Generalized Agreement for Bidirectional Word Alignment

Chunyang Liu[†], **Yang Liu**[†*], **Huanbo Luan**[†], **Maosong Sun**[†], and **Heng Yu**[‡]
[†] State Key Laboratory of Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology
Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
[‡] Samsung R&D Institute of China, Beijing 100028, China
{liuchunyang2012,liuyang.china,luanhuanbo}@gmail.com, sms@tsinghua.edu.cn
h0517.yu@samsung.com

Abstract

While agreement-based joint training has proven to deliver state-of-the-art alignment accuracy, the produced word alignments are usually restricted to one-to-one mappings because of the hard constraint on agreement. We propose a general framework to allow for arbitrary loss functions that measure the disagreement between asymmetric alignments. The loss functions can not only be defined between asymmetric alignments but also between alignments and other latent structures such as phrase segmentations. We use a Viterbi EM algorithm to train the joint model since the inference is intractable. Experiments on Chinese-English translation show that joint training with generalized agreement achieves significant improvements over two state-of-the-art alignment methods.

1 Introduction

Word alignment is a natural language processing task that aims to specify the correspondence between words in two languages (Brown et al., 1993). It plays an important role in statistical machine translation (SMT) as word-aligned bilingual corpora serve as the input of translation rule extraction (Koehn et al., 2003; Chiang, 2007; Galley et al., 2006; Liu et al., 2006).

Although state-of-the-art generative alignment models (Brown et al., 1993; Vogel et al., 1996) have been widely used in practical SMT systems, they fail to model the symmetry of word alignment. While word alignments in real-world bilingual data usually exhibit complicated mappings (i.e., mixed with one-to-one, one-to-many, many-to-one, and many-to-many links), these models assume that each target word is aligned to exactly one source word. To alleviate this problem, heuristic methods (e.g., grow-diag-final) have been proposed to combine two asymmetric alignments (source-to-target and target-to-source) to generate symmetric bidirectional alignments (Och and Ney, 2003; Koehn and Hoang, 2007).

Instead of using heuristic symmetrization, Liang et al. (2006) introduce a principled approach that encourages the agreement between asymmetric alignments in two directions. The basic idea is to favor links on which both unidirectional models agree. They associate two models via the agreement constraint and show that agreement-based joint training improves alignment accuracy significantly.

However, enforcing agreement in joint training faces a major problem: the two models are restricted to one-to-one alignments (Liang et al., 2006). This significantly limits the translation accuracy, especially for distantly-related language pairs such as Chinese-English (see Section 5). Although posterior decoding can potentially address this problem, Liang et al. (2006) find that many-to-many alignments occur infrequently because posteriors are sharply peaked around the Viterbi alignments. We believe that this happens because their model imposes a *hard* constraint on agreement: the two models must share the same alignment when estimating the parameters by calculating the products of alignment posteriors (see Section 2).

In this work, we propose a general framework for imposing agreement constraints in joint training of unidirectional models. The central idea is to use the expectation of a loss function, which measures the disagreement between two models, to replace the original probability of agreement. This allows for many possible ways to quantify agreement. Experiments on Chinese-English translation show that our approach outperforms two state-of-the-art baselines significantly.

*Corresponding author: Yang Liu.

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1828–1836,
Lisbon, Portugal, 17-21 September 2015.

