Better Word Alignments with Supervised ITG Models

Aria Haghighi, John Blitzer, John DeNero and **Dan Klein**
Computer Science Division, University of California at Berkeley
{aria42,blitzer,denero,klein}@cs.berkeley.edu

Abstract

This work investigates supervised word alignment methods that exploit inversion transduction grammar (ITG) constraints. We consider maximum margin and conditional likelihood objectives, including the presentation of a new normal form grammar for canonicalizing derivations. Even for non-ITG sentence pairs, we show that it is possible learn ITG alignment models by simple relaxations of structured discriminative learning objectives. For efficiency, we describe a set of pruning techniques that together allow us to align sentences two orders of magnitude faster than naive bitext CKY parsing. Finally, we introduce many-to-one block alignment features, which significantly improve our ITG models. Altogether, our method results in the best reported AER numbers for Chinese-English and a performance improvement of 1.1 BLEU over GIZA++ alignments.

1 Introduction

Inversion transduction grammar (ITG) constraints (Wu, 1997) provide coherent structural constraints on the relationship between a sentence and its translation. ITG has been extensively explored in unsupervised statistical word alignment (Zhang and Gildea, 2005; Cherry and Lin, 2007a; Zhang et al., 2008) and machine translation decoding (Cherry and Lin, 2007b; Petrov et al., 2008). In this work, we investigate large-scale, discriminative ITG word alignment.

Past work on discriminative word alignment has focused on the family of at-most-one-to-one matchings (Melamed, 2000; Taskar et al., 2005; Moore et al., 2006). An exception to this is the work of Cherry and Lin (2006), who discriminatively trained one-to-one ITG models, albeit with limited feature sets. As they found, ITG approaches offer several advantages over general matchings. First, the additional structural constraint can result in superior alignments. We confirm and extend this result, showing that one-to-one ITG models can perform as well as, or better than, general one-to-one matching models, either using heuristic weights or using rich, learned features.

A second advantage of ITG approaches is that they admit a range of training options. As with general one-to-one matchings, we can optimize margin-based objectives. However, unlike with general matchings, we can also efficiently compute expectations over the set of ITG derivations, enabling the training of conditional likelihood models. A major challenge in both cases is that our training alignments are often not one-to-one ITG alignments. Under such conditions, directly training to maximize margin is unstable, and training to maximize likelihood is ill-defined, since the target alignment derivations don't exist in our hypothesis class. We show how to adapt both margin and likelihood objectives to learn good ITG aligners.

In the case of likelihood training, two innovations are presented. The simple, two-rule ITG grammar exponentially over-counts certain alignment structures relative to others. Because of this, Wu (1997) and Zens and Ney (2003) introduced a normal form ITG which avoids this over-counting. We extend this normal form to null productions and give the first extensive empirical comparison of simple and normal form ITGs, for posterior decoding under our likelihood models. Additionally, we show how to deal with training instances where the gold alignments are outside of the hypothesis class by instead optimizing the likelihood of a set of minimum-loss alignments.

Perhaps the greatest advantage of ITG models is that they straightforwardly permit block-

structured alignments (i.e. phrases), which general matchings cannot efficiently do. The need for block alignments is especially acute in Chinese-English data, where oracle AERs drop from 10.2 without blocks to around 1.2 with them. Indeed, blocks are the primary reason for gold alignments being outside the space of one-to-one ITG alignments. We show that placing linear potential functions on many-to-one blocks can substantially improve performance.

Finally, to scale up our system, we give a combination of pruning techniques that allows us to sum ITG alignments two orders of magnitude faster than naive inside-outside parsing.

All in all, our discriminatively trained, block ITG models produce alignments which exhibit the best AER on the NIST 2002 Chinese-English alignment data set. Furthermore, they result in a 1.1 BLEU-point improvement over GIZA++ alignments in an end-to-end Hiero (Chiang, 2007) machine translation system.

2 Alignment Families

In order to structurally restrict attention to reasonable alignments, word alignment models must constrain the set of alignments considered. In this section, we discuss and compare alignment families used to train our discriminative models.

Initially, as in Taskar et al. (2005) and Moore et al. (2006), we assume the score $\mathbf{a}$ of a potential alignment $\mathbf{a}$) decomposes as

$$s(\mathbf{a}) = \sum_{(i,j)\in\mathbf{a}} s_{ij} + \sum_{i\notin\mathbf{a}} s_{i\epsilon} + \sum_{j\notin\mathbf{a}} s_{\epsilon j} \qquad (1)$$

where s_{ij} are word-to-word potentials and $s_{i\epsilon}$ and $s_{\epsilon j}$ represent English null and foreign null potentials, respectively.

