# Wider Pipelines: $N$-Best Alignments and Parses in MT Training

**Ashish Venugopal** and **Andreas Zollmann** and **Noah Smith** and **Stephan Vogel**
School of Computer Science, Carnegie Mellon University, Pittsburgh
interACT Lab, University of Karlsruhe
{ashishv,zollmann,nasmith,vogel+}@cs.cmu.edu

## Abstract

State-of-the-art statistical machine translation systems use hypotheses from several maximum *a posteriori* inference steps, including word alignments and parse trees, to identify translational structure and estimate the parameters of translation models. While this approach leads to a modular pipeline of independently developed components, errors made in these "single-best" hypotheses can propagate to downstream estimation steps that treat these inputs as clean, trustworthy training data. In this work we integrate $N$-best alignments and parses by using a probability distribution over these alternatives to generate posterior fractional counts for use in downstream estimation. Using these fractional counts in a DOP-inspired syntax-based translation system, we show significant improvements in translation quality over a single-best trained baseline.

## 1 Introduction

Modern statistical machine translation systems are becoming more accurate, but also more complex. To cope with increased system complexity, it is convenient to carve systems into modules that can be separately developed, improved, and tested. In this paper, we explore the cost of such modularization on overall system performance by increasing the amount of information that flows between the training modules of one competitive machine translation approach. Specifically, we consider the pipelining of **word alignment** and **syntactic parsing** information in the construction of transation rules and the estimation of statistics used to decode with those rules.

As Chiang (2005) and Koehn et al. (2003) note, lexical "phrase-based" translation models suffer from sparse data effects when translating conceptual elements that span or skip across several source language words. Phrase-based models also rely on simple distance and lexical distortion models to represent the reordering effects across language pairs. Such models are typically applied over limited source sentence ranges for reasons of model strength (i.e., translation constraints that help prevent errors) and decoding time efficiency (Och and Ney, 2004).

Hierarchically structured models as in Chiang (2005) define weighted transduction **rules**, interpretable as components of a probablistic synchronous grammar (Aho and Ullman, 1969), that represent translation and re-ordering operations. As in monolingual parsing models, such rules make use of nonterminal categories to extend the domain of locality, beyond string-local effects, for resolving ambiguity and making translation decisions. Chiang (2005) uses a single nonterminal category ($X$), while others use syntactically-motivated nonterminal categories, thus bearing the "syntax-based" designation (Galley et al., 2006; Zollmann and Venugopal, 2006). Chiang (2005) and Venugopal et al. (2007) demonstrate efficient translation with probabilistic synchronous CFGs (hereafter, PSCFGs), and Marcu et al. (2006) present results that show significant improvements in translation quality over a phrase based system.

Current phrase-based and hierarchically structured systems rely on the output of a sequential "pipeline" of maximum *a posteriori* inference steps to identify hidden translation structure and estimate the parameters of their translation models. The first step in this pipeline typically involves learning word-alignments (Brown et al., 1993) over parallel sentence aligned training data. The outputs of this step are the model's most probable word-to-word correspondences within each parallel sentence pair.

These alignments are used as the input to a phrase extraction step, where multi-word phrase pairs are identified and scored (with multiple features) based on statistics computed across the training data. The most successful methods extract phrases that adhere to heuristic constraints (Koehn et al., 2003; Och and Ney, 2004). Thus, errors made within the single-best alignment are propagated (1) to the identification of phrases, since errors in the alignment affect which phrases are extracted, and (2) to the estimation of phrase weights, since each extracted phrase is counted as evidence for relative frequency estimates. Methods like those described in Wu (1997) and Marcu and Wong (2002) address this problem by jointly modeling alignment and phrase identification, yet have not achieved the same empirical results as surface heuristic based methods, or require substantially more computational effort to train. See also DeNero et al. (2006).

In this work we describe an approach that "widens" the pipeline, rather than performing two steps jointly. We present $N$-best alignments and parses to the downstream phrase extraction algorithm and define a probability distribution over these alternatives to generate expected, possibly fractional counts for the extracted translation rules, under that distribution. These fractional counts are then used when assigning weights to rules.

This technique is directly applicable to both flat- and hierarchically-structured translation models. In syntax-based translation, single-best target language parse trees (given by a statistical parser) are used to assign syntactic categories within each rule, and to constrain the combination of those rules. Decisions made during the parsing step of the pipeline affect the choice of nonterminals used for each rule in the PSCFG. Presenting $N$-best parse alternatives to the rule extraction process allows the identification of more diverse structures for use during translation and, perhaps, better generalization ability.

