Discriminative Pruning for Discriminative ITG Alignment

Shujie Liu[†], **Chi-Ho Li**[‡] and **Ming Zhou**[‡]
[†]School of Computer Science and Technology
Harbin Institute of Technology, Harbin, China
shujieliu@mtlab.hit.edu.cn
[‡]Microsoft Research Asia, Beijing, China
{chl, mingzhou}@microsoft.com

Abstract

While Inversion Transduction Grammar (ITG) has regained more and more attention in recent years, it still suffers from the major obstacle of speed. We propose a discriminative ITG pruning framework using Minimum Error Rate Training and various features from previous work on ITG alignment. Experiment results show that it is superior to all existing heuristics in ITG pruning. On top of the pruning framework, we also propose a discriminative ITG alignment model using hierarchical phrase pairs, which improves both F-score and Bleu score over the baseline alignment system of GIZA++.

1 Introduction

Inversion transduction grammar (ITG) (Wu, 1997) is an adaptation of SCFG to bilingual parsing. It does synchronous parsing of two languages with phrasal and word-level alignment as by-product. For this reason ITG has gained more and more attention recently in the word alignment community (Zhang and Gildea, 2005; Cherry and Lin, 2006; Haghighi *et al.*, 2009).

A major obstacle in ITG alignment is speed. The original (unsupervised) ITG algorithm has complexity of $O(n^6)$. When extended to supervised/discriminative framework, ITG runs even more slowly. Therefore all attempts to ITG alignment come with some pruning method. For example, Haghighi *et al.* (2009) do pruning based on the probabilities of links from a simpler alignment model (viz. HMM); Zhang and Gildea (2005) propose Tic-tac-toe pruning, which is based on the Model 1 probabilities of word pairs inside and outside a pair of spans.

As all the principles behind these techniques have certain contribution in making good pruning decision, it is tempting to incorporate all these features in ITG pruning. In this paper, we propose a novel discriminative pruning framework for discriminative ITG. The pruning model uses no more training data than the discriminative ITG parser itself, and it uses a log-linear model to integrate all features that help identify the correct span pair (like Model 1 probability and HMM posterior). On top of the discriminative pruning method, we also propose a discriminative ITG alignment system using hierarchical phrase pairs.

In the following, some basic details on the ITG formalism and ITG parsing are first reviewed (Sections 2 and 3), followed by the definition of pruning in ITG (Section 4). The "Discriminative Pruning for Discriminative ITG" model (DPDI) and our discriminative ITG (DITG) parsers will be elaborated in Sections 5 and 6 respectively. The merits of DPDI and DITG are illustrated with the experiments described in Section 7.

2 Basics of ITG

The simplest formulation of ITG contains three types of rules: terminal unary rules $X \rightarrow e/f$, where e and f represent words (possibly a null word, ε) in the English and foreign language respectively, and the binary rules $X \rightarrow [X, X]$ and $X \rightarrow \langle X, X \rangle$, which refer to that the component English and foreign phrases are combined in the same and inverted order respectively.

From the viewpoint of word alignment, the terminal unary rules provide the links of word pairs, whereas the binary rules represent the reordering factor. One of the merits of ITG is that it is less biased towards short-distance reordering.

Such a formulation has two drawbacks. First of all, it imposes a 1-to-1 constraint in word alignment. That is, a word is not allowed to align to more than one word. This is a strong limitation as no idiom or multi-word expression is allowed to align to a single word on the other side. In fact there have been various attempts in relaxing the 1-to-1 constraint. Both ITG alignment

Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 316–324,
Uppsala, Sweden, 11-16 July 2010. ©2010 Association for Computational Linguistics

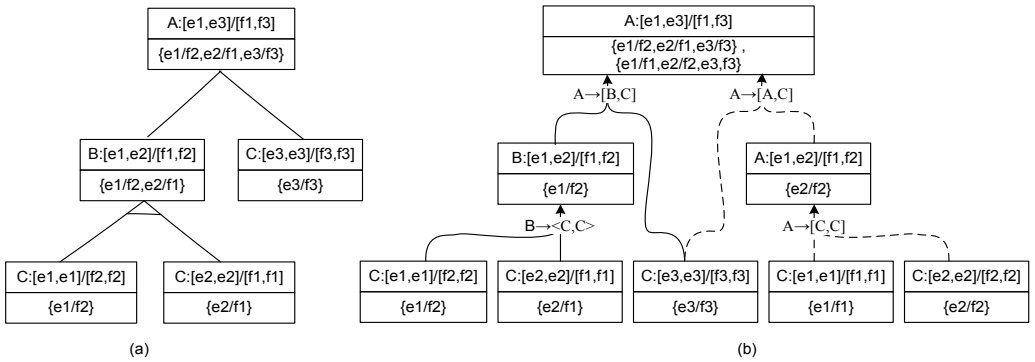

Figure 1: Example ITG parses in graph (a) and hypergraph (b).

approaches with and without this constraint will be elaborated in Section 6.

