PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 98 OCTOBER 2012 37–50

Hierarchical Phrase-Based Translation with Jane 2

Matthias Huck, Jan-Thorsten Peter, Markus Freitag, Stephan Peitz, Hermann Ney

Human Language Technology and Pattern Recognition Group, RWTH Aachen University

Abstract

In this paper, we give a survey of several recent extensions to hierarchical phrase-based machine translation that have been implemented in version 2 of Jane, RWTH's open source statistical machine translation toolkit. We focus on the following techniques: Insertion and deletion models, lexical scoring variants, reordering extensions with non-lexicalized reordering rules and with a discriminative lexicalized reordering model, and soft string-to-dependency hierarchical machine translation. We describe the fundamentals of each of these techniques and present experimental results obtained with Jane 2 to confirm their usefulness in state-of-the-art hierarchical phrase-based translation (HPBT).

1. Introduction

Jane (Vilar et al., 2010a) is an open source translation toolkit which has been developed at RWTH Aachen University and is freely available for non-commercial use. Jane provides efficient C++ implementations for hierarchical phrase extraction, optimization of log-linear feature weights, and parsing-based search algorithms. A modular design and flexible extension mechanisms allow for easy integration of novel features and translation approaches.

In hierarchical phrase-based translation (Chiang, 2005, 2007), a weighted synchronous context-free grammar is induced from parallel text. In addition to contiguous *lexical* phrases, *hierarchical* phrases with usually up to two gaps are extracted. Hierarchical decoding is carried out with a search procedure which is based on CYK+ parsing (Chappelier and Rajman, 1998). Standard features that are typically inte-

Corresponding author: huck@cs.rwth-aachen.de
Cite as: Matthias Huck, Jan-Thorsten Peter, Markus Freitag, Stephan Peitz, Hermann Ney. Hierarchical Phrase-Based Translation with Jane 2. The Prague Bulletin of Mathematical Linguistics No. 98, 2012, pp. 37–50.
doi: 10.2478/v10108-012-0007-8.

grated into hierarchical baseline setups are: phrase translation probabilities and lexical smoothing probabilities, each in both source-to-target and target-to-source translation directions, word and phrase penalty, binary features marking hierarchical phrases, glue rule, and rules with non-terminals at the boundaries, and an n-gram language model. Other common and simple features are source-to-target and target-to-source phrase length ratios and binary features marking phrases that have been seen more than a certain number of times—one, two, three or five times, for instance—in the training data.

Jane additionaly implements a number of advanced techniques. These range from discriminative word lexicon (DWL models and triplet lexicon models (Mauser et al., 2009; Huck et al., 2010) over syntactic enhancements like parse matching (Vilar et al., 2008), preference grammars (Venugopal et al., 2009; Stein et al., 2010) and pseudo-syntactic enhancements like poor man's syntax (Vilar et al., 2010b) to a variety of search strategies with diverse pruning approaches and language model (LM) score estimation heuristics (Huang and Chiang, 2007; Vilar and Ney, 2009, 2011). Log-linear parameter weights can be optimized with either the downhill simplex algorithm (Nelder and Mead, 1965), Och's minimum error rate training (MERT) (Och, 2003), or the margin infused relaxed algorithm (MIRA) (Chiang et al., 2009).

The purpose of this paper is to present some features that have been added to Jane in version 2, namely insertion and deletion models (Section 2), lexical scoring variants (Section 3), reordering extensions (Section 4), and soft string-to-dependency features (Section 5). We will not address Jane's basic functionality or any other non-standard techniques that are available in Jane. Many of them have been discussed in depth in previous publications (Stein et al., 2011; Vilar et al., 2012). We refer the reader to those and to the manual included in the Jane package. Advice on how to employ most of the features implemented in Jane can likewise be found in the manual. Jane 2 is available for download at `http://www.hltpr.rwth-aachen.de/jane/`.

