A Hybrid Approach to Skeleton-based Translation

Tong Xiao†‡, Jingbo Zhu†‡, Chunliang Zhang†‡
† Northeastern University, Shenyang 110819, China
‡ Hangzhou YaTuo Company, 358 Wener Rd., Hangzhou 310012, China
{xiaotong,zhujingbo,zhangcl}@mail.neu.edu.cn

Abstract

In this paper we explicitly consider sentence skeleton information for Machine Translation (MT). The basic idea is that we translate the key elements of the input sentence using a skeleton translation model, and then cover the remain segments using a full translation model. We apply our approach to a state-of-the-art phrase-based system and demonstrate very promising BLEU improvements and TER reductions on the NIST Chinese-English MT evaluation data.

1 Introduction

Current Statistical Machine Translation (SMT) approaches model the translation problem as a process of generating a derivation of atomic translation units, assuming that every unit is drawn out of the same model. The simplest of these is the phrase-based approach (Och et al., 1999; Koehn et al., 2003) which employs a global model to process any sub-strings of the input sentence. In this way, all we need is to increasingly translate a sequence of source words each time until the entire sentence is covered. Despite good results in many tasks, such a method ignores the roles of each source word and is somewhat different from the way used by translators. For example, an important-first strategy is generally adopted in human translation - we translate the key elements/structures (or skeleton) of the sentence first, and then translate the remaining parts. This especially makes sense for some languages, such as Chinese, where complex structures are usually involved.

Note that the source-language structural information has been intensively investigated in recent studies of syntactic translation models. Some of them developed syntax-based models on complete syntactic trees with Treebank annotations (Liu et al., 2006; Huang et al., 2006; Zhang et al., 2008), and others used source-language syntax as soft constraints (Marton and Resnik, 2008; Chiang, 2010). However, these approaches suffer from the same problem as the phrase-based counterpart and use the single global model to handle different translation units, no matter they are from the skeleton of the input tree/sentence or other not-so-important sub-structures.

In this paper we instead explicitly model the translation problem with sentence skeleton information. In particular,

- We develop a skeleton-based model which divides translation into two sub-models: a skeleton translation model (i.e., translating the key elements) and a full translation model (i.e., translating the remaining source words and generating the complete translation).
- We develop a skeletal language model to describe the possibility of translation skeleton and handle some of the long-distance word dependencies.
- We apply the proposed model to Chinese-English phrase-based MT and demonstrate promising BLEU improvements and TER reductions on the NIST evaluation data.

2 A Skeleton-based Approach to MT

2.1 Skeleton Identification

The first issue that arises is how to identify the skeleton for a given source sentence. Many ways are available. E.g., we can start with a full syntactic tree and transform it into a simpler form (e.g., removing a sub-tree). Here we choose a simple and straightforward method: a skeleton is obtained by dropping all unimportant words in the original sentence, while preserving the grammaticality. See the following for an example skeleton of a Chinese sentence.

Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Short Papers), pages 563–568,
Baltimore, Maryland, USA, June 23-25 2014. ©2014 Association for Computational Linguistics

Original Sentence (subscripts represent indices):

每[1] 吨[2] 海水淡化[3] 处理[4] 的[5]
per ton seawater desalination treatment of

成本[6] 在[7] 5[8] 元[9] 的[10] 基础[11] 上[12]
the cost 5 yuan of from

进一步[13] 下降[14] 。[15]
has been further reduced .

(The cost of seawater desalination treatment has been further reduced from 5 yuan per ton.)

Sentence Skeleton (subscripts represent indices):

成本[6] 进一步[13] 下降[14] 。[15]
the cost has been further reduced .

(The cost has been further reduced.)

Obviously the skeleton used in this work can be viewed as a simplified sentence. Thus the problem is in principle the same as sentence simplification/compression. The motivations of defining the problem in this way are two-fold. First, as the skeleton is a well-formed (but simple) sentence, all current MT approaches are applicable to the skeleton translation problem. Second, obtaining simplified sentences by word deletion is a well-studied issue (Knight and Marcu, 2000; Clarke and Lapata, 2006; Galley and McKeown, 2007; Cohn and Lapata, 2008; Yamangil and Shieber, 2010; Yoshikawa et al., 2012). Many good sentence simpliciation/compression methods are available to our work. Due to the lack of space, we do not go deep into this problem. In Section 3.1 we describe the corpus and system employed for automatic generation of sentence skeletons.

