Expected BLEU Training for Graphs: BBN System Description for WMT11 System Combination Task

Antti-Veikko I. Rosti* and **Bing Zhang** and **Spyros Matsoukas** and **Richard Schwartz**
Raytheon BBN Technologies, 10 Moulton Street, Cambridge, MA 02138, USA
{arosti,bzhang,smatsouk,schwartz}@bbn.com

Abstract

BBN submitted system combination outputs for Czech-English, German-English, Spanish-English, and French-English language pairs. All combinations were based on confusion network decoding. The confusion networks were built using incremental hypothesis alignment algorithm with flexible matching. A novel bi-gram count feature, which can penalize bi-grams not present in the input hypotheses corresponding to a source sentence, was introduced in addition to the usual decoder features. The system combination weights were tuned using a graph based expected BLEU as the objective function while incrementally expanding the networks to bi-gram and 5-gram contexts. The expected BLEU tuning described in this paper naturally generalizes to hypergraphs and can be used to optimize thousands of weights. The combination gained about 0.5-4.0 BLEU points over the best individual systems on the official WMT11 language pairs. A 39 system multi-source combination achieved an 11.1 BLEU point gain.

1 Introduction

The confusion networks for the BBN submissions to the WMT11 system combination task were built using incremental hypothesis alignment algorithm with flexible matching (Rosti et al., 2009). A novel bi-gram count feature was used in addition to the standard decoder features. The N-best list based expected BLEU tuning (Rosti et al., 2010), similar to the one proposed by Smith and Eisner (2006), was extended to operate on word lattices. This method is closely related to the consensus BLEU (CoBLEU) proposed by Pauls et al. (2009). The minimum operation used to compute the clipped counts (matches) in the BLEU score (Papineni et al., 2002) was replaced by a differentiable function, so there was no need to use sub-gradient ascent as in CoBLEU. The expected BLEU (xBLEU) naturally generalizes to hypergraphs by simply replacing the forward-backward algorithm with inside-outside algorithm when computing the expected n-gram counts and sufficient statistics for the gradient.

The gradient ascent optimization of the xBLEU appears to be more stable than the gradient-free direct 1-best BLEU tuning or N-best list based minimum error rate training (Och, 2003), especially when tuning a large number of weights. On the official WMT11 language pairs with up to 30 weights, there was no significant benefit from maximizing xBLEU. However, on a 39 system multi-source combination (43 weights total), it yielded a significant gain over gradient-free BLEU tuning and N-best list based expected BLEU tuning.

2 Hypothesis Alignment and Features

The incremental hypothesis alignment with flexible matching (Rosti et al., 2009) produces a confusion network for each system output acting as a skeleton hypothesis for the ith source sentence. A confusion network is a graph where all paths visit all

*This work was supported by DARPA/I2O Contract No. HR0011-06-C-0022 under the GALE program (Approved for Public Release, Distribution Unlimited). The views, opinions, and/or findings contained in this article/presentation are those of the author/presenter and should not be interpreted as representing the official views or policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the Department of Defense.

Proceedings of the 6th Workshop on Statistical Machine Translation, pages 159–165,
Edinburgh, Scotland, UK, July 30–31, 2011. ©2011 Association for Computational Linguistics

vertices. Consecutive vertices are connected by one or more edges representing alternatives. Each edge l is associated with a token and a set of scores. A token may be a word, punctuation symbol, or special NULL token indicating a deletion in the alignment. The set of scores includes a vector of N_s system specific confidences, s_{iln}, indicating whether the token was aligned from the output of the system n.[1] Other scores may include a language model (LM) score as well as non-NULL and NULL token indicators (Rosti et al., 2007). As Rosti et al. (2010) described, the networks for all skeletons are connected to a start and end vertex with NULL tokens in order to form a joint lattice with multiple parallel networks. The edges connecting the start vertex to the initial vertices in each network have a heuristic prior estimated from the alignment statistics at the confidence corresponding to the skeleton system. The edges connecting the final vertices of each network to the end vertex have all system confidences set to one, so the final edge does not change the score of any path.

