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

Ashish Venugopal, Andreas Zollmann, Noah A. Smith, and Stephan Vogel
Language Technologies Institute
Carnegie Mellon University
Pittsburgh, PA 15213, USA
{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 translation 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 Ullmann, 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) and Zollmann et al. (2008) present results that show significant improvements in translation quality over a phrase based system on languages where long distance re-ordering effects exist.

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), Marcu and Wong (2002), and DeNero et al. (2006) 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.

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 integrated our 'wider-pipeline' model into the PSCFG grammar construction process of the publicly available Syntax-Augmented Machine Translation system (Zollmann and Venugopal, 2006). We first review PSCFG grammars (Section 2), and then, in Section 3, present a method of integrating PSCFG 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 Chinese-English data track (Paul, 2006).

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 γ is equal to the count $\#\mathrm{NT}(\alpha)$ of nonterminal tokens in α,
- $\sim$: $\{1, \ldots, \#\mathrm{NT}(\gamma)\} \rightarrow \{1, \ldots, \#\mathrm{NT}(\alpha)\}$ is a one-to-one mapping from nonterminal tokens in γ to nonterminal tokens in α, 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 γ from left to right starting with 1, and by indexing the NT occurrences in α by the indices of their corresponding counterparts in γ. 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-inspired "slash" categories (Steedman, 2000) and also concatenated "plus" categories.

We now briefly describe the identification and estimation of PSCFG rules from parallel sentence aligned corpora under the framework proposed by Zollmann and Venugopal (2006). Note, however, that this paper's contribution of integrating evidence from N-best alignments and/or parses can be applied to any of the other PSCFG approaches mentioned above in a straight-forward manner.

2.1 Grammar Construction

Zollmann and Venugopal (2006) describe a process to generate a PSCFG given parallel sentence pairs $\langle f, e \rangle$, a parse tree π 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. Koehn et al. (2003; Och and Ney (2004).

Each phrase in $Phrases(a)$ is first annotated with a syntactic category to produce initial **rules**, where γ is set to the source side of the phrase, α is set to the target side of the phrase, and X is assigned based on the corresponding target side span in π. If the target span of the phrase does not match a constituent in π, 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.". The system described in Zollmann and Venugopal (2006) can be used to create a Syntax Augmented grammar as well as a purely hierarchical grammar that uses a single generic nonterminal symbol Chiang (2005). The Syntax Augmented system also generates a purely hierarchical variant for each syntactic rule that is identified, giving the decoder the option of using labelled or non-labelled rules during translation. These initial rules form the lexical basis for generalized rules that include labeled syntactic categories in γ and α. Following the DOP-inspired (Scha, 1990) 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

$$\begin{aligned} N \quad \rightarrow \quad & f_1 \dots f_{i-1} M f_{u+1} \dots f_m / \\ & e_1 \dots e_{j-1} M e_{v+1} \dots e_n \end{aligned}$$

where the two instances of M are mapped under $\sim$. The recursive form of this generalization operation allows the generation of rules with multiple nonterminal symbols. Since we only conside initial phrases up to a fixed length (10 in this work), and only allow a fixed number of nonterminals per rule (2), this operation has a runtime that is polynomial as a function of $|Phrases(a)|$.

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:

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

where $\mathrm{tgt}(\cdot)$ maps a derivation to its target yield and $\mathrm{src}(\cdot)$ maps a derivation to its source yield.

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:

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

where ϕ_i is a feature function on rules, p_{LM} is a g-gram probability of the target yield $\mathrm{tgt}(D)$, and $Z(\lambda)$ is a normalization constant chosen such that the probabilities sum up to one.[1] 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 λ_i are trained in concert

[1] Note that we never need to actually compute $Z(\lambda)$ since we are merely interested in the maximum-probability derivation.

with the language model weight λ_{LM} via minimum-error training (MER) (Och, 2003). Here, we focus on the estimation of the feature *values* ϕ during the grammar construction process. The feature values are statistics estimated from rule counts.

