Improving the Minimum Bayes' Risk Combination of Machine Translation Systems

Jesús González-Rubio, Francisco Casacuberta

Departamento de sistemas informáticos y computación
Universitat Politècnica de València, camino de Vera s/n, 46022 Valencia, Spain
{jegonzalez,fcn}@dsic.upv.es

Abstract

We investigate the problem of combining the outputs of different translation systems into a minimum Bayes' risk consensus translation. We explore different risk formulations based on the BLEU score, and provide a dynamic programming decoding algorithm for each of them. In our experiments, these algorithms generated consensus translations with better risk, and more efficiently, than previous proposals.

1. Introduction

Machine translation (MT) is a fundamental technology and a core component of language processing systems. However, MT systems are still far from perfect [1]. The combination of multiple MT systems is a promising research direction to improve the quality of current MT technology. The key idea of system combination [2] is that it is often very difficult to find the real best system for the task at hand, while different systems can exhibit complementary strengths and limitations. Thus, a proper combination of systems could be more effective than using a single monolithic system.

A simple, yet effective, system combination method for MT was proposed by González-Rubio et al., [3]. The authors describe minimum Bayes' risk system combination (MBRSC), a method to combine the outputs of multiple MT systems into a consensus translation with maximum expected BLEU [4] score. Previous combination methods either implement sophisticated decision functions to select one of the provided translations [5, 6, 7], or generate new consensus translations by combining the best subsequences of the provided translations by means of a Viterbi-like search on a confusion network [8, 9, 10]. MBRSC aims at gathering together the advantages of sentence-selection and subsequence-combination methods. In comparison to sentence-selection methods, MBRSC also implements a sophisticated minimum Bayes' risk (MBR) classifier, and additionally, it is able to generate new consensus translations that include the "best" subsequences from different individual translations. Regarding subsequence-combination methods, MBRSC can also generate new consensus translations different from the provided translations, and also, the final consensus translation has the best expected score with respect to the widespread BLEU score.

Despite these advantages, the original implementation of MBRSC [3] (§2) presented some flaws, e.g. the proposed gradient ascent decoding, that, in our opinion, prevents the method from revealing its full potential. Here, we propose new decoding algorithms for MBRSC based on the dynamic programming [11] (DP) paradigm. We study two different approaches to compute the BLEU-based risk. On the one hand, we instantiate DP decoding to use the original BLEU risk over expected counts (§3) so our results are comparable to those in [3]. In practice, this approach is implemented as a beam search [12]. On the other hand, we implement an actual exact DP decoding using the linear approximation to the BLEU score proposed in [13] to compute the risk (§4). Then, we provide an extensive empirical study (§5) of the proposed decoding algorithms in comparison to the original MBRSC proposal. Finally, we conclude with a summary of our contributions.

2. Minimum Bayes' Risk System Combination

2.1. MBRSC Model and Decision Function

We now describe the original MBRSC proposal in [3]. Given K MT systems, MBRSC models the probability of a sentence $\mathbf{y}$ to be a translation of a source sentence $\mathbf{x}$ as a weighted ensemble [14]:

$$P(\mathbf{y} \mid \mathbf{x}) = \sum_{k=1}^{K} \alpha_k \cdot P_k(\mathbf{y} \mid \mathbf{x}) \tag{1}$$

where $P_k(\mathbf{y} \mid \mathbf{x})$ denotes the probability distribution over translations modeled by system k. Free parameters $\{\alpha_1, \ldots, \alpha_K\}$ are scaling factors that denote the relative importance of each system ($\sum_{k=1}^{K} \alpha_k = 1$).