1.1. Notational Conventions

In hierarchical phrase-based translation, we deal with rules $X \rightarrow \langle \alpha, \beta, \sim \rangle$ where $\langle \alpha, \beta \rangle$ is a bilingual phrase pair that may contain symbols from a non-terminal set, i.e. $\alpha \in (\mathcal{N} \cup V_F)^+$ and $\beta \in (\mathcal{N} \cup V_E)^+$, where V_F and V_E are the source and target vocabulary, respectively, and $\mathcal{N}$ is a non-terminal set which is shared by source and target. The left-hand side of the rule is a non-terminal symbol $X \in \mathcal{N}$, and the $\sim$ relation denotes a one-to-one correspondence between the non-terminals in α and in β. Let J_α denote the number of terminal symbols in α and I_β the number of terminal symbols in β. Indexing α with j, i.e. the symbol α_j, $1 \leq j \leq J_\alpha$, denotes the j-th terminal symbol on the source side of the phrase pair $\langle \alpha, \beta \rangle$, and analogous with β_i, $1 \leq i \leq I_\beta$, on the target side.

2. Insertion and Deletion Models

Insertion and deletion models are designed as a means to avoid the omission of content words in the hypotheses. In our case, they are implemented as phrase-level feature functions which count the number of inserted or deleted words (Huck and Ney, 2012). An English word is considered inserted or deleted based on lexical probabilities with the words on the foreign language side of the phrase. Lexical translation probabilities from different types of lexicon models may be employed within the insertion and deletion scoring functions, e.g. a model which is extracted from word-aligned training data and—given the word alignment matrix—relies on pure relative frequencies (henceforth denoted as *RF word lexicon*) (Koehn et al., 2003), or the IBM model 1 lexicon (henceforth denoted as *IBM-1*) (Brown et al., 1993).

We define insertion and deletion models, each in both source-to-target and target-to-source direction, by giving phrase-level scoring functions for the features. In the Jane 2 implementation, the feature values are precomputed and written to the phrase table. The features are then incorporated directly into the log-linear model combination of the decoder.

2.1. Insertion Models

The insertion model in source-to-target direction $t_{s2tIns}(\cdot)$ counts the number of inserted words on the target side β of a hierarchical rule with respect to the source side α of the rule:

$$t_{s2tIns}(\alpha, \beta) = \sum_{i=1}^{I_\beta} \prod_{j=1}^{J_\alpha} \left[p(\beta_i|\alpha_j) < \tau_{\alpha_j} \right] \tag{1}$$

Here, $[\cdot]$ denotes a true or false statement: The result is 1 if the condition is true and 0 if the condition is false. The model considers an occurrence of a target word e an insertion iff no source word f exists within the phrase where the lexical translation probability $p(e|f)$ is greater than a corresponding threshold τ_f.

In an analogous manner to the source-to-target direction, the insertion model in target-to-source direction $t_{t2sIns}(\cdot)$ counts the number of inserted words on the source side α of a hierarchical rule with respect to the target side β of the rule:

$$t_{t2sIns}(\alpha, \beta) = \sum_{j=1}^{J_\alpha} \prod_{i=1}^{I_\beta} \left[p(\alpha_j|\beta_i) < \tau_{\beta_i} \right] \tag{2}$$

Target-to-source lexical translation probabilities $p(f|e)$ are thresholded with values τ_e which may be distinct for each target word e. The model considers an occurrence of a source word f an insertion iff no target word e exists within the phrase with $p(f|e)$ greater than or equal to τ_e.

2.2. Deletion Models

The deletion model in source-to-target direction $t_{s2tDel}(\cdot)$ counts the number of deleted words on the source side α of a hierarchical rule with respect to the target side β of the rule:

$$t_{s2tDel}(\alpha, \beta) = \sum_{j=1}^{J_\alpha} \prod_{i=1}^{I_\beta} \left[p(\beta_i|\alpha_j) < \tau_{\alpha_j} \right] \tag{3}$$

It considers an occurrence of a source word f a deletion iff no target word e exists within the phrase with $p(e|f)$ greater than or equal to τ_f.

The target-to-source deletion model $t_{t2sDel}(\cdot)$ correspondingly considers an occurrence of a target word e a deletion iff no source word f exists within the phrase with $p(f|e)$ greater than or equal to τ_e:

$$t_{t2sDel}(\alpha, \beta) = \sum_{i=1}^{I_\beta} \prod_{j=1}^{J_\alpha} [p(\alpha_j|\beta_i) < \tau_{\beta_i}] \tag{4}$$

2.3. Thresholding Methods for Insertion and Deletion Models

We introduce thresholding methods for insertion and deletion models which set thresholds based on the characteristics of the lexicon model that is applied. We restrict ourselves to the description of the source-to-target direction.

individual τ_f is a distinct value for each f, computed as the arithmetic average of all entries $p(e|f)$ of any e with the given f in the lexicon model.

global The same value $\tau_f = \tau$ is used for all f. We compute this global threshold by averaging over the individual thresholds.

histogram n τ_f is a distinct value for each f. τ_f is set to the value of the $n+1$-th largest probability $p(e|f)$ of any e with the given f.

all All entries with probabilities larger than the floor value are not thresholded. This variant may be considered as *histogram* ∞.

median τ_f is a median-based distinct value for each f, i.e. it is set to the value that separates the higher half of the entries from the lower half of the entries $p(e|f)$ for the given f.