We evaluate our proposed alignments ($\mathbf{a}$) against hand-annotated alignments, which are marked with *sure* ($\mathbf{s}$) and *possible* ($\mathbf{p}$) alignments. The *alignment error rate* (AER) is given by,

$$AER(\mathbf{a},\mathbf{s},\mathbf{p}) = \mathbf{1} - \frac{|\mathbf{a}\cap\mathbf{s}| + |\mathbf{a}\cap\mathbf{p}|}{|\mathbf{a}| + |\mathbf{s}|}$$

2.1 1-to-1 Matchings

The class of at most 1-to-1 alignment matchings, $\mathcal{A}_{1\text{-}1}$, has been considered in several works (Melamed, 2000; Taskar et al., 2005; Moore et al., 2006). The alignment that maximizes a set of potentials factored as in Equation (1) can be found in $O(n^3)$ time using a bipartite matching algorithm (Kuhn, 1955).[1] On the other hand, summing over $\mathcal{A}_{1\text{-}1}$ is $\#P$-hard (Valiant, 1979).

Initially, we consider heuristic alignment potentials given by Dice coefficients

$$Dice(e,f) = \frac{2C_{ef}}{C_e + C_f}$$

where C_{ef} is the joint count of words (e,f) appearing in aligned sentence pairs, and C_e and C_f are monolingual unigram counts.

We extracted such counts from 1.1 million French-English aligned sentence pairs of Hansards data (see Section 6.1). For each sentence pair in the Hansards test set, we predicted the alignment from $\mathcal{A}_{1\text{-}1}$ which maximized the sum of Dice potentials. This yielded 30.6 AER.

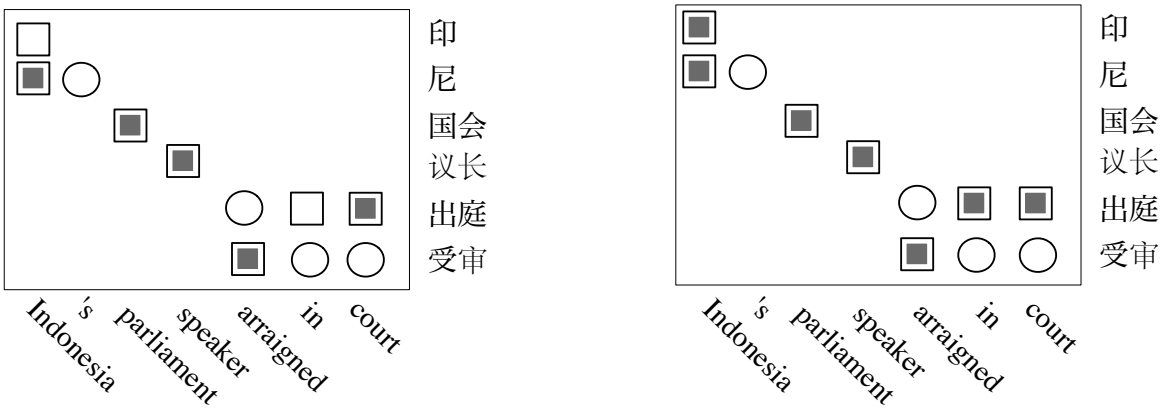
2.2 Inversion Transduction Grammar

Wu (1997)'s inversion transduction grammar (ITG) is a synchronous grammar formalism in which derivations of sentence pairs correspond to alignments. In its original formulation, there is a single non-terminal X spanning a bitext cell with an English and foreign span. There are three rule types: Terminal unary productions $X \rightarrow \langle e,f\rangle$, where e and f are an aligned English and foreign word pair (possibly with one being null); normal binary rules $X \rightarrow X^{(L)}X^{(R)}$, where the English and foreign spans are constructed from the children as $\langle X^{(L)}X^{(R)}, X^{(L)}X^{(R)}\rangle$; and inverted binary rules $X \rightarrow\!\!\!\!\sim\ X^{(L)}X^{(R)}$, where the foreign span inverts the order of the children $\langle X^{(L)}X^{(R)}, X^{(R)}X^{(L)}\rangle$.[2] In general, we will call a bitext cell a *normal* cell if it was constructed with a normal rule and *inverted* if constructed with an inverted rule.

Each ITG derivation yields some alignment. The set of such ITG alignments, $\mathcal{A}_{ITG}$, are a strict subset of $\mathcal{A}_{1\text{-}1}$ (Wu, 1997). Thus, we will view ITG as a constraint on $\mathcal{A}_{1\text{-}1}$ which we will argue is generally beneficial. The maximum scoring alignment from $\mathcal{A}_{ITG}$ can be found in $O(n^6)$ time with synchronous CFG parsing; in practice, we can make ITG parsing efficient using a variety of pruning techniques. One computational advantage of $\mathcal{A}_{ITG}$ over $\mathcal{A}_{1\text{-}1}$ alignments is that summation over $\mathcal{A}_{ITG}$ is tractable. The corresponding

[1] We shall use n throughout to refer to the maximum of foreign and English sentence lengths.

[2] The superscripts on non-terminals are added only to indicate correspondence of child symbols.



(a) Max-Matching Alignment

(b) Block ITG Alignment

Figure 1: Best alignments from (a) 1-1 matchings and (b) block ITG (BITG) families respectively. The 1-1 matching is the best possible alignment in the model family, but cannot capture the fact that *Indonesia* is rendered as two words in Chinese or that *in court* is rendered as a single word in Chinese.

dynamic program allows us to utilize likelihood-based objectives for learning alignment models (see Section 4).