2.3 Feature Value Statistics

The features ϕ 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:

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

In our notation, lhs returns the left-hand side of a rule, src returns the source side γ, and tgt returns the target side α of a rule r. The function ul removes all syntactic labels from its arguments, but retains ordering notation. For example, ul(NP+AUX$_1$does not go) $= \Box_1$ does not go.

The last two features represent the same kind of relative frequency estimates commonly used in phrase-based systems. The 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, ϕ 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 γ and α. 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.

ϕ contains several boolean features that indicate whether: (a.) the rule is purely lexical in α and γ, (b.) the rule is purely *non-lexical* in α and γ, (c.) the ratio of lexical source and target words in the rule is between 1/5 and 5. ϕ also contains a feature that reflects the number of target lexical symbols and a feature that is 1 for each rule, allowing the decoder to prefer shorter (or longer) derivations based on the corresponding weight in λ.

3 N-best Evidence

The PSCFG 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). Improving or diversifying the set of initial phrases also affects the rules with nonterminals that are identified via the procedure described above. Since PSCFG systems rely on rules with nonterminal symbols to represent reordering operations, the set of these initial phrases has the potential to have a profound impact on translation quality. The quality of the parses affects the syntactic categories assigned to the left-hand-side and nonterminal symbols of each rule. These categories play an important role in constraining the decoding process to grammatically feasible target parse trees.

Several recent studies explore the relationship between the quality of the initial models in the "pipeline" and final translation quality. 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. Ganchev et al. (2008) demonstrate significant improvements in both alignment quality (as measured by alignment error rate (Och and Ney, 2003)) and translation quality when using a posterior decoding method to select alignments (as opposed to the single-best Viterbi alignment). Xue et al. (2006) apply n-best alignments to improve phrase-based translation, while Dyer et al. (2008) and Mi et al. (2008) widen the pipeline by considering word-lattice and forest-based translation rather than translating the single-best hypothesis from a previous stage in the pipeline.

Our approach considers alignment and parse quality for a fixed training data size and model complexity. The alignment model and the parser are capable of generating N-best alternative candidates along with corresponding probabilities for each candidate. 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 toward the integration of N-best evidence into the grammar construction 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 extracted rule.

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:

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

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, π_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-and" described in that work. Since we require N-best alignments, we first extract N-best alignments in each direction, and then perform the refinement technique to all N^2 bidirectional alignment pairs. The resulting alignments are assigned the probability $(p_f.p_r)^\alpha$ where p_f is the candidate probabilty for the forward alignment and p_r is the candidate probability to the reverse alignment.

We then remove any duplicate refined alignments (the refined alignment with the highest probability is retained) that came about due to the refinement process. Finally, we select the top N alignments from this set of refined alignments.

The selection of α controls the entropy of the resulting distribution over candidate alignments (after normalization). Higher values of $\alpha > 1$ make the distribution more peaked (affecting the estimation of features on rules from these alignments), while values of $0 \leq \alpha < 1$ make the distribution more uniform. A more peaked distribution favors rules from the top alignments, while a more uniform one gives rules from lower performing aligments more of a chance to participate in translation. We can also use this same technique to control the distribution over parses.

4 Translation Results

4.1 Experimental Setup

We present results on the IWSLT 2007 and 2008 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) as well as the evaluation corpora from the evaluation years preceding 2007. The development data consists of 489 sentences (average length of 10.6 words) from the 2006 evaluation, the 2007 test set contains 489 sentence (average length of 6.47 words) sentences and the 2008 test set contains 507 sentences (average length of 5.59 words). Word alignment was trained using the GIZA++ toolkit, and N-best parses generated by the Charniak (2000) parser, without additional re-ranking.[2] N-best alignments were generated from source to target and target to source, refined as described above.

Initial phrases of up to length 10 were identified using the heuristics proposed by Koehn et al. (2003). Rules with up to 2 nonterminals are extracted using the SAMT toolkit (Zollmann and Venugopal, 2006), modified to handle N-best alignments and parses and posterior counting. Note that lexical weights (Koehn et al., 2003) as described above are assigned to ϕ 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[3] and translation experiments use the decoder and MER trainer available in the same toolkit. We use the cube-pruning option (Chiang, 2007) in these experiments.