Secondly, the simple ITG leads to redundancy if word alignment is the sole purpose of applying ITG. For instance, there are two parses for three consecutive word pairs, viz. $[a/a'\,[b/b'\,c/c']\,]$ and $[[a/a'\,b/b']\,c/c']$. The problem of redundancy is fixed by adopting ITG normal form. In fact, normal form is the very first key to speeding up ITG. The ITG normal form grammar as used in this paper is described in Appendix A.

3 Basics of ITG Parsing

Based on the rules in normal form, ITG word alignment is done in a similar way to chart parsing (Wu, 1997). The base step applies all relevant terminal unary rules to establish the links of word pairs. The word pairs are then combined into span pairs in all possible ways. Larger and larger span pairs are recursively built until the sentence pair is built.

Figure 1(a) shows one possible derivation for a toy example sentence pair with three words in each sentence. Each node (rectangle) represents a pair, marked with certain phrase category, of foreign span (F-span) and English span (E-span) (the upper half of the rectangle) and the associated alignment hypothesis (the lower half). Each graph like Figure 1(a) shows only one derivation and also only one alignment hypothesis.

The various derivations in ITG parsing can be compactly represented in hypergraph (Klein and Manning, 2001) like Figure 1(b). Each hypernode (rectangle) comprises both a span pair (upper half) and the list of possible alignment hypotheses (lower half) for that span pair. The hyperedges show how larger span pairs are derived from smaller span pairs. Note that a hypernode may have more than one alignment hypothesis, since a hypernode may be derived through more than one hyperedge (e.g. the topmost hypernode in Figure 1(b)). Due to the use of normal form, the hypotheses of a span pair are different from each other.

4 Pruning in ITG Parsing

The ITG parsing framework has three levels of pruning:

1) To discard some unpromising span pairs;
2) To discard some unpromising F-spans and/or E-spans;
3) To discard some unpromising alignment hypotheses for a particular span pair.

The second type of pruning (used in Zhang *et. al.* (2008)) is very radical as it implies discarding too many span pairs. It is empirically found to be highly harmful to alignment performance and therefore not adopted in this paper.

The third type of pruning is equivalent to minimizing the beam size of alignment hypotheses in each hypernode. It is found to be well handled by the K-Best parsing method in Huang and Chiang (2005). That is, during the bottom-up construction of the span pair repertoire, each span pair keeps only the best alignment hypothesis. Once the complete parse tree is built, the k-best list of the topmost span is obtained by minimally expanding the list of alignment hypotheses of minimal number of span pairs.

The first type of pruning is equivalent to minimizing the number of hypernodes in a hypergraph. The task of ITG pruning is defined in this paper as the first type of pruning; i.e. the search for, given an F-span, the minimal number of E-spans which are the most likely counterpart of that F-span.[1] The pruning method should maintain a balance between efficiency (run as quickly as possible) and performance (keep as many correct span pairs as possible).

[1] Alternatively it can be defined as the search of the minimal number of E-spans per F-span. That is simply an arbitrary decision on how the data are organized in the ITG parser.

A naïve approach is that the required pruning method outputs a score given a span pair. This score is used to rank all E-spans for a particular F-span, and the score of the correct E-span should be in general higher than most of the incorrect ones.

5 The DPDI Framework

DPDI, the discriminative pruning model proposed in this paper, assigns score to a span pair $(\bar{f}, \bar{e})$ as probability from a log-linear model:

$$P(\bar{e}|\bar{f}) = \frac{exp(\sum_i \lambda_i \Psi_i(\bar{f}, \bar{e}))}{\sum_{\bar{e}' \in E} exp(\sum_i \lambda_i \Psi_i(\bar{f}, \bar{e}'))} \quad (1)$$

where each $\Psi_i(\bar{f}, \bar{e})$ is some feature about the span pair, and each λ is the weight of the corresponding feature. There are three major questions to this model:

1) How to acquire training samples? (Section 5.1)
2) How to train the parameters λ ? (Section 5.2)
3) What are the features? (Section 5.3)

5.1 Training Samples

Discriminative approaches to word alignment use manually annotated alignment for sentence pairs. Discriminative pruning, however, handles not only a sentence pair but every possible span pair. The required training samples consist of various F-spans and their corresponding E-spans.

Rather than recruiting annotators for marking span pairs, we modify the parsing algorithm in Section 3 so as to produce span pair annotation out of sentence-level annotation. In the base step, only the word pairs listed in sentence-level annotation are inserted in the hypergraph, and the recursive steps are just the same as usual.