Using the same heuristic Dice potentials on the Hansards test set, the maximal scoring alignment from $\mathcal{A}_{ITG}$ yields 28.4 AER—2.4 better than $\mathcal{A}_{1\text{-}1}$ —indicating that ITG can be beneficial as a constraint on heuristic alignments.

2.3 Block ITG

An important alignment pattern disallowed by $\mathcal{A}_{1\text{-}1}$ is the many-to-one alignment block. While not prevalent in our hand-aligned French Hansards dataset, blocks occur frequently in our hand-aligned Chinese-English NIST data. Figure 1 contains an example. Extending $\mathcal{A}_{1\text{-}1}$ to include blocks is problematic, because finding a maximal 1-1 matching over phrases is NP-hard (DeNero and Klein, 2008).

With ITG, it is relatively easy to allow contiguous many-to-one alignment blocks without added complexity.[3] This is accomplished by adding additional unary terminal productions aligning a foreign phrase to a single English terminal or vice versa. We will use BITG to refer to this block ITG variant and $\mathcal{A}_{BITG}$ to refer to the alignment family, which is neither contained in nor contains $\mathcal{A}_{1\text{-}1}$. For this alignment family, we expand the alignment potential decomposition in Equation (1) to incorporate block potentials $s_{\overline{e}f}$ and $s_{e\overline{f}}$ which represent English and foreign many-to-one alignment blocks, respectively.

One way to evaluate alignment families is to consider their oracle AER. In the 2002 NIST Chinese-English hand-aligned data (see Section 6.2), we constructed oracle alignment potentials as follows: s_{ij} is set to +1 if (i, j) is a sure or possible alignment in the hand-aligned data, -1 otherwise. All null potentials ($s_{i\epsilon}$ and $s_{\epsilon j}$) are set to 0. A max-matching under these potentials is generally a minimal loss alignment in the family. The oracle AER computed in this was is 10.1 for $\mathcal{A}_{1\text{-}1}$ and 10.2 for $\mathcal{A}_{ITG}$. The $\mathcal{A}_{BITG}$ alignment family has an oracle AER of 1.2. These basic experiments show that $\mathcal{A}_{ITG}$ outperforms $\mathcal{A}_{1\text{-}1}$ for heuristic alignments, and $\mathcal{A}_{BITG}$ provide a much closer fit to true Chinese-English alignments than $\mathcal{A}_{1\text{-}1}$.

3 Margin-Based Training

In this and the next section, we discuss learning alignment potentials. As input, we have a training set $\mathcal{D} = (\mathbf{x}_1, \mathbf{a}_1^*), \ldots, (\mathbf{x}_n, \mathbf{a}_n^*)$ of hand-aligned data, where $\mathbf{x}$ refers to a sentence pair. We will assume the score of a alignment is given as a linear function of a feature vector $\phi(\mathbf{x}, \mathbf{a})$. We will further assume the feature representation of an alignment, $\phi(\mathbf{x}, \mathbf{a})$ decomposes as in Equation (1),

$$\sum_{(i,j)\in\mathbf{a}} \phi_{ij}(\mathbf{x}) + \sum_{i\notin\mathbf{a}} \phi_{i\epsilon}(\mathbf{x}) + \sum_{j\notin\mathbf{a}} \phi_{\epsilon j}(\mathbf{x})$$

In the framework of loss-augmented margin learning, we seek a $\mathbf{w}$ such that $\mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{a}^*)$ is larger than $\mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{a}) + L(\mathbf{a}, \mathbf{a}^*)$ for all $\mathbf{a}$ in an alignment family, where $L(\mathbf{a}, \mathbf{a}^*)$ is the loss between a proposed alignment $\mathbf{a}$ and the gold alignment $\mathbf{a}^*$. As in Taskar et al. (2005), we utilize a

[3] In our experiments we limited the block size to 4.

loss that decomposes across alignments. Specifically, for each alignment cell (i, j) which is not a possible alignment in $\mathbf{a}^*$, we incur a loss of 1 when $\mathbf{a}_{ij} \neq \mathbf{a}^*_{ij}$; note that if (i, j) is a possible alignment, our loss is indifferent to its presence in the proposal alignment.

A simple loss-augmented learning procedure is the margin infused relaxed algorithm (MIRA) (Crammer et al., 2006). MIRA is an online procedure, where at each time step $t + 1$, we update our weights as follows:

$$\begin{aligned} \mathbf{w}_{t+1} &= \text{argmin}_{\mathbf{w}} ||\mathbf{w} - \mathbf{w}_t||_2^2 \quad (2) \\ &\text{s.t. } \mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{a}^*) \geq \mathbf{w} \cdot \phi(\mathbf{x}, \hat{\mathbf{a}}) + L(\hat{\mathbf{a}}, \mathbf{a}^*) \\ &\text{where } \hat{\mathbf{a}} = \arg\max_{\mathbf{a} \in \mathcal{A}} \mathbf{w}_t \cdot \phi(\mathbf{x}, \mathbf{a}) \end{aligned}$$

In our data sets, many $\mathbf{a}^*$ are not in $\mathcal{A}_{1\text{-}1}$ (and thus not in $\mathcal{A}_{ITG}$), implying the minimum in-family loss must exceed 0. Since MIRA operates in an online fashion, this can cause severe stability problems. On the Hansards data, the simple averaging technique described by Collins (2002) yields a reasonable model. On the Chinese NIST data, however, where almost no alignment is in $\mathcal{A}_{1\text{-}1}$, the update rule from Equation (2) is completely unstable, and even the averaged model does not yield high-quality results.

