A Comparison of Update Strategies for Large-Scale Maximum Expected BLEU Training

Joern Wuebker, Sebastian Muehr, Patrick Lehnen, Stephan Peitz and Hermann Ney
Human Language Technology and Pattern Recognition Group
RWTH Aachen University
Aachen, Germany
{surname}@cs.rwth-aachen.de

Abstract

This work presents a flexible and efficient discriminative training approach for statistical machine translation. We propose to use the RPROP algorithm for optimizing a maximum expected BLEU objective and experimentally compare it to several other updating schemes. It proves to be more efficient and effective than the previously proposed growth transformation technique and also yields better results than stochastic gradient descent and AdaGrad. We also report strong empirical results on two large scale tasks, namely BOLT Chinese→English and WMT German→English, where our final systems outperform results reported by Setiawan and Zhou (2013) and on matrix.statmt.org. On the WMT task, discriminative training is performed on the full training data of 4M sentence pairs, which is unsurpassed in the literature.

1 Introduction

The main advantage of learning parameters in a discriminative fashion is the possibility to directly optimize towards a quality or error measure on the task that is being performed. This stands in contrast to the generative approach, where parameters are chosen to maximize likelihood under a generative story, which often bears little correspondence with the actual application of the model.

In statistical machine translation (SMT), extending the generative noisy-channel formulation (Brown et al., 1993) as a discriminative, log-linear combination of multiple models (Och, 2003) has become the state of the art. However, most of the component models are still estimated by heuristics or generative training. In this paper, a flexible, efficient and easy to implement discriminative training scheme for SMT is presented. It can be applied to any kind and any number of features. We use the RPROP algorithm to optimize a maximum expected BLEU objective. n-best lists approximate the infeasibly large space of translation hypotheses. They are generated with the application of leave-one-out to make them more representative with respect to unseen data.

We make the following main contributions:

1. We propose to apply the RPROP algorithm for maximum expected BLEU training and perform an experimental comparison with growth transformation (GT) (He and Deng, 2012; Setiawan and Zhou, 2013), stochastic gradient descent (Auli et al., 2014) and AdaGrad (Green et al., 2013). RPROP yields superior performance, reaching a total improvement of 1.2 BLEU points over our IWSLT German→English baseline using 5.22M features.

2. In terms of time and memory efficiency, RPROP clearly outperforms GT. The latter needs to update a much larger number of features due to its renormalization component. On the IWSLT data, RPROP is 6.4 times faster than GT and requires a third of the memory.

3. On the WMT German→English task, we perform discriminative training on 4M sentence

Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, pages 1516–1526,
Denver, Colorado, May 31 – June 5, 2015.

pairs, which, to the best of our knowledge, is 2.4 times the size of the largest training set reported in previous work (1.66M sentences in (Simianer et al., 2012)). This proves the scalability of our approach.

4. On two large scale tasks our experiments show good improvements over strong baselines which include recurrent language modeling components. On the Chinese→English DARPA BOLT task, we achieve nearly twice the improvement reported in (Setiawan and Zhou, 2013) on the same test sets which results in a superior final system. Finally, the best single system reported on matrix.statmt.org is outperformed by 0.8 BLEU points on the WMT German→English *newstest2013* set.

Our experiments also prove that leave-one-out impacts translation quality.

This paper is organized as follows. We review related work in Section 2 and present the translation system in Section 3. In Section 4 we describe the different discriminative update strategies applied in this work and Section 5 derives the complete maximum expected BLEU training algorithm. Finally, experimental results are given in Section 6 and we conclude with Section 7.

2 Related Work

Discriminative training is one of the most active research areas in SMT and it can be integrated into the pipeline at various stages.

Och (2003) proposed to apply minimum error rate training (MERT) to optimize the different feature weights in the log-linear model combination on a small development data set. This is still considered to be the state of the art, but is only capable of optimizing a handful of features. More recently, MIRA (Watanabe et al., 2007; Chiang et al., 2008) and PRO (Hopkins and May, 2011) have been presented as optimization procedures that can replace MERT and scale to thousands of parameters.

In a different line of work, Liang et al. (2006) describe a fully discriminative training pipeline, where more than one million features are tuned on the training data using a perceptron-style update algorithm. The Direct Translation Model 2 introduced by Ittycheriah and Roukos (2007) is similar in that it also trains millions of features on the training data. However, the weights are estimated based on a maximum entropy model and the underlying translation paradigm differs from the standard phrase-based model. Gao and He (2013) use gradient ascent to train Markov random field models for phrase translation. These models are interpreted as undirected phrase compatibility scores rather than translation probabilities. Thus, as in our work, they are not subject to a sum-to-one constraint. Simianer et al. (2012) propose a distributed setup for large-scale discriminative training with joint feature selection. The training corpus is divided into several shards, on which features are updated via perceptron-style gradient descent. The authors present results showing that training on large data sets improves results over just using a small development corpus. Another approach based on the AdaGrad method that scales to large numbers of sparse features is proposed in (Green et al., 2013; Green et al., 2014). Different from our work, the authors use either the tuning sets or a small subsample of the training data (15k sentences) for discriminative training.

A notably different idea is pursued by Yu et al. (2013), who present a large-scale training procedure that explicitly minimizes search errors. This is achieved by force-decoding the training data and updating at the point where the correct derivation drops off the beam.