2.2 Base Model

Next we describe our approach to integrating skeleton information into MT models. We start with an assumption that the 1-best skeleton is provided by the skeleton identification system. Then we define skeleton-based translation as a task of searching for the best target string $\hat{t}$ given the source string and its skeleton τ:

$$\hat{t} = \arg\max_{t} \mathrm{P}(t|\tau, s) \quad (1)$$

As is standard in SMT, we further assume that 1) the translation process can be decomposed into a derivation of phrase-pairs (for phrase-based models) or translation rules (for syntax-based models); 2) and a linear function $g(\cdot)$ is used to assign a model score to each derivation. Let $d_{s,\tau,t}$ (or d for short) denote a translation derivation. The above problem can be redefined in a Viterbi fashion - we find the derivation $\hat{d}$ with the highest model score given s and τ:

$$\hat{d} = \arg\max_{d} g(d) \quad (2)$$

In this way, the MT output can be regarded as the target-string encoded in $\hat{d}$.

To compute $g(d)$, we use a linear combination of a skeleton translation model $g_{skel}(d)$ and a full translation model $g_{full}(d)$:

$$g(d) = g_{skel}(d) + g_{full}(d) \quad (3)$$

where the skeleton translation model handles the translation of the sentence skeleton, while the full translation model is the baseline model and handles the original problem of translating the whole sentence. The motivation here is straightforward: we use an additional score $g_{skel}(d)$ to model the problem of skeleton translation and interpolate it with the baseline model. See Figure 1 for an example of applying the above model to phrase-based MT. In the figure, each source phrase is translated into a target phrase, which is represented by linked rectangles. The skeleton translation model focuses on the translation of the sentence skeleton, i.e., the solid (red) rectangles; while the full translation model computes the model score for all those phrase-pairs, i.e., all solid and dashed rectangles.

Another note on the model. Eq. (3) provides a very flexible way for model selection. While we will restrict ourself to phrase-based translation in the following description and experiments, we can choose different models/features for $g_{skel}(d)$ and $g_{full}(d)$. E.g., one may introduce syntactic features into $g_{skel}(d)$ due to their good ability in capturing structural information; and employ a standard phrase-based model for $g_{full}(d)$ in which not all segments of the sentence need to respect syntactic constraints.

2.3 Model Score Computation

In this work both the skeleton translation model $g_{skel}(d)$ and full translation model $g_{full}(d)$ resemble the usual forms used in phrase-based MT, i.e., the model score is computed by a linear combination of a group of phrase-based features and language models. In phrase-based MT, the translation problem is modeled by a derivation of phrase-pairs. Given a translation model m, a language model lm and a vector of feature weights $\mathbf{w}$, the model score of a derivation d is computed by