We instead use a variant of MIRA similar to Chiang et al. (2008). First, rather than update towards the hand-labeled alignment $\mathbf{a}^*$, we update towards an alignment which achieves minimal loss within the family.[4] We call this best-in-class alignment $\mathbf{a}^*_p$. Second, we perform loss-augmented inference to obtain $\hat{\mathbf{a}}$. This yields the modified QP,

$$\begin{aligned} \mathbf{w}_{t+1} &= \text{argmin}_{\mathbf{w}} ||\mathbf{w} - \mathbf{w}_t||_2^2 \quad (3) \\ &\text{s.t. } \mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{a}^*_p) \geq \mathbf{w} \cdot \phi(\mathbf{x}, \hat{\mathbf{a}}) + L(\mathbf{a}, \mathbf{a}^*_p) \\ &\text{where } \hat{\mathbf{a}} = \arg\max_{\mathbf{a} \in \mathcal{A}} \mathbf{w}_t \cdot \phi(\mathbf{x}, \mathbf{a}) + \lambda L(\mathbf{a}, \mathbf{a}^*_p) \end{aligned}$$

By setting $\lambda = 0$, we recover the MIRA update from Equation (2). As λ grows, we increase our preference that $\hat{\mathbf{a}}$ have high loss (relative to $\mathbf{a}^*_p$) rather than high model score. With this change, MIRA is stable, but still performs suboptimally. The reason is that initially the score for all alignments is low, so we are biased toward only using very high loss alignments in our constraint. This slows learning and prevents us from finding a useful weight vector. Instead, in all the experiments we report here, we begin with $\lambda = 0$ and slowly increase it to $\lambda = 0.5$.

[4] There might be several alignments which achieve this minimal loss; we choose arbitrarily among them.

4 Likelihood Objective

An alternative to margin-based training is a likelihood objective, which learns a conditional alignment distribution $P_{\mathbf{w}}(\mathbf{a}|\mathbf{x})$ parametrized as follows,

$$\log P_{\mathbf{w}}(\mathbf{a}|\mathbf{x}) = \mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{a}) - \log \sum_{\mathbf{a}' \in \mathcal{A}} \exp(\mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{a}'))$$

where the log-denominator represents a sum over the alignment family $\mathcal{A}$. This alignment probability only places mass on members of $\mathcal{A}$. The likelihood objective is given by,

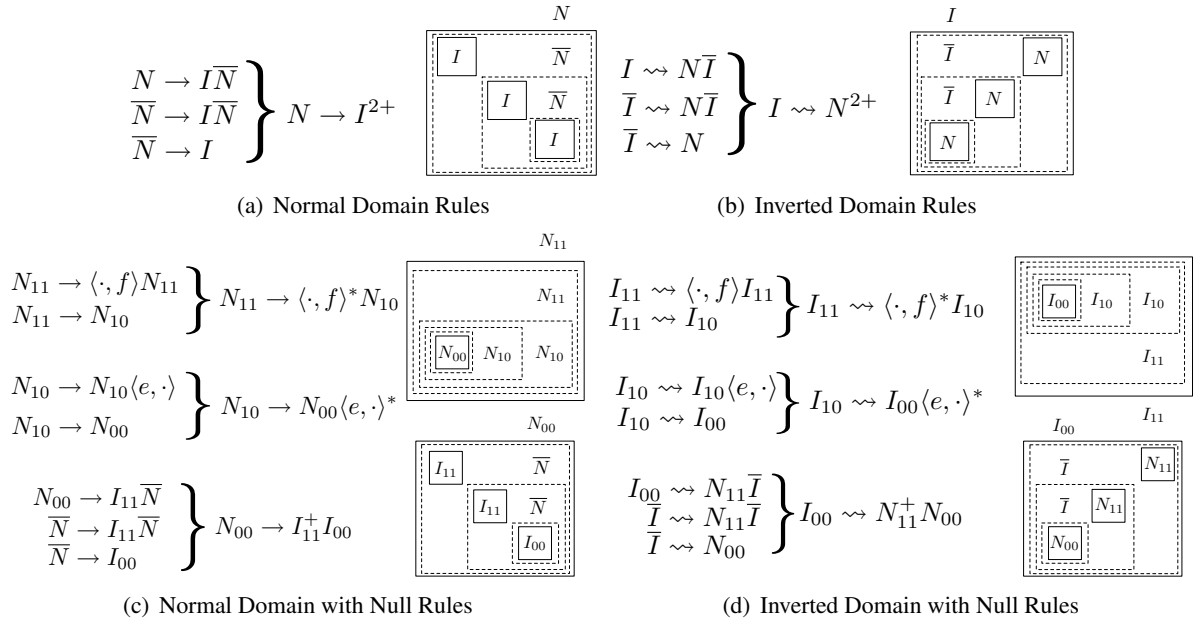
$$\max_{\mathbf{w}} \sum_{(\mathbf{x}, \mathbf{a}^*) \in \mathcal{A}} \log P_{\mathbf{w}}(\mathbf{a}^*|\mathbf{x})$$

Optimizing this objective with gradient methods requires summing over alignments. For $\mathcal{A}_{ITG}$ and $\mathcal{A}_{BITG}$, we can efficiently sum over the set of ITG *derivations* in $O(n^6)$ time using the inside-outside algorithm. However, for the ITG grammar presented in Section 2.2, each alignment has multiple grammar derivations. In order to correctly sum over the set of ITG alignments, we need to alter the grammar to ensure a bijective correspondence between alignments and derivations.