We integrate $N$-best alignments and $N$-best parses into the PSCFG grammar induction process within syntax-augmented machine translation estimation suggested in Zollmann and Venugopal (2006). We first recapture their approach more formally in Section 2, and then, in Section 3, extend their grammar extraction method to integrate rules extracted from $N$-best alignments and parses and allow the posterior fractional counts to influence the rule weights.

In Section 4, we show how the widened pipeline improves translation performance on a limited-domain domain speech translation task, the IWSLT-06 Chinese-English data track (Paul, 2006). We explore the impact on translation quality when considering lower probability alignments and parses from the $N$-best lists (according to their respective models) and show significant improvements when combining these alternatives under our estimation method.

## 2 Synchronous Grammars for SMT

Probabilistic synchronous context-free grammars (PSCFGs) are defined by a source terminal set (source vocabulary) $\mathcal{T}_S$, a target terminal set (target vocabulary) $\mathcal{T}_T$, a shared nonterminal set $\mathcal{N}$ and induce rules of the form

$$X \rightarrow \langle \gamma, \alpha, \sim, w \rangle$$

where

- $X \in \mathcal{N}$ is a nonterminal,
- $\gamma \in (\mathcal{N} \cup \mathcal{T}_S)^*$ is a sequence of nonterminals and source terminals,
- $\alpha \in (\mathcal{N} \cup \mathcal{T}_T)^*$ is a sequence of nonterminals and target terminals,
- the count $\#\mathrm{NT}(\gamma)$ of nonterminal tokens in $\gamma$ is equal to the count $\#\mathrm{NT}(\alpha)$ of nonterminal tokens in $\alpha$,
- $\sim$: $\{1, \ldots, \#\mathrm{NT}(\gamma)\} \rightarrow \{1, \ldots, \#\mathrm{NT}(\alpha)\}$ is a one-to-one mapping from nonterminal tokens in $\gamma$ to nonterminal tokens in $\alpha$, and
- $w \in [0, \infty)$ is a nonnegative real-valued weight assigned to the rule.

In our notation, we will assume $\sim$ to be implicitly defined by indexing the NT occurrences in $\gamma$ from left to right starting with 1, and by indexing the NT occurrences in $\alpha$ by the indices of their corresponding counterparts in $\gamma$. Syntax-oriented PSCFG approaches often ignore source structure, instead focusing on generating syntactically well-formed target derivations. Chiang (2005) uses a single nonterminal category, Galley et al. (2006) use syntactic constituents for the PSCFG nonterminal set, and

Zollmann and Venugopal (2006) take advantage of CCG (Steedman, 1999) inspired "slash" and "plus" categories.

We now describe the identification and estimation of PSCFG rules from parallel sentence aligned corpora under the framework proposed by Zollmann and Venugopal (2006), followed by our extensions to integrate evidence from $N$-best alignments and parses.

### 2.1 Grammar Induction

Zollmann and Venugopal (2006) describe a process to generate a PSCFG given parallel sentence pairs $\langle f, e \rangle$, a parse tree $\pi$ for each $e$, the maximum *a posteriori* word alignment $a$ over $\langle f, e \rangle$, and a set of phrase pairs $Phrases(a)$ identified by any alignment-driven phrase induction technique such as e.g. (Och and Ney, 2004).

Each phrase in $Phrases(a)$ is first annotated with a syntactic category to produce initial **rules**, where $\gamma$ is set to the source side of the phrase, $\alpha$ is set to the target side of the phrase, and $X$ is assigned based on the corresponding target side span in $\pi$. If the target span of the phrase does not match a constituent in $\pi$, heuristics are used to assign categories that correspond to partial rewriting of the tree. These heuristics first consider concatenation operations, forming categories like "NP+VP", and then resort to CCG style "slash" categories like "NP/NN." Preference for the concatenation operations over the slash categories is based on the assumption that categories closer to the leaves of the tree are more accurate and more strongly tied to the words than categories higher up the tree.