4.2 Cumulative (N, N')-Best

We measure translation quality using the mixed-cased IBM-BLEU (Papineni et al., 2002) 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 track the number of rules relevant for the first sentence in the IWSLT 2007 test set (grammars are subsampled on a per-sentence basis to keep memory requirements low during decoding). We also note the number of seconds required to translate each test

[2] 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.

[3] As BTEC is a very domain-specific corpus, training the language model on large available monolingual corpora (e.g., from the news-domain) is of limited utility.

set. Due to time and resource constraints we limit our evaluation to varying the number of alignments and parses separately, and we limit N' to 10 (due to the significant increase in decoding time that results from adding more nonterminal labels to the grammar).

As noted above, many rules extracted based on N-best alignments cannot participate in the decoding process because lexical weight features can have costs of infinity if the underlying word based models $\hat{p}(s|t)$ and $\hat{p}(t|s)$, estimated based on "single-best" alignments, yield zero probabilities. Smoothing these models alleviates the problem, but does not fix it at its root. In the spirit of softening our pipelined decisions, we create lexical weight features based on the IBM Model 4 tables output by GIZA++ at the end of its training, instead of single-best alignment relative frequencies. Using these IBM Model 4 weights allows a larger number of rules to be added to the grammar since more rules have non-zero lexical weights.

We also investigate the impact of the shape N-best probability distribution used to estimate features ϕ by varying α.

N-best alignments. Table 1 shows translation results on the IWSLT translation task for the Development (IWSLT 2006) and two test corpora (IWSLT 2007 and 2008) using the Syntax Augmented grammar. In this table we vary the number of alternative alignments, consider first-best (1), 5, 10 and 50 best alternatives. We also experiment with lexical weights from the first-best alignment ($lex = 1st$) and directly from IBM Model 4 ($lex = m4$), while α controls the entropy of the normalized distribution over alternative alignments.

For the Syntax-Augmented grammar, using $lex = m4$ slightly increases the number of rules in the grammar, but only adds benefit for the 2007 test set. We continue to use $lex = m4$ for the remaining experiments since we do not want to discard rules based on the lexical weights. Increasing $N = 1$ to $N = 5$ brings significant improvements in translation quality on all 3 evaluation corpora, while increasing N further to $N = 10$ and $N = 50$ retain the improvements but at the cost of a significantly larger grammar and decoding times. Varying α to modify the entropy of the alignment distribution does not seem to have a consistent impact on translation quality; some test sets show improvements while others suffer.

N-best alignments (hierarchical grammar). Similar results with the purely hierarchical grammar are shown in Table 2. We see clear improvements when moving to $N = 5$, and even further small improvement up to $N = 10$, but a slight degradation going further to $N = 50$. Again, we do not see a clear benefit from varying α. Surprisingly, while Dev. scores are significantly lower with the purely hierarchical grammar compared to the Syntax Augmented grammar, unseen test set scores are very similar, and achieved at significantly lower decoding times. Since the number of features in ϕ are very similar for both models, it is unlikely that this discrepancy is solely due to overfitting during MER training. It is more likely that this discrepancy is related to the relative lengths of each evaluation corpus. The development corpus contains longer sentences on average than the evaluation corpora. The number of rules used in purely hierarchical grammar is significantly lower than in the Syntax Augmented grammar, and increasing N does not exhibit the same growth in the number of rules either. The Syntax Augmented grammar grows much faster since rule identified from alternative alignment candiates have syntactic nonterminal symbols and are less likely to be duplicates of already identified rules.

N'-best parses. Table 3 summarizes results when varying the number of alternative parses. These experiments use $\alpha = 1$, $lex = m4$ and 1-best alignments only. We also additionally track the number of nonterminal labels represented in the grammar. Using additional evidence from N'-best parses seems to have a overall slight negative impact on translation quality while taking significantly longer to perform decoding. The growth in the number of nonterminal labels and as a consequence the number of rules has a dramatic impact on decoding time and likely contributes to additional search errors. The one corpus where alternative parses ($N' = 10$) produces results comparable to using N best alignments is IWSLT 2008, which is also the corpus with the shortest sentences on average, thus reducing the potential impact of search error.