4.1 ITG Normal Form

There are two ways in which ITG derivations double count alignments. First, n-ary productions are not binarized to remove ambiguity; this results in an exponential number of derivations for diagonal alignments. This source of overcounting is considered and fixed by Wu (1997) and Zens and Ney (2003), which we briefly review here. The resulting grammar, which does not handle null alignments, consists of a symbol N to represent a bitext cell produced by a normal rule and I for a cell formed by an inverted rule; alignment terminals can be either N or I. In order to ensure unique derivations, we stipulate that a N cell can be constructed only from a sequence of smaller inverted cells I. Binarizing the rule $N \rightarrow I^{2+}$ introduces the intermediary symbol $\overline{N}$ (see Figure 2(a)). Similarly for inverse cells, we insist an I cell only be built by an inverted combination of N cells; binarization of $I \rightsquigarrow N^{2+}$ requires the introduction of the intermediary symbol $\overline{I}$ (see Figure 2(b)).

Null productions are also a source of double counting, as there are many possible orders in



(a) Normal Domain Rules

(b) Inverted Domain Rules

(c) Normal Domain with Null Rules

(d) Inverted Domain with Null Rules

Figure 2: Illustration of two unambiguous forms of ITG grammars: In (a) and (b), we illustrate the normal grammar without nulls (presented in Wu (1997) and Zens and Ney (2003)). In (c) and (d), we present a normal form grammar that accounts for null alignments.

which to attach null alignments to a bitext cell; we address this by adapting the grammar to force a null attachment order. We introduce symbols N_{00}, N_{10}, and N_{11} to represent whether a normal cell has taken no nulls, is accepting foreign nulls, or is accepting English nulls, respectively. We also introduce symbols I_{00}, I_{10}, and I_{11} to represent inverse cells at analogous stages of taking nulls. As Figures 2 (c) and (d) illustrate, the directions in which nulls are attached to normal and inverse cells differ. The N_{00} symbol is constructed by one or more 'complete' inverted cells I_{11} terminated by a no-null I_{00}. By placing I_{00} in the lower right hand corner, we allow the larger N_{00} to unambiguously attach nulls. N_{00} transitions to the N_{10} symbol and accepts any number of $\langle e, \cdot \rangle$ English terminal alignments. Then N_{10} transitions to N_{11} and accepts any number of $\langle \cdot, f \rangle$ foreign terminal alignments. An analogous set of grammar rules exists for the inverted case (see Figure 2(d) for an illustration). Given this normal form, we can efficiently compute model expectations over ITG alignments without double counting.[5] To our knowledge, the alteration of the normal form to accommodate null emissions is novel to this work.

[5]The complete grammar adds sentinel symbols to the upper left and lower right, and the root symbol is constrained to be a N_{00}.

4.2 Relaxing the Single Target Assumption

A crucial obstacle for using the likelihood objective is that a given $\mathbf{a}^*$ may not be in the alignment family. As in our alteration to MIRA (Section 3), we could replace $\mathbf{a}^*$ with a minimal loss in-class alignment $\mathbf{a}_p^*$. However, in contrast to MIRA, the likelihood objective will implicitly penalize proposed alignments which have loss equal to $\mathbf{a}_p^*$. We opt instead to maximize the probability of the set of alignments $\mathcal{M}(\mathbf{a}^*)$ which achieve the same optimal in-class loss. Concretely, let m^* be the minimal loss achievable relative to $\mathbf{a}^*$ in $\mathcal{A}$. Then,

$$\mathcal{M}(\mathbf{a}^*) = \{\mathbf{a} \in \mathcal{A} | L(\mathbf{a}, \mathbf{a}^*) = m^*\}$$

When $\mathbf{a}^*$ is an ITG alignment (i.e., m^* is 0), $\mathcal{M}(\mathbf{a}^*)$ consists only of alignments which have all the sure alignments in $\mathbf{a}^*$, but may have some subset of the possible alignments in $\mathbf{a}^*$. See Figure 3 for a specific example where $m^* = 1$.