To illustrate this annotation process, we consider the following French-English sentence pair and selected phrase pairs obtained by phrase induction on an automatically produced alignment $a$:

$$\begin{aligned} f &= \text{il ne va pas} \\ e &= \text{he does not go} \\ \text{il} &: \text{he} \\ \text{va} &: \text{go} \\ \text{ne va pas} &: \text{not go} \\ \text{il ne va pas} &: \text{he does not go} \end{aligned}$$

The alignment $a$ with the associated target side parse tree is shown in Fig. 1 in the alignment visualization style defined by Galley et al. (2006). Matching the target span of each phrase with the parse $\pi$, we generate the following initial rules.

$$\begin{aligned} \text{PRP} &\rightarrow \text{il, he} \\ \text{VB} &\rightarrow \text{va, go} \\ \text{RB+VB} &\rightarrow \text{ne va pas, not go} \\ \text{S} &\rightarrow \text{il ne va pas, he does not go} \end{aligned}$$

Note that the third rule illustrates the use of concatenation categories to identify syntactic categories. These initial rules form the lexical basis for generalized rules that include labeled syntactic categories in $\gamma$ and $\alpha$. Following the Data-Oriented Parsing (Scha, 1990) inspired rule generalization technique proposed by Chiang (2005), one can now generalize each **identified** rule (initial or already partially generalized)

$$N \rightarrow f_1 \dots f_m / e_1 \dots e_n$$

for which there is an **initial** rule

$$M \rightarrow f_i \dots f_u / e_j \dots e_v$$

where $1 \leq i < u \leq m$ and $1 \leq j < v \leq n$, to obtain a new rule

$$N \rightarrow f_1 \dots f_{i-1} M_k f_{u+1} \dots f_m / e_1 \dots e_{j-1} M_k e_{v+1} \dots e_n$$

where $k$ is an index for the nonterminal $M$ that indicates the one-to-one correspondence between the new $M$ tokens on the two sides (it is not in the space of word indices like $i, j, u, v, m, n$). The recursive form of this generalization operation allows the generation of rules with multiple nonterminal symbols. Note that since we only generalize over initial rules, this operation has polynomial runtime as a function of $|Phrases(a)|$.

The initial rules listed above can be generalized to additionally extract the following rules from $f, e$.

$$\begin{aligned} \text{S} &\rightarrow \text{PRP}_1 \text{ ne va pas , PRP}_1 \text{ does not go} \\ \text{S} &\rightarrow \text{il ne VB}_1 \text{ pas , he does not VB}_1 \\ \text{S} &\rightarrow \text{il RB+VB}_1, \text{ he does RB+VB}_1 \\ \text{S} &\rightarrow \text{PRP}_1 \text{ RB+VB}_2, \text{ PRP}_1 \text{ does RB+VB}_2 \\ \text{RB+VB} &\rightarrow \text{ne VB}_1 \text{ pas , not VB}_1 \end{aligned}$$

Fig. 2 uses regions to identify the labeled, source and target side span for all initial rules extracted on our example sentence pair and parse. Under this representation, the generalization operation can be viewed as a process that selects a region, and proceeds to subtract out any sub-region to form a generalized rule.

Note that its is possible to extract the same rule from a given sentence multiple times, making estimates derived from counts over these rules inconsistent. In this work, we do not double-count any rules that can be extracted multiple times from one sentence pair, even if these multiple rules represent structures found in different parts of the sentence. This decision biases our counting against rules that represent high frequency words. We prefer this option rather than over-counting rules that represent low frequency words, since differences in low-frequency estimates have greater effect on the translation model, which is log-linear, as we will see in the following section.

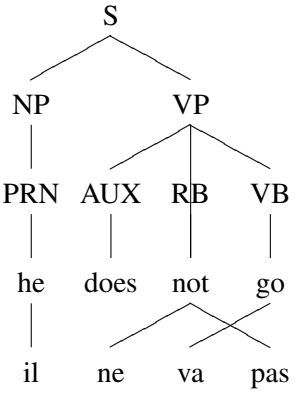


Figure 1: Alignment graph (word alignment and target parse tree) for a French-English sentence pair.

