are 2 layers deep. The parsing errors mostly result from the segmentation errors, which are mainly caused by syntactic parsing errors. On the other hand, the polysemous cue words, such as “而(but, and, thus)” may lead ambiguity for relation recognition, because they can be clues for different relations.

Task	**Precision**	**Recall**	**F1**
Segmentation	0.74	0.83	0.78
Labeling	0.71	0.78	0.75

Table 2: Segmentation and labeling result.

5.3 Results of Translation

Table 3 presents the translation comparison results. In this table, XD represents the method in (Xiong et al., 2009). D1 stands for Decoder-1, and D2 for Decoder-2. Values with boldface are the highest scores in comparison. D2 performs best on the test data with 2.3/0.77/1.43/1.16 points. Compared with XD, our results also outperform by 0.52 points on the whole test data.

Observing and comparing the translation results, we find that our translation results are more readable by maintaining the semantic integrality of the *edu*s and by giving more appreciate reorganization of the translated *edu*s.

Testing Set	**Baseline**	**XD**	**D1**	**D2**
NIST04	29.39	31.52	31.34	**31.69**
NIST05	29.86	29.80	30.28	**30.63**
CWMT08	24.31	25.24	**25.74**	**25.74**
ALL	27.85	28.49	28.66	**29.01**

Table 3: Comparison with related models.

[2] http://code.google.com/p/berkeleyparser/

[3] LDC category number : LDC2000T50, LDC2002E18, LDC2003E07, LDC2004T07, LDC2005T06, LDC2002L27, LDC2005T10 and LDC2005T34

[4] http://code.google.com/p/giza-pp/

[5] China Workshop on Machine Translation 2008

[6] www.statmt.org/moses/index.php?n=Main.HomePage

6 Conclusion and Future Work

In this paper, we present an RST-based translation framework for modeling semantic structures in translation model, so as to maintain the semantically functional integrity and hierarchical relations of *edu*s during translating. With respect to the existing models, we think our translation framework works more similarly to what human does, and we believe that this research is a crucial step towards discourse-oriented translation.

In the next step, we will study on the implicit discourse relations for Chinese and further modify the RST-based framework. Besides, we will try to combine other current translation models such as syntactic model and hierarchical model into our framework. Furthermore, the more accurate evaluation metric for discourse-oriented translation will be further studied.

Acknowledgments

The research work has been funded by the Hi-Tech Research and Development Program (“863” Program) of China under Grant No. 2011AA01A207, 2012AA011101, and 2012AA011102 and also supported by the Key Project of Knowledge Innovation Program of Chinese Academy of Sciences under Grant No.KGZD-EW-501.

References

David A Duverle and Helmut Prendinger. 2009. A novel discourse parser based on support vector machine classification. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*: Volume 2-Volume 2, pages 665–673. Association for Computational Linguistics.

Zhengxian Gong, Min Zhang, and Guodong Zhou. 2011. Cache-based document-level statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 909–919. Association for Computational Linguistics.

Hugo Hernault, Danushka Bollegala, and Mitsuru Ishizuka. 2010a. A sequential model for discourse segmentation. *Computational Linguistics and Intelligent Text Processing,* pages 315–326.

Hugo Hernault, Helmut Prendinger, Mitsuru Ishizuka, et al. 2010b. Hilda: A discourse parser using support vector machine classification. *Dialogue & Discourse*, 1(3).

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. Association for Computational Linguistics.

William C Mann and Sandra A Thompson. 1986. Rhetorical structure theory: Description and construction of text structures. *Technical report, DTIC Document.*

William C Mann and Sandra A Thompson. 1987. Rhetorical structure theory: A framework for the analysis of texts. *Technical report, DTIC Document*.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Daniel Marcu, Lynn Carlson, and Maki Watanabe. 2000. The automatic translation of discourse structures. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, pages 9–17. Morgan Kaufmann Publishers Inc.

David Reitter. 2003. Simple signals for complex rhetorics: On rhetorical analysis with rich-feature support vector models. *Language*, 18:52.

Radu Soricut and Daniel Marcu. 2003. Sentence level discourse parsing using syntactic and lexical information. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*-Volume 1, pages 149–156. Association for Computational Linguistics.

Billy TM Wong and Chunyu Kit. 2012. Extending machine translation evaluation metrics with lexical cohesion to document level. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, page 1060–1068. Association for Computational Linguistics.

Tong Xiao, Jingbo Zhu, Shujie Yao, and Hao Zhang. 2011. Document-level consistency verification in machine translation. In *Machine Translation Summit*, volume 13, pages 131–138.

Hao Xiong, Wenwen Xu, Haitao Mi, Yang Liu, and Qun Liu. 2009. Sub-sentence division for tree-based machine translation. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 137–140. Association for Computational Linguistics.

Naiwen Xue, Fei Xia, Fu-Dong Chiou, and Marta Palmer. 2005. The Penn Chinese treebank: Phrase structure annotation of a large corpus. *Natural Language Engineering*, 11(2):207.

Ming Yue. 2008. Rhetorical structure annotation of Chinese news commentaries. *Journal of Chinese Information Processing*, 4:002.