3. Lexical Scoring

Lexical scoring on phrase level is the standard technique for phrase table smoothing in statistical machine translation (Koehn et al., 2003; Zens and Ney, 2004). Jane 2 supports lexical smoothing as well as source-to-target sentence level lexical scoring within search with many types of lexicon models (Huck et al., 2011). Phrase-level lexical scores do not have to be calculated on demand for each hypothesis expansion,

but can again be precomputed in advance and written to the phrase table. We present four scoring variants for lexical smoothing with RF word lexicons or IBM-1 which are provided by Jane 2. We describe the source-to-target directions. The target-to-source scores are computed similarly.

3.1. Phrase-Level Scoring Variants

The first scoring variant $t_{\text{Norm}}(\cdot)$ uses an IBM-1 or RF lexicon model $p(e|f)$ to rate the quality of a target side β given the source side α of a phrase with an included length normalization:

$$t_{\text{Norm}}(\alpha, \beta) = \sum_{i=1}^{I_\beta} \log \left(\frac{p(\beta_i|\text{NULL}) + \sum_{j=1}^{J_\alpha} p(\beta_i|\alpha_j))}{1 + J_\alpha} \right) \tag{5}$$

By dropping the length normalization we arrive at the second variant $t_{\text{NoNorm}}(\cdot)$:

$$t_{\text{NoNorm}}(\alpha, \beta) = \sum_{i=1}^{I_\beta} \log \left(p(\beta_i|\text{NULL}) + \sum_{j=1}^{J_\alpha} p(\beta_i|\alpha_j)) \right) \tag{6}$$

The third scoring variant $t_{\text{NoisyOr}}(\cdot)$ is the noisy-or model proposed by Zens and Ney (Zens and Ney, 2004):

$$t_{\text{NoisyOr}}(\alpha, \beta) = \sum_{i=1}^{I_\beta} \log \left(1 - \prod_{j=1}^{J_\alpha} (1 - p(\beta_i|\alpha_j)) \right) \tag{7}$$

The fourth scoring variant $t_{\text{Moses}}(\cdot)$ is due to Koehn, Och and Marcu (Koehn et al., 2003) and is the standard method in the open-source Moses system (Koehn et al., 2007):

$$t_{\text{Moses}}(\alpha, \beta, \{a_{ij}\}) = \sum_{i=1}^{I_\beta} \log \left(\begin{cases} \frac{1}{|\{a_i\}|} \sum_{j \in \{a_i\}} p(\beta_i|\alpha_j)) & \text{if} \quad |\{a_i\}| > 0 \\ p(\beta_i|\text{NULL}) & \text{otherwise} \end{cases} \right) \tag{8}$$

This last variant requires the availability of word alignments $\{a_{ij}\}$ for phrase pairs $\langle \alpha, \beta \rangle$. We store the most frequent alignment during phrase extraction and use it to compute $t_{\text{Moses}}(\cdot)$.

Note that all of these scoring methods generalize to hierarchical phrase pairs which may be only partially lexicalized. Unseen events are scored with a small floor value.

Source-to-target sentence-level scores are calculated analogous to Eq. (5), but with the difference that the quality of the target side β of a rule currently chosen to expand a partial hypothesis is rated given the whole input sentence f_1^J instead of the source side α of the rule only.

4. Reordering Extensions

In hierarchical phrase-based machine translation, reordering is modeled implicitely as part of the translation model. Hierarchical phrase-based decoders conduct phrase reorderings based on the one-to-one relation between the non-terminals on source and target side within hierarchical translation rules. Recently, some authors have been able to improve translation quality by augmenting the hierarchical grammar with more flexible reordering mechanisms based on additional non-lexicalized reordering rules (He et al., 2010b; Sankaran and Sarkar, 2012; Li et al., 2012). Extensions with lexicalized reordering models have also been presented in the literature lately (He et al., 2010b,a).