## 2.2 Decoding

Given a source sentence $f$, the translation task under a PSCFG grammar can be expressed analogously to monolingual parsing with a CFG. We find the most likely derivation $D$ of the input source sentence while reading off the English translation from this derivation:

$$\hat{e} = \text{tgt}\left(\underset{D:\text{src}(D)=f}{\arg\max}\ p(D)\right) \quad (1)$$

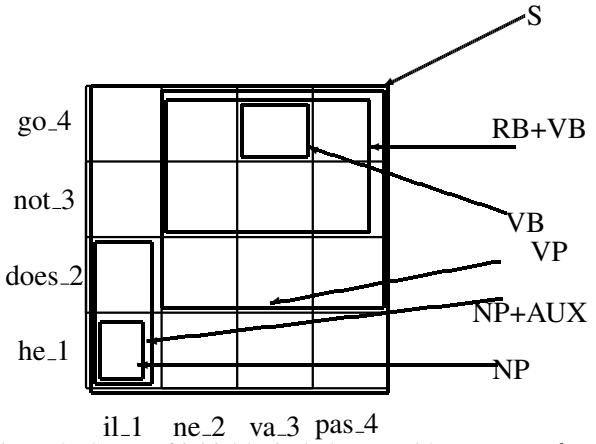


Figure 2: Spans of initial lexical phrases with respect to $f, e$. Each phrase is labeled with a category derived from the tree in Figure 1.

where $\text{tgt}(D)$ refers to the target terminal symbols generated by the derivation $D$ and $\text{src}(D)$ refers to the source terminal symbols spanned by $D$.

Our distribution $p$ over derivations is defined by a log-linear model. The probability of a derivation $D$ is defined in terms of the rules $r$ that are used in $D$:

$$p(D) = \frac{p_{LM}(\text{tgt}(D))^{\lambda_{\text{LM}}} \times \prod_{r \in D} \prod_i \phi_i(r)^{\lambda_i}}{Z(\lambda)} \quad (2)$$

where $\phi_i$ refers to features defined on each rule, $p_{\text{LM}}$ is a $g$-gram LM probability applied to the target terminal symbols generated by the derivation $D$, and $Z(\lambda)$ is a normalization constant chosen such that the probabilities sum up to one. The computational challenges of this search task (compounded by the integration of the language model) are addressed elsewhere (Chiang, 2007; Venugopal et al., 2007). All feature weights $\lambda_i$ are trained in concert with the language model weight via minimum-error training (Och, 2003). Here, we focus on the estimation of the feature *values* $\phi$ during the grammar induction process. The feature values are statistics estimated from rule counts.

## 2.3 Feature Value Statistics

The features $\phi$ represent multiple criteria by which the decoding process can judge the quality of each rule and, by extension, each derivation. We include both real-valued and boolean-valued features for each rule. The following probabilistic quantities are estimated and used as feature values:

- $\hat{p}(r|\,\text{lhs}(X))$ : Probability of a rule given its l.h.s

category

- $\hat{p}(r|\,\mathrm{src}(r))$ : Probability of a rule given its source side
- $\hat{p}(r|\,\mathrm{tgt}(r))$ : Probability of a rule given its target side
- $\hat{p}(\mathrm{ul}(\mathrm{src}(r)), \mathrm{ul}(\mathrm{tgt}(r))|\,\mathrm{ul}(\mathrm{src}(r))$ : Probability of the unlabeled source and target side of the rule given its unlabeled source side.
- $\hat{p}(\mathrm{ul}(\mathrm{src}(r)), \mathrm{ul}(\mathrm{tgt}(r))|\,\mathrm{ul}(\mathrm{src}(r)))$ : Probability of the unlabeled source and target side of the rule given its unlabeled target side.

where $\mathrm{lhs}$ returns the left-hand-side of a rule, $\mathrm{src}$ returns the source side $\gamma$, and $\mathrm{tgt}$ returns the target side $\alpha$ of a rule $r$. The function $\mathrm{ul}$ removes all syntactic labels from its arguments, but retains ordering notation. For example, $\mathrm{ul}(\text{NP+AUX}_1\text{does not go}) = \square_1$ does not go.

The last two features are extensions to the feature set suggested by (Zollmann and Venugopal, 2006). They represent the same kind of relative frequency estimates commonly used in phrase based systems. The $\mathrm{ul}$ function allows us to calculate these estimates for rules with nonterminals as well.

To estimate these probabilistic features, we use maximum likelihood estimates based on counts of the rules extracted from the training data. For example, $\hat{p}(r|lhs(r))$ is estimated by computing $\#(r)/\#(\mathrm{lhs}(r))$, aggregating counts from all extracted rules.