Given a loss function $\mathrm{L}(\mathbf{y}, \mathbf{y}')$ between a candidate translation $\mathbf{y}$ and a reference translation $\mathbf{y}'$, the optimal decision function for the ensemble model of MBRSC is an instance of the MBR classifier [15]:

$$\begin{aligned}\hat{\mathbf{y}} &= \arg\min_{\mathbf{y}\in\mathcal{Y}} \mathrm{R}(\mathbf{y} \mid \mathbf{x}) \\ &= \arg\min_{\mathbf{y}\in\mathcal{Y}} \mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}\left[\mathrm{L}(\mathbf{y}, \mathbf{y}')\right] \\ &= \arg\min_{\mathbf{y}\in\mathcal{Y}} \sum_{\mathbf{y}'\in\mathcal{Y}} P(\mathbf{y}' \mid \mathbf{x}) \cdot \mathrm{L}(\mathbf{y}, \mathbf{y}') \end{aligned} \quad (2)$$

where $\mathrm{R}(\mathbf{y} \mid \mathbf{x})$ denotes the Bayes' risk, namely the expected loss ($\mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}\left[\mathrm{L}(\mathbf{y}, \mathbf{y}')\right]$), of translation $\mathbf{y}$, and $\mathcal{Y}$ denotes the whole target language.

MBRSC uses the widespread BLEU [4] metric as loss function. The BLEU score $\mathrm{B}(\mathbf{y}, \mathbf{y}')$ between a candidate translation $\mathbf{y}$ and a reference $\mathbf{y}'$ is given by:

$$\mathrm{B}(\mathbf{y}, \mathbf{y}') = \left(\prod_{n=1}^{4} \rho_n(\mathbf{y}, \mathbf{y}')\right)^{\frac{1}{4}} \cdot \phi(\mathbf{y}, \mathbf{y}') \quad (3)$$

where $\rho_n(\mathbf{y}, \mathbf{y}')$ is the precision of n-grams of size n between $\mathbf{y}$ and $\mathbf{y}'$, and $\phi(\mathbf{y}, \mathbf{y}')$ is a brevity penalty, that penalizes short translations:

$$\rho_n(\mathbf{y}, \mathbf{y}') = \frac{\sum_{\mathbf{w}\in\mathcal{W}_n(\mathbf{y})} \min(\#_{\mathbf{w}}(\mathbf{y}), \#_{\mathbf{w}}(\mathbf{y}'))}{\sum_{\mathbf{w}\in\mathcal{W}_n(\mathbf{y})} \#_{\mathbf{w}}(\mathbf{y})} \quad (4)$$

$$\phi(\mathbf{y}, \mathbf{y}') = \min\left(\exp\left(1 - \frac{|\,\mathbf{y}'\,|}{|\,\mathbf{y}\,|}\right), 1\right) \quad (5)$$

where $\mathcal{W}_n(\mathbf{y})$ is the set of n-grams of size n in $\mathbf{y}$, $\#_{\mathbf{w}}(\mathbf{y})$ is the count of n-gram $\mathbf{w}$ in $\mathbf{y}$, and $|\,\mathbf{y}\,|$ denotes the length of translation $\mathbf{y}$.

BLEU is a percentage with a value of one denoting an exact match between $\mathbf{y}$ and $\mathbf{y}'$. Thus, we rewrite the MBRSC decision function in Equation (2) substituting the $\arg\min_{\mathbf{y}\in\mathcal{Y}}$ operator by an $\arg\max_{\mathbf{y}\in\mathcal{Y}}$:

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}\in\mathcal{Y}} \sum_{k=1}^{K} \alpha_k \cdot \underbrace{\left(\sum_{\mathbf{y}'\in\mathcal{Y}} P_k(\mathbf{y}' \mid \mathbf{x}) \cdot \mathrm{B}(\mathbf{y}, \mathbf{y}')\right)}_{\text{system-specific loss}} \quad (6)$$

This formulation assumes that all systems share the same domain of translations ($\mathcal{Y}$) which in practice it is not always true. In practice, MBRSC takes as input a representation, e.g. an N-best list, of the candidate translations of each system and assumes that any other translation not in the provided representation has zero probability of being generated by that system.