Jane 2 offers both the facility to incorporate grammar-based mechanisms to perform reorderings that do not result from the application of hierarchical rules (Vilar et al., 2010a) and the optional integration of a discriminative lexicalized reordering model (Zens and Ney, 2006; Huck et al., 2012). Jane 2 furthermore enables the computation of distance-based distortion costs.

4.1. Non-Lexicalized Reordering Rules

In order to allow for a more flexible arrangement of phrases in the hypotheses, a single swap rule

$$X \rightarrow \left\langle X^{\sim 0} X^{\sim 1}, X^{\sim 1} X^{\sim 0} \right\rangle \tag{9}$$

may be added supplementary to the standard initial rule and glue rule. The swap rule enables adjacent phrases to be transposed.

Other, more complex modifications to the grammar outright replace the standard initial rule and glue rule and implement jumps across blocks of symbols. Specific jump rules put jumps across blocks on source side into effect. Blocks that are skipped by the jump rules are translated without further jumps. Reordering within these windows is possible with hierarchical rules only.

4.2. Discriminative Lexicalized Reordering Model

The discriminative lexicalized reordering model (*discrim. RO*) tries to predict the orientation of neighboring blocks. We use two orientation classes *left* and *right*, in the same manner as described by Zens and Ney (2006). The reordering model is applied at the phrase boundaries only, where words which are adjacent to gaps within hierarchical phrases are defined as boundary words as well. The orientation probability is modeled in a maximum entropy framework (Berger et al., 1996). The feature set of the model may consist of binary features based on the source word at the current source position, on the word class at the current source position, on the target word at the current target position, and on the word class at the current target position. The reordering model is trained with the generalized iterative scaling (GIS)

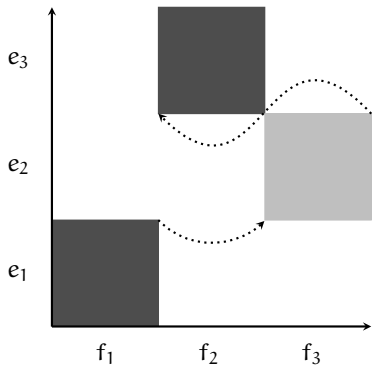

Figure 1. Illustration of an embedding of a lexical phrase (light) in a hierarchical phrase (dark), with orientations scored with the neighboring blocks.

algorithm (Darroch and Ratcliff, 1972) with the maximum class posterior probability as training criterion, and it is smoothed with a gaussian prior (Chen and Rosenfeld, 1999).

For each rule application during hierarchical decoding, the reordering model is applied at all boundaries where lexical blocks are placed side by side within the partial hypothesis. For this purpose, we need to access neighboring boundary words and their aligned source words and source positions. Note that, as hierarchical phrases are involved, several block joinings may take place at once during a single rule application. Figure 1 gives an illustration with an embedding of a lexical phrase (light) in a hierarchical phrase (dark). The gap in the hierarchical phrase $\langle f_1 f_2 X^{\sim 0}, e_1 X^{\sim 0} e_3 \rangle$ is filled with the lexical phrase $\langle f_3, e_2 \rangle$. The discriminative reordering model scores the orientation of the lexical phrase with regard to the neighboring block of the hierarchical phrase which precedes it within the target sequence (here: right orientation), and the block of the hierarchical phrase which succeeds the lexical phrase with regard to the latter (here: left orientation).

5. Soft String-to-Dependency Hierarchical Machine Translation

String-to-dependency hierarchical machine translation (Shen et al., 2008, 2010) employs target-side dependency features to capture syntactically motivated relations between words even across longer distances. It implements enhancements to the hierarchical phrase-based paradigm that allow for an integration of knowledge obtained from dependency parses of the training material. Jane realizes a non-restrictive approach that does not prohibit the production of hypotheses with malformed dependency relations (Stein et al., 2010). Jane includes a spectrum of soft string-to-dependency features: invalidity markers for extracted phrase dependency structures, penalty features for construction errors of the dependency tree assembled during de-

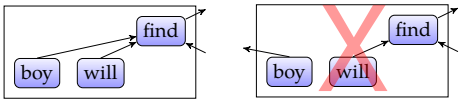

Figure 2. Fixed on head structure (left) and a counterexample (right).