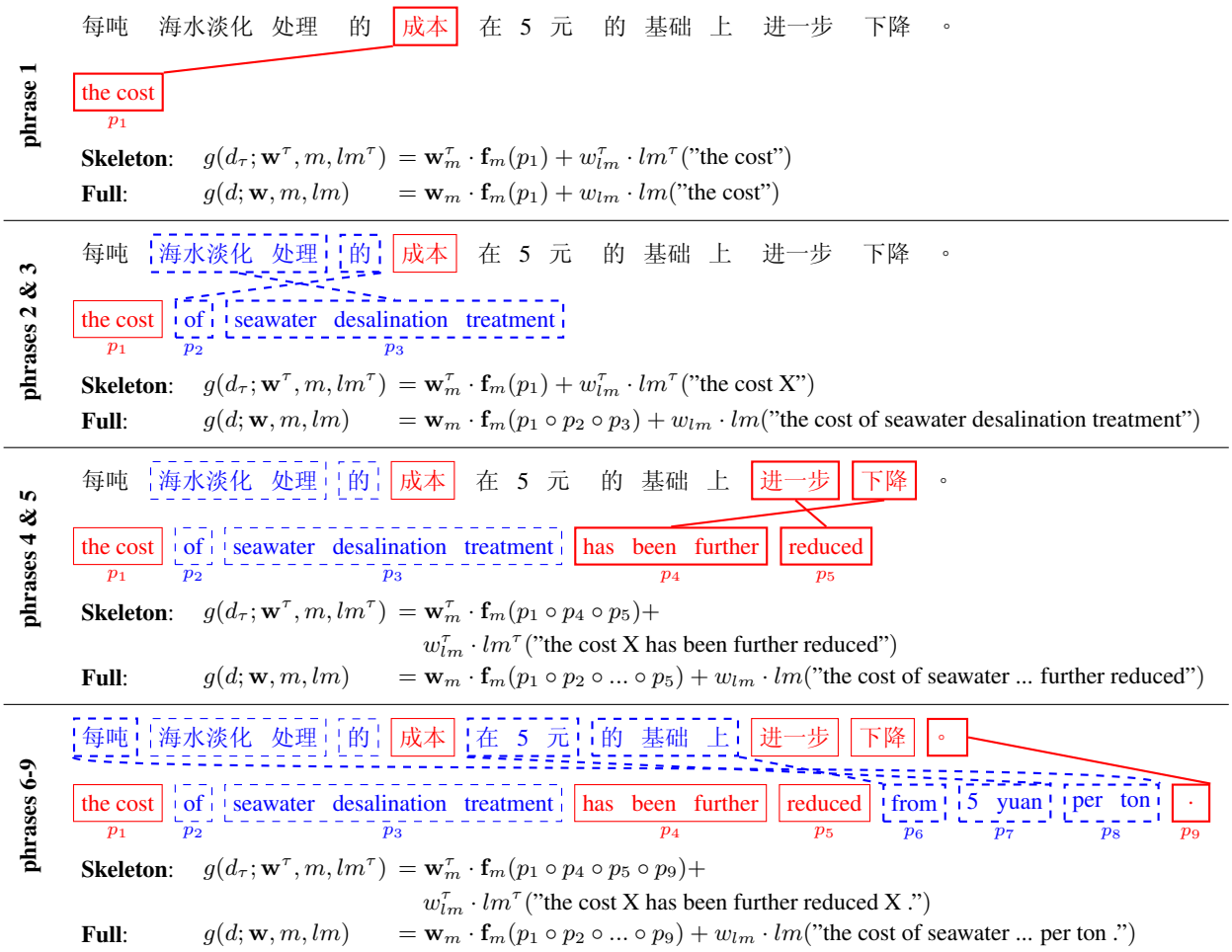

Figure 1: Example derivation and model scores for a sentence in LDC2006E38. The solid (red) rectangles represent the sentence skeleton, and the dashed (blue) rectangles represent the non-skeleton segments. X represents a slot in the translation skeleton. ∘ represents composition of phrase-pairs.

$$g(d; \mathbf{w}, m, lm) = \mathbf{w}_m \cdot \mathbf{f}_m(d) + w_{lm} \cdot lm(d) \quad (4)$$

where $\mathbf{f}_m(d)$ is a vector of feature values defined on d, and $\mathbf{w}_m$ is the corresponding weight vector. $lm(d)$ and w_{lm} are the score and weight of the language model, respectively.

To ease modeling, we only consider *skeleton-consistent* derivations in this work. A derivation d is skeleton-consistent if no phrases in d cross skeleton boundaries (e.g., a phrase where two of the source words are in the skeleton and one is outside). Obviously, from any skeleton-consistent derivation d we can extract a skeleton derivation d_τ which covers the sentence skeleton exactly. For example, in Figure 1, the derivation of phrase-pairs $\{p_1, p_2, ..., p_9\}$ is skeleton-consistent, and the skeleton derivation is formed by $\{p_1, p_4, p_5, p_9\}$.

Then, we can simply define $g_{skel}(d)$ and $g_{full}(d)$ as the model scores of d_τ and d:

$$g_{skel}(d) \triangleq g(d_\tau; \mathbf{w}^\tau, m, lm^\tau) \quad (5)$$
$$g_{full}(d) \triangleq g(d; \mathbf{w}, m, lm) \quad (6)$$

This model makes the skeleton translation and full translation much simpler because they perform in the same way of string translation in phrase-based MT. Both $g_{skel}(d)$ and $g_{full}(d)$ share the same translation model m which can easily learned from the bilingual data[1]. On the other hand, it has different feature weight vectors for individual models (i.e., $\mathbf{w}$ and $\mathbf{w}^\tau$).