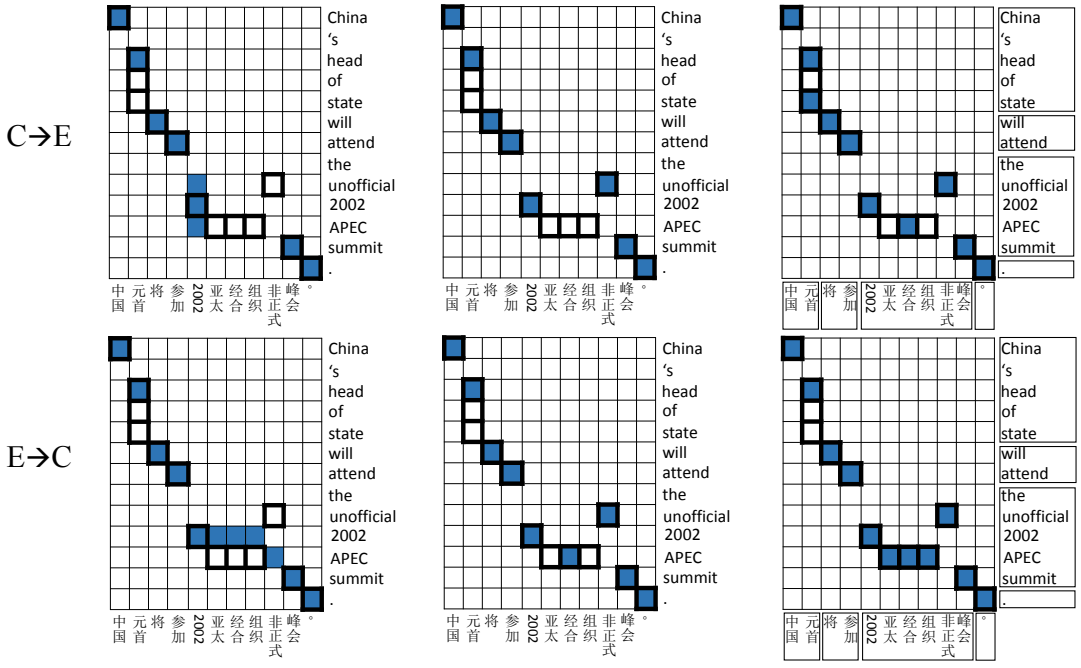

Figure 1: Comparison of (a) independent training without agreement, (b) joint training with agreement, and (c) joint training with generalized agreement. Bold squares are gold-standard links and solid squares are model predictions. The Chinese and English sentences are segmented into phrases in (c). Joint training with agreement achieves a high precision but generally only produces one-to-one alignments. We propose generalized agreement to account for not only the consensus between asymmetric alignments, but also the conformity of alignments to other latent structures such as phrase segmentations.

2 Background

2.1 Asymmetric Alignment Models

Given a source-language sentence $\mathbf{e} \equiv e_1^I = e_1, \ldots, e_I$ and a target-language sentence $\mathbf{f} \equiv f_1^J = f_1, \ldots, f_J$, a source-to-target translation model (Brown et al., 1993; Vogel et al., 1996) can be defined as

$$P(\mathbf{f}|\mathbf{e};\boldsymbol{\theta}_1) = \sum_{\mathbf{a}_1} P(\mathbf{f}, \mathbf{a}_1|\mathbf{e};\boldsymbol{\theta}_1) \quad (1)$$

where $\mathbf{a}_1$ denotes the source-to-target alignment and $\boldsymbol{\theta}_1$ is the set of source-to-target translation model parameters.

Likewise, the target-to-source translation model is given by

$$P(\mathbf{e}|\mathbf{f};\boldsymbol{\theta}_2) = \sum_{\mathbf{a}_2} P(\mathbf{e}, \mathbf{a}_2|\mathbf{f};\boldsymbol{\theta}_2) \quad (2)$$

where $\mathbf{a}_2$ denotes the target-to-source alignment and $\boldsymbol{\theta}_2$ is the set of target-to-source translation model parameters.

Given a training set $D = \{\langle \mathbf{f}^{(s)}, \mathbf{e}^{(s)} \rangle\}_{s=1}^S$, the two models are trained independently to maximize the log-likelihood of the training data for each direction, respectively:

$$\mathcal{L}(\boldsymbol{\theta}_1) = \sum_{s=1}^{S} \log P(\mathbf{f}^{(s)}|\mathbf{e}^{(s)};\boldsymbol{\theta}_1) \quad (3)$$

$$\mathcal{L}(\boldsymbol{\theta}_2) = \sum_{s=1}^{S} \log P(\mathbf{e}^{(s)}|\mathbf{f}^{(s)};\boldsymbol{\theta}_2) \quad (4)$$

One key limitation of these generative models is that they are **asymmetric**: each target word is restricted to be aligned to exactly one source word (including the empty cept) in the source-to-target direction and vice versa. This is undesirable because most real-world word alignments are **symmetric**, in which one-to-one, one-to-many, many-to-one, and many-to-many links are usually mixed. See Figure 1(a) for example. Therefore, a number of heuristic symmetrization methods such as intersection, union, and grow-diag-final have been proposed to combine asym-

metric alignments (Och and Ney, 2003; Koehn and Hoang, 2007).

2.2 Alignment by Agreement

Rather than using heuristic symmetrization methods, Liang et al. (2006) propose a principled approach to jointly training of the two models via enforcing **agreement**:

$$\begin{aligned} & J(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \\ = & \sum_{s=1}^{S} \log P(\mathbf{f}^{(s)}|\mathbf{e}^{(s)}; \boldsymbol{\theta}_1) + \\ & \log P(\mathbf{e}^{(s)}|\mathbf{f}^{(s)}; \boldsymbol{\theta}_2) + \\ & \log \sum_{\mathbf{a}} P(\mathbf{a}|\mathbf{f}^{(s)}, \mathbf{e}^{(s)}; \boldsymbol{\theta}_1) \times \\ & P(\mathbf{a}|\mathbf{e}^{(s)}, \mathbf{f}^{(s)}; \boldsymbol{\theta}_2) \end{aligned} \quad (5)$$

Note that the last term in Eq. (5) encourages the two models to agree on asymmetric alignments. While this strategy significantly improves alignment accuracy, the joint model is prone to generate one-to-one alignments because it imposes a *hard* constraint on agreement: the two models must share the same alignment when estimating the parameters by calculating the products of alignment posteriors. In Figure 1(b), the two one-to-one alignments are almost identical except for one link. This makes the posteriors to be sharply peaked around the Viterbi alignments (Liang et al., 2006). As a result, the lack of many-to-many alignments limits the benefits of joint training to end-to-end machine translation.

3 Generalized Agreement for Bidirectional Alignment

Our intuition is that the agreement between two alignments can be defined as a loss function, which enables us to consider various ways of quantification (Section 3.1) and even to incorporate the dependency between alignments and other latent structures such as phrase segmentations (Section 3.2).