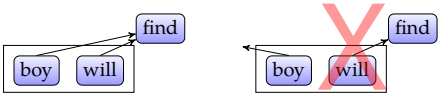

Figure 3. Floating with children structure (left) and a counterexample (right).

coding, and dependency LM features. Dependency trees over translation hypotheses are built on-the-fly during the decoding process from information gathered in the training phase and stored in the phrase table. The soft string-to-dependency features are applied to rate the quality of the constructed tree structures. With version 2 of Jane, dependency LM scoring is—like the other features—directly integrated into the decoder (Peter et al., 2011).

5.1. Dependency Structures in Translation

A dependency models a linguistic relationship between two words, like e.g. the subject of a sentence that depends on the verb. String-to-dependency machine translation demands the creation of dependency structures over hypotheses produced by the decoder. This can be achieved by parsing the training material and carrying the dependency structures over to the translated sentences by augmenting the entries in the phrase table with dependency information. However, the dependency structures seen on phrase level during phrase extraction are not guaranteed to be applicable for the assembling of a dependency tree during decoding. Many of the extracted phrases may be covered by structures where some of the dependencies contradict each other. Dependency structures over extracted phrases which can be considered uncritical in this respect are called *valid*. Valid dependency structures are of two basic types: *fixed on head* or *floating with children*. An example and a counterexample for each type are shown in Figures 2 and 3, respectively. In an approach without hard restrictions, all kinds of structures are allowed, but invalid ones are penalized. Merging heuristics allow for a composition of malformed dependency structures.

A soft approach means that we will not be able to construct a well-formed tree for all translations and that we have to cope with merging errors. During decoding, the previously extracted dependencies are used to build a dependency tree for each hypothesis. While in the optimal case the child phrase merges seamlessly into the parent phrase, often the dependencies will contradict each other and we have to devise strategies for these errors. An example of an ideal case is shown in Figure 4, and a phrase that breaks the previous dependency structure is shown in Figure 5. As a remedy, whenever the direction of a dependency within the child phrase points to the opposite direction of the parent phrase gap, we select the parental direction, but penalize the merging error. In a restrictive approach, the problem can be avoided

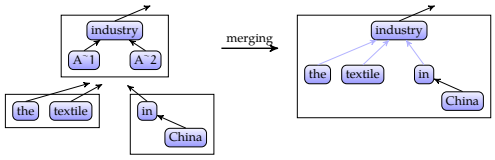

Figure 4. Merging two phrases without merging errors. All dependency pointers point into the same directions as the parent-dependencies.

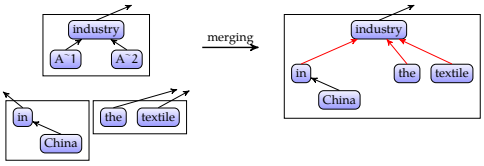

Figure 5. Merging two phrases with one left and two right merging errors. The dependency pointers point into other directions as the parent-dependencies.

by requiring the decoder to always obey the dependency directions of the extracted phrases while assembling the dependency tree.

5.2. Dependency Language Model

Jane computes several language model scores for a given tree: for each node as well as for the left and right-hand side dependencies of each node. For each of these scores, Jane also increments a distinct word count, to be included in the log-linear model, for a total of six features. Note that, while in a well-formed tree only one root can exist, we might end up with a forest rather than a single tree if several branches cannot be connected properly. In this case, the scores are computed on each resulting (partial) tree but treated as if they were computed on a single tree.

6. Experimental Evaluation

We present empirical results obtained with the different models on the Chinese→English 2008 NIST task.

We work with a parallel training corpus of 3.0M Chinese-English sentence pairs (77.5M Chinese / 81.0M English running words). The English target side of the data is lowercased, truecasing is part of the postprocessing pipeline. Word alignments are created by aligning the data in both directions with GIZA++ and symmetrizing the two trained alignments (Och and Ney, 2003). We rely on the Stanford Dependency Parser (Klein and Manning, 2003) to create dependency annotation on the target side of the training data. When extracting phrases, we apply several restrictions, in particular a maximum length of 10 on source and target side for lexical phrases, a length limit of five (including non-terminal symbols) for hierarchical phrases, and no more than two gaps per phrase. The language model is a 4-gram with modified Kneser-Ney smoothing which was trained with the SRILM toolkit (Stolcke, 2002).