As in phrase-based translation model estimation, $\phi$ also contains two lexical weights $\hat{p}_w(\mathrm{lex}(\mathrm{src}(r))|\,\mathrm{lex}(\mathrm{tgt}(r)))$ and $\hat{p}_w(\mathrm{lex}(\mathrm{tgt}(r))|\,\mathrm{lex}(\mathrm{src}(r)))$ (Koehn et al., 2003) that are based on the lexical symbols of $\gamma, \alpha$. These weights are estimated based on an pair of statistical lexicons that represent $\hat{p}(s|t), \hat{p}(t|s)$, where $s$ and $t$ are single words in the source and target vocabulary. These word-level translation models are typically estimated by maximum likelihood, considering the word-to-word links from "single-best" alignments as evidence.

$\phi$ also contains several boolean and count features: the rule is purely lexical in $\alpha$ and $\gamma$; the rule is purely *non-lexical* in $\alpha$ and $\gamma$, the rule has significant differences in the number of lexical source and target words; and the rule generates more or less target words than other derivations. The last two features are commonly used in phrase based systems to ensure that the target translations are of sufficient length to perform well against $n$-gram automatic translation evaluation metrics like BLEU (Papineni et al., 2002).

## 3 $N$-best Evidence

The rule extraction procedure described above relies on high quality word alignments and parses. The quality of the alignments affects the set of phrases that can be identified by the heuristics in (Koehn et al., 2003). In addition, alignment quality plays a role in determining the set of valid compositions that can create complex rules, which represent both translation as well as reordering operations across language pairs. The quality of the parses affects the syntactic categories assigned to each complex rule and its respective composed arguments. These categories play an important role in constraining the decoding process to grammatically feasible target parse trees.

Quirk and Corston-Oliver (2006) show improvements in translation quality when the quality of parsing is improved by adding additional training data within the "treelet" paradigm introduced by Quirk et al. (2005). Koehn et al. (2003) show that translation quality in a phrase based system does not vary significantly when increasing the complexity of the model used for alignment (ranging from IBM model 1 through 4), but that increasing the amount of parallel training data does improvement alignment quality.

Our approach considers alignment and parse quality for a fixed training data size and model complexity. Variance in quality is judged by the models that generates, respectively, the alignments and parses, and is reflected in the probabilities assigned to the $N$-best alternatives. Informal examination of the highest probability alignment and target parse tree reveals two important arguments in favor of integrating $N$-best hypotheses into the rule extraction process. Firstly, there are often multiple reasonable alignments and parses that can model the bilingual sentence pair and the target sentence. We can expect that rules extracted from more diverse, correct evidence can improve translation quality on new sentences, since more (good) rules will be extracted. Secondly, where there is a high degree of agreement

across each alternative in the $N$-best lists, the remaining differences between alternatives are often the source of error or ambiguity.

Attempts to reduce the use (in decoding) of rules extracted from sections of the alignment and parse that are not consistent with other alternatives could reduce errors made during translation. Put another way, the more complete hypotheses a word-link or constituent appears in, and the more probable those hypotheses, the more we should trust rules that use these links.

Our approach towards the integration of $N$-best evidence into the grammar induction process allows us to take advantage of the diversity found in the $N$ best alternatives, while reducing the negative impact of errors made in these alternatives.

### 3.1 Counting from $N$-Best Lists

In this work we propose extraction of complex rules over $N$-best alignments and $N'$-best parses, making use of probability distributions over these alternatives to assign fractional posterior counts to each complex rule that can be extracted.

Taking the alignment $N$-best list to define a posterior distribution over alignments and the parse $N'$-best list to define a posterior over parse trees, we can estimate the posterior probability of each rule that might be extracted for each (alignment, tree) pair. Assuming that the alignment module gives alignments $a_1, ..., a_N$, with posterior probabilities $p(a_1 \mid e, f), ..., p(a_N \mid e, f)$, we approximate the posterior by renormalizing:

$$\hat{p}(a_i) = p(a_i \mid e, f) \Big/ \sum_{j=1}^{N} p(a_j \mid e, f) \qquad (3)$$

The same is applied to the parser's $N'$-best parses, $\pi_1, ..., \pi_{N'}$.