A single word confidence is produced from the confidence vector by taking an inner product with the system weights σ_n which are constrained to sum to one,[2] $\sum_n \sigma_n = 1$. The total edge score is produced by a log-linear interpolation of the word confidence with other features f_{ilm}:

$$s_{il} = \log\Big(\sum_{n=1}^{N_s} \sigma_n s_{iln}\Big) + \sum_m \lambda_m f_{ilm} \qquad (1)$$

The usual features f_{ilm} include the LM score as well as non-NULL and NULL token indicators. Based on an analysis of the system combination outputs, a large number of bi-grams not present in any input hypothesis are often produced, some of which are clearly ungrammatical despite the LM. These novel bi-grams are due to errors in hypothesis alignment and the confusion network structure where any word from the incoming edges of a vertex can be followed by any word from the outgoing edges. After expanding and re-scoring the joint lattice with a bi-gram, a new feature indicating the presence of a novel bi-gram may be added on the edges. A negative weight for this feature discourages novel bi-grams in the output during decoding.

3 Weight Optimization

The most common objective function used in machine translation is the BLEU-N score (Papineni et al., 2002) defined as follows:[3]

$$\text{BLEU} = \prod_{n=1}^{N} \left(\frac{\sum_i m_i^n}{\sum_i h_i^n}\right)^{\frac{1}{N}} \phi\left(1 - \frac{\sum_i r_i}{\sum_i h_i^1}\right) \qquad (2)$$

where N is the maximum n-gram order (typically $N = 4$), m_i^n is the number of n-gram matches (clipped counts) between the hypothesis e_i and reference $\hat{e}_i$ for segment i, h_i^n is the number of n-grams in the hypothesis, r_i is the reference length,[4] and $\phi(x) = \min(1.0, e^x)$ is the brevity penalty. Using g^n to represent an arbitrary n-gram, c_{ig^n} to represent the count of g^n in hypothesis e_i, and $\hat{c}_{ig^n}$ to represent the count of g^n in reference $\hat{e}_i$, the BLEU statistics can be defined as follows:

$$m_i^n = \sum_{g^n} \min(c_{ig^n}, \hat{c}_{ig^n}) \qquad (3)$$

$$h_i^n = \sum_{g^n} c_{ig^n} \qquad (4)$$

The unigram count h_i^1 is simply the hypothesis length and higher order n-gram counts can be obtained by $h_i^n = h_i^{n-1} - 1$. The reference n-gram counts for each sentence can be stored in an n-gram trie for efficient scoring.[5]

The BLEU score is not differentiable due to the minimum operations on the matches m_i^n and brevity penalty $\phi(x)$. Therefore gradient-free optimization algorithms, such as Powell's method or downhill simplex (Press et al., 2007), are often employed in weight tuning (Och, 2003). System combination weights tuned using the downhill simplex method to directly optimize 1-best BLEU score of the decoder outputs served as the first baseline in the experiments. The distributed optimization approach used here was first described in (Rosti et al., 2010).

[1] The confidences are binary when aligning 1-best outputs. More elaborate confidences may be estimated from N-best lists; see for example Rosti et al. (2007).

[2] See (Rosti et al., 2010) for a differentiable constraint.

[3] Superscripts indicate the n-gram order in all variables in this paper. They are used as exponents only for the constant e.

[4] If multiple references are available, r_i is the reference length closest to the hypothesis length, h_i^1.

[5] If multiple references are available, the maximum n-gram counts are stored.

A set of system combination weights was first tuned for unpruned lattices re-scored with a bi-gram LM. Another set of re-scoring weights was tuned for 300-best lists re-scored with a 5-gram LM.