Optimum values for scaling factors α_k are estimated by minimum error rate training [16] optimizing BLEU on a separate development set.

2.2. MBRSC Decoding

The direct implementation of Equation (6) has a high temporal complexity in $O(|\,\mathcal{Y}\,|^2 \cdot I)$, where $|\,\mathcal{Y}\,|$ denotes the number of candidate translations, and I represents the maximum translation length given that $\mathrm{B}(\mathbf{y}, \mathbf{y}')$ can be computed in $O(\max(|\,\mathbf{y}\,|, |\,\mathbf{y}'\,|))$ time. Since the number of candidate translations may be quite large, an exhaustive enumeration of all of them is often unfeasible. González-Rubio et. al [3] address this challenge by dividing Equation (6) into two sub-problems: the computation of the risk, namely the expected BLEU score, of each translation, and the actual search for the optimal consensus translation ($\arg\max_{\mathbf{y}\in\mathcal{Y}}$).

Given that BLEU references the reference translation $\mathbf{y}'$ only via its n-gram counts (see Equation (3)), MBRSC follows [17] to formalize an efficient alternative to the exact risk in Equation (6). Instead of computing the expected BLEU score of translation $\mathbf{y}$, MBRSC computes the BLEU score of $\mathbf{y}$ with respect to the expected n-gram counts $\mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\#_{\mathbf{w}}(\mathbf{y}')]$ in the alternative candidate translations of $\mathbf{x}$:

$$\begin{aligned}\mathrm{R}(\mathbf{y} \mid \mathbf{x}) &= \mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\mathrm{B}(\mathbf{y}, \mathbf{y}')] \\ &\approx \widetilde{\mathrm{B}}(\mathbf{y}, \mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\#_{\mathbf{w}}(\mathbf{y}')]) \\ &= \left(\prod_{n=1}^{4} \widetilde{\rho_n}(\mathbf{y}, \mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\#_{\mathbf{w}}(\mathbf{y}')])\right)^{\frac{1}{4}} \cdot \\ &\quad \widetilde{\phi}(\mathbf{y}, \mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\#_{\mathbf{w}}(\mathbf{y}')]) \end{aligned} \quad (7)$$

where $P(\mathbf{y}' \mid \mathbf{x})$ is the ensemble probability in Equation (1), and $\rho_n(\mathbf{y}, \mathbf{y}')$ and $\phi(\mathbf{y}, \mathbf{y}')$ are reformulated as functions of expected n-gram counts.

Regarding the actual search, MBRSC implements a two-step algorithm. First, it performs a conventional MBR sentence-selection decoding [18] to obtain an initial consensus translation. Then, a gradient ascent algorithm refines that initial solution by the iterative application of different edit operations (substitution, insertion, and deletion of single words) searching for an improvement in risk. Algorithm 1 depicts this gradient ascent decoding algorithm. Since the risk ($\mathrm{R}(\mathbf{y} \mid \mathbf{x})$ in Equation (7)) can be computed in $O(I)$[1], the com-

[1] Expected n-gram counts can be computed in advance.

Algorithm 1: MBRSC gradient ascent search [3].

input : $\mathbf{y}_0$ (initial solution)
Σ (target language vocabulary)
I (maximum translation length)
output : $\hat{\mathbf{y}}, \mathrm{R}(\hat{\mathbf{y}} \mid \mathbf{x})$ (best translation and its score)
auxiliary : $\mathrm{R}(\mathbf{y} \mid \mathbf{x})$ (expected BLEU score of $\mathbf{y}$)
$\mathrm{sub}(\mathbf{y}, y, i)$ (replaces i^{th} word of $\mathbf{y}$ by y)
$\mathrm{del}(\mathbf{y}, i)$ (deletes the i^{th} word of $\mathbf{y}$)
$\mathrm{ins}(\mathbf{y}, y, i)$ (inserts y as the i^{th} word of $\mathbf{y}$)