If the sentence-level annotation satisfies the alignment constraints of ITG, then each F-span will have only one E-span in the parse tree. However, in reality there are often the cases where a foreign word aligns to more than one English word. In such cases the F-span covering that foreign word has more than one corresponding E-spans. Consider the example in Figure 2, where the golden links in the alignment annotation are $e1/f1$, $e2/f1$, and $e3/f2$; i.e. the foreign word $f1$ aligns to both the English words $e1$ and $e2$. Therefore the F-span $[f1, f1]$ aligns to the E-span $[e1, e1]$ in one hypernode and to the E-span $[e2, e2]$ in another hypernode. When such situation happens, we calculate the product of the inside and outside probability of each alignment hypothesis of the span pair, based on the probabilities of the links from some simpler alignment model[2]. The E-span with the most probable hypothesis is selected as the alignment of the F-span.

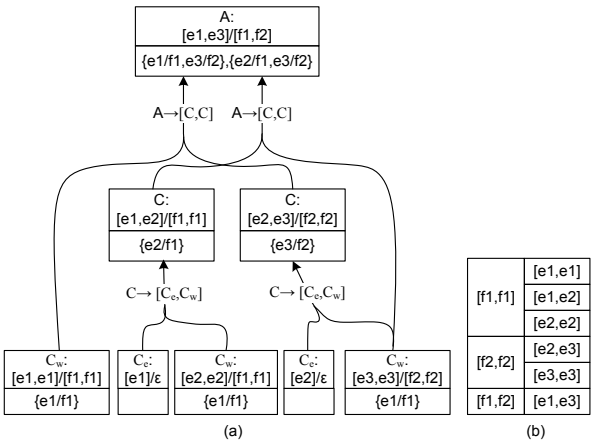

Figure 2: Training sample collection.

Table (b) lists, for the hypergraph in (a), the candidate E-spans for each F-span.

It should be noted that this automatic span pair annotation may violate some of the links in the original sentence-level alignment annotation. We have already seen how the 1-to-1 constraint in ITG leads to the violation. Another situation is the 'inside-out' alignment pattern (c.f. Figure 3). The ITG reordering constraint cannot be satisfied unless one of the links in this pattern is removed.

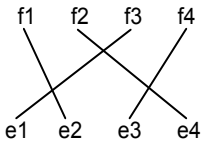

Figure 3: An example of inside-out alignment

The training samples thus obtained are positive training samples. If we apply some classifier for parameter training, then negative samples are also needed. Fortunately, our parameter training does not rely on any negative samples.

5.2 MERT for Pruning

Parameter training of DPDI is based on Minimum Error Rate Training (MERT) (Och, 2003), a widely used method in SMT. MERT for SMT estimates model parameters with the objective of minimizing certain measure of translation errors (or maximizing certain performance measure of translation quality) for a development corpus. Given an SMT system which produces, with

[2] The formulae of the inside and outside probability of a span pair will be elaborated in Section 5.3. The simpler alignment model we used is HMM.

model parameters λ_1^M, the K-best candidate translations $\hat{e}(f_s; \lambda_1^M)$ for a source sentence f_s, and an error measure $E(r_s, e_{s,k})$ of a particular candidate $e_{s,k}$ with respect to the reference translation r_s, the optimal parameter values will be:

$$\hat{\lambda}_1^M = \underset{\lambda_1^M}{argmin}\left\{\sum_{s=1}^{S} E\left(r_s, \hat{e}(f_s; \lambda_1^M)\right)\right\}$$

$$= \underset{\lambda_1^M}{argmin}\left\{\sum_{s=1}^{S}\sum_{k=1}^{K} E\left(r_s, e_{s,k}\right)\delta(\hat{e}(f_s; \lambda_1^M), e_{s,k})\right\}$$

DPDI applies the same equation for parameter tuning, with different interpretation of the components in the equation. Instead of a development corpus with reference translations, we have a collection of training samples, each of which is a pair of F-span (f_s) and its corresponding E-span (r_s). These samples are acquired from some manually aligned dataset by the method elaborated in Section 5.1. The ITG parser outputs for each f_s a K-best list of E-spans $\hat{e}(f_s; \lambda_1^M)$ based on the current parameter values λ_1^M.

The error function is based on the presence and the rank of the correct E-span in the K-best list:

$$E\left(r_s, \hat{e}(f_s; \lambda_1^M)\right) = \begin{cases} -rank(r_s) \; if \; r_s \in \hat{e}(f_s; \lambda_1^M) \\ penalty \quad otherwise \end{cases} \tag{2}$$

where $rank(r_s)$ is the (0-based) rank of the correct E-span r_s in the K-best list $\hat{e}(f_s; \lambda_1^M)$. If r_s is not in the K-best list at all, then the error is defined to be $penalty$, which is set as *-100000* in our experiments. The rationale underlying this error function is to keep as many correct E-spans as possible in the K-best lists of E-spans, and push the correct E-spans upward as much as possible in the K-best lists.