3.1 Graph expected BLEU

Gradient-free optimization algorithms work well with a relatively small number of weights. Weight optimization for a 44 system combination in Rosti et al. (2010) was shown to be unstable with downhill simplex algorithm. Instead, an N-best list based expected BLEU tuning with gradient ascent yielded better results. This served as the second baseline in the experiments. The objective function is defined by replacing the n-gram statistics with expected n-gram counts and matches as in (Smith and Eisner, 2006), and brevity penalty with a differentiable approximation:

$$\varphi(x) = \frac{e^x - 1}{1 + e^{1000x}} + 1 \tag{5}$$

An N-best list represents a subset of the search space and multiple decoding iterations with N-best list merging is required to improve convergence. In this work, expected BLEU tuning is extended for lattices by replacing the minimum operation in n-gram matches with another differentiable approximation. The expected n-gram statistics for path j, which correspond to the standard statistics in Equations 3 and 4, are defined as follows:

$$\bar{m}_i^n = \sum_{g^n} \mu\Big(\sum_{j \in \mathcal{J}_i} P_{ij} c_{ijg^n}, \hat{c}_{ig^n} \Big) \tag{6}$$

$$\bar{h}_i^n = \sum_{g^n} \sum_{j \in \mathcal{J}_i} P_{ij} c_{ijg^n} \tag{7}$$

where $\mathcal{J}_i$ is the set of all paths in a lattice or all derivations in a hypergraph for the ith source sentence, P_{ij} is the posterior of path j, and c_{ijg^n} is the count of n-grams g^n in hypothesis e_{ij} on path j. The path posterior and approximate minimum are defined by:

$$P_{ij} = \frac{\prod_{l \in j} e^{\gamma s_{il}}}{\sum_{j' \in \mathcal{J}_i} \prod_{l \in j'} e^{\gamma s_{il}}} \tag{8}$$

$$\mu(x, c) = \frac{x - c}{1 + e^{1000(x-c)}} + c \tag{9}$$

where s_{il} is the total score on edge l defined in Equation 1 and γ is an edge score scaling factor. The scaling factor affects the shape of the edge posterior distribution; $\gamma > 1.0$ makes the edge posteriors on the 1-best path higher than edge posteriors on other paths and $\gamma < 1.0$ makes the posteriors on all paths more uniform.

The graph expected BLEU can be factored as $\text{xBLEU} = e^P B$ where:

$$P = \frac{1}{N} \sum_{n=1}^{N} \Big(\log \sum_i \bar{m}_i^n - \log \sum_i \bar{h}_i^n \Big) \tag{10}$$

$$B = \varphi\Big(1 - \frac{\sum_i r_i}{\sum_i \bar{h}_i^1}\Big) \tag{11}$$

and r_i is the reference length.[6] This objective function is closely related to CoBLEU (Pauls et al., 2009). Unlike CoBLEU, xBLEU is differentiable and standard gradient ascent algorithms can be used to find weights that maximize the objective.

Note, the expected counts can be expressed in terms of edge posteriors as:

$$\sum_{j \in \mathcal{J}_i} P_{ij} c_{ijg^n} = \sum_{l \in \mathcal{L}_i} p_{il} \delta(c_{il}^n, g^n) \tag{12}$$

where $\mathcal{L}_i$ is the set of all edges for the ith sentence, p_{il} is the edge posterior, $\delta(x, c)$ is the Kronecker delta function which is 1 if $x = c$ and 0 if $x \neq c$, and c_{il}^n is the n-gram context of edge l. The edge posteriors can be computed via standard forward-backward algorithm for lattices or inside-outside algorithm for hypergraphs. As with the BLEU statistics, only expected unigram counts $\bar{h}_i^1$ need to be accumulated for the hypothesis n-gram counts in Equation 7 as $\bar{h}_i^n = \bar{h}_i^{n-1} - 1$ for $n > 1$. Also, the expected n-gram counts for each graph can be stored in an n-gram trie for efficient gradient computation.