In (Blunsom et al., 2008), conditional random fields (CRFs) are trained within a hierarchical phrase-based translation framework. The hierarchical phrase-based paradigm is used to model the search space in model estimation and search, leaving the hypothesis weighting to CRF features. They constrain search by a beam width for gradient estimation and update the model with the help of L-BFGS. In a similar way Lavergne et al. (2011) use the n-gram based approach (Casacuberta and Vidal, 2004; Mariño et al., 2006) to model the reordering, phrase alignment, and the language model. A CRF is applied to estimate the phrase weights. Model updates are carried out by the RPROP algorithm (Riedmiller and Braun, 1993). However, both approaches only improve over constrained baselines.

Our work is inspired by (He and Deng, 2012; Setiawan and Zhou, 2013), where the authors propose to train the standard phrasal and lexical channel models with the growth transformation (GT) algorithm. They use n-best lists on the training data and optimize a maximum expected BLEU objective, that provides a clear training criterion, which is missing e.g. in MIRA estimation. Auli et al. (2014) report good results by applying the same objective function to reordering features, which are trained with stochastic gradient descent (SGD).

Our work differs in several key aspects: (i) We propose to apply the RPROP algorithm, which yields superior results to GT, SGD and AdaGrad in our experimental comparison. (ii) For the first time, we apply maximum expected BLEU training on a data set as large as four million sentence pairs. (iii) We apply a leave-one-out heuristic (Wuebker et al., 2010) to make better use of the training data. (iv) We apply phrasal, lexical, reordering and triplet features. (v) Finally, we do not run MERT after each training iteration, which is expensive for large translation systems.

3 Statistical Translation System

Our work can be applied to any statistical machine translation paradigm and we will present results on a standard phrase-based translation system (Koehn et al., 2003) and a hierarchical phrase-based translation system (Chiang, 2005). The translation process is implemented as a weighted log-linear combination of several models $h_{m,\Theta}(E,F)$, where $E = e_1,\ldots,e_I$ denotes the translation hypothesis, $F = f_1,\ldots,f_J$ the source sentence, m a model index, and Θ the model parameters. These models include the phrase translation and lexical smoothing scores in both directions, language model (LM) score, distortion penalty, word penalty and phrase penalty (Och and Ney, 2004). Given a source sentence F, the models $h_{m,\Theta}(E,F)$ and the corresponding log-linear feature weights λ_m, the translation decoder searches for the best scoring translation $\hat{E}$:

$$\hat{E} = \arg\max_{E} \{f_\Theta(E,F)\} \tag{1}$$

$$f_\Theta(E,F) = \sum_{m\in M} \lambda_m h_{m,\Theta}(E,F) \tag{2}$$

where $\ldots,\lambda_m,\ldots$ are the model weighting parameters. In practice, the Viterbi approximation is applied and for simplicity, in the following we will assume the particular derivation for a translation hypothesis to be included in the variable E. The log-linear feature weights are optimized with minimum error rate training (MERT) (Och, 2003).

4 Update Strategies

4.1 Previously Proposed Algorithms

The **Growth Transformation (GT)** or Extended Baum-Welch Algorithm was proposed by He and Deng (2012) for maximum expected BLEU training of the standard phrasal and lexical channel models. It is an algorithm to iteratively optimize polynomials of random variables that are subject to sum-to-one contraints and is therefore suitable for training probability distributions. The disadvantage is that each parameter update requires a renormalization step, which artificially blows up the number of features that need to be changed and has a significant impact on time and memory efficiency. The update formulas are derived in (He and Deng, 2012).

Stochastic Gradient Descent (SGD) is a well-known and frequently applied training scheme, which is used for maximum expected BLEU training of reordering models by (Auli et al., 2014). It performs the following update:

$$\vartheta^{(t+1)} = \vartheta^{(t)} + \eta \cdot \nabla_\vartheta^{(t)} \tag{3}$$

Here, the disadvantage is its high sensitivity to the fixed learning rate η. However, as it does not subject the features to sum-to-one-contraints, it is considerably more time and memory efficient than GT.

As an improvement to SGD, **AdaGrad** (Duchi et al., 2011) is designed for large, sparse feature sets and makes use of an adaptive learning rate. It was proposed for MT training by (Green et al., 2013). Although its main area of application are online algorithms, it is also applicable in our offline setting and is more robust than SGD due to the adaptive learning rate. Following (Green et al., 2013), we apply the approximation with a diagonal outer product matrix, which is computationally cheap. This results in the update equations

$$\vartheta^{(t+1)} = \vartheta^{(t)} + \eta \cdot G_t^{-\frac{1}{2}} \cdot \nabla_\vartheta^{(t)} \tag{4}$$

$$G_t = G_{t-1} + (\nabla_\vartheta^{(t)})^2 \tag{5}$$

4.2 RPROP

The resilient backpropagation algorithm (RPROP) proposed by Riedmiller and Braun (1993) is a gradient-based optimization algorithm that emprirically learns the step size without taking the slope into account, making it highly robust and avoiding the need for a learning rate. If the gradient switches algebraic sign compared to the previous iteration, the last step is reverted and the step size reduced. If the sign remains the same, the step size is increased. Formally, given a set of parameters Θ and an objective function $O(\Theta)$, in iteration t each parameter $\vartheta \in \Theta$ is updated according to