This new error measure leads to a change in details of the training algorithm. In MERT for SMT, the interval boundaries at which the performance or error measure changes are defined by the *upper envelope* (illustrated by the dash line in Figure 4(a)), since the performance/error measure depends on the best candidate translation. In MERT for DPDI, however, the error measure depends on the correct E-span rather than the E-span leading to the highest system score. Thus the interval boundaries are the intersections between the correct E-span and all other candidate E-spans (as shown in Figure 4(b)). The rank of the correct E-span in each interval can then be figured out as shown in Figure 4(c). Finally, the error measure in each interval can be calculated by Equation (2) (as shown in Figure 4(d)). All other steps in MERT for DPDI are the same as that for SMT.

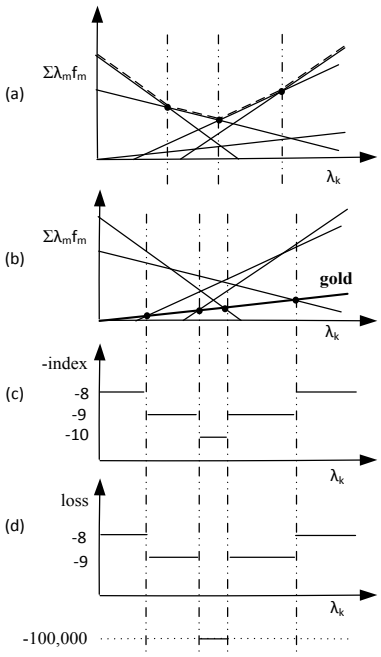

Figure 4: MERT for DPDI

Part (a) shows how intervals are defined for SMT and part (b) for DPDI. Part (c) obtains the rank of correct E-spans in each interval and part (d) the error measure. Note that the beam size (max number of E-spans) for each F-span is 10.

5.3 Features

The features used in DPDI are divided into three categories:

1) Model 1-based probabilities. Zhang and Gildea (2005) show that Model 1 (Brown *et al.*, 1993; Och and Ney., 2000) probabilities of the word pairs inside and outside a span pair ($[e_{i1}, e_{i2}]/[f_{j1}, f_{j2}]$) are useful. Hence these two features:
   a) Inside probability (i.e. probability of word pairs within the span pair):

$$p_{inc}\left(e_{i1,i2}|f_{j1,j2}\right) = \prod_{i\in(i1,i2)}\sum_{j\in(j1,j2)}\frac{1}{(j2-j1)}p_{M1}(e_i|f_j)$$

   b) Outside probability (i.e. probability of the word pairs outside the span pair):

$$p_{out}\left(e_{i1,i2}|f_{j1,j2}\right) = \prod_{i\notin(i1,i2)}\sum_{j\notin(j1,j2)}\frac{1}{(J-j2+j1)}p_{M1}(e_i|f_j)$$

   where J is the length of the foreign sentence.
2) Heuristics. There are four features in this category. The features are explained with the

example of Figure 5, in which the span pair in interest is $[e2, e3]/[f1, f2]$. The four links are produced by some simpler alignment model like HMM. The word pair $e2/f1$ is the only link in the span pair. The links $e4/f2$ and $e3/f3$ are inconsistent with the span pair.[3]

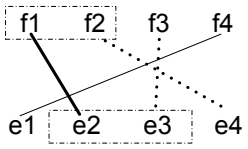

Figure 5: Example for heuristic features

a) Link ratio: $\frac{2\times\#links}{flen+elen}$
where $\#links$ is the number of links in the span pair, and $flen$ and $elen$ are the length of the foreign and English spans respectively. The feature value of the example span pair is (2*1)/(2+2)=0.5.

b) inconsistent link ratio: $\frac{2\times\#links_{incon}}{flen+elen}$
where $\#links_{incon}$ is the number of links which are inconsistent with the phrase pair according to some simpler alignment model (e.g. HMM). The feature value of the example is (2*2)/(2+2) =1.0.

c) Length ratio: $\left|\frac{flen}{elen} - ratio_{avg}\right|$
where $ratio_{avg}$ is defined as the average ratio of foreign sentence length to English sentence length, and it is estimated to be around 1.15 in our training dataset. The rationale underlying this feature is that the ratio of span length should not be too deviated from the average ratio of sentence length. The feature value for the example is |2/2-1.15|=0.15.

d) Position Deviation: $\left|pos_{\bar{f}} - pos_{\bar{e}}\right|$
where $pos_{\bar{f}}$ refers to the position of the F-span in the entire foreign sentence, and it is defined as $\frac{1}{2J}\left(start_{\bar{f}} + end_{\bar{f}}\right)$, $start_{\bar{f}}$ /$end_{\bar{f}}$ being the position of the first/last word of the F-span in the foreign sentence. $pos_{\bar{e}}$ is defined similarly. The rationale behind this feature is the monotonic assumption, i.e. a phrase of the foreign sentence usually occupies roughly the same position of the equivalent English phrase. The feature value for the example is |(1+2)/(2*4)-(2+3)/(2*4)| =0.25.