Given a single alignment-parse pair, we can extract rules as described in Section 2.1. Our approach is to extract rules from the cross-product $\{a_1, ..., a_N\} \times \{\pi_1, ..., \pi_{N'}\}$, incrementing the partial count of each rule extracted by $\hat{p}(a_i) \cdot \hat{p}(\pi_j)$. A rule $r$'s total count for the sentence pair $\langle f, e \rangle$ is:

$$\sum_{i=1}^{N} \sum_{j=1}^{N'} \hat{p}(a_i) \cdot \hat{p}(\pi_j) \cdot \begin{cases} 1 & \text{if } r \text{ can be extracted from} \\ & \quad e, f, a_i, \pi_j \\ 0 & \text{otherwise} \end{cases} \qquad (4)$$

In practice, this can be computed more efficiently through structure-sharing. Note that if $N = N' = 1$, this counting method generalizes the original counting method.

Note that GIZA++ (Och and Ney, 2003) can infer the $N$-best word alignments under IBM Model 4 and the Charniak parser (Charniak, 2000) outputs its $N'$-best parses, with their associated probabilities.

Instead of using the simple counts for rules given the derivation inferred using the maximum *a posteriori* estimated alignment and parse $(a_1, \pi_1)$, we now use the expected counts under the approximate posterior. These posteriors encode (in a principled way) a measurement of confidence in substructures used to generate each rule. Possible rule instances supported by more and more likely alignments and parses should, intuitively, receive higher counts (approaching 1 as certainty increases, supported by more and higher-probability alternatives), while rule instances that rely on low probability or fewer alignments and parses will get lower counts (approaching 0 as certainty increases).

### 3.2 Refined Alignments

Work by Och and Ney (2004) and Koehn et al. (2003) demonstrates the value of generating word alignments in both source-to-target and target-to-source directions in order to facilitate the extraction of phrases with many-to-many word relationships. We follow Koehn et al. (2003) in generating a refined bidirectional alignment using the heuristic algorithm "grow-diag-final" described in that work. Since we require $N$-best alignments, we first extract $N$-best alignments in each direction, then perform the refinement technique to all $N^2$ bidirectional alignment pairs. By taking the geometric mean of the probabilities of the alignments in each direction, we can assign probabilities to the resulting refined alignments and remove all duplicate alignments that came about due to the refinement process. We then select the top $N$ alignments from this set of refined alignments. The geometric mean could of course be tuned to favor one direction; we did not explore such tuning.

## 4 Translation Results

### 4.1 Experimental Setup

We present results on the IWSLT 2006 Chinese-to-English translation task, based on the Full BTEC corpus of travel expressions with 120K parallel sentences (906K source words and 1.2M target words). The evaluation test set contains 500 sentences with an average length of 10.3 Chinese words. Word alignment was trained using the GIZA++ toolkit, and 100 parses generated by the Charniak (2000) parser, without additional re-ranking.[1] 100-best alignments were generated from source to target and target to source, refined as described above and then made unique.

Initial phrases were identified using the heuristics proposed by Koehn et al. (2003). Rules were extracted using the toolkit made available in Zollmann and Venugopal (2006) and modified to handle $N$-best alignments and posterior counting. Note that lexical weights (Koehn et al., 2003) as described above are assigned to $\phi$ for rule based on the based on "single-best" word alignments. Rules that receive zero probability value for their lexical weights are immediately discarded, since they would then have a prohibitively high cost when used during translation. Rules extracted from single-best evidence as well as $N$ best evidence can be discarded in this way.

The n-gram language model is trained on the target side of the parallel training corpus and translation experiments run with the decoder and MER trainer available in the same toolkit. We use the Cube-Pruning (Chiang, 2007) option for translation experiments.

### 4.2 Cumulative $(N, N')$-Best

We measure translation quality using the BLEU metric as we vary the size of $N$ and $N'$ for alignments and parses respectively. Each value of $N$ implies that the first $N$ alternatives have been considered when building the grammar. For each grammar we also consider the number of rules (after selecting only those relevant to the development and test data) as well as the number of syntactic categories represented in the grammar. We also note the number of seconds required to translate the evaluation data.

Table 1 summarizes results across each grammar configuration for the IWSLT limited domain task. Due to time and resource constraints, we limit our evaluation to varying the number of alignments and the number of parses used separately. We limit $N'$, the number of alternative parses considered, to 10 due to the dramatic increases in runtime incurred by adding parse trees. This result is as expected: using alternative parse information directly increases the number of nonterminals available in the grammar. As mentioned in Chiang (2007), the number of nonterminals is the dominant factor in parsing runtime after the $n$-gram language model is integrated into search.