We use the cube pruning algorithm (Huang and Chiang, 2007) to carry out the search. A maximum length constraint of 10 is applied to all non-terminals but the initial symbol S. Model weights are optimized against Bleu (Papineni et al., 2002) with

	MT06 (Dev)		MT08 (Test)	
	BLEU [%]	TER [%]	BLEU [%]	TER [%]
Baseline 1 (with s2t+t2s RF word lexicons, $t_{Norm}(\cdot)$)	32.6	61.2	25.2	66.6
+ s2t+t2s insertion model (RF, individual)	32.9	61.4	25.7	66.2
+ s2t+t2s deletion model (RF, histogram 10)	32.9	61.4	26.0	66.1
+ sentence-level s2t IBM-1, $t_{Norm}(\cdot)$	32.9	61.6	25.7	66.6
+ phrase-level s2t IBM-1, $t_{Norm}(\cdot)$	33.0	61.4	26.4	66.1
+ phrase-level t2s IBM-1, $t_{Norm}(\cdot)$	33.4	60.7	26.5	65.7
+ phrase-level s2t+t2s IBM-1, $t_{Norm}(\cdot)$	33.8	60.5	26.9	65.4
+ discrim. RO	33.0	61.3	25.8	66.0
+ swap rule + binary swap feature	33.2	61.3	26.2	66.1
+ jump rules + distance-based distortion costs	33.2	61.0	26.4	66.0
+ insertion model + discrim. RO + DWL + triplets	35.0	59.5	27.8	64.4
Soft string-to-dependency	33.5	60.8	26.0	65.7
— only valid phrases	32.8	62.0	25.4	67.1
— no merging errors	32.5	61.5	25.5	66.4
Baseline 2 (no phrase table smoothing)	32.0	62.2	24.3	67.8
+ phrase-level s2t+t2s RF word lexicons, $t_{Norm}(\cdot)$	32.6	61.2	25.2	66.6
+ phrase-level s2t+t2s RF word lexicons, $t_{NoNorm}(\cdot)$	32.7	61.8	25.6	66.7
+ phrase-level s2t+t2s RF word lexicons, $t_{NoisyOr}(\cdot)$	32.4	61.2	25.5	66.4
+ phrase-level s2t+t2s RF word lexicons, $t_{Moses}(\cdot)$	32.7	61.8	25.4	66.9

Table 1. Experimental results for the NIST Chinese→English translation task (truecase). s2t denotes source-to-target scoring, t2s target-to-source scoring.

MERT on 100-best lists. We employ MT06 as development set, MT08 is used as unseen test set. Translation quality is measured in truecase with BLEU and TER (Snover et al., 2006). The empirical results are presented in Table 1. By incorporating a combination of several of the advanced methods provided by Jane 2 (insertion model, discrim. RO, DWL, triplets), we are able to achieve a performce gain of +2.6% BLEU/ -2.2% TER absolute over a standard hierarchical baseline (*Baseline 1*) .

7. Conclusion

Jane is a stable and efficient state-of-the-art statistical machine translation toolkit that is freely available to the scientific community. It implements the standard hierarchical phrase-based translation approach with many extensions that further enhance the performance of the system. Version 2 of Jane features novel techniques like insertion and deletion models, lexical scoring variants, discriminative reordering extensions, and soft string-to-dependency hierarchical machine translation. We found them to be useful to achieve competitive results on large-scale tasks, and we hope that fellow researchers will benefit from the release of our toolkit.

Acknowledgments

This work was partly achieved as part of the Quaero Programme, funded by OSEO, French State agency for innovation, and partly funded by the European Union under the FP7 project T4ME Net, Contract No. 249119.

Bibliography

Berger, Adam L., Stephen A. Della Pietra, and Vincent J. Della Pietra. A Maximum Entropy Approach to Natural Language Processing. *Computational Linguistics*, 22(1):39–72, Mar. 1996.

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

Chappelier, Jean-Cédric and Martin Rajman. A Generalized CYK Algorithm for Parsing Stochastic CFG. In *Proc. of the First Workshop on Tabulation in Parsing and Deduction*, pages 133–137, Apr. 1998.

Chen, Stanley F. and Ronald Rosenfeld. A Gaussian Prior for Smoothing Maximum Entropy Models. Technical Report CMUCS-99-108, Carnegie Mellon University, Pittsburgh, PA, USA, Feb. 1999.

Chiang, David. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 263–270, Ann Arbor, MI, USA, June 2005.