3) HMM-based probabilities. Haghighi *et al.* (2009) show that posterior probabilities from the HMM alignment model is useful for pruning. Therefore, we design two new features by replacing the link count in link ratio and inconsistent link ratio with the sum of the link's posterior probability.

6 The DITG Models

The discriminative ITG alignment can be conceived as a two-staged process. In the first stage DPDI selects good span pairs. In the second stage good alignment hypotheses are assigned to the span pairs selected by DPDI. Two discriminative ITG (DITG) models are investigated. One is word-to-word DITG (henceforth W-DITG), which observes the 1-to-1 constraint on alignment. Another is DITG with hierarchical phrase pairs (henceforth HP-DITG), which relaxes the 1-to-1 constraint by adopting hierarchical phrase pairs in Chiang (2007).

Each model selects the best alignment hypotheses of each span pair, given a set of features. The contributions of these features are integrated through a log linear model (similar to Liu *et al.*, 2005; Moore, 2005) like Equation (1). The discriminative training of the feature weights is again MERT (Och, 2003). The MERT module for DITG takes alignment F-score of a sentence pair as the performance measure. Given an input sentence pair and the reference annotated alignment, MERT aims to maximize the F-score of DITG-produced alignment. Like SMT (and unlike DPDI), it is the upper envelope which defines the intervals where the performance measure changes.

6.1 Word-to-word DITG

The following features about alignment link are used in W-DITG:

1) Word pair translation probabilities trained from HMM model (Vogel, *et.al.*, 1996) and IBM model 4 (Brown *et.al.*, 1993; Och and Ney, 2000).
2) Conditional link probability (Moore, 2005).
3) Association score rank features (Moore *et al.*, 2006).
4) Distortion features: counts of inversion and concatenation.
5) Difference between the relative positions of the words. The relative position of a word in a sentence is defined as the posi-

[3] An inconsistent link connects a word within the phrase pair to some word outside the phrase pair. C.f. Deng *et al.* (2008)

tion of the word divided by sentence length.

6) Boolean features like whether a word in the word pair is a stop word.

6.2 DITG with Hierarchical Phrase Pairs

The 1-to-1 assumption in ITG is a serious limitation as in reality there are always segmentation or tokenization errors as well as idiomatic expressions. Wu (1997) proposes a bilingual segmentation grammar extending the terminal rules by including phrase pairs. Cherry and Lin (2007) incorporate phrase pairs in phrase-based SMT into ITG, and Haghighi *et al.* (2009) introduce Block ITG (BITG), which adds 1-to-many or many-to-1 terminal unary rules.

It is interesting to see if DPDI can benefit the parsing of a more realistic ITG. HP-DITG extends Cherry and Lin's approach by not only employing simple phrase pairs but also hierarchical phrase pairs (Chiang, 2007). The grammar is enriched with rules of the format: $X \rightarrow \bar{e}_i / \bar{f}_i$ where $\bar{e}_i$ and $\bar{f}_i$ refer to the English and foreign side of the *i*-th (simple/hierarchical) phrase pair respectively.

As example, if there is a simple phrase pair $X \rightarrow \langle North\ Korea, 北\ 朝鲜 \rangle$, then it is transformed into the ITG rule $C \rightarrow$ "*North Korea*"/"北 朝鲜". During parsing, each span pair does not only examine all possible combinations of sub-span pairs using binary rules, but also checks if the yield of that span pair is exactly the same as that phrase pair. If so, then the alignment links within the phrase pair (which are obtained in standard phrase pair extraction procedure) are taken as an alternative alignment hypothesis of that span pair.

For a hierarchical phrase pair like $X \rightarrow \langle X_1\ of\ X_2, X_2\ 的\ X_1 \rangle$, it is transformed into the ITG rule $C \rightarrow$ "X_1 *of* X_2"/"X_2 的 X_1" during parsing, each span pair checks if it contains the lexical anchors *"of"* and "的", and if the remaining words in its yield can form two sub-span pairs which fit the reordering constraint among X_1 and X_2. (Note that span pairs of any category in the ITG normal form grammar can substitute for X_1 or X_2.) If both conditions hold, then the span pair is assigned an alignment hypothesis which combines the alignment links among the lexical anchors ($like\ of$/的) and those links among the sub-span pairs.

HP-ITG acquires the rules from HMM-based word-aligned corpus using standard phrase pair extraction as stated in Chiang (2007). The rule probabilities and lexical weights in both English-to-foreign and foreign-to-English directions are estimated and taken as features, in addition to those features in W-DITG, in the discriminative model of alignment hypothesis selection.