$$\vartheta^{(t+1)} = \begin{cases} \vartheta^{(t-1)} & \text{, if } \quad \nabla_\vartheta^{(t-1)} \cdot \nabla_\vartheta^{(t)} < 0 \\ \vartheta^{(t)} + \Delta\vartheta^{(t)} & \text{, else if } \quad \nabla_\vartheta^{(t)} > 0 \\ \vartheta^{(t)} - \Delta\vartheta^{(t)} & \text{, else if } \quad \nabla_\vartheta^{(t)} < 0 \\ \vartheta^{(t)} & \text{, else} \end{cases}$$

where $\nabla_\vartheta^{(t)} := \frac{\delta O(\Theta^{(t)})}{\delta \vartheta}$ denotes the derivative of the objective function. The step size $\Delta\vartheta^{(t)} > 0$ grows or decreases depending on the sign of the gradient:

$$\Delta\vartheta^{(t)} = \begin{cases} \eta^+ \cdot \Delta\vartheta^{(t-1)} & \text{, if } \quad \nabla_\vartheta^{(t-1)} \cdot \nabla_\vartheta^{(t)} > 0 \\ \eta^- \cdot \Delta\vartheta^{(t-1)} & \text{, if } \quad \nabla_\vartheta^{(t-1)} \cdot \nabla_\vartheta^{(t)} < 0 \\ \Delta\vartheta^{(t-1)} & \text{, else} \end{cases}$$

The strength parameters $0 < \eta^- < 1 \leq \eta^+$ usually have little impact and are fixed to $\eta^- = 0.5$ and $\eta^+ = 1.2$ throughout this work. The RPROP algorithm is simple and easy to implement. It has proven effective for a number of tasks, e.g. in (Wiesler et al., 2013; Heigold et al., 2011; Lavergne et al., 2011; Hahn et al., 2011). Different from growth transformation (cf. Sec. 4.1), it does not assume a probability distribution and performs its updates without a sum-to-one constraint.

Compared to SGD and AdaGrad, RPROP's practical advantage is the absence of a learning rate that needs to be tuned. Further, we see its theoretical advantage in the empirically learned step size. In the first iterations, RPROP's updates are considerably smaller than with the other strategies, resulting in a more careful exploration of the search space. In higher iterations, the update steps for good features keep growing and we observe an exponential increase of the objective function. In contrast, GT, SGD, and AdaGrad determine the size of their update step based on the slope of the gradient, which we believe to be misleading given the complex topology of the feature space in MT.

5 Training

5.1 Maximum Expected BLEU

Following (He and Deng, 2012), we want to optimize a maximum expected BLEU objective. We denote the universe of possible sentences in the source language as $\mathbb{F}$ and in the target language as $\mathbb{E}$. The expected BLEU score under parameter set Θ with respect to the joint probability distribution $p_\Theta(\cdot, \cdot)$ is defined as

$$\langle \beta \rangle_\Theta = \sum_{F \in \mathbb{F}} \sum_{E \in \mathbb{E}} p_\Theta(E, F) \beta(E) \tag{6}$$

Here, $\beta(E)$ is the BLEU score for target sentence E (assuming the reference translation to be part of the mapping β) and we use the notation $\langle \cdot \rangle$ to denote the expectation. Enumerating all possible source and target sentences F, E is infeasible. Therefore, we estimate the empirical expectation on a corpus $\mathbb{C} \subset \mathbb{E} \times \mathbb{F}$. We denote the source sentences in $\mathbb{C}$ as $\mathbb{C}_F$ and the size of the corpus as $N = |\mathbb{C}|$. The joint probability $p_\Theta(E, F)$ is decomposed with the help of the Bayes Theorem, resulting in:

$$\langle \beta \rangle_\Theta = \sum_{F \in \mathbb{C}_F} p(F) \sum_{E \in \mathbb{E}_\Theta(F)} p_\Theta(E|F) \beta(E) \tag{7}$$

For $p(F) = \frac{N_F}{N}$ we assume the empirical distribution within the training corpus, where N_F is the count of sentence F. The summation over all $E \in \mathbb{E}$ is sampled with a subset $\mathbb{E}_\Theta(F)$ of the most likely hypotheses with respect to the parameterized probability $p_\Theta(E, F)$, which in practice is an n-best list generated by the decoder. Iterating over the corpus $\mathbb{C} = \{(E_1, F_1), \ldots, (E_n, F_n), \ldots, (E_N, F_N)\}$ finally results in

$$\langle \beta \rangle_\Theta = \frac{1}{N} \sum_{n=1}^{N} \sum_{E \in \mathbb{E}_\Theta(F_n)} p_\Theta(E|F_n) \beta(E)$$

We use the same unclipped sentence-level BLEU-4 score with smoothed 3-gram and 4-gram precisions as in (He and Deng, 2012), which we denote as $\beta(E) = \text{BLEU}(E, E_n^*)$ with respect to the reference translation E_n^*.

The normalized posterior translation probability $p_\Theta(E|F)$ from source sentence F to target sentence E approximates a maximum entropy model normalized on sentence level:

$$p_\Theta(E|F) = \frac{e^{-f_\Theta(E,F)}}{\sum_{E' \in \mathbb{E}_\Theta(F)} e^{-f_\Theta(E',F)}} \tag{8}$$

The denominator of this probability does not depend on the output sentence. Thus, the $\arg\max$ of Equation 8 is equal to the $\arg\max$ of the translation score in Equation 1.