System	# Rules (1 sent.)	Dev	2007	2008	2007 Time (s)	2008 Time (s)
$N = 1$ (lex=1st)	400K	0.309	0.355	0.453	8108	8367
$N = 1$ ($\alpha = 1$ lex=m4)	420K	0.301	0.361	0.440	8024	8250
$N = 5$ ($\alpha = 1$ lex=m4)	680K	0.322	0.374	0.470	15376	15577
$N = 10$ ($\alpha = 1$ lex=m4)	900K	0.313	0.382	0.467	19298	19469
$N = 50$ ($\alpha = 1$ lex=m4)	1500K	0.316	0.370	0.478	29500	30894
$N = 10$ ($\alpha = 0.5$ lex=m4)	900K	0.315	0.395	0.477	20398	20118
$N = 50$ ($\alpha = 0.5$ lex=m4)	1500K	0.317	0.373	0.477	33682	34760
$N = 10$ ($\alpha = 2$ lex=m4)	900K	0.313	0.375	0.464	15117	15070
$N = 50$ ($\alpha = 2$ lex=m4)	1500K	0.315	0.373	0.488	26590	27126

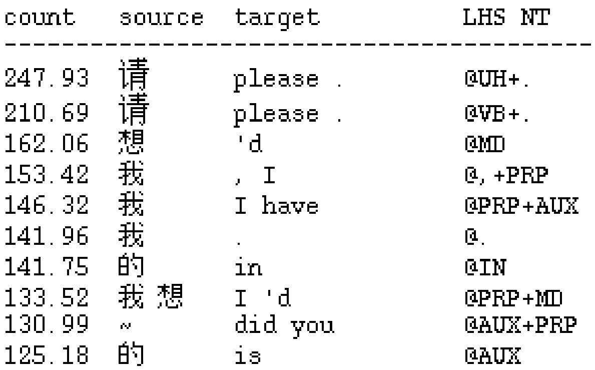
Table 1: Grammar statistics and translation quality (IBM-BLEU) on development (IWSLT 2006) and test set (IWSLT 2007, 2008) when integrating N-best alignments for alternative Syntax Augmented grammar configurations. # Rules reflect rules that are applicable to the first sentence in IWSLT 2007. Decoding times in seconds are cumulative over all sentences in respective test set.

System	# Rules (1 sent.)	Dev	2007	2008	2007 Time (s)	2008 Time (s)
Hier $N = 1$	10K	0.277	0.367	0.460	895	1451
Hier $N = 5$ ($\alpha = 1$)	12K	0.286	0.374	0.472	906	1476
Hier $N = 10$ ($\alpha = 1$)	13K	0.291	0.382	0.477	944	1516
Hier $N = 50$ ($\alpha = 1$)	14K	0.282	0.384	0.463	979	1596
Hier $N = 10$ ($\alpha = 0.5$)	13K	0.285	0.399	0.476	963	1547
Hier $N = 50$ ($\alpha = 0.5$)	14K	0.283	0.376	0.470	982	1599
Hier $N = 10$ ($\alpha = 2$)	13K	0.284	0.372	0.467	965	1570
Hier $N = 50$ ($\alpha = 2$)	14K	0.290	0.374	0.459	921	1483

Table 2: Grammar statistics and translation quality (IBM-BLEU) on development (IWSLT 2006) and test sets (IWSLT 2007, 2008) when integrating N-best alignments for purely hierarchical grammar configurations. # Rules reflect rules that are applicable to the first sentence in IWSLT 2007. Decoding times in seconds are cumulative over all sentences in respective test set.