7 Evaluation

DPDI is evaluated against the baselines of Tic-tac-toe (TTT) pruning (Zhang and Gildea, 2005) and Dynamic Program (DP) pruning (Haghighi *et al.*, 2009; DeNero *et al.*, 2009) with respect to Chinese-to-English alignment and translation. Based on DPDI, HP-DITG is evaluated against the alignment systems GIZA++ and BITG.

7.1 Evaluation Criteria

Four evaluation criteria are used in addition to the time spent on ITG parsing. We will first evaluate pruning regarding the pruning decisions themselves. That is, the first evaluation metric, pruning error rate (henceforth PER), measures how many correct E-spans are discarded. The major drawback of PER is that not all decisions in pruning would impact on alignment quality, since certain F-spans are of little use to the entire ITG parse tree.

An alternative criterion is the upper bound on alignment F-score, which essentially measures how many links in annotated alignment can be kept in ITG parse. The calculation of F-score upper bound is done in a bottom-up way like ITG parsing. All leaf hypernodes which contain a correct link are assigned a score (known as hit) of 1. The hit of a non-leaf hypernode is based on the sum of hits of its daughter hypernodes. The maximal sum among all hyperedges of a hypernode is assigned to that hypernode. Formally,

$$hit\big(X[\bar{f}, \bar{e}]\big) = \max_{Y,Z,\bar{f}_1,\bar{e}_1,\bar{f}_2,\bar{e}_2} \big(hit\big(Y[\bar{f}_1, \bar{e}_1]\big) + hit[\bar{f}_2, \bar{e}_2]\big)$$

$$hit(C_w[u, v]) = \begin{cases} 1 & if\ \langle u, v \rangle \in R \\ 0 & otherwise \end{cases}$$

$$hit(C_e) = 0; hit\big(C_f\big) = 0$$

where X, Y, Z are variables for the categories in ITG grammar, and R comprises the golden links in annotated alignment. C_w, C_e, C_f are defined in Appendix A.

Figure 6 illustrates the calculation of the hit score for the example in Section 5.1/Figure 2. The upper bound of recall is the hit score divided by the total number of golden links. The upper

ID	pruning	beam size	pruning/total time cost	PER	F-UB	F-score
1	DPDI	10	72''/3'03''	**4.9%**	**88.5%**	**82.5%**
2	TTT	10	**58''/2'38''**	8.6%	87.5%	81.1%
3	TTT	20	53''/6'55''	5.2%	**88.6%**	82.4%
4	DP	--	11''/6'01''	12.1%	86.1%	80.5%

Table 1: Evaluation of DPDI against TTT (Tic-tac-toe) and DP (Dynamic Program) for W-DITG

ID	pruning	beam size	pruning/total time cost	PER	F-UB	F-score
1	DPDI	10	72''/5'18''	**4.9%**	**93.9%**	**87.0%**
2	TTT	10	**58''/4'51''**	8.6%	93.0%	84.8%
3	TTT	20	53''/12'5''	5.2%	**94.0%**	86.5%
4	DP	--	11''/15'39''	12.1%	91.4%	83.6%

Table 2: Evaluation of DPDI against TTT (Tic-tac-toe) and DP (Dynamic Program) for HP-DITG.

bound of precision, which should be defined as the hit score divided by the number of links produced by the system, is almost always 1.0 in practice. The upper bound of alignment F-score can thus be calculated as well.

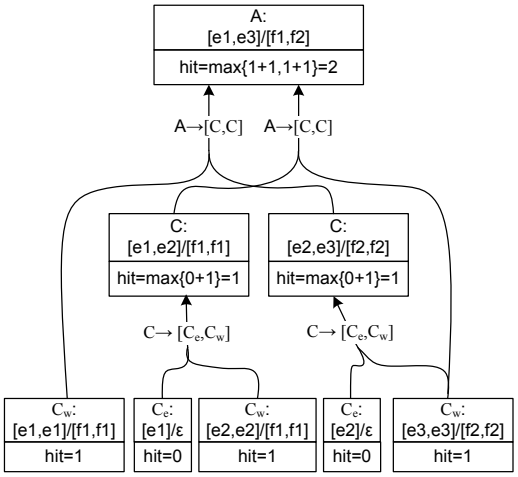

Figure 6: Recall Upper Bound Calculation

Finally, we also do end-to-end evaluation using both F-score in alignment and Bleu score in translation. We use our implementation of hierarchical phrase-based SMT (Chiang, 2007), with standard features, for the SMT experiments.

7.2 Experiment Data

Both discriminative pruning and alignment need training data and test data. We use the manually aligned Chinese-English dataset as used in Haghighi *et al.* (2009). The 491 sentence pairs in this dataset are adapted to our own Chinese word segmentation standard. 250 sentence pairs are used as training data and the other 241 are test data. The corresponding numbers of F-spans in training and test data are 4590 and 3951 respectively.