3.1 Agreement between Word Alignments

The key idea of generalizing agreement is to leverage *loss functions* that measure the difference between two unidirectional alignments. For example, the last term in Eq. (5) can be re-written as

$$\begin{aligned} & \sum_{\mathbf{a}} P(\mathbf{a}|\mathbf{f}^{(s)}, \mathbf{e}^{(s)}; \boldsymbol{\theta}_1) P(\mathbf{a}|\mathbf{e}^{(s)}, \mathbf{f}^{(s)}; \boldsymbol{\theta}_2) \\ = & \sum_{\mathbf{a}_1} \sum_{\mathbf{a}_2} P(\mathbf{a}_1|\mathbf{f}^{(s)}, \mathbf{e}^{(s)}; \boldsymbol{\theta}_1) \times \\ & P(\mathbf{a}_2|\mathbf{e}^{(s)}, \mathbf{f}^{(s)}; \boldsymbol{\theta}_2) \times \\ & \delta(\mathbf{a}_1, \mathbf{a}_2) \end{aligned} \quad (6)$$

Note that the last term in Eq. (6) is actually the expected value of agreement:

$$\mathbb{E}_{\mathbf{a}_1|\mathbf{f}^{(s)}, \mathbf{e}^{(s)}; \boldsymbol{\theta}_1} \left[\mathbb{E}_{\mathbf{a}_2|\mathbf{e}^{(s)}, \mathbf{f}^{(s)}; \boldsymbol{\theta}_2} \left[\delta(\mathbf{a}_1, \mathbf{a}_2) \right] \right] \quad (7)$$

Our idea is to replace $\delta(\mathbf{a}_1, \mathbf{a}_2)$ in Eq. (6) with an arbitrary loss function $\Delta(\mathbf{a}_1, \mathbf{a}_2)$ that measures the difference between $\mathbf{a}_1$ and $\mathbf{a}_2$. This gives the new joint training objective with generalized agreement:

$$\begin{aligned} & J(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) \\ = & \sum_{s=1}^{S} \log P(\mathbf{f}^{(s)}|\mathbf{e}^{(s)}; \boldsymbol{\theta}_1) + \\ & \log P(\mathbf{e}^{(s)}|\mathbf{f}^{(s)}; \boldsymbol{\theta}_2) - \\ & \log \sum_{\mathbf{a}_1} \sum_{\mathbf{a}_2} P(\mathbf{a}_1|\mathbf{f}^{(s)}, \mathbf{e}^{(s)}; \boldsymbol{\theta}_1) \times \\ & P(\mathbf{a}_2|\mathbf{e}^{(s)}, \mathbf{f}^{(s)}; \boldsymbol{\theta}_2) \times \\ & \Delta(\mathbf{a}_1, \mathbf{a}_2) \end{aligned} \quad (8)$$

Obviously, Liang et al. (2006)'s training objective is a special case of our framework. We refer to its loss function as **hard matching**:

$$\Delta_{\text{HM}}(\mathbf{a}_1, \mathbf{a}_2) = 1 - \delta(\mathbf{a}_1, \mathbf{a}_2) \quad (9)$$

We are interested in developing a soft version of the hard matching loss function because this will help to produce many-to-many symmetric alignments. For example, in Figure 1(c), the two alignments share most links but still allow for disagreed links to capture one-to-many and many-to-one links. Note that the union of the two asymmetric alignments is almost the same with the gold-standard alignment in this example.

While there are many possible ways to define a **soft matching** loss function, we choose the difference between disagreed and agreed link counts because it is easy and efficient to calculate during search:

$$\Delta_{\text{SM}}(\mathbf{a}_1, \mathbf{a}_2) = |\mathbf{a}_1 \cup \mathbf{a}_2| - 2|\mathbf{a}_1 \cap \mathbf{a}_2| \quad (10)$$

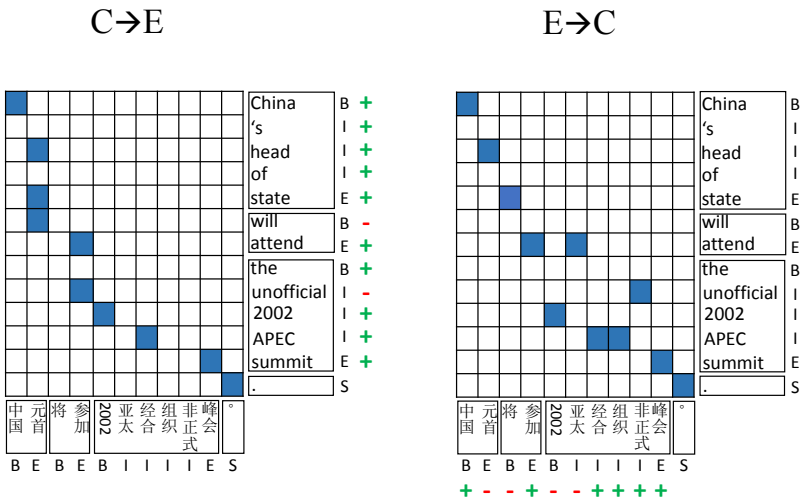

Figure 2: Generalized agreement between word alignments and phrase segmentations. The Chinese and English sentences are segmented into phrases using B (beginning), I (internal), E (ending), S (single) labels. We expect that word alignment does not violate the phrase segmentation. The word "unofficial" in the C → E alignment is labeled with "-" because "unofficial" and "2002" belong to the same English phrase but their counterparts are separated in two Chinese phrases. Words that do not violate the phrase alignment are labeled with "+". See Section 3.2 for details.