Maximum Entropy models tend to generalize poorly, which can be circumvented by regularization. He and Deng (2012) use Kullback-Leibler regularization, raising the need of having normalized models $h_{m,\Theta}(E,F)$. We employ the more general L_2-regularization and the objective function is defined as

$$O(\Theta) = log\langle\beta\rangle_\Theta - \tau \cdot \sum_{\vartheta \in \Theta} \vartheta^2 \tag{9}$$

including the hyper parameter τ controlling the degree of regularization. The derivative of the objective function, which is needed for the gradient-based training methods, directly follows:

$$\frac{\delta O(\Theta)}{\delta \vartheta} = -\tau \cdot 2\vartheta + \frac{1}{\langle\beta\rangle_\Theta} \cdot \frac{\delta\langle\beta\rangle_\Theta}{\delta\vartheta} \tag{10}$$

With $\frac{\partial h_{m,\Theta}(E,F)}{\partial \vartheta} = \#_\vartheta(E,F)$ the number of times feature ϑ fires in the derivation for translation hypothesis E given source sentence F, the derivative of $p_\Theta(E|F)$ is defined as (for ease of notation $\mathbb{E}_\Theta(F_n)$ is represented by $\mathbb{E}_n$)

$$\begin{aligned}\frac{\partial p_\Theta(E|F)}{\partial\vartheta} &= -p_\Theta(E|F) \cdot \qquad (11)\\ &\Big(\#_\vartheta(E,F) - \sum_{E' \in \mathbb{E}_n} p_\Theta(E'|F)\#_\vartheta(E',F)\Big)\end{aligned}$$

And the derivative of the expected BLEU is

$$\begin{aligned}\frac{\delta\langle\beta\rangle_\Theta}{\delta\vartheta} &= \frac{1}{N}\sum_{n=1}^{N}\sum_{E\in\mathbb{E}_n}\beta(E)\frac{\partial p_\Theta(E|F_n)}{\partial\vartheta}\\ &= -\frac{1}{N}\sum_{n=1}^{N}\Bigg(\sum_{E\in\mathbb{E}_n} p_\Theta(E|F)\beta(E)\#_\vartheta(E,F)\\ &-\Big(\sum_{E\in\mathbb{E}_n} p_\Theta(E|F)\beta(E)\Big)\cdot\\ &\Big(\sum_{E\in\mathbb{E}_n} p_\Theta(E|F)\#_\vartheta(E,F)\Big)\Bigg) \qquad (12)\end{aligned}$$

This can be more compactly expressed by local expectations $\langle\cdot\rangle_n$ of the BLEU score and the feature count $\#_\vartheta$:

$$\frac{\delta\langle\beta\rangle_\Theta}{\delta\vartheta} = -\frac{1}{N}\sum_{n=1}^{N}(\langle\beta\#_\vartheta\rangle_n - \langle\beta\rangle_n\langle\#_\vartheta\rangle_n)$$

In our implementation, $\#_\vartheta$ is moved to the front of the equation to obtain common factors that can be used by all parameter updates:

$$\begin{aligned}\frac{\delta\langle\beta\rangle_\Theta}{\delta\vartheta} =& \frac{1}{N}\sum_{n=1}^{N}\sum_{E\in\mathbb{E}_n}\#_\vartheta(E,F)\cdot\\ & p_\Theta(E|F)(\langle\beta\rangle_n - \beta(E)) \qquad (13)\end{aligned}$$

5.2 Leave-one-out

Although He and Deng (2012) claim that it is not necessary, we apply a leave-one-out heuristic similar to (Wuebker et al., 2010) when generating the n-best lists on the training data. The authors have shown this to effectively counteract over-fitting effects and we argue that it helps to bring out the full potential of our discriminative training procedure.

When we decode the training data of our translation model, very long and rare phrases can be used to translate the sentence. The translation probability for these phrases, which are often singletons, are generally over-estimated by the heuristic count model. When they are too dominant in the n-best lists they effectively render the training data useless, as they are unlikely to generalize to unseen data. The idea of leave-one-out is that for decoding each sentence, the global counts of the relative frequency estimates are reduced by the local counts extracted from the current sentence pair. This way, the above mentioned rare phrases are penalized and the decoder is encouraged to use more general phrases taken from the remainder of the training data. Singleton phrases are given a fixed penalty. In this work, we apply leave-one-out with all update strategies.

5.3 Features

Maximum expected BLEU training facilitates training of arbitrary features. In this work we apply four types of features. (a) A discriminative phrase table, i.e. one feature for each phrase pair. (b) Lexical features, i.e. one feature for each source-target word pair that appear within the same phrase. (c) Source and target triplet features (Hasan et al., 2008), i.e. triples of one source and two target words or one target and two source words appearing within a single phrase pair. (d) The hierarchical lexicalized reordering model (Galley and Manning, 2008), i.e. one feature for each combination of phrase pair, orientation (monotone (M), swap (S) or discontinuous (D)) and orientation direction (forward or backward). GT is only applied with feature set (a), where we reestimate the two phrasal channel models as was done in (He and Deng, 2012). With the other update algorithms we follow the approach taken in (Auli et al., 2014) and condense each feature type into a small number of models for the log-linear combination, which is afterwards tuned with MERT. (a) and (b) result in a single additional model, (c) in two models (source and target triplets) and (d) in six models ($\{$forward,backward$\}\times\{$M,S,D$\}$).