3.2 Gradient of graph expected BLEU

The gradient of the xBLEU with respect to weight λ can be factored as:

$$\frac{\partial \text{xBLEU}}{\partial \lambda} = \sum_i \sum_{l \in \mathcal{L}_i} \frac{\partial s_{il}}{\partial \lambda} \sum_{j \in \mathcal{J}_i} \frac{\partial \text{xBLEU}}{\partial \log P_{ij}} \frac{\partial \log P_{ij}}{\partial s_{il}} \tag{13}$$

where the gradient of the log-path-posterior with respect to the edge score is given by:

$$\frac{\partial \log P_{ij}}{\partial s_{il}} = \gamma\Big(\delta(l \in j) - p_{il}\Big) \tag{14}$$

[6] If multiple reference are available, r_i is the reference length closest to the expected hypothesis length $\bar{h}_i^1$.

$$\frac{\partial \text{xBLEU}}{\partial \lambda} = \gamma e^P \left(\frac{B}{N} \sum_{n=1}^{N} \sum_i \left(\frac{\hat{m}_{ik}^n - m_{ik}^n}{m^n} - \frac{\hat{h}_{ik}^n - h_{ik}^n}{h^n} \right) \right) + C\varphi'(1-C) \sum_i \frac{\hat{h}_{ik}^1 - h_{ik}^1}{h^1} \quad (15)$$

and $\delta(l \in j)$ is one if edge l is on path j, and zero otherwise. Using the factorization $\text{xBLEU} = e^P B$, Equation 13 can be expressed using sufficient statistics as shown in Equation 15, where $\varphi'(x)$ is the derivative of $\varphi(x)$ with respect to x, $m^n = \sum_i \bar{m}_i^n$, $h^n = \sum_i \bar{h}_i^n$, $C = \sum r_i / \sum_i \bar{h}_i^1$, and the remaining sufficient statistics are given by:

$$\mu'_{ig^n} = \mu'\Big(\sum_{j \in \mathcal{J}_i} P_{ij} c_{ijg^n}, \hat{c}_{ig^n} \Big)$$

$$m_{ik}^n = \Big(\sum_{l \in \mathcal{L}_i} p_{il} \frac{\partial s_{il}}{\partial \lambda} \Big) \Big(\sum_{j \in \mathcal{J}_i} P_{ij} \sum_{g^n} \mu'_{ig^n} c_{ijg^n} \Big)$$

$$\hat{m}_{ik}^n = \sum_{l \in \mathcal{L}_i} \frac{\partial s_{il}}{\partial \lambda} \sum_{j : l \in \mathcal{J}_i} P_{ij} \sum_{g^n} \mu'_{ig^n} c_{ijg^n}$$

$$h_{ik}^n = \Big(\sum_{l \in \mathcal{L}_i} p_{il} \frac{\partial s_{il}}{\partial \lambda} \Big) \Big(\sum_{j \in \mathcal{J}_i} P_{ij} \sum_{g^n} c_{ijg^n} \Big)$$

$$\hat{h}_{ik}^n = \sum_{l \in \mathcal{L}_i} \frac{\partial s_{il}}{\partial \lambda} \sum_{j : l \in \mathcal{J}_i} P_{ij} \sum_{g^n} c_{ijg^n}$$

where $\mu'(x, c)$ is the derivative of $\mu(x, c)$ with respect to x, and the parentheses in the equations for m_{ik}^n and h_{ik}^n signify that the second terms do not depend on the edge l.

3.3 Forward-backward algorithm under expectation semiring

The sufficient statistics for graph expected BLEU can be computed using expectation semirings (Li and Eisner, 2009). Instead of computing single forward/backward or inside/outside scores, additional n-gram elements are tracked for matches and counts. For example in a bi-gram graph, the elements for edge l are represented by a 5-tuple[7] $s_l = \langle p_l, r_{lh}^1, r_{lh}^2, r_{lm}^1, r_{lm}^2 \rangle$ where $p_l = e^{\gamma s_{il}}$ and:

$$r_{lh}^n = \sum_{g^n} \delta(c_{il}^n, g^n) e^{\gamma s_{il}} \quad (16)$$

$$r_{lm}^n = \sum_{g^n} \mu'_{ig^n} e^{\gamma s_{il}} \quad (17)$$

Assuming the lattice is topologically sorted, the forward algorithm[8] under expectation semiring for a 3-tuple[9] $s_l = \langle p_l, r_{lh}^1, r_{lm}^1 \rangle$ is defined by:

$$\alpha_0 = \langle 1, 0, 0 \rangle \quad (18)$$

$$\alpha_v = \bigoplus_{l \in \mathcal{I}_v} \alpha_{u(l)} \otimes s_l \quad (19)$$

where $\mathcal{I}_v$ is the set of all edges with target vertex v and $u(l)$ is the source vertex for edge l, and the operations are defined by:

$$s_1 \oplus s_2 = \langle p_1 + p_2, r_{1h}^1 + r_{2h}^1, r_{1m}^1 + r_{2m}^1 \rangle$$
$$s_1 \otimes s_2 = \langle p_1 p_2, p_1 r_{2h}^1 + p_2 r_{1h}^1, p_1 r_{2m}^1 + p_2 r_{1m}^1 \rangle$$

The backward algorithm for β_u can be implemented via the forward algorithm in reverse through the graph. The sufficient statistics for the gradient can be accumulated during the backward pass noting that:

$$\sum_{j \in \mathcal{J}_i} P_{ij} \sum_{g^n} \mu'_{ig^n} c_{ijg^n} = \frac{r_m^n(\beta_0)}{p(\beta_0)} \quad (20)$$

$$\sum_{j \in \mathcal{J}_i} P_{ij} \sum_{g^n} c_{ijg^n} = \frac{r_h^n(\beta_0)}{p(\beta_0)} \quad (21)$$

where $r_m^n(\cdot)$ and $r_h^n(\cdot)$ extract the nth order r elements from the tuple for matches and counts, respectively, and $p(\cdot)$ extracts the p element. The statistics for the paths traveling via edge l can be computed by:

$$\sum_{j : l \in \mathcal{J}_i} P_{ij} \sum_{g^n} \mu'_{ig^n} c_{ijg^n} = \frac{r_m^n(\alpha_u \otimes s_l \otimes \beta_v)}{p(\beta_0)} \quad (22)$$

$$\sum_{j : l \in \mathcal{J}_i} P_{ij} \sum_{g^n} c_{ijg^n} = \frac{r_h^n(\alpha_u \otimes s_l \otimes \beta_v)}{p(\beta_0)} \quad (23)$$

where the u and v subscripts in α_u and β_v are the start and end vertices for edge l. To avoid underflow, all the computations can be carried out in log domain.

[7] The sentence index i is dropped for brevity.

[8] For inside-outside algorithm, see (Li and Eisner, 2009).

[9] A 3-tuple for uni-gram counts is used as an example in order to save space. In a 5-tuple for bi-gram counts, all r elements are computed independently of other r elements with the same operations. Similarly, tri-gram counts require 7-tuples and four-gram counts require 9-tuples.

tune	cz-en		de-en		es-en		fr-en	
System	TER	BLEU	TER	BLEU	TER	BLEU	TER	BLEU
worst	66.03	18.09	69.03	16.28	60.56	21.02	62.75	21.83
best	53.75	28.36	58.39	24.28	50.26	30.55	50.48	30.87
latBLEU	53.99	29.25	56.70	26.49	48.34	34.55	48.90	33.90
nbExpBLEU	54.43	29.04	56.36	27.33	48.44	34.73	48.58	34.23
latExpBLEU	53.89	29.37	56.24	27.36	48.27	34.93	48.53	34.24

test	cz-en		de-en		es-en		fr-en	
System	TER	BLEU	TER	BLEU	TER	BLEU	TER	BLEU
worst	65.35	17.69	69.03	15.83	61.22	19.79	62.36	21.36
best	52.21	29.54	58.00	24.16	50.15	30.14	50.15	30.32
latBLEU	52.80	29.89	55.87	26.22	48.29	33.91	48.51	32.93
nbExpBLEU	52.97	29.93	55.77	26.52	48.39	33.86	48.25	32.94
latExpBLEU	52.68	29.99	55.74	26.62	48.30	34.10	48.17	32.91

Table 1: Case insensitive TER and BLEU scores on `newssyscombtune` (tune) and `newssyscombtest` (test) for combinations of outputs from four source languages. Three tuning methods were used: lattice BLEU (latBLEU), N-best list based expected BLEU (nbExpBLEU), and lattice expected BLEU (latExpBLEU).