For language modeling, lm is the standard n-gram language model adopted in the baseline system. lm^τ is a skeletal language for estimating the well-formedness of the translation skeleton. Here a translation skeleton is a target string where all segments of non-skeleton translation are generalized to a symbol X. E.g., in Figure 1, the trans-

[1]In $g_{skel}(d)$, we compute the reordering model score on the skeleton though it is learned from the full sentences. In this way the reordering problems in skeleton translation and full translation are distinguished and handled separately.

lation skeleton is '*the cost X has been further reduced X* .', where two Xs represent non-skeleton segments in the translation. In such a way of string representation, the skeletal language model can be implemented as a standard n-gram language model, that is, a string probability is calculated by a product of a sequence of n-gram probabilities (involving normal words and X). To learn the skeletal language model, we replace non-skeleton parts of the target sentences in the bilingual corpus to Xs using the source sentence skeletons and word alignments. The skeletal language model is then trained on these generalized strings in a standard way of n-gram language modeling.

By substituting Eq. (4) into Eqs. (5) and (6), and then Eqs. (3) and (2), we have the final model used in this work:

$$\hat{d} = \arg\max_{d} \Big(\mathbf{w}_m \cdot \mathbf{f}_m(d) + w_{lm} \cdot lm(d) + \mathbf{w}_m^{\tau} \cdot \mathbf{f}_m(d_\tau) + w_{lm}^{\tau} \cdot lm^{\tau}(d_\tau) \Big) \quad (7)$$

Figure 1 shows the translation process and associated model scores for the example sentence. Note that this method does not require any new translation models for implementation. Given a baseline phrase-based system, all we need is to learn the feature weights $\mathbf{w}$ and $\mathbf{w}^{\tau}$ on the development set (with source-language skeleton annotation) and the skeletal language model lm^{τ} on the target-language side of the bilingual corpus. To implement Eq. (7), we can perform standard decoding while "doubly weighting" the phrases which cover a skeletal section of the sentence, and combining the two language models and the translation model in a linear fashion.

3 Evaluation

3.1 Experimental Setup

We experimented with our approach on Chinese-English translation using the NiuTrans open-source MT toolkit (Xiao et al., 2012). Our bilingual corpus consists of 2.7M sentence pairs. All these sentences were aligned in word level using the GIZA++ system and the "grow-diag-final-and" heuristics. A 5-gram language model was trained on the Xinhua portion of the English Gigaword corpus in addition to the target-side of the bilingual data. This language model was used in both the baseline and our improved systems. For our skeletal language model, we trained a 5-gram language model on the target-side of the bilingual data by generalizing non-skeleton segments to Xs. We used the newswire portion of the NIST MT06 evaluation data as our development set, and used the evaluation data of MT04 and MT05 as our test sets. We chose the default feature set of the NiuTrans.Phrase engine for building the baseline, including phrase translation probabilities, lexical weights, a 5-gram language model, word and phrase bonuses, a ME-based lexicalized reordering model. All feature weights were learned using minimum error rate training (Och, 2003).

Our skeleton identification system was built using the t3 toolkit[2] which implements a state-of-the-art sentence simplification system. We used the NEU Chinese sentence simplification (NEUCSS) corpus as our training data (Zhang et al., 2013). It contains the annotation of sentence skeleton on the Chinese-language side of the Penn Parallel Chinese-English Treebank (LDC2003E07). We trained our system using the Parts 1-8 of the NEUCSS corpus and obtained a 65.2% relational F1 score and 63.1% compression rate in held-out test (Part 10). For comparison, we also manually annotated the MT development and test data with skeleton information according to the annotation standard provided within NEUCSS.

3.2 Results

Table 1 shows the case-insensitive IBM-version BLEU and TER scores of different systems. We see, first of all, that the MT system benefits from our approach in most cases. In both the manual and automatic identification of sentence skeleton (rows 2 and 4), there is a significant improvement on the "All" data set. However, using different skeleton identification results for training and inference (row 3) does not show big improvements due to the data inconsistency problem.