5.4 Efficient Implementation

The expected BLEU $\langle\beta\rangle_\Theta$ is efficiently computed in one iteration over the full n-best list. As can be seen from Equation 13, the derivative $\frac{\delta\langle\beta\rangle_\Theta}{\delta\vartheta}$ is additive with respect to each firing instance of feature ϑ in the n-best list. The additive factor only depends on the current sentence pair. Therefore, for each sentence of the training data we iterate through its n-best list once to compute the expectation of the sentence-level BLEU score $\langle\beta\rangle_n$ and then a second time to update the current derivative for each time the feature fires. The only thing that needs to be kept in memory is a list of the current derivatives for each parameter ϑ.

```
1. Create the baseline system and run MERT
2. Generate $n$-best list on training corpus
3. Compute sentence-level BLEU $\beta(E_n)$
   for each hypothesis $E_n$ in the list
4. Initialize parameters with $\vartheta = 0, \forall\vartheta \in \Theta$
5. Iterate:
   a) Compute the derivatives $\frac{\delta O(\Theta)}{\delta\vartheta}$
   b) Perform update and output $\Theta^{(t)}$
6. Run MERT on dev with each table $\Theta^{(t)}$
7. Select best $\Theta^{(t)}$ on dev
8. Evaluate on test sets
```

Figure 1: The complete training algorithm.

5.5 Complete Training Algorithm

The complete training and evaluation procedure is shown in Figure 1. We start by building a baseline translation system with MERT-optimized model weights λ. With the baseline system we generate n-best lists on the training data. Now, for each translation hypothesis E_n of the n-best list, we compute the sentence-level BLEU score $\beta(E_n)$ and initialize the parameter set for training with the count model. Next, we run the training algorithm for a fixed number of iterations[1] and output the updated feature values $\Theta^{(t)}$ after each iteration t. Finally, we run MERT with each $\Theta^{(t)}$, select the best table on `dev` and evaluate on our test sets.

6 Experiments

6.1 Setup

The experiments are carried out on the IWSLT 2013 German→English shared translation task.[2] For rapid experimentation, the translation model is trained on the in-domain TED portion of the bilingual data, which is also used for maximum expected BLEU training. However, we use a large 4-gram LM with modified Kneser-Ney smoothing (Kneser and Ney, 1995; Chen and Goodman, 1998), trained with the SRILM toolkit (Stolcke, 2002). As additional data sources for the LM we use the complete News Commentary, Europarl v7 and Common Crawl corpora as well as selected parts of the Shuffled News

[1] Note that we keep the λ weights fixed throughout all iterations of maximum expected BLEU training.

[2] http://www.iwslt2013.org

	IWSLT		BOLT		WMT	
	German	**English**	**Chinese**	**English**	**German**	**English**
Sentences	138K		4.08M		4.09M	
Run. Words	2.63M	2.70M	78.3M	85.9M	105M	104M
Vocabulary	75.4K	50.2K	384K	817K	659K	649K

Table 1: Statistics for the bilingual training data of the IWSLT 2013 German→English, the DARPA BOLT Chinese→English and the WMT 2014 German→English tasks.

and LDC English Gigaword corpora. The selection is based on cross-entropy difference (Moore and Lewis, 2010). This makes for a total of 1.7 billion running words for LM training. The baseline further contains a hierarchical reordering model (HRM) (Galley and Manning, 2008) and a 7-gram word class language model (Wuebker et al., 2013). On IWSLT, all results are averages over three independent MERT runs, and we evaluate statistical significance with *MultEval* (Clark et al., 2011).

To confirm our findings, additional experiments are run on two large-scale tasks over strong baselines including recurrent neural language models. On the DARPA BOLT Chinese→English task we use our internal evaluation system as a baseline. It is a powerful hierarchical phrase-based SMT engine with 19 dense features, including an LSTM recurrent neural language model (Sundermeyer et al., 2012) and a hierarchical reordering model (Huck et al., 2013). The 5-gram backoff LM is in total trained on 2.9 billion running words. We use the same data for tuning and testing as Setiawan and Zhou (2013), namely 1275 (tune) and 1239[3] sentences of web data taken from LDC2010E30, the NIST MT06 evaluation set and an additional single-reference test set from the discussion forum (df) domain containing 1124 sentence pairs. Maximum expected BLEU training is performed on the discussion forum portion of the training data, consisting of 67.8K sentence pairs. On the German→English task of the *9th Workshop on Statistical Machine Translation*[4], both translation model and maximum expected BLEU training is performed on all available bilingual data. Our baseline is a phrase-based translation engine with a 4-gram backoff LM trained on 2.5 billion words with `lmplz` (Heafield et al., 2013), a recurrent neural LM, a 7-gram word class LM and the HRM.

IWSLT de-en	**# feat.**	**test**
baseline	18	30.4
GT (He and Deng, 2012)	6.08M	30.9
SGD (Auli et al., 2014)	921K	30.8
AdaGrad (Green et al., 2013)	921K	31.1
RPROP (this work)	921K	**31.3**
RPROP w/o leave-one-out	921K	30.7
RPROP all features	5.22M	**31.6**

Table 2: Results for the IWSLT 2013 German→English task in BLEU [%]. The comparison between update strategies is done with feature set (a) and *RPROP all features* uses feature sets (a)-(d). GT, SGD, AdaGrad and RPROP are trained with leave-one-out, unless otherwise specified.

Bilingual data statistics for all tasks are given in Table 1. We use the machine translation toolkit *Jane* (Vilar et al., 2010; Wuebker et al., 2012) and evaluate with case-insensitive BLEU [%] (Papineni et al., 2002) in all experiments.