3.2 Agreement between Word Alignments and Phrase Segmentations

Our framework is very general and can be extended to include the agreement between word alignment and other latent structures such as phrase segmentations.

The words in a Chinese sentence often constitute phrases that are translated as units in English and vice versa. Inspired by the alignment consistency constraint widely used in translation rule extraction (Koehn et al., 2003), we make the following assumption to impose a structural agreement constraint between word alignment and phrase segmentation: *source words in one source phrase should be aligned to target words belonging to the same target phrase and vice versa.*

For example, consider the C → E alignment in Figure 2. We segment Chinese and English sentences into phrases, which are sequences of consecutive words. Since "2002" and "APEC" belong to the same English phrase, their counterparts on the Chinese side should also belong to one phrase.

While this assumption can potentially improve the correlation between word alignment and phrase-based translation, a question naturally arises: how to segment sentences into phrases? Instead of leveraging chunking, we treat phrase segmentation as a latent variable and train the joint alignment and segmentation model from unlabeled data in an unsupervised way.

Formally, given a target-language sentence $\mathbf{f} \equiv f_1^J = f_1, \ldots, f_J$, we introduce a latent variable $\mathbf{b} \equiv b_1^J = b_1, \ldots, b_J$ to denote a *phrase segmentation*. Each label $b_j \in \{B, I, E, S\}$, where B denotes the beginning word of a phrase, I denotes the internal word, E denotes the ending word, and S denotes the one-word phrase. Figure 2 shows the label sequences for the sentence pair.

We use a first-order HMM to model phrase segmentation of a target sentence:

$$P(\mathbf{f}; \boldsymbol{\lambda}_1) = \sum_{\mathbf{b}_1} P(\mathbf{f}, \mathbf{b}_1; \boldsymbol{\lambda}_1) \quad (11)$$

Similarly, the hidden Markov model for the phrase segmentation of the source sentence can be defined as

$$P(\mathbf{e}; \boldsymbol{\lambda}_2) = \sum_{\mathbf{b}_2} P(\mathbf{e}, \mathbf{b}_2; \boldsymbol{\lambda}_2) \quad (12)$$

Then, we can combine word alignment and phrase segmentation and define the joint training objective as

$$\begin{aligned} & J(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) \\ = & \sum_{s=1}^{S} \log P(\mathbf{f}^{(s)} | \mathbf{e}^{(s)}; \boldsymbol{\theta}_1) + \end{aligned}$$

1: **procedure** VITERBIEM(D)
2: Initialize $\boldsymbol{\Theta}^{(0)}$
3: **for all** $k = 1, \ldots, K$ **do**
4: $\hat{\mathbf{H}}^{(k)} \leftarrow$ SEARCH($D, \boldsymbol{\Theta}^{(k-1)}$)
5: $\boldsymbol{\Theta}^{(k)} \leftarrow$ UPDATE($D, \hat{\mathbf{H}}^{(k)}$)
6: **end for**
7: **return** $\hat{\mathbf{H}}^{(K)}, \boldsymbol{\Theta}^{(K)}$
8: **end procedure**

Algorithm 1: A Viterbi EM algorithm for learning the joint word alignment and phrase segmentation model from bilingual corpus. D is a bilingual corpus, $\boldsymbol{\Theta}^{(k)}$ is the set of model parameters at the k-th iteration, $\mathbf{H}^{(k)}$ is the set of Viterbi latent variables at the k-th iteration.

$$\begin{aligned} &\log P(\mathbf{e}^{(s)}|\mathbf{f}^{(s)}; \boldsymbol{\theta}_2) + \\ &\log P(\mathbf{f}^{(s)}; \boldsymbol{\lambda}_1) + \\ &\log P(\mathbf{e}^{(s)}; \boldsymbol{\lambda}_2) - \\ &\log \mathcal{E}(\mathbf{f}^{(s)}, \mathbf{e}^{(s)}, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) \end{aligned} \quad (13)$$

where the expected loss is given by

$$\begin{aligned} &\mathcal{E}(\mathbf{f}^{(s)}, \mathbf{e}^{(s)}, \boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2) \\ = &\sum_{\mathbf{a}_1} \sum_{\mathbf{a}_2} \sum_{\mathbf{b}_1} \sum_{\mathbf{b}_2} P(\mathbf{a}_1|\mathbf{f}^{(s)}, \mathbf{e}^{(s)}; \boldsymbol{\theta}_1) \times \\ &\quad P(\mathbf{a}_2|\mathbf{e}^{(s)}, \mathbf{f}^{(s)}; \boldsymbol{\theta}_2) \times \\ &\quad P(\mathbf{b}_1|\mathbf{f}^{(s)}; \boldsymbol{\lambda}_1) \times \\ &\quad P(\mathbf{b}_2|\mathbf{e}^{(s)}; \boldsymbol{\lambda}_2) \times \\ &\quad \Delta(\mathbf{a}_1, \mathbf{a}_2, \mathbf{b}_1, \mathbf{b}_2) \end{aligned} \quad (14)$$