In SMT experiments, the bilingual training dataset is the NIST training set excluding the Hong Kong Law and Hong Kong Hansard, and our 5-gram language model is trained from the Xinhua section of the Gigaword corpus. The NIST'03 test set is used as our development corpus and the NIST'05 and NIST'08 test sets are our test sets.

7.3 Small-scale Evaluation

The first set of experiments evaluates the performance of the three pruning methods using the small 241-sentence set. Each pruning method is plugged in both W-DITG and HP-DITG. IBM Model 1 and HMM alignment model are re-implemented as they are required by the three ITG pruning methods.

The results for W-DITG are listed in Table 1. Tests 1 and 2 show that with the same beam size (i.e. number of E-spans per F-span), although DPDI spends a bit more time (due to the more complicated model), DPDI makes far less incorrect pruning decisions than the TTT. In terms of F-score upper bound, DPDI is 1 percent higher. DPDI achieves even larger improvement in actual F-score.

To enable TTT achieving similar F-score or F-score upper bound, the beam size has to be doubled and the time cost is more than twice the original (c.f. Tests 1 and 3 in Table 1) .

The DP pruning in Haghighi *et.al.* (2009) performs much poorer than the other two pruning methods. In fact, we fail to enable DP achieve the same F-score upper bound as the other two methods before DP leads to intolerable memory consumption. This may be due to the use of different HMM model implementations between our work and Haghighi *et.al.* (2009).

Table 2 lists the results for HP-DITG. Roughly the same observation as in W-DITG can be made. In addition to the superiority of DPDI, it can also be noted that HP-DITG achieves much higher F-score and F-score upper bound. This shows that

hierarchical phrase is a powerful tool in rectifying the 1-to-1 constraint in ITG.

Note also that while TTT in Test 3 gets roughly the same F-score upper bound as DPDI in Test 1, the corresponding F-score is slightly worse. A possible explanation is that better pruning not only speeds up the parsing/alignment process but also guides the search process to focus on the most promising region of the search space.

7.4 Large-scale End-to-End Experiment

ID	Pruning	beam size	time cost	Bleu-05	Bleu-08
1	DPDI	10	1092h	**38.57**	**28.31**
2	TTT	10	**972h**	37.96	27.37
3	TTT	20	2376h	38.13	27.58
4	DP	--	2068h	37.43	27.12

Table 3: Evaluation of DPDI against TTT and DP for HP-DITG

ID	WA-Model	F-Score	Bleu-05	Bleu-08
1	HMM	80.1%	36.91	26.86
2	Giza++	84.2%	37.70	27.33
3	BITG	85.9%	37.92	27.85
4	HP-DITG	**87.0%**	**38.57**	**28.31**

Table 4: Evaluation of DPDI against HMM, Giza++ and BITG

Table 3 lists the word alignment time cost and SMT performance of different pruning methods. HP-DITG using DPDI achieves the best Bleu score with acceptable time cost. Table 4 compares HP-DITG to HMM (Vogel, et al., 1996), GIZA++ (Och and Ney, 2000) and BITG (Haghighi *et al.*, 2009). It shows that HP-DITG (with DPDI) is better than the three baselines both in alignment F-score and Bleu score. Note that the Bleu score differences between HP-DITG and the three baselines are statistically significant (Koehn, 2004).

An explanation of the better performance by HP-DITG is the better phrase pair extraction due to DPDI. On the one hand, a good phrase pair often fails to be extracted due to a link inconsistent with the pair. On the other hand, ITG pruning can be considered as phrase pair selection, and good ITG pruning like DPDI guides the subsequent ITG alignment process so that less links inconsistent to good phrase pairs are produced. This also explains (in Tables 2 and 3) why DPDI with beam size 10 leads to higher Bleu than TTT with beam size 20, even though both pruning methods lead to roughly the same alignment F-score.

8 Conclusion and Future Work

This paper reviews word alignment through ITG parsing, and clarifies the problem of ITG pruning. A discriminative pruning model and two discriminative ITG alignments systems are proposed. The pruning model is shown to be superior to all existing ITG pruning methods, and the HP-DITG alignment system is shown to improve state-of-the-art alignment and translation quality.

The current DPDI model employs a very limited set of features. Many features are related only to probabilities of word pairs. As the success of HP-DITG illustrates the merit of hierarchical phrase pair, in future we should investigate more features on the relationship between span pair and hierarchical phrase pair.

Appendix A. The Normal Form Grammar

Table 5 lists the ITG rules in normal form as used in this paper, which extend the normal form in Wu (1997) so as to handle the case of alignment to null.