Chiang, David. Hierarchical Phrase-Based Translation. *Computational Linguistics*, 33(2): 201–228, June 2007.

Chiang, David, Kevin Knight, and Wei Wang. 11,001 new Features for Statistical Machine Translation. In *Proc. of the Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics (HLT-NAACL)*, pages 218–226, Boulder, CO, USA, June 2009.

Darroch, John N. and Douglas Ratcliff. Generalized Iterative Scaling for Log-Linear Models. *Annals of Mathematical Statistics*, 43:1470–1480, 1972.

He, Zhongjun, Yao Meng, and Hao Yu. Maximum Entropy Based Phrase Reordering for Hierarchical Phrase-based Translation. In *Proc. of the Conf. on Empirical Methods for Natural Language Processing (EMNLP)*, pages 555–563, Cambridge, MA, USA, Oct. 2010a.

He, Zhongjun, Yao Meng, and Hao Yu. Extending the Hierarchical Phrase Based Model with Maximum Entropy Based BTG. In *Proc. of the Conf. of the Assoc. for Machine Translation in the Americas (AMTA)*, Denver, CO, USA, Oct./Nov. 2010b.

Huang, Liang and David Chiang. Forest Rescoring: Faster Decoding with Integrated Language Models. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 144–151, Prague, Czech Republic, June 2007.

Huck, Matthias and Hermann Ney. Insertion and Deletion Models for Statistical Machine Translation. In *Proc. of the Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics (HLT-NAACL)*, pages 347–351, Montréal, Canada, June 2012.

Huck, Matthias, Martin Ratajczak, Patrick Lehnen, and Hermann Ney. A Comparison of Various Types of Extended Lexicon Models for Statistical Machine Translation. In *Proc. of the Conf. of the Assoc. for Machine Translation in the Americas (AMTA)*, Denver, CO, USA, Oct./Nov. 2010.

Huck, Matthias, Saab Mansour, Simon Wiesler, and Hermann Ney. Lexicon Models for Hierarchical Phrase-Based Machine Translation. In *Proc. of the Int. Workshop on Spoken Language Translation (IWSLT)*, pages 191–198, San Francisco, CA, USA, Dec. 2011.

Huck, Matthias, Stephan Peitz, Markus Freitag, and Hermann Ney. Discriminative Reordering Extensions for Hierarchical Phrase-Based Machine Translation. In *Proc. of the 16th Annual Conf. of the European Assoc. for Machine Translation*, pages 313–320, Trento, Italy, May 2012.

Klein, Dan and Christopher D. Manning. Accurate Unlexicalized Parsing. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 423–430, Sapporo, Japan, July 2003.

Koehn, Philipp, Franz Joseph Och, and Daniel Marcu. Statistical Phrase-Based Translation. In *Proc. of the Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics (HLT-NAACL)*, pages 127–133, Edmonton, Canada, May/June 2003.

Koehn, P., H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 177–180, Prague, Czech Republic, June 2007.

Li, Junhui, Zhaopeng Tu, Guodong Zhou, and Josef van Genabith. Using Syntactic Head Information in Hierarchical Phrase-Based Translation. In *Proc. of the Workshop on Statistical Machine Translation (WMT)*, pages 232–242, Montréal, Canada, June 2012.

Mauser, Arne, Saša Hasan, and Hermann Ney. Extending Statistical Machine Translation with Discriminative and Trigger-Based Lexicon Models. In *Proc. of the Conf. on Empirical Methods for Natural Language Processing (EMNLP)*, pages 210–218, Singapore, Aug. 2009.

Nelder, John A. and Roger Mead. A Simplex Method for Function Minimization. *The Computer Journal*, 7:308–313, 1965.

Och, Franz Josef. Minimum Error Rate Training for Statistical Machine Translation. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 160–167, Sapporo, Japan, July 2003.

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

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 311–318, Philadelphia, PA, USA, July 2002.

Peter, Jan-Thorsten, Matthias Huck, Hermann Ney, and Daniel Stein. Soft String-to-Dependency Hierarchical Machine Translation. In *International Workshop on Spoken Language Translation*, pages 246–253, San Francisco, CA, USA, Dec. 2011.