We define a new loss function **segmentation violation** to measure the degree that an alignment violates phrase segmentations.

$$\begin{aligned} &\Delta_{\text{SV}}(\mathbf{a}_1, \mathbf{a}_2, \mathbf{b}_1, \mathbf{b}_2) \\ = &\sum_{j=1}^{J-1} \beta(\mathbf{a}_1, j, \mathbf{b}_1, \mathbf{b}_2) + \sum_{i=1}^{I-1} \beta(\mathbf{a}_2, i, \mathbf{b}_2, \mathbf{b}_1) \end{aligned} \quad (15)$$

where $\beta(\mathbf{a}_1, j, \mathbf{b}_1, \mathbf{b}_2)$ evaluates whether two links $l_1 = (j, a_j)$ and $l_2 = (j+1, a_{j+1})$ violate the phrase segmentation:

1. $\mathbf{f}_j$ and $\mathbf{f}_{j+1}$ belong to one phrase but $\mathbf{e}_{a_j}$ and $\mathbf{e}_{a_{j+1}}$ belong to two phrases, or
2. $\mathbf{f}_j$ and $\mathbf{f}_{j+1}$ belong to two phrases but $\mathbf{e}_{a_j}$ and $\mathbf{e}_{a_{j+1}}$ belong to one phrase.

The β function returns 1 if there is violation and 0 otherwise.

1: **procedure** SEARCH(D, $\boldsymbol{\Theta}$)
2: $\hat{\mathbf{H}} \leftarrow \emptyset$
3: **for all** $s \in \{1, \ldots, S\}$ **do**
4: $\hat{\mathbf{a}}_1 \leftarrow$ ALIGN($\mathbf{f}^{(s)}, \mathbf{e}^{(s)}, \boldsymbol{\theta}_1$)
5: $\hat{\mathbf{a}}_2 \leftarrow$ ALIGN($\mathbf{e}^{(s)}, \mathbf{f}^{(s)}, \boldsymbol{\theta}_2$)
6: $\hat{\mathbf{b}}_1 \leftarrow$ SEGMENT($\mathbf{f}^{(s)}, \boldsymbol{\lambda}_1$)
7: $\hat{\mathbf{b}}_2 \leftarrow$ SEGMENT($\mathbf{e}^{(s)}, \boldsymbol{\lambda}_2$)
8: $\mathbf{h}_0 \leftarrow \langle \hat{\mathbf{a}}_1, \hat{\mathbf{a}}_2, \hat{\mathbf{b}}_1, \hat{\mathbf{b}}_2 \rangle$
9: $\hat{\mathbf{h}} \leftarrow$HILLCLIMB($\mathbf{f}^{(s)}, \mathbf{e}^{(s)}, \mathbf{h}_0, \boldsymbol{\Theta}$)
10: $\hat{\mathbf{H}} \leftarrow \hat{\mathbf{H}} \cup \{\hat{\mathbf{h}}\}$
11: **end for**
12: **return** $\hat{\mathbf{H}}$
13: **end procedure**

Algorithm 2: A search algorithm for finding the Viterbi latent variables. $\hat{\mathbf{a}}_1$ and $\hat{\mathbf{a}}_2$ denote Viterbi alignments, $\hat{\mathbf{b}}_1$ and $\hat{\mathbf{b}}_2$ denote Viterbi segmentations. They form a starting point $\mathbf{h}_0$ for the hill climbing algorithm, which keeps changing alignments and segmentations until the model score does not increase. $\hat{\mathbf{h}}$ is the final set of Viterbi latent variables for one sentence.

In Figure 2, we use "+" to label words that do not violate the phrase segmentations and "-" to label violations.

In practice, we combine the two loss functions to enable word alignment and phrase segmentation to benefit each other in a joint search space:

$$\begin{aligned} &\Delta_{\text{SM+SV}}(\mathbf{a}_1, \mathbf{a}_2, \mathbf{b}_1, \mathbf{b}_2) \\ = &\Delta_{\text{SM}}(\mathbf{a}_1, \mathbf{a}_2) + \Delta_{\text{SV}}(\mathbf{a}_1, \mathbf{a}_2, \mathbf{b}_1, \mathbf{b}_2) \end{aligned} \quad (16)$$

4 Training

Liang et al. (2006) indicate that it is intractable to train the joint model. For simplicity and efficiency, they exploit a simple heuristic procedure that leverages the product of posterior marginal probabilities. The intuition behind the heuristic is that links on which two models disagree should be discounted because the products of the marginals are small (Liang et al., 2006).