Our modified likelihood objective is given by,

$$\max_{\mathbf{w}} \sum_{(\mathbf{x}, \mathbf{a}^*) \in \mathcal{D}} \log \sum_{\mathbf{a} \in \mathcal{M}(\mathbf{a}^*)} P_{\mathbf{w}}(\mathbf{a}|\mathbf{x})$$

Note that this objective is no longer convex, as it involves a logarithm of a summation, however we still utilize gradient-based optimization. Summing and obtaining feature expectations over $\mathcal{M}(\mathbf{a}^*)$ can be done efficiently using a constrained variant

	MIRA						Likelihood					
	1-1			ITG			ITG-S			ITG-N		
Features	P	R	AER	P	R	AER	P	R	AER	P	R	AER
Dice,dist	85.9	82.6	15.6	86.7	82.9	15.0	**89.2**	**85.2**	**12.6**	87.8	82.6	14.6
+lex,ortho	89.3	86.0	12.2	90.1	86.4	11.5	**92.0**	**90.6**	**8.6**	90.3	88.8	10.4
+joint HMM	95.8	93.8	**5.0**	**96.0**	93.2	5.2	95.5	**94.2**	**5.0**	95.6	94.0	5.1

Table 1: Results on the French Hansards dataset. Columns indicate models and training methods. The rows indicate the feature sets used. ITG-S uses the simple grammar (Section 2.2). ITG-N uses the normal form grammar (Section 4.1). For MIRA (Viterbi inference), the highest-scoring alignment is the same, regardless of grammar.

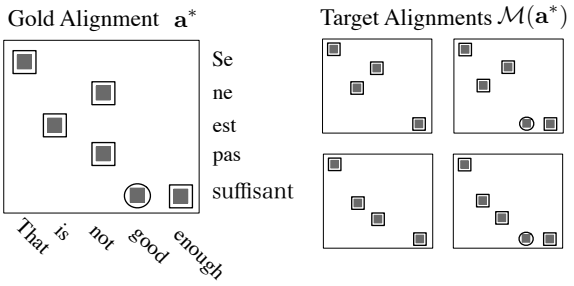


Figure 3: Often, the gold alignment $\mathbf{a}^*$ isn't in our alignment family, here $\mathcal{A}_{BITG}$. For the likelihood objective (Section 4.2), we maximize the probability of the set $\mathcal{M}(\mathbf{a}^*)$ consisting of alignments $\mathcal{A}_{BITG}$ which achieve minimal loss relative to $\mathbf{a}^*$. In this example, the minimal loss is 1, and we have a choice of removing either of the sure alignments to the English word *not*. We also have the choice of whether to include the possible alignment, yielding 4 alignments in $\mathcal{M}(\mathbf{a}^*)$.

of the inside-outside algorithm where sure alignments not present in $\mathbf{a}^*$ are disallowed, and the number of missing sure alignments is appended to the state of the bitext cell.[6]

One advantage of the likelihood-based objective is that we can obtain posteriors over individual alignment cells,

$$P_{\mathbf{w}}((i,j)|\mathbf{x}) = \sum_{\mathbf{a} \in \mathcal{A}:(i,j)\in\mathbf{a}} P_{\mathbf{w}}(\mathbf{a}|\mathbf{x})$$

We obtain posterior ITG alignments by including all alignment cells (i, j) such that $P_{\mathbf{w}}((i, j)|\mathbf{x})$ exceeds a fixed threshold t. Posterior thresholding allows us to easily trade-off precision and recall in our alignments by raising or lowering t.

5 Dynamic Program Pruning

Both discriminative methods require repeated model inference: MIRA depends upon loss-augmented Viterbi parsing, while conditional likelihood uses the inside-outside algorithm for computing cell posteriors. Exhaustive computation of these quantities requires an $O(n^6)$ dynamic program that is prohibitively slow even on small supervised training sets. However, most of the search space can safely be pruned using posterior predictions from a simpler alignment models. We use posteriors from two jointly estimated HMM models to make pruning decisions during ITG inference (Liang et al., 2006). Our first pruning technique is broadly similar to Cherry and Lin (2007a). We select high-precision alignment links from the HMM models: those word pairs that have a posterior greater than 0.9 in either model. Then, we prune all bitext cells that would invalidate more than 8 of these high-precision alignments.

Our second pruning technique is to prune all one-by-one (word-to-word) bitext cells that have a posterior below 10^{-4} in both HMM models. Pruning a one-by-one cell also indirectly prunes larger cells containing it. To take maximal advantage of this indirect pruning, we avoid explicitly attempting to build each cell in the dynamic program. Instead, we track bounds on the spans for which we have successfully built ITG cells, and we only iterate over larger spans that fall within those bounds. The details of a similar bounding approach appear in DeNero et al. (2009).

In all, pruning reduces MIRA iteration time from 175 to 5 minutes on the NIST Chinese-English dataset with negligible performance loss. Likelihood training time is reduced by nearly two orders of magnitude.

6 Alignment Quality Experiments

We present results which measure the quality of our models on two hand-aligned data sets. Our first is the English-French Hansards data set from the 2003 NAACL shared task (Mihalcea and Pedersen, 2003). Here we use the same 337/100 train/test split of the labeled data as Taskar et al.

[6] Note that alignments that achieve the minimal loss would not introduce any alignments not either sure or possible, so it suffices to keep track only of the number of sure recall errors.