1	$S \rightarrow A \mid B \mid C$
2	$A \rightarrow [A\ B] \mid [A\ C] \mid [B\ B] \mid [BC] \mid [C\ B] \mid [C\ C]$
3	$B \rightarrow \langle A\ A\rangle \mid \langle A\ C\rangle \mid \langle B\ A\rangle \mid \langle B\ C\rangle$
	$B \rightarrow \langle C\ A\rangle \mid \langle C\ C\rangle$
4	$C \rightarrow C_w \mid C_{fw} \mid C_{ew}$
5	$C \rightarrow [C_{ew}\ C_{fw}]$
6	$C_w \rightarrow u/v$
7	$C_e \rightarrow \varepsilon/v; C_f \rightarrow u/\varepsilon$
8	$C_{em} \rightarrow C_e \mid [C_{em}\ C_e]; C_{fm} \rightarrow C_f \mid [C_{fm}\ C_f]$
9	$C_{ew} \rightarrow [C_{em}\ C_w]; C_{fw} \rightarrow [C_{fm}\ C_w]$

Table 5: ITG Rules in Normal Form

In these rules, S is the Start symbol; A is the category for concatenating combination whereas B for inverted combination. Rules (2) and (3) are inherited from Wu (1997). Rules (4) divide the terminal category C into subcategories. Rule schema (6) subsumes all terminal unary rules for some English word u and foreign word v, and rule schemas (7) are unary rules for alignment to null. Rules (8) ensure all words linked to null are combined in left branching manner, while rules (9) ensure those words linked to null combine with some following, rather than preceding, word pair. (Note: Accordingly, all sentences must be ended by a special token $\langle end\rangle$, otherwise the last word(s) of a sentence cannot be linked to null.) If there are both English and foreign words linked to null, rule (5) ensures that those English

words linked to null precede those foreign words linked to null.

References

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

Colin Cherry and Dekang Lin. 2006. *Soft Syntactic Constraints for Word Alignment through Discriminative Training.* In *Proceedings of* ACL-COLING.

Colin Cherry and Dekang Lin. 2007. *Inversion Transduction Grammar for Joint Phrasal Translation Modeling.* In *Proceedings of* SSST, NAACL-HLT, Pages:17-24.

David Chiang. 2007. *Hierarchical Phrase-based Translation. Computational Linguistics*, 33(2).

John DeNero, Mohit Bansal, Adam Pauls, and Dan Klein. 2009. *Efficient Parsing for Transducer Grammars.* In *Proceedings of* NAACL, Pages:227-235.

Alexander Fraser and Daniel Marcu. 2006. *Semi-Supervised Training for StatisticalWord Alignment.* In *Proceedings of* ACL, Pages:769-776.

Aria Haghighi, John Blitzer, John DeNero, and Dan Klein. 2009. *Better Word Alignments with Supervised ITG Models.* In *Proceedings of* ACL, Pages: 923-931.

Liang Huang and David Chiang. 2005. *Better k-best Parsing.* In *Proceedings of* IWPT 2005, Pages:173-180.

Franz Josef Och and Hermann Ney. 2000. *Improved statistical alignment models.* In *Proceedings of* ACL. Pages: 440-447

Franz Josef Och. 2003. *Minimum error rate training in statistical machine translation.* In *Proceedings of* ACL, Pages:160-167.

Dan Klein and Christopher D. Manning. 2001. *Parsing and Hypergraphs.* In *Proceedings of* IWPT, Pages:17-19

Philipp Koehn. 2004. *Statistical Significance Tests for Machine Translation Evaluation.* In *Proceedings of* EMNLP, Pages: 388-395.

Yang Liu, Qun Liu and Shouxun Lin. 2005. *Log-linear models for word alignment.* In *Proceedings of* ACL, Pages: 81-88.

Robert Moore. 2005. *A Discriminative Framework for Bilingual Word Alignment.* In *Proceedings of* EMNLP 2005, Pages: 81-88.

Robert Moore, Wen-tau Yih, and Andreas Bode. 2006. *Improved Discriminative Bilingual Word Alignment.* In *Proceedings of* ACL, Pages: 513-520.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. *HMM-based word alignment in statistical translation.* In *Proceedings of* COLING, Pages: 836-841.

Stephan Vogel. 2005. *PESA: Phrase Pair Extraction as Sentence Splitting.* In *Proceedings of* MT Summit.

Dekai Wu. 1997. *Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora. Computational Linguistics*, 23(3).

Hao Zhang and Daniel Gildea. 2005. *Stochastic Lexicalized Inversion Transduction Grammar for Alignment.* In *Proceedings of* ACL.

Hao Zhang, Chris Quirk, Robert Moore, and Daniel Gildea. 2008. *Bayesian learning of non-compositional phrases with synchronous parsing.* In *Proceedings of* ACL, Pages: 314-323.