Unfortunately, it is hard to develop a similar heuristic for our model that allows for arbitrary loss functions. Alternatively, we resort to a Viterbi EM algorithm, as shown in Algorithm 1. The algorithm takes the training data $D = \{\langle \mathbf{f}^{(s)}, \mathbf{e}^{(s)} \rangle\}_{s=1}^{S}$ as input (line 1). We use $\boldsymbol{\Theta}^{(k)} = \langle \boldsymbol{\theta}_1^{(k)}, \boldsymbol{\theta}_2^{(k)}, \boldsymbol{\lambda}_1^{(k)}, \boldsymbol{\lambda}_2^{(k)} \rangle$ to denote the set of model parameters at the k-th iteration. After initializing the model parameters (line 2), the algorithm alternates between searching for the Viterbi alignments

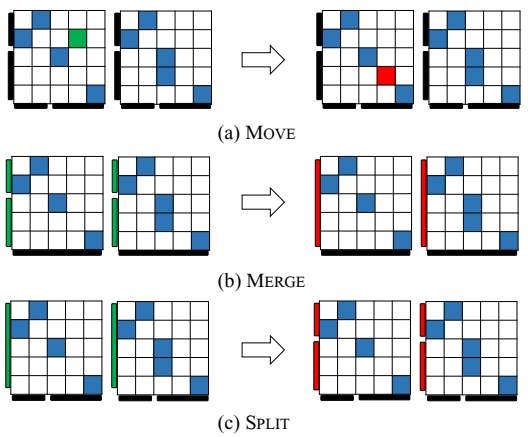

Figure 3: Operators used in the HILLCLIMB procedure.

and segmentations $\hat{\mathbf{H}}^{(k)}$ using the SEARCH procedure (line 4) and updating model parameters using the UPDATE procedure (line 5). The algorithm terminates after running for K iterations.

It is challenging to search for the Viterbi alignments and segmentations because of complicated structural dependencies. As shown in Algorithm 2, our strategy is first to find Viterbi alignments and segmentations independently using the ALIGN and SEGMENT procedures (lines 4-7), which then serve as a starting point for the HILLCLIMB procedure (lines 8-9).

Figure 3 shows three operators we use in the HILLCLIMB procedure. The MOVE operator moves a link in an alignment, the MERGE operator merges two phrases into one phrase, and the SPLIT operator splits one phrase into two smaller phrases. Note that each operator can be further divided into two variants: one for the source side and another for the target side.

5 Experiments

5.1 Setup

We evaluate our approach on Chinese-English alignment and translation tasks.

The training corpus consists of 1.2M sentence pairs with 32M Chinese words and 35.4M English words. We used the SRILM toolkit (Stolcke, 2002) to train a 4-gram language model on the Xinhua portion of the English GIGAWORD corpus, which contains 398.6M words. For alignment evaluation, we used the Tsinghua Chinese-English word alignment evaluation data set.[1] The evaluation metric is alignment error rate (AER) (Och and Ney, 2003). For translation evaluation, we used the NIST 2006 dataset as the development set and the NIST 2002, 2003, 2004, 2005, and 2008 datasets as the test sets. The evaluation metric is case-insensitive BLEU (Papineni et al., 2002).

We used both phrase-based (Koehn et al., 2003) and hierarchical phrase-based (Chiang, 2007) translation systems to evaluate whether our approach improves translation performance. For the phrase-based model, we used the open-source toolkit Moses (Koehn and Hoang, 2007). For the hierarchical phrase-based model, we used an in-house re-implementation on par with state-of-the-art open-source decoders.

We compared our approach with two state-of-the-art generative alignment models:

1. GIZA++ (Och and Ney, 2003): unsupervised training of IBM models (Brown et al., 1993) and the HMM model (Vogel et al., 1996) using EM,

2. BERKELEY (Liang et al., 2006): unsupervised training of joint HMMs using EM.

For GIZA++, we trained IBM Model 4 in two directions with the default setting and used the grow-diag-final heuristic to generate symmetric alignments. For BERKELEY, we trained joint HMMs using the default setting. The hyper-parameter of posterior decoding was optimized on the development set.

We used first-order HMMs for both word alignment and phrase segmentation. Our joint alignment and segmentation model were trained using the Viterbi EM algorithm for five iterations. Note that the Chinese-to-English and English-to-Chinese alignments are generally non-identical but share many links (see Figure 1(c)). Then, we used the grow-diag-final heuristic to generate symmetric alignments.

5.2 Comparison with GIZA++ and BERKELEY

Table 1 shows the comparison of our approach with GIZA++ and BERKELEY in terms of AER and BLEU. GIZA++ trains two asymmetric models independently and uses the grow-diag-final (i.e., GDF) for symmetrization. BERKELEY

[1] http://nlp.csai.tsinghua.edu.cn/~ly/systems/TsinghuaAligner/TsinghuaAligner.html

system	training	agreement	loss	sym.	AER	BLEU
GIZA++	indep.	N/A	N/A	GDF	21.35	24.46
BERKELEY	joint	word-word	HM	PD	20.52	24.54
this work	joint	word-word	SM	GDF	22.19	25.11
		word-word, word-phrase	SM+SV		22.01	25.78

Table 1: Comparison with GIZA++ and BERKELEY. "word-word" denotes the agreement between Chinese-to-English and English-to-Chinese word alignments. "word-phrase" denotes the agreement between word alignments and phrase segmentations. "HM" denotes the hard matching loss function, "SM" denotes soft matching, and "SV" denotes segmentation violation. "GDF" denotes grow-diag-final. "PD" denotes posterior decoding. The BLEU scores are evaluated on NIST08 test set.