Another interesting question is whether the skeletal language model really contributes to the improvements. To investigate it, we removed the skeletal language model from our skeleton-based translation system (with automatic skeleton identification on both the development and test sets). Seen from row $-lm^{\tau}$ of Table 1, the removal of the skeletal language model results in a significant drop in both BLEU and TER performance. It indicates that this language model is very beneficial to our system. For comparison, we removed

[2] http://staffwww.dcs.shef.ac.uk/people/T.Cohn/t3/

Entry			MT06 (Dev)		MT04		MT05		All	
system	dev-skel	test-skel	BLEU	TER	BLEU	TER	BLEU	TER	BLEU	TER
baseline	-	-	35.06	60.54	38.53	61.15	34.32	62.82	36.64	61.54
SBMT	manual	manual	**35.71**	**59.60**	38.99	**60.67**	**35.35**	**61.60**	**37.30**	**60.73**
SBMT	manual	auto	**35.72**	**59.62**	38.75	61.16	35.02	**62.20**	37.03	61.19
SBMT	auto	auto	35.57	**59.66**	**39.21**	**60.59**	**35.29**	**61.89**	**37.33**	**60.80**
$-lm^\tau$	auto	auto	35.23	**60.17**	38.86	60.78	34.82	**62.46**	36.99	61.16
$-m^\tau$	auto	auto	35.50	**59.69**	39.00	**60.69**	**35.10**	**62.03**	37.12	**60.90**
s-space	-	-	35.00	60.50	38.39	61.20	34.33	62.90	36.57	61.58
s-feat.	-	-	35.16	60.50	38.60	61.17	34.25	62.88	36.70	61.58

Table 1: BLEU4[%] and TER[%] scores of different systems. Boldface means a significant improvement ($p < 0.05$). SBMT means our skeleton-based MT system. $-lm^\tau$ (or $-m^\tau$) means that we remove the skeletal language model (or translation model) from our proposed approach. s-space means that we restrict the baseline system to the search space of skeleton-consistent derivations. s-feat. means that we introduce an indicator feature for skeleton-consistent derivations into the baseline system.

the skeleton-based translation model from our system as well. Row $-m^\tau$ of Table 1 shows that the skeleton-based translation model can contribute to the overall improvement but there is no big differences between baseline and $-m^\tau$.

Apart from showing the effects of the skeleton-based model, we also studied the behavior of the MT system under the different settings of search space. Row s-space of Table 1 shows the BLEU and TER results of restricting the baseline system to the space of skeleton-consistent derivations, i.e., we remove both the skeleton-based translation model and language model from the SBMT system. We see that the limited search space is a little harmful to the baseline system. Further, we regarded skeleton-consistent derivations as an indicator feature and introduced it into the baseline system. Seen from row s-feat., this feature does not show promising improvements. These results indicate that the real improvements are due to the skeleton-based model/features used in this work, rather than the "well-formed" derivations.

4 Related Work

Skeleton is a concept that has been used in several sub-areas in MT for years. For example, in confusion network-based system combination it refers to the backbone hypothesis for building confusion networks (Rosti et al., 2007; Rosti et al., 2008); Liu et al. (2011) regard skeleton as a shortened sentence after removing some of the function words for better word deletion. In contrast, we define sentence skeleton as the key segments of a sentence and develop a new MT approach based on this information.

There are some previous studies on the use of sentence skeleton or related information in MT (Mellebeek et al., 2006a; Mellebeek et al., 2006b; Owczarzak et al., 2006). In spite of their good ideas of using skeleton skeleton information, they did not model the skeleton-based translation problem in modern SMT pipelines. Our work is a further step towards the use of sentence skeleton in MT. More importantly, we develop a complete approach to this issue and show its effectiveness in a state-of-the-art MT system.

5 Conclusion and Future Work

We have presented a simple but effective approach to integrating the sentence skeleton information into a phrase-based system. The experimental results show that the proposed approach achieves very promising BLEU improvements and TER reductions on the NIST evaluation data. In our future work we plan to investigate methods of integrating both syntactic models (for skeleton translation) and phrasal models (for full translation) in our system. We also plan to study sophisticated reordering models for skeleton translation, rather than reusing the baseline reordering model which is learned on the full sentences.

Acknowledgements

This work was supported in part by the National Science Foundation of China (Grants 61272376 and 61300097), and the China Postdoctoral Science Foundation (Grant 2013M530131). The authors would like to thank the anonymous reviewers for their pertinent and insightful comments.

References

David Chiang. 2010. Learning to Translate with Source and Target Syntax. In *Proc. of* ACL 2010, pages 1443-1452.