	MIRA									Likelihood					
	1-1			ITG			BITG			BITG-S			BITG-N		
Features	P	R	AER	P	R	AER	P	R	AER	P	R	AER	P	R	AER
Dice, dist, blcks, dict, lex	85.7	63.7	26.8	**86.2**	65.8	25.2	85.0	73.3	21.1	85.7	73.7	20.6	85.3	**74.8**	**20.1**
+HMM	90.5	69.4	21.2	**91.2**	70.1	20.3	90.2	80.1	15.0	87.3	82.8	14.9	88.2	**83.0**	**14.4**

Table 2: Word alignment results on Chinese-English. Each column is a learning objective paired with an alignment family. The first row represents our best model without external alignment models and the second row includes features from the jointly trained HMM. Under likelihood, BITG-S uses the simple grammar (Section 2.2). BITG-N uses the normal form grammar (Section 4.1).

(2005); we compute external features from the same unlabeled data, 1.1 million sentence pairs. Our second is the Chinese-English hand-aligned portion of the 2002 NIST MT evaluation set. This dataset has 491 sentences, which we split into a training set of 150 and a test set of 191. When we trained external Chinese models, we used the same unlabeled data set as DeNero and Klein (2007), including the bilingual dictionary.

For likelihood based models, we set the L2 regularization parameter, σ^2, to 100 and the threshold for posterior decoding to 0.33. We report results using the simple ITG grammar (ITG-S, Section 2.2) where summing over derivations double counts alignments, as well as the normal form ITG grammar (ITG-N,Section 4.1) which does not double count. We ran our annealed loss-augmented MIRA for 15 iterations, beginning with λ at 0 and increasing it linearly to 0.5. We compute Viterbi alignments using the averaged weight vector from this procedure.

6.1 French Hansards Results

The French Hansards data are well-studied data sets for discriminative word alignment (Taskar et al., 2005; Cherry and Lin, 2006; Lacoste-Julien et al., 2006). For this data set, it is not clear that improving alignment error rate beyond that of GIZA++ is useful for translation (Ganchev et al., 2008). Table 1 illustrates results for the Hansards data set. The first row uses dice and the same distance features as Taskar et al. (2005). The first two rows repeat the experiments of Taskar et al. (2005) and Cherry and Lin (2006), but adding ITG models that are trained to maximize conditional likelihood. The last row includes the posterior of the jointly-trained HMM of Liang et al. (2006) as a feature. This model alone achieves an AER of 5.4. No model significantly improves over the HMM alone, which is consistent with the results of Taskar et al. (2005).

6.2 Chinese NIST Results

Chinese-English alignment is a much harder task than French-English alignment. For example, the HMM aligner achieves an AER of 20.7 when using the competitive thresholding heuristic of DeNero and Klein (2007). On this data set, our block ITG models make substantial performance improvements over the HMM, and moreover these results do translate into downstream improvements in BLEU score for the Chinese-English language pair. Because of this, we will briefly describe the features used for these models in detail. For features on one-by-one cells, we consider Dice, the distance features from (Taskar et al., 2005), dictionary features, and features for the 50 most frequent lexical pairs. We also trained an HMM aligner as described in DeNero and Klein (2007) and used the posteriors of this model as features. The first two columns of Table 2 illustrate these features for ITG and one-to-one matchings.

For our block ITG models, we include all of these features, along with variants designed for many-to-one blocks. For example, we include the average Dice of all the cells in a block. In addition, we also created three new block-specific features types. The first type comprises bias features for each block length. The second type comprises features computed from N-gram statistics gathered from a large monolingual corpus. These include features such as the number of occurrences of the phrasal (multi-word) side of a many-to-one block, as well as pointwise mutual information statistics for the multi-word parts of many-to-one blocks. These features capture roughly how "coherent" the multi-word side of a block is.

The final block feature type consists of phrase shape features. These are designed as follows: For each word in a potential many-to-one block alignment, we map an individual word to *X* if it is not one of the 25 most frequent words. Some example features of this type are,

- **English Block:** [*the X*, *X*], [*in X of*, *X*]
- **Chinese Block:** [一 *X*, *X*] [*X* 人, *X*]

For English blocks, for example, these features capture the behavior of phrases such as *in spite of* or *in front of* that are rendered as one word in Chinese. For Chinese blocks, these features capture the behavior of phrases containing classifier phrases like 一 个 or 一 份, which are rendered as English indefinite determiners.

The right-hand three columns in Table 2 present supervised results on our Chinese English data set using block features. We note that almost all of our performance gains (relative to both the HMM and 1-1 matchings) come from BITG and block features. The maximum likelihood-trained normal form ITG model outperforms the HMM, even without including any features derived from the unlabeled data. Once we include the posteriors of the HMM as a feature, the AER decreases to 14.4. The previous best AER result on this data set is 15.9 from Ayan and Dorr (2006), who trained stacked neural networks based on GIZA++ alignments. Our results are not directly comparable (they used more labeled data, but did not have the HMM posteriors as an input feature).

6.3 End-To-End MT Experiments

We further evaluated our alignments in an end-to-end Chinese to English translation task using the publicly available hierarchical pipeline JosHUa (Li and Khudanpur, 2008). The pipeline extracts a Hiero-style synchronous context-free grammar (Chiang, 2007), employs suffix-array based rule extraction (Lopez, 2007), and tunes model parameters with minimum error rate training (Och, 2003). We trained on the FBIS corpus using sentences up to length 40, which includes 2.7 million English words. We used a 5-gram language model trained on 126 million words of the Xinhua section of the English Gigaword corpus, estimated with SRILM (Stolcke, 2002). We tuned on 300 sentences of the NIST MT04 test set.