6.2 Experimental Results

Table 2 shows the **IWSLT** results. We first compare the performance of the four update algorithms, for simplicity only on the discriminative phrase table features. Different from previous work the n-best lists of the training data were generated with leave-one-out, unless otherwise stated. In all cases we tested different values for the regularization parameter τ and in the case of SGD and AdaGrad also for the learning rate η. We selected the best configurations based on a validation set (test2011). For AdaGrad we also experimented with FOBOS regularization and feature selection (Duchi and Singer, 2009), but did not observe improved results. As

[3] named *dev* in (Setiawan and Zhou, 2013)

[4] `http://statmt.org/wmt14/`

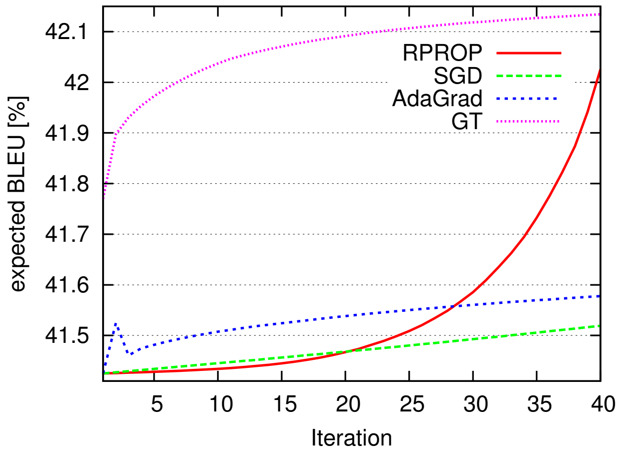

Figure 2: Expected BLEU value on IWSLT German→English for the different update strategies. Note that growth transformation (GT) applies a different regularization term and is therefore not directly comparable with the other techniques.

expected, we found that in all cases regularization is not strictly necessary - results are barely affected as long as τ is sufficiently small - and that SGD is much more sensitive to η than AdaGrad. Further, SGD and RPROP need around 25 iterations to reach good results, where 5-10 iterations are sufficient for GT and AdaGrad. For a fair comparison, however, we run all algorithms for 40 iterations and select the best one on a seletion set, namely iterations 19 (AdaGrad), 23 (GT), 29 (RPROP) and 35 (SGD). Figure 2 shows how the expected BLEU function evolves in training with different update strategies. Although the value for GT is not directly comparable to the others due to a different regularization term, the respective characteristics are clearly visible. SGD exhibits a linear growth pattern, GT resembles a logarithmic and RPROP an exponential function. After initially overshooting and then retracting as the regularization kicks in, AdaGrad also displays logarithmic characteristics.

In terms of BLEU RPROP performs best, followed by AdaGrad, GT and SGD, where the RPROP-AdaGrad and AdaGrad-GT differences are small (0.2% BLEU absolute) but statistically significant on the 95% level. Altogether, RPROP improves over the baseline by 0.9 BLEU points, which is statistically significant at the 99% level. In an additional experiment we verified that leave-one-out has a clear impact on the results. The BLEU difference between RPROP with and without leave-one-out is 0.6% absolute. By adding lexical, triplet and reordering features, we get an additional gain and observe a total improvement of 1.2 BLEU points over the baseline system.

BOLT zh-en	**# feat.**	**df**	**web**	**MT06**
baseline	19	18.0	34.1	39.7
SGD	12.4M	18.0	34.3	39.8
AdaGrad	12.4M	18.3	34.7	40.1
RPROP	12.4M	**18.7**	**34.8**	**40.5**
Setiawan&Zhou (GT)	150M	-	32.7	40.3

Table 3: Results for the BOLT Chinese→English task in BLEU [%] on the discussion forum test set (df), the mixed web test set and NIST MT06. The baseline is our BOLT evaluation system and contains a recurrent neural LM. We compare with (Setiawan and Zhou, 2013) who applied maximum expected BLEU training with growth transformation (GT). Note that the number of features reported by Setiawan and Zhou (2013) is artificially blown up due to renormalization.

Efficiency comparison. 921K discriminative phrase table features are active in our training data. Due to the renormalization component, this results in a total of 6.08M features that are updated with GT using the same data. Consequently, it is less time and space efficient than the other algorithms. With our implementation, GT needed around 16 hours and 6.7G memory for 40 iterations, where RPROP, AdaGrad and SGD finished after less than 2.5 hours and required 2.1G memory.

For the **BOLT** task, we directly compare with the GT-trained system in (Setiawan and Zhou, 2013) using the same tune set for MERT and reporting results on the same test sets, see Table 3. With RPROP we achieve nearly twice the improvement reported by Setiawan on both *web* and *MT06* using feature sets (a)-(c)[5]. Our baseline on *web* is already much stronger and RPROP training yields +0.7 BLEU points, as opposed to +0.44 reported by Setiawan. On *MT06* our baseline system is slightly worse, but with the larger gain received by RPROP our final system outperforms the one reported by Se-

[5] Reordering features are not applicable to our hiero system.