4.3 Grammar Rules

Figure 1 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 left-hand-side nonterminal symbol. 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.



```
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 1: Top rules extracted by our method, but not the baseline.

5 Conclusion

In this work we have demonstrated the feasibility and benefits of widening the MT pipeline to include

System	# Rules (1 sent.)	# Labels	Dev	2007	2008	2007 Time (s)	2008 Time (s)
$N' = 1$	420K	10K	0.301	0.361	0.440	8024	8250
$N' = 5$	800K	15K	0.300	0.358	0.447	16930	15102
$N' = 10$	1079K	18K	0.299	0.361	0.460	26944	23662

Table 3: Grammar statistics and translation quality (IBM-BLEU) on development (IWSLT 2006) and test sets (IWSLT 2007, 2008) and when integrating N-best parses with the Syntax Augmented grammar. # Rules reflect rules that are applicable to the first sentence in IWSLT 2007. Decoding times in seconds are cumulative over all sentences in respective test set. All experiments in this table use $lex = m4$, $\alpha = 1$ and 1-best alignments.

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 when considering N-best alignments, while N'-best parses seem to have no impact on translation quality. Using a relatively small number of additional alternative alignments results in significant improvements in quality, with minimal impact on the number of rules in the grammar and the translation runtime for a hierarchical system, but at significantly increased grammar size and runtime for a syntax-augmented system. In future work we plan to focus on methods to take better advantage of the syntactic labels from alternative parse candidates.

Acknowledgments This work has been partly funded by GALE HR0011-06-2-0001. N. Smith is supported by NSF IIS-0836431 and an IBM faculty award.

References

Alfred V. Aho and Jeffrey D. Ullmann. 1969. Syntax directed translations and the pushdown assembler. *Journal of Computer and System Sciences*.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*.

Eugene Charniak. 2000. A maximum entropy-inspired parser. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics Conference (HLT/NAACL)*.

David Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of the Annual Meeting of the Association for Compuational Linguistics (ACL)*.

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

John DeNero, Dan Gillick, James Zhang, and Dan Klein. 2006. Why generative phrase models underperform surface heuristics. In *Proceedings of the Workshop on Statistical Machine Translation, ACL*.

Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008. Generalizing word lattice translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Michael Galley, Mark Hopkins, Kevin Knight, and Daniel Marcu. 2006. Scalable inferences and training of context-rich syntax translation models. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics Conference (HLT/NAACL)*.

Kuzman Ganchev, Joao V. Graca, and Ben Taskar. 2008. Better alignments = better translations? In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

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

Daniel Marcu and William Wong. 2002. A phrase-based, joint probability model for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Daniel Marcu, Wei Wang, Abdessamad Echihabi, and Kevin Knight. 2006. SPMT: Statistical machine translation with syntactified target language phrases. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Sydney, Australia.

Haitao Mi, Liang Huang, and Qun Liu. 2008. Forest-based translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Franz J. Och and Hermann Ney. 2003. A systematic comparison of various alignment models. *Computational Linguistics*.

Franz J. Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*.

Franz J. Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the Annual Meeting of the Association for Compuational Linguistics (ACL)*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the Annual Meeting of the Association for Compuational Linguistics (ACL)*.

Michael Paul. 2006. Overview of the IWSLT 2006 evaluation campaign. In *Proceedings of the International Workshop on Spoken Language Translation (IWSLT)*.

Chris Quirk and Simon Corston-Oliver. 2006. The impact of parse quality on syntactically-informed statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Chris Quirk, Arul Menezes, and Collin Cherry. 2005. Dependency tree translation: Syntactically informed mt. In *Proceedings of the Annual Meeting of the Association for Compuational Linguistics (ACL)*.

Remko Scha. 1990. Taaltheorie en taaltechnologie; competence en performance (language theory and language technology; competence and performance). *Computertoepassingen in de Neerlandistiek*.

Mark Steedman. 2000. *The Syntactic Process*. MIT Press.

Ashish Venugopal, Andreas Zollmann, and Stephan Vogel. 2007. An efficient two-pass approach to synchronous-cfg driven statistical mt. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics Conference (HLT/NAACL)*.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*.

Yong-Zeng Xue, Sheng Li, Tie-Jun Zhao, Mu-Yun Yang, and Jun Li. 2006. Bilingual phrase extraction from n-best alignments. In *Proceedings of the International Conference on Innovative Computing, Information and Control (ICICIC)*.

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

Andreas Zollmann, Ashish Venugopal, Franz J. Och, and Jay Ponte. 2008. A systematic comparison of phrase-based, hierarchical and a systematic comparison of phrase-based, hierarchical and syntax-augmented statistical MT. In *Proceedings of the Conference on Computational Linguistics (COLING)*.