Results on the NIST MT05 test set appear in Table 3. We compared four sets of alignments. The GIZA++ alignments[7] are combined across directions with the grow-diag-final heuristic, which outperformed the union. The joint HMM alignments are generated from competitive posterior thresholding (DeNero and Klein, 2007). The ITG Viterbi alignments are the Viterbi output of the ITG model with all features, trained to maximize log likelihood. The ITG Posterior alignments result from applying competitive thresholding to alignment posteriors under the ITG model. Our supervised ITG model gave a 1.1 BLEU increase over GIZA++.

[7] We used a standard training regimen: 5 iterations of model 1, 5 iterations of HMM, 3 iterations of Model 3, and 3 iterations of Model 4.

Alignments			Translations	
Model	Prec	Rec	Rules	BLEU
GIZA++	62	84	1.9M	23.22
Joint HMM	79	77	4.0M	23.05
Viterbi ITG	90	80	3.8M	24.28
Posterior ITG	81	83	4.2M	**24.32**

Table 3: Results on the NIST MT05 Chinese-English test set show that our ITG alignments yield improvements in translation quality.

7 Conclusion

This work presented the first large-scale application of ITG to discriminative word alignment. We empirically investigated the performance of conditional likelihood training of ITG word aligners under simple and normal form grammars. We showed that through the combination of relaxed learning objectives, many-to-one block alignment potential, and efficient pruning, ITG models can yield state-of-the art word alignments, even when the underlying gold alignments are highly non-ITG. Our models yielded the lowest published error for Chinese-English alignment and an increase in downstream translation performance.

References

Necip Fazil Ayan and Bonnie Dorr. 2006. Going beyond AER: An extensive analysis of word alignments and their impact on MT. In *ACL*.

Colin Cherry and Dekang Lin. 2006. Soft syntactic constraints for word alignment through discriminative training. In *ACL*.

Colin Cherry and Dekang Lin. 2007a. Inversion transduction grammar for joint phrasal translation modeling. In *NAACL-HLT 2007*.

Colin Cherry and Dekang Lin. 2007b. A scalable inversion transduction grammar for joint phrasal translation modeling. In *SSST Workshop at ACL*.

David Chiang, Yuval Marton, and Philip Resnik. 2008. Online large-margin training of syntactic and structural translation features. In *EMNLP*.

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

Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *EMNLP*.

Koby Crammer, Ofer Dekel, Shai S. Shwartz, and Yoram Singer. 2006. Online passive-aggressive algorithms. *Journal of Machine Learning Research*.

John DeNero and Dan Klein. 2007. Tailoring word alignments to syntactic machine translation. In *ACL*.

John DeNero and Dan Klein. 2008. The complexity of phrase alignment problems. In *ACL Short Paper Track*.

John DeNero, Mohit Bansal, Adam Pauls, and Dan Klein. 2009. Efficient parsing for transducer grammars. In *NAACL*.

Kuzman Ganchev, Joao Graca, and Ben Taskar. 2008. Better alignments = better translations? In *ACL*.

H. W. Kuhn. 1955. The Hungarian method for the assignment problem. *Naval Research Logistic Quarterly*.

Simon Lacoste-Julien, Ben Taskar, Dan Klein, and Michael Jordan. 2006. Word alignment via quadratic assignment. In *NAACL*.

Zhifei Li and Sanjeev Khudanpur. 2008. A scalable decoder for parsing-based machine translation with equivalent language model state maintenance. In *SSST Workshop at ACL*.

Percy Liang, Dan Klein, and Dan Klein. 2006. Alignment by agreement. In *NAACL-HLT*.

Adam Lopez. 2007. Hierarchical phrase-based translation with suffix arrays. In *EMNLP*.

I. Dan Melamed. 2000. Models of translational equivalence among words. *Computational Linguistics*.

Rada Mihalcea and Ted Pedersen. 2003. An evaluation exercise for word alignment. In *HLT/NAACL Workshop on Building and Using Parallel Texts*.

Robert C. Moore, Wen tau Yih, and Andreas Bode. 2006. Improved discriminative bilingual word alignment. In *ACL-COLING*.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *ACL*.

Slav Petrov, Aria Haghighi, and Dan Klein. 2008. Coarse-to-fine syntactic machine translation using language projections. In *Empirical Methods in Natural Language Processing*.

Andreas Stolcke. 2002. Srilm: An extensible language modeling toolkit. In *ICSLP 2002*.

Ben Taskar, Simon Lacoste-Julien, and Dan Klein. 2005. A discriminative matching approach to word alignment. In *NAACL-HLT*.

L. G. Valiant. 1979. The complexity of computing the permanent. *Theoretical Computer Science*, 8:189–201.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23.

Richard Zens and Hermann Ney. 2003. A comparative study on reordering constraints in statistical machine translation. In *ACL*.

Hao Zhang and Dan Gildea. 2005. Stochastic lexicalized inversion transduction grammar for alignment. In *ACL*.

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