WMT de-en	# feat.	newstest2013
baseline	19	28.3
RPROP	45.0M	**28.9**
matrix.statmt.org	14	28.1

Table 4: Results for the WMT German→English task in BLEU [%]. The baseline contains a recurrent neural LM. We compare with the best single system that is reported on `matrix.statmt.org`, which was submitted by the Unversity of Edinburgh.

tiawan by 0.2 BLEU points. We would like to stress that this is not a domain adaptation effect, as maximum expected BLEU training was performed on discussion forum (df) data. On the *df* test set, on the other hand, we probably can observe domain adaptation via RPROP training. The improvement here is 0.7% BLEU absolute with a single reference, as opposed to four references on *web* and *MT06*. We also report results training the same feature sets with SGD and AdaGrad, confirming results we observed on IWSLT. Here, SGD yields only minor improvements. AdaGrad performs better, but still 0.1 - 0.4 BLEU points worse than RPROP. Running GT is infeasible in our hierarchical phrase-based setup.

Table 4 shows the results on the **WMT** task. This is our largest setting, where max. exp. BLEU training is performed on the full training data with more than 4M sentence pairs which, to the best of our knowledge, is unsurpassed in the literature. Altogether, training took more than one month, about 3/4 of which were for generating n-best lists by decoding the training data. The triplet features did not finish in time, so we applied the feature sets (a), (b) and (d), 45M features in total. With a renormalization step as in GT, this number would grow to 309M. On *newstest2013*, our baseline already outperforms the best single system reported on matrix.statmt.org by 0.2 BLEU points. The discriminatively trained features yield an additional improvement of 0.6% BLEU absolute on this high-end system.

7 Conclusion

We have experimentally compared several update strategies for maximum expected BLEU training. The RPROP algorithm proposed in this work shows superior performance compared to AdaGrad, growth transformation (GT) and stochastic gradient descent. In terms of time and memory efficiency, GT is clearly inferior to the other algorithms due to renormalization. Applying phrasal, lexical, triplet and reordering features, the baseline is improved by 1.2% BLEU absolute on the IWSLT German→English task. On two large scale tasks we achieve clearly superior performance compared to results reported in the literature. On BOLT Chinese→English our discriminative training yields nearly twice the improvement reported by Setiawan and Zhou (2013), resulting in a superior final system. On WMT German→English, we outperform the best single system reported on matrix.statmt.org by 0.8% BLEU absolute. Here, we perform maximum expexted BLEU training on more than 4M sentence pairs, which is the largest number reported in the literature to date.

Acknowledgments

The research leading to these results has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement n$^{\text{o}}$ 287658. This material is also partially based upon work supported by the DARPA BOLT project under Contract No. HR0011- 12-C-0015. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA. We further thank the anonymous reviewers for valuable comments.

References

Michael Auli, Michel Galley, and Jianfeng Gao. 2014. Large-scale Expected BLEU Training of Phrase-based Reordering ModelsI. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1250–1260, Doha, Qatar, October.

Phil Blunsom, Trevor Cohn, and Miles Osborne. 2008. A discriminative latent variable model for statistical machine translation. In *Proc. of ACL-HLT*.

Peter F. Brown, Stephan 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, June.

Francisco Casacuberta and Enrique Vidal. 2004. Machine translation with inferred stochastic finite-state transducers. *Computational Linguistics*, 30(2):205–225.

Stanley F. Chen and Joshuo Goodman. 1998. An Empirical Study of Smoothing Techniques for Language Modeling. Technical Report TR-10-98, Computer Science Group, Harvard University, Cambridge, MA, August.

D. Chiang, Y. Marton, and P. Resnik. 2008. Online large-margin training of syntactic and structural translation features. In *Conference on Empirical Methods in Natural Language Processing*, pages 224–233, Honolulu, Hawaii.

David Chiang. 2005. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 263–270, Ann Arbor, Michigan, USA, June.

Jonathan H. Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *49th Annual Meeting of the Association for Computational Linguistics:shortpapers*, pages 176–181, Portland, Oregon, June.

John Duchi and Yoram Singer. 2009. Efficient online and batch learning using forward backward splitting. *Journal of Machine Learning Research*, 10:2899–2934, December.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, July.

Michel Galley and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 848–856, Stroudsburg, PA, USA. Association for Computational Linguistics.

Jianfeng Gao and Xiaodong He. 2013. Training MRF-Based Phrase Translation Models using Gradient Ascent. In *Proceedings of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies conference (NAACL HLT)*, pages 450–459, Atlanta, Georgia, USA, Jun.

Spence Green, Sida Wang, Daniel Cer, and Christopher D. Manning. 2013. Fast and adaptive online training of feature-rich translation models. In *51st Annual Meeting of the Association for Computational Linguistics*, pages 311–321, Sofia, Bulgaria, August.

Spence Green, Daniel Cer, and Christopher D. Manning. 2014. An empirical comparison of features and tunign for phrase-based machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 466–476, Baltimore, Maryland, USA, June.

Stefan Hahn, Marco Dinarelli, Christian Raymond, Fabrice Lefevre, Patrick Lehnen, Renato De Mori, Alessandro Moschitti, Hermann Ney, and Giuseppe Riccardi. 2011. Comparing Stochastic Approaches to Spoken Language Understanding in Multiple Languages. *IEEE Transactions on Audio, Speech, and Language Processing*, 19(6):1569–1583, August.

Saša Hasan, Juri Ganitkevitch, Hermann Ney, and Jesús Andrés-Ferrer. 2008. Triplet lexicon models for statistical machine translation. In *Conference on Empirical Methods in Natural Language Processing*, pages 372–381, Honolulu, Hawaii, October. Association for Computational Linguistics.

Xiaodong He and Li Deng. 2012. Maximum Expected BLEU Training of Phrase and Lexicon Translation Models. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 292–301, Jeju, Republic of Korea, Jul.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 690–696, Sofia, Bulgaria, August.