3.4 Entropy on a graph

Expanding the joint lattice to n-gram orders above $n = 2$ is often impractical without pruning. If the edge posteriors are not reliable, which is usually the case for unoptimized weights, pruning might remove good quality paths from the graph. As a compromise, an incremental expansion strategy may be adopted by first expanding and re-scoring the lattice with a bi-gram, optimizing weights for xBLEU-2, and then expanding and re-scoring the lattice with a 5-gram. Pruning should be more reliable with the edge posteriors computed using the tuned bi-gram weights. A second set of weights may be tuned with the 5-gram graph to maximize xBLEU-4.

When the bi-gram weights are tuned, it may be beneficial to increase the edge score scaling factor to focus the edge posteriors to the 1-best path. On the other hand, a lower scaling factor may be beneficial when tuning the 5-gram weights. Rosti et al. (2010) determined the scaling factor automatically by fixing the perplexity of the merged N-best lists used in tuning. Similar strategy may be adopted in incremental n-gram expansion of the lattices.

Entropy on a graph can also be computed using the expectation semiring formalism (Li and Eisner, 2009) by defining $s_l = \langle p_l, r_l \rangle$ where $p_l = e^{\gamma s_{il}}$ and $r_l = \log p_l$. The entropy is given by:

$$H_i = \log p(\beta_0) - \frac{r(\beta_0)}{p(\beta_0)} \tag{24}$$

where $p(\beta_0)$ and $r(\beta_0)$ extract the p and r elements from the 2-tuple β_0, respectively. The average target entropy over all sentences was set manually to 3.0 in the experiments based on the tuning convergence and size of the pruned 5-gram lattices.

4 Experimental Evaluation

System outputs for all language pairs with English as the target were combined (`cz-en`, `de-en`, `es-en`, and `fr-en`). Unpruned English bi-gram and 5-gram language model components were trained using the WMT11 corpora: `EuroParl`, `GigaFrEn`, `UNDoc_Es`, `UNDoc_Fr`, `NewsCommentary`, `News2007`, `News2008`, `News2009`, `News2010`, and `News2011`. Additional six Gigaword v4 components included: `AFP`, `APW`, `XIN+CNA`, `LTW`, `NYT`, and `Headlines+Datelines`. The total number of words used to train the LMs was about 6.4 billion. Interpolation weights for the sixteen components were tuned to minimize perplexity on the `newstest2010-ref.en` development set. The modified Kneser-Ney smoothing (Chen and

Goodman, 1998) was used in training. Experiments using a LM trained on the system outputs and interpolated with the general LM were also conducted. The interpolation weights between 0.1 and 0.9 were tried, and the weight yielding the highest BLEU score on the tuning set was selected. A tri-gram true casing model was trained on all the LM training data. This model was used to restore the case of the lower-case system combination output.

All twelve 1-best system outputs on `cz-en`, 26 outputs on `de-en`, 16 outputs on `es-en`, and 24 outputs on `fr-en` were combined. Three different weight optimization methods were tried. First, lattice based 1-best BLEU optimization of the bi-gram decoding weights followed by N-best list based BLEU optimization of 5-gram re-scoring weights using 300-best lists, both using downhill simplex. Second, N-best list based expected BLEU optimization of the bi-gram and 5-gram weights using 300-best lists with merging between bi-gram decoding iterations. Third, lattice based expected BLEU optimization of bi-gram and 5-gram decoding weights. The L-BFGS (Liu and Nocedal, 1989) algorithm was used in gradient ascent. Results for all four single source experiments are shown in Table 1, including case insensitive TER (Snover et al., 2006) and BLEU scores for the worst and best systems, and the system combination outputs for the three tuning methods. The gains on tuning and test sets were consistent, though relatively smaller on `cz-en` due to a single system (`online-B`) dominating the other systems by about 5-6 BLEU points. The tuning method had very little influence on the test set scores apart from `de-en` where the lattice BLEU optimization yields slightly lower BLEU scores. This seems to suggest that the gradient free optimization is not as stable with a larger number of weights.[10] The novel bi-gram feature did not have significant influence on the TER or BLEU scores, but the number of novel bi-grams was reduced by up to 100%.