alignment	loss	translation	NIST06	NIST02	NIST03	NIST04	NIST05	NIST08
GIZA++	N/A	phrase	29.57	31.82	31.67	32.20	30.48	24.46
		hier.	30.72	33.90	33.12	33.54	32.28	24.72
BERKELEY	HM	phrase	29.87	32.21	32.48	32.06	30.59	24.54
		hier.	29.52	33.59	32.70	32.95	29.52	24.29
this work	SM	phrase	30.04^{*}	32.75^{**++}	32.35^{**}	32.47^{*+}	30.86^{*+}	25.11^{**++}
		hier.	30.71^{++}	34.50^{**++}	33.89^{**++}	34.02^{*++}	32.83^{**++}	24.32
	SM+SV	phrase	30.60^{**++}	33.37^{**++}	33.24^{**++}	33.15^{**++}	31.57^{**++}	25.78^{**++}
		hier.	30.88^{++}	34.53^{**++}	34.04^{**++}	33.66^{++}	32.93^{**++}	25.17^{*++}

Table 2: Results on (hierarchical) phrase-based translation. The evaluation metric is case-insensitive BLEU. "HM" denotes the hard matching loss function, "SM" denotes soft matching, and "SV" denotes segmentation violation. "*": significantly better than GIZA++ ($p < 0.05$). "**": significantly better than GIZA++ ($p < 0.01$). "+": significantly better than BERKELEY ($p < 0.05$). "++": significantly better than BERKELEY ($p < 0.01$).

trains two models jointly with the hard-matching (i.e., HM) loss function and uses posterior decoding for symmetrization.

For our approach, we distinguish between two variants:

1. Imposing agreement between word alignments (i.e., word-word) that uses the soft matching loss function (i.e., SM) (see Section 3.1);

2. Imposing agreement between word alignments and phrase segmentations (i.e., word-word, word-phrase) that uses both the soft matching and segmentation violation loss functions (i.e., SM+SV) (see Section 3.2).

We used the grow-diag-final heuristic for symmetrization.

For the alignment evaluation, we find that our approach achieves higher AER scores than the two baseline systems. One possible reason is that links in the intersection of two symmetric alignments or two symmetric models agree usually correspond to sure links in the gold-standard annotation. Our approach loosens the hard constraint on agreement and makes the posteriors less peaked around the Viterbi alignments.

For the translation evaluation, we used the phrase-based system Moses to report BLEU scores on the NIST 2008 test set. We find that both the two variants of our approach significantly outperforms the two baselines ($p < 0.01$).

5.3 Results on (Hierarchical) Phrase-based Translation

Table 2 shows the results on phrase-based and hierarchical phrase-based translation systems. We find that our approach systematically outperforms GIZA++ and BERKELEY on all NIST datasets.

In particular, generalizing the agreement to model the discrepancy between word alignment and phrase segmentation is consistently beneficial for improving translation quality, suggesting that it is important to introduce structural constraints into word alignment to increase the correlation between alignment and translation.

While "SM+SV" improves over "SM" significantly on phrase-based translation, the margins on the hierarchical phrase-based system are relatively smaller. One possible reason is that the "SV"

system	loss	$\|\mathcal{A}_{C\to E}\|$	$\|\mathcal{A}_{E\to C}\|$	$\|\mathcal{A}_{C\to E} \cap \mathcal{A}_{E\to C}\|$	F1
GIZA++	N/A	29.39M	27.64M	17.07M	59.86
BERKELEY	HM	29.12M	28.09M	21.30M	74.46
this work	SM	29.84M	29.31M	20.24M	68.42
	SM+SV	30.04M	29.50M	20.54M	69.00

Table 3: Agreement evaluation of GIZA++, BERKELEY and our approach. The F1 score reflects how well two asymmetric alignments agree with each other.

loss function can better account for phrase-based rather than hierarchical phrase-based translation. It is possible to design new loss functions tailored to hierarchical phrase-based translation.

We also find that the BLEU scores of BERKELEY on hierarchical phrase-based translation are much lower than those on phrase-based translation. This might result from the fact that BERKELEY is prone to produce one-to-one alignments, which are not optimal for hierarchical phrase-based translation.

5.4 Agreement Evaluation

Table 3 compares how well two asymmetric models agree with each other among GIZA++, BERKELEY and our approach. We use F1 score to measure the degree of agreement:

$$\frac{2|\mathcal{A}_{C\to E} \cap \mathcal{A}_{E\to C}|}{|\mathcal{A}_{C\to E}| + |\mathcal{A}_{E\to C}|} \quad (17)$$

where $\mathcal{A}_{C\to E}$ is the set of Chinese-to-English alignments on the training data and $\mathcal{A}_{E\to C}$ is the set of English-to-Chinese alignments.

It is clear that independent training leads to low agreement and joint training results in high agreement. BERKELEY achieves the highest value of agreement because of the hard constraint.

6 Related Work

This work is inspired by two lines of research: (1) agreement-based learning and (2) joint modeling of multiple NLP tasks.