Georg Heigold, Stefan Hahn, Patrick Lehnen, and Hermann Ney. 2011. EM-Style Optimization of Hidden Conditional Random Fields for Grapheme-to-Phoneme Conversion. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 4920–4923, Prague, Czech Republic, May.

Mark Hopkins and Jonathan May. 2011. Tuning as ranking. In *In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1352–1362, Edinburgh, Scotland, July.

Matthias Huck, Joern Wuebker, Felix Rietig, and Hermann Ney. 2013. A phrase orientation model for hierarchical machine translation. In *Workshop on Statistical Machine Translation*, pages 452–463, Sofia, Bulgaria, August.

Abraham Ittycheriah and Salim Roukos. 2007. Direct Translation Model 2. In *Proceedings of the North American Chapter of the Association for Computational Linguistics - Human Language Technologies conference (NAACL HLT)*, pages 57–64, Rochester, NY, USA, Apr.

Reinerd Kneser and Hermann Ney. 1995. Improved backing-off for M-gram language modeling. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processingw*, volume 1, pages 181–184, May.

P. Koehn, F. J. Och, and D. Marcu. 2003. Statistical Phrase-Based Translation. In *Proceedings of the 2003 Meeting of the North American chapter of the Association for Computational Linguistics (NAACL-03)*, pages 127–133, Edmonton, Alberta.

Thomas Lavergne, Alexandre Allauzen, Josep Maria Crego, and François Yvon. 2011. From n-gram-based to crf-based translation models. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 542–553, Edinburgh, Scotland, July. Association for Computational Linguistics.

Percy Liang, Alexandre Buchard-Côté, Dan Klein, and Ben Taskar. 2006. An End-to-End Discriminative Approach to Machine Translation. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 761–768, Sydney, Australia.

José B Mariño, Rafael E Banchs, Josep M Crego, Adrià de Gispert, Patrik Lambert, José A R Fonollosa, and Marta R Costa-jussà. 2006. N-gram-based Machine Translation. *Comput. Linguist.*, 32(4):527–549, December.

R.C. Moore and W. Lewis. 2010. Intelligent Selection of Language Model Training Data. In *ACL (Short Papers)*, pages 220–224, Uppsala, Sweden, July.

Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449, December.

Franz Josef Och. 2003. Minimum Error Rate Training in Statistical Machine Translation. In *Proc. of the 41th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 160–167, Sapporo, Japan, July.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA, July.

Martin Riedmiller and Heinrich Braun. 1993. A direct adaptive method for faster backpropagation learning: The rprop algorithm. In *IEEE International Conference on Neural Networks*, pages 586–591.

Hendra Setiawan and Bowen Zhou. 2013. Discriminative training of 150 million translation parameters and its application to pruning. In *NAACL-HLT 2013*, pages 335–341, Atlanta, Georgia, June.

Patrick Simianer, Stefan Riezler, and Chris Dyer. 2012. Joint Feature Selection in Distributed Stochastic Learning for Large-Scale Discriminative Training in SMT. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11–21, Jeju Island, Korea, July. Association for Computational Linguistics.

Andreas Stolcke. 2002. SRILM – An Extensible Language Modeling Toolkit. In *Proc. of the Int. Conf. on Speech and Language Processing (ICSLP)*, volume 2, pages 901–904, Denver, CO, September.

Martin Sundermeyer, Ralf Schlüter, and Hermann Ney. 2012. Lstm neural networks for language modeling. In *Interspeech*, Portland, OR, USA, September.

David Vilar, Daniel Stein, Matthias Huck, and Hermann Ney. 2010. Jane: Open source hierarchical translation, extended with reordering and lexicon models. In *ACL 2010 Joint Fifth Workshop on Statistical Machine Translation and Metrics MATR*, pages 262–270, Uppsala, Sweden, July.

Taro Watanabe, Jun Suzuki, Hajime Tsukada, and Hideki Isozaki. 2007. Online large-margin training for statistical machine translation. In *Proceedings of the 2007 Conference on Empirical Methods in Natural Language Processing*, pages 764–773, Prague, Czech Republic, June.

Simon Wiesler, Alexander Richard, Ralf Schlüter, and Hermann Ney. 2013. A Critical Evaluation of Stochastic Algorithms for Convex Optimization. In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 6955–6959, Vancouver, Canada, May.

Joern Wuebker, Arne Mauser, and Hermann Ney. 2010. Training phrase translation models with leaving-one-out. In *Proceedings of the 48th Annual Meeting of the Assoc. for Computational Linguistics*, pages 475–484, Uppsala, Sweden, July.

Joern Wuebker, Matthias Huck, Stephan Peitz, Malte Nuhn, Markus Freitag, Jan-Thorsten Peter, Saab Mansour, and Hermann Ney. 2012. Jane 2: Open source phrase-based and hierarchical statistical machine translation. In *International Conference on Computational Linguistics*, pages 483–491, Mumbai, India, December.

Joern Wuebker, Stephan Peitz, Felix Rietig, and Hermann Ney. 2013. Improving statistical machine translation with word class models. In *Conference on Empirical Methods in Natural Language Processing*, pages 1377–1381, Seattle, USA, October.

Heng Yu, Liang Huang, Haitao Mi, and Kai Zhao. 2013. Max-violation perceptron and forced decoding for scalable mt training. In *Conference on Empirical Methods in Natural Language Processing*, pages 1112–1123, Seattle, USA, October.