Finally, experiments combining 39 system outputs by taking the top half of the outputs from each language pair were performed. The selection was based on case insensitive BLEU scores on the tuning set. Table 2 shows the scores for seven combinations using the three tuning methods with or without the system output biased LM, and finally without the novel bi-gram count feature. There is a clear advantage from the expected BLEU tuning on the tuning set, and lattice tuning yields better scores than N-best list based tuning. The difference between `latBLEU` and `nbExpBLEU` without `biasLM` is not quite as large on the test set but `latExpBLEU` yields significant gains over both. The `biasLM` also yields significant gains on all but `latBLEU` tuning. Finally, removing the novel bi-gram count feature results in a significant loss, probably due to the large number of input hypotheses. The number of novel bi-grams in the test set output was reduced to zero when using this feature.

`xx-en`	**tune**		**test**	
System	TER	BLEU	TER	BLEU
worst	62.81	21.19	62.92	20.29
best	51.11	30.87	50.80	30.32
latBLEU	40.95	40.75	41.06	39.81
+biasLM	41.18	40.90	41.16	39.90
nbExpBLEU	40.81	41.36	41.05	40.15
+biasLM	40.72	41.99	40.65	40.89
latExpBLEU	40.57	41.68	40.62	40.60
+biasLM	40.42	42.23	40.52	41.38
-nBgF	40.85	41.41	40.88	40.55

Table 2: Case insensitive TER and BLEU scores on `newssyscombtune` (tune) and `newssyscombtest` (test) for `xx-en` combination. Combinations using lattice BLEU tuning (latBLEU), N-best list based expected BLEU tuning (nbExpBLEU), and lattice expected BLEU tuning (latExpBLEU) with and without the system output biased LM (`biasLM`) are shown. Final row, marked `nBgF`, corresponds to the above tuning without the novel bi-gram feature.

5 Conclusions

The BBN submissions for WMT11 system combination task were described in this paper together with a differentiable objective function, graph expected BLEU, which scales well for a large number of weights and can be generalized to hypergraphs. System output biased language model and a novel bi-gram count feature also gave significant gains on a 39 system multi-source combination.

[10] A total number of 30 weights, 26 system and 4 feature weights, were tuned for `de-en`.

References

Stanley F. Chen and Joshua Goodman. 1998. An empirical study of smoothing techniques for language modeling. Technical Report TR-10-98, Computer Science Group Harvard University.

Zhifei Li and Jason Eisner. 2009. First- and second-order expectation semirings with applications to minimum-risk training on translation forests. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 40–51.

Dong C. Liu and Jorge Nocedal. 1989. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(3):503–528.

Franz J. Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Adam Pauls, John DeNero, and Dan Klein. 2009. Consensus training for consensus decoding in machine translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1418–1427.

William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. 2007. *Numerical recipes: the art of scientific computing*. Cambridge University Press, 3rd edition.

Antti-Veikko I. Rosti, Spyros Matsoukas, and Richard Schwartz. 2007. Improved word-level system combination for machine translation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 312–319.

Antti-Veikko I. Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. 2009. Incremental hypothesis alignment with flexible matching for building confusion networks: BBN system description for WMT09 system combination task. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 61–65.

Antti-Veikko I. Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. 2010. BBN system description for WMT10 system combination task. In *Proceedings of the Fifth Workshop on Statistical Machine Translation*, pages 321–326.

David A. Smith and Jason Eisner. 2006. Minimum risk annealing for training log-linear models. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 787–794.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciula, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas*, pages 223–231.