Sankaran, Baskaran and Anoop Sarkar. Improved Reordering for Shallow-*n* Grammar based Hierarchical Phrase-based Translation. In *Proc. of the Human Language Technology Conf. /*

North American Chapter of the Assoc. for Computational Linguistics (HLT-NAACL), pages 533–537, Montréal, Canada, June 2012.

Shen, Libin, Jinxi Xu, and Ralph Weischedel. A New String-to-Dependency Machine Translation Algorithm with a Target Dependency Language Model. In *Proc. of the Annual Meeting of the Assoc. for Computational Linguistics (ACL)*, pages 577–585, Columbus, OH, USA, June 2008.

Shen, Libin, Jinxi Xu, and Ralph Weischedel. String-to-Dependency Statistical Machine Translation. *Computational Linguistics*, 36(4):649–671, Dec. 2010.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proc. of the Conf. of the Assoc. for Machine Translation in the Americas (AMTA)*, pages 223–231, Cambridge, MA, USA, Aug. 2006.

Stein, Daniel, Stephan Peitz, David Vilar, and Hermann Ney. A Cocktail of Deep Syntactic Features for Hierarchical Machine Translation. In *Proc. of the Conf. of the Assoc. for Machine Translation in the Americas (AMTA)*, Denver, CO, USA, Oct./Nov. 2010.

Stein, Daniel, David Vilar, Stephan Peitz, Markus Freitag, Matthias Huck, and Hermann Ney. A Guide to Jane, an Open Source Hierarchical Translation Toolkit. *The Prague Bulletin of Mathematical Linguistics*, (95):5–18, Apr. 2011.

Stolcke, Andreas. SRILM – an Extensible Language Modeling Toolkit. In *Proc. of the Int. Conf. on Spoken Language Processing (ICSLP)*, volume 3, Denver, CO, USA, Sept. 2002.

Venugopal, Ashish, Andreas Zollmann, N.A. Smith, and Stephan Vogel. Preference Grammars: Softening Syntactic Constraints to Improve Statistical Machine Translation. In *Proc. of the Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics (HLT-NAACL)*, pages 236–244, Boulder, CO, USA, June 2009.

Vilar, David and Hermann Ney. On LM Heuristics for the Cube Growing Algorithm. In *Proc. of the Annual Conf. of the European Assoc. for Machine Translation (EAMT)*, pages 242–249, Barcelona, Spain, May 2009.

Vilar, David and Hermann Ney. Cardinality pruning and language model heuristics for hierarchical phrase-based translation. *Machine Translation*, Nov. 2011. URL `http://dx.doi.org/10.1007/s10590-011-9119-4`.

Vilar, David, Daniel Stein, and Hermann Ney. Analysing Soft Syntax Features and Heuristics for Hierarchical Phrase Based Machine Translation. In *Proc. of the Int. Workshop on Spoken Language Translation (IWSLT)*, pages 190–197, Waikiki, HI, USA, Oct. 2008.

Vilar, David, Daniel Stein, Matthias Huck, and Hermann Ney. Jane: Open Source Hierarchical Translation, Extended with Reordering and Lexicon Models. In *Proc. of the Workshop on Statistical Machine Translation (WMT)*, pages 262–270, Uppsala, Sweden, July 2010a.

Vilar, David, Daniel Stein, Stephan Peitz, and Hermann Ney. If I Only Had a Parser: Poor Man's Syntax for Hierarchical Machine Translation. In *Proc. of the Int. Workshop on Spoken Language Translation (IWSLT)*, pages 345–352, Paris, France, Dec. 2010b.

Vilar, David, Daniel Stein, Matthias Huck, and Hermann Ney. Jane: an advanced freely available hierarchical machine translation toolkit. *Machine Translation*, 2012. URL `http://dx.doi.org/10.1007/s10590-011-9120-y`.

Zens, Richard and Hermann Ney. Improvements in Phrase-Based Statistical Machine Translation. In *Proc. of the Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics (HLT-NAACL)*, pages 257–264, Boston, MA, USA, May 2004.

Zens, Richard and Hermann Ney. Discriminative Reordering Models for Statistical Machine Translation. In *Proc. of the Human Language Technology Conf. / North American Chapter of the Assoc. for Computational Linguistics (HLT-NAACL)*, pages 55–63, New York City, NY, USA, June 2006.

Address for correspondence:
Matthias Huck
huck@cs.rwth-aachen.de
Human Language Technology and Pattern Recognition Group
Computer Science Department
RWTH Aachen University
D-52056 Aachen, Germany