6.1 Agreement-based Learning

The key idea of agreement-based learning is to train a set of models jointly by encouraging them to agree on the hidden variables (Liang et al., 2006; Liang et al., 2008). This can also be seen as a particular form of posterior constraint or posterior regularization (Graça et al., 2007; Ganchev et al., 2010). The agreement is prior knowledge and indirect supervision, which helps to train a more reasonable model with biased guidance.

While agreement-based learning provides a principled approach to training a generative model, it constrains that the sub-models must share the same output space. Our work extends (Liang et al., 2006) to introduce arbitrary loss functions that can encode prior knowledge. As a result, Liang et al. (2006)'s model is a special case of our framework. Another difference is that our framework allows for including the agreement between word alignment and other structures such as phrase segmentations and parse trees.

6.2 Joint Modeling of Multiple NLP Tasks

It is well accepted that different NLP tasks can help each other by providing additional information for resolving ambiguities. As a result, joint modeling of multiple NLP tasks has received intensive attention in recent years, including phrase segmentation and alignment (Zhang et al., 2003), alignment and parsing (Burkett et al., 2010), tokenization and translation (Xiao et al., 2010), parsing and translation (Liu and Liu, 2010), alignment and named entity recognition (Chen et al., 2010; Wang et al., 2013).

Among them, Zhang et al. (2003)'s integrated search algorithm for phrase segmentation and alignment is most close to our work. They use Point-wise Mutual Information to identify possible phrase pairs. The major difference is we train models jointly instead of integrated decoding.

7 Conclusion

We have presented generalized agreement for bidirectional word alignment. The loss functions can be defined both between asymmetric alignments and between alignments and other latent structures such as phrase segmentations. We develop a Viterbi EM algorithm to train the joint model. Experiments on Chinese-English translation show that joint training with generalized agreement achieves

significant improvements over two baselines for (hierarchical) phrase-based MT systems. In the future, we plan to investigate more loss functions to account for syntactic constraints.

Acknowledgments

Yang Liu and Maosong Sun are supported by the 863 Program (2015AA011808), the National Natural Science Foundation of China (No. 61331013 and No. 61432013), and Samsung R&D Institute of China. Huanbo Luan is supported by the National Natural Science Foundation of China (No. 61303075). This research is also supported by the Singapore National Research Foundation under its International Research Centre@Singapore Funding Initiative and administered by the IDM Programme. We sincerely thank the reviewers for their valuable suggestions. We also thank Yue Zhang, Meng Zhang and Shiqi Shen for their insightful discussions.

References

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

David Burkett, John Blitzer, and Dan Klein. 2010. Joint parsing and alignment with weakly synchronized grammars. In *Proceedings of NAACL-HLT 2010*.

Yufeng Chen, Chengqing Zong, and Keh-Yih Su. 2010. On jointly recognizing and aligning bilingual named entities. In *Proceedings of ACL 2010*.

David Chiang. 2007. Hierarchical phrase-based translation. *Computational Linguistics*, 33(2):201–228.

Michel Galley, Jonathan Graehl, Kevin Knight, Daniel Marcu, Steve DeNeefe, Wei Wang, and Ignacio Thayer. 2006. Scalable inference and training of context-rich syntactic translation models. In *Proceedings of COLING-ACL 2006*, pages 961–968, Sydney, Australia, July.

Kuzmann Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. 2010. Posterior regularization for structured latent variable models. *Journal of Machine Learning Research*.

Joao V Graça, Kuzman Ganchev, and Ben Taskar. 2007. Expectation maximization and posterior constraints. In *Proceedings of NIPS 2007*.

Philipp Koehn and Hieu Hoang. 2007. Factored translation models. In *Proceedings of EMNLP-CoNLL 2007*, pages 868–876, Prague, Czech Republic, June.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of HLT-NAACL 2003*, pages 127–133, Edmonton, Canada, May.

Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proceedings of HLT-NAACL 2006*, pages 104–111, New York City, USA, June.

Percy Liang, Dan Klein, and Michael I. Jordan. 2008. Agreement-based learning. In *Advances in Neural Information Processing Systems (NIPS)*.

Yang Liu and Qun Liu. 2010. Joint parsing and translation. In *Proceedings of COLING 2010*.

Yang Liu, Qun Liu, and Shouxun Lin. 2006. Tree-to-string alignment template for statistical machine translation. In *Proceedings of COLING/ACL 2006*.

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

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

Andreas Stolcke. 2002. Srilm - an extensible language modeling toolkit. In *Proceedings of ICSLP 2002*.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. Hmm-based word alignment in statistical translation. In *Proceedings of COLING 1996*.

Mengqiu Wang, Wanxiang Che, and Christopher D. Manning. 2013. Joint word alignment and bilingual named entity recognition using dual decomposition. In *Proceedings of ACL 2013*.

Xinyan Xiao, Yang Liu, Young-Sook Hwang, Qun Liu, and Shouxun Lin. 2010. Joint tokenization and translation. In *Proceedings of COLING 2010*.

Ying Zhang, Stephan Vogel, and Alex Waibel. 2003. Integrated phrase segmentation and alignment algorithm for statistical machine translation. In *Proceedings of Natural Language Processing and Knowledge Engineering, 2003*. IEEE.