James Clarke and Mirella Lapata. 2006. Models for Sentence Compression: A Comparison across Domains, Training Requirements and Evaluation Measures. In *Proc. of* ACL/COLING 2006, pages 377-384.

Trevor Cohn and Mirella Lapata. 2008. Sentence Compression Beyond Word Deletion. In *Proc. of* COLING 2008, pages 137-144.

Jason Eisner. 2003. Learning Non-Isomorphic Tree Mappings for Machine Translation. In *Proc. of* ACL 2003, pages 205-208.

Michel Galley and Kathleen McKeown. 2007. Lexicalized Markov Grammars for Sentence Compression. In *Proc. of* HLT:NAACL 2007, pages 180-187.

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

Kevin Knight and Daniel Marcu. 2000. Statistical-based summarization-step one: sentence compression. In *Proc. of* AAAI 2000, pages 703-710.

Philipp Koehn, Franz J. Och and Daniel Marcu. 2003. Statistical Phrase-Based Translation. In *Proc. of* NAACL 2003, pages 48-54.

Yang Liu, Qun Liu and Shouxun Lin. 2006. Tree-to-String Alignment Template for Statistical Machine Translation. In *Proc. of* ACL/COLING 2006, pages 609-616.

Shujie Liu, Chi-Ho Li and Ming Zhou. 2011. Statistic Machine Translation Boosted with Spurious Word Deletion. In *Proc. of* Machine Translation Summit XIII, pages 72-79.

Yuval Marton and Philip Resnik. 2008. Soft Syntactic Constraints for Hierarchical Phrased-Based Translation. In *Proc. of* ACL:HLT 2008, pages 1003-1011.

Bart Mellebeek, Karolina Owczarzak, Josef van Genabith and Andy Way. 2006. Multi-Engine Machine Translation by Recursive Sentence Decomposition. In *Proc. of* AMTA 2006, pages 110-118.

Bart Mellebeek, Karolina Owczarzak, Declan Groves, Josef Van Genabith and Andy Way. 2006. A Syntactic Skeleton for Statistical Machine Translation. In *Proc. of* EAMT 2006, pages 195-202.

Franz J. Och, Christoph Tillmann and Hermann Ney. 1999. Improved Alignment Models for Statistical Machine Translation. In *Proc. of* EMNLP/VLC 1999, pages 20-28.

Franz J. Och. 2003. Minimum error rate training in statistical machine translation. In *Proc. of* ACL 2003, pages 160-167.

Karolina Owczarzak, Bart Mellebeek, Declan Groves, Josef van Genabith and Andy Way. 2006. Wrapper Syntax for Example-Based Machine Translation. In *Proc. of* AMTA2006, pages 148-155.

Antti-Veikko I. Rosti, Spyros Matsoukas and Richard Schwartz. 2007. Improved Word-Level System Combination for Machine Translation. In *Proc. of* ACL 2007, pages 312-319.

Antti-Veikko I. Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. 2008. Incremental hypothesis alignment for building confusion networks with application to machine translation system combination. In *Proc. of* Third Workshop on Statistical Machine Translation, pages 183 - 186.

Tong Xiao, Jingbo Zhu, Hao Zhang and Qiang Li 2012. NiuTrans: An Open Source Toolkit for Phrase-based and Syntax-based Machine Translation. In *Proc. of* ACL 2012, system demonstrations, pages 19-24.

Elif Yamangil and Stuart M. Shieber. 2010. Bayesian Synchronous Tree-Substitution Grammar Induction and Its Application to Sentence Compression. In *Proc. of* ACL 2010, pages 937-947.

Katsumasa Yoshikawa, Ryu Iida, Tsutomu Hirao and Manabu Okumura. 2012. Sentence Compression with Semantic Role Constraints. In *Proc. of* ACL 2012, pages 349-353.

Min Zhang, Hongfei Jiang, Aiti Aw, Haizhou Li, Chew Lim Tan and Sheng Li. 2008. A Tree Sequence Alignment-based Tree-to-Tree Translation Model. In *Proc. of* ACL:HLT 2008, pages 559-567.

Chunliang Zhang, Minghan Hu, Tong Xiao, Xue Jiang, Lixin Shi and Jingbo Zhu. 2013. Chinese Sentence Compression: Corpus and Evaluation. In *Proc. of* Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data, pages 257-267.