The baseline result, where the "single-best" pipeline is used, achieves a development set score of 23.67% and a test set score of 19.78%. On both development and test data, methods that integrate additional evidence from the $N$-best alternatives achieve significant improvements (1.29 points) over the baseline. There is more impact from considering alternative alignments rather than parses, at least at the levels considered here. In both cases, the number of nonterminals increases with the number of additional alternatives considered, increasing runtime as expected. Scores on development data using alternative alignments show a clearer trend of improvements as more alternatives are considered. This effect is likely due to parameters $\lambda$ being optimally learned on development data to make the best use of additional rules from the alternative evidence. Nevertheless, all configurations of $N$-best evidence do show marked improvement over the "single-best" baseline.

### 4.3 Widening the Lexicon

As noted above, rules that have no support from the lexical weight features are immediately discarded. The underlying word based models $\hat{p}(s|t)$ and $\hat{p}(t|st)$ are estimated based on "single-best" alignments. In the spirit of softening our pipelined decisions, we add an *additional* pair of lexical weights (in each direction) to $\phi$ based on the IBM Model 4 tables output by GIZA++ at the end of its training. Using these IBM Model 4 weights allows a significantly larger number of rules to be added to the grammar since more rules have non-

[1] Reranking might be used to change estimates of $\hat{p}(\tau_i)$, but would not change the set of rules extracted—only the fractional counts.

| $N, N'$ | #Rules | #NTs | Dev | Test | Time |
|---|---|---|---|---|---|
| 1, 1 | 300K | 1771 | 23.7 | 19.8 | 1145 |
| 1..5, 1 | 490K | 1894 | 24.3 | 21.0 | 2086 |
| 1..10, 1 | 582K | 1947 | 24.3 | 20.1 | 2563 |
| 1..25, 1 | 747K | 2026 | 24.4 | 20.1 | 3840 |
| 1..50, 1 | 911K | 2072 | 24.8 | 21.1 | 5132 |
| 1, 1..5 | 616K | 2393 | 23.9 | 20.0 | 4291 |
| 1, 1..10 | 850K | 2633 | 24.0 | 20.1 | 7237 |

Table 1: Grammar statistics and translation quality (IBM-BLEU) on development and test set and when integrating $N$-best alignments an $N'$-best parses. Decoding time in seconds is on all 500 sentences

| $N, N'$ | #Rules | #NTs | Dev | Test | Time |
|---|---|---|---|---|---|
| 1, 1 | 311K | 1781 | 23.7 | 21.2 | 1,369 |
| 1..10, 1 | 1m | 2212 | 26.0 | 22.2 | 13,406 |
| 1..50, 1 | 2.3m | 2526 | 26.2 | 21.4 | 53,492 |
| 1, 1..10 | 652K | 2407 | 25.9 | X | 13,396 |

Table 2: Grammar statistics and translation quality when integrating $N$-best alignments an $N'$-best parses using IBM Model 4 lexical weights. All missing values (X) will be available in the final version.

zero lexical weight. Table 2 summarizes grammar statistics and translation quality for grammars that use these "widened" lexicons. There are significantly more rules used in all the $N$-best evidence based grammars as a result of using the Model 4 lexicon. Despite a modest increase in the number of rules used in the baseline system, translation quality is significantly improved by using the IBM Model 4 lexical weight rather than models based on single-best alignments only. The grammar built on $N = 10, N' = 1$ achieves the best results on evaluation data, again more than 1 point over the baseline result. Results with $N = 50, N' = 1$, however, show signs of parameter over-fitting to the development data. We believe that this is due to the sheer number of available rules that can be used to generate translation alternatives for MER training—there are too many parameters and too little data to estimate them.[2]

[2]This effect could be mitigated during the normalization (Section 3.1) by controlling the entropy of the resulting distribution over alternative alignments. We could make the distributions over $N$-best lists "peakier" to increase the penalty of using rules from low-probability alignments during translation.

| count | source | target | LHS NT |
|---|---|---|---|
| 247.93 | 请 | please . | @UH+. |
| 210.69 | 请 | please . | @VB+. |
| 162.06 | 想 | 'd | @MD |
| 153.42 | 我 | , I | @,+PRP |
| 146.32 | 我 | I have | @PRP+AUX |
| 141.96 | 我 | . | @. |
| 141.75 | 的 | in | @IN |
| 133.52 | 我 想 | I 'd | @PRP+MD |
| 130.99 | ~ | did you | @AUX+PRP |
| 125.18 | 的 | is | @AUX |

Figure 3: Top rules extracted by our method, but not the baseline.

### 4.4 Grammar Rules

Figure 3 shows the most frequently occurring rules that exist only in the best performing $N = 10, N' = 1$ grammar, and not in the baseline (Model-4 lexicon) grammar. We show the estimated counts on these rules as well as their source, target and l.h.s. These rules are particularly interesting when considering the domain of this translation task. The source side of the training data contains no punctuation (since it is transcribed speech), while the target side does (since they were manually generated translations). The system therefore attempts to generate punctuation during translation. Consider the first example, where the Chinese word for "please" (often found at the beginning of a sentence) is aligned to the English "please ." (at the end of the sentence as indicated by the punctuation). This rule is extracted from a lower-probability alignment with high levels of distortion. This pattern was not seen in any single-best alignments.

## 5 Conclusion

In this work we have demonstrated the feasibility and benefits of widening the MT pipeline to include additional evidence from $N$-best alignments and parses. We integrate this diverse knowledge under a principled model that uses a probability distribution over these alternatives. We achieve significant improvements in translation quality over grammars built on "single-best" evidence alone.

**Acknowledgments** This work has been partly funded by the European Union under the inte-

grated project TC-STAR - Technology and Corpora for Speech to Speech Translation (IST-2002-FP6-506738, http://www.tc-star.org).

## References

A.V. Aho and J.D Ullman. 1969. Syntax directed translations and the pushdown assembler. *Journal of Computer and System Sciences*, 3:37–56.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Comput. Linguist.*, 19(2):263–311.

Eugene Charniak. 2000. A maximum entropy-inspired parser. In *Proc. of the North American Association for Computational Linguistics (HLT/NAACL)*.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proc. of ACL*.

David Chiang. 2007. Hierarchical phrase based translation. *Computational Linguistics*. To appear.

John DeNero, Dan Gillick, James Zhang, and Dan Klein. 2006. Why generative phrase models underperform surface heuristics. In *Proceedings on the Workshop on Statistical Machine Translation*, pages 31–38, New York City, June. Association for Computational Linguistics.

Michael Galley, M. Hopkins, Kevin Knight, and Daniel Marcu. 2006. Scalable inferences and training of context-rich syntax translation models. In *Proc. of NAACL-HLT*.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proc. of HLT/NAACL*.

Daniel Marcu and William Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proc. of the Conference on Empirical Methods in Natural Language Processing*, Philadephia, PA, July 6-7.

Daniel Marcu, Wei Wang, Abdessamad Echihabi, and Kevin Knight. 2006. SPMT: Statistical Machine Translation with Syntactified Target Language Phrases. In *Proc. of EMNLP*, Sydney, Australia.

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

Franz Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Comput. Linguist.*

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proc. of ACL*, Sapporo, Japan, July 6-7.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proc. of ACL*.

Michael Paul. 2006. Overview of the IWSLT 2006 Evaluation Campaign. In *Proc. of the International Workshop on Spoken Language Translation*, pages 1–15, Kyoto, Japan.

Chris Quirk and Simon Corston-Oliver. 2006. The impact of parse quality on syntactically-informed statistical machine translation. In *Proc. of EMNLP*, Sydney, Australia.

Chris Quirk, Arul Menezes, and Collin Cherry. 2005. Dependency tree translation: Syntactically informed mt. In *Proc. of ACL*.

R. Scha. 1990. Taaltheorie en taaltechnologie; competence en performance. In Q.A.M. de Kort and G.L.J. Leerdam, editors, *Computertoepassingen in de Neerlandistiek, LVVN-jaarboek*, pages 7–22, Almere, The Netherlands. English translation as: Language Theory and Language Technology; Competence and Performance; http://iaaa.nl/rs/LeerdamE.html.

Mark Steedman. 1999. Alternative quantifier scope in CCG. In *Proc. of ACL*.

Ashish Venugopal, Andreas Zollmann, and Stephan Vogel. 2007. An efficient two-pass approach to synchronous CFG driven MT. In *Proc. of HLT/NAACL*, Rochester,NY, April 22-April 28.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–404.

Andreas Zollmann and Ashish Venugopal. 2006. Syntax augmented machine translation via chart parsing. In *Proc. of the Workshop on Statistical Machine Translation, HLT/NAACL*, New York, June.