begin
  $\hat{\mathbf{y}} \leftarrow \mathbf{y}_0$;
  repeat
    $\mathbf{y}_c \leftarrow \hat{\mathbf{y}}$;
    for $1 \leq i \leq |\,\mathbf{y}_c\,|$ **do**
      $\hat{\mathbf{y}}_s \leftarrow \mathbf{y}_c$; $\hat{\mathbf{y}}_i \leftarrow \mathbf{y}_c$;
      for $y \in \Sigma$ **do**
        $\mathbf{y}_s \leftarrow \mathrm{sub}(\mathbf{y}_c, y, i)$;
        if $\mathrm{R}(\mathbf{y}_s \mid \mathbf{x}) \geq \mathrm{R}(\hat{\mathbf{y}}_s \mid \mathbf{x})$ **then**
          $\hat{\mathbf{y}}_s \leftarrow \mathbf{y}_s$;
        $\mathbf{y}_i \leftarrow \mathrm{ins}(\mathbf{y}_c, y, i)$;
        if $\mathrm{R}(\mathbf{y}_i \mid \mathbf{x}) \geq \mathrm{R}(\hat{\mathbf{y}}_i \mid \mathbf{x})$ **then**
          $\hat{\mathbf{y}}_i \leftarrow \mathbf{y}_i$;
      $\hat{\mathbf{y}}_d \leftarrow \mathrm{del}(\mathbf{y}_c, i)$;
      $\hat{\mathbf{y}} \leftarrow \arg\max_{\mathbf{y}' \in \{\hat{\mathbf{y}}, \hat{\mathbf{y}}_s, \hat{\mathbf{y}}_i, \hat{\mathbf{y}}_d\}} \mathrm{R}(\mathbf{y}' \mid \mathbf{x})$
  until $(\mathrm{R}(\hat{\mathbf{y}} \mid \mathbf{x}) \leq \mathrm{R}(\mathbf{y}_c \mid \mathbf{x}))$ || $(\hat{\mathbf{y}} \geq I)$;
  return $\hat{\mathbf{y}}, \mathrm{R}(\hat{\mathbf{y}} \mid \mathbf{x})$;
end

plexity of the main loop is $O(I^2 \cdot |\Sigma|)$, and usually only a moderate number of iterations (< 10) are needed to converge. Hence, the complete two-step decoding has a complexity in $O(N^2 + I^2 \cdot |\Sigma|)$, where N is the number of translations under consideration in the preliminary sentence-selection decoding.

3. MBRSC Dynamic Programming Decoding

The main drawback of the originally proposed gradient ascent decoding is that it is sensitive to an initial solution which makes it prone to get stuck in local optima. Next, we propose a more sophisticated approach by formalizing MBRSC decoding as a DP problem.

Under the DP framework, decoding is interpreted as a sequence of decisions that incrementally generate new translation hypotheses. Starting with an empty hypothesis, hypotheses of size i are expanded with one more target word $y \in \Sigma$ to create new hypotheses of size $i+1$. This search space can be represented as a directed acyclic graph where the states denote partial hypotheses and the edges are labeled with expansion words.

Among all possible translations, we are interested in that of the higher expected BLEU score. In this case, since two hypotheses sharing the same n-gram counts are indistinguishable, each state of the graph can be represented by a specific bag (namely a specific multiset) $\mathcal{N}$ of n-grams. We define $Q(\mathcal{N}, \mathbf{y}) = q$ where q is the maximum score of a path leading from the initial state to the state $(\mathcal{N})$, and $\mathbf{y}$ is the corresponding translation hypothesis. We also define $\widehat{Q} = \hat{q}$ as the final state of the optimal translation $\hat{\mathbf{y}}$. Finally, the following DP recursion equations allow us to retrieve the path of maximum score in such a search graph:

$$Q(\emptyset, \text{""}) = 0$$

$$Q(\mathcal{N}_e, \mathbf{y}_e) = \max_{\substack{y \in \Sigma \cup \{\$\}: \\ \forall (\mathcal{N}_p, \mathbf{y}_p),\ \mathbf{y}_e = \mathbf{y}_p\, y \\ \mathcal{N}_e = \mathcal{N}_p \cup \Theta(\mathbf{y}_p, y)}} \widetilde{\mathrm{B}}(\mathbf{y}_e, \mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\#_{\mathbf{w}}(\mathbf{y}')])$$

$$\widehat{Q} = \max_{\substack{\forall (\mathcal{N}_p, \mathbf{y}_p) \\ \hat{\mathbf{y}} = \mathbf{y}_p\, \$}} \widetilde{\mathrm{B}}(\hat{\mathbf{y}}, \mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\#_{\mathbf{w}}(\mathbf{y}')])$$

where the end-of-sentence symbol, \$, denotes a complete translation, and function $\Theta(\mathbf{y}_p, y)$ returns the new n-grams generated when expanding hypothesis $\mathbf{y}_p$ with word y. For example, given the hypothesis $\mathbf{y}_p$="we are faced with" and the expansion word y="enormous", the expanded hypothesis $\mathbf{y}_e$="we are faced with enormous" contains four[2] n-grams more than $\mathbf{y}_p$: "enormous", "with enormous", "faced with enormous", and "are faced with enormous".

In the DP recursion equations, all target language words are considered as potential expansion options for every hypothesis. However, not all word sequences form correct natural language sentences. E.g., given the example above, it is clear that word y="enormous" can be a valid expansion option while word y="with" cannot. Thus, we consider $y \in \Sigma \cup \{\$\}$ as a valid expansion word for hypothesis $\mathbf{y}_p$ only if at least one of the new n-grams ($\mathbf{w} \in \Theta(\mathbf{y}_p, y)$) in the resulting expanded hypothesis $\mathbf{y}_e = \mathbf{y}_p\, y$ has an expected count above zero:

$$\Delta(\mathbf{y}_p) = \{y \mid \exists\, \mathbf{w} \in \Theta(\mathbf{y}_p, y)\ \wedge \mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\#_{\mathbf{w}}(\mathbf{y}')] > 0\}$$

Unfortunately, due to the exponential number of states[3], we cannot expect to efficiently implement the recursion equations above. In practice, we use a beam search algorithm [12] with pruning. Specifically, for each size i, we keep only the M best-scoring hypotheses and discard the rest of them. To assure a fair competition between hypotheses, the score of each of them

[2] BLEU considers n-grams up to size four.

[3] The number is exponential in the size of the vocabulary [19].

Algorithm 2: Beam search for MBRSC.

input : $\mathbf{x}$ (source language sentence),
M (pruning parameter),
I (maximum translation length)
output : $\hat{y}, \hat{q}$ (optimal translation and its score)
auxiliary : $\Theta(\mathbf{y}, y)$ (new n-grams after expanding hypothesis $\mathbf{y}$ with word y),
$\Delta(\mathbf{y})$ (expansion words for hypothesis $\mathbf{y}$),
$\overline{\mathrm{R}}(\mathbf{y} \mid \mathbf{x})$ (complete score of $\mathbf{y}$),
$\Pi(i, N)$ (non-pruned states of size i)

```
1  begin
2    Q(∅, "") ← 0;  ŷ ← "";  Q̂ ← 0;
3    for i = 0 to I do
4      forall (N_p, y_p) ∈ Π(i, N) do
5        forall y ∈ Δ(y_p) do
6          y_e ← y_p y;  q_e ← R̄(y_e | x);
7          if y == $ then
8            q̂ ← Q̂;
9            if q_e > q̂ then
10             ŷ ← y_e;  Q̂ ← q_e;
11         else
12           N_e ← N_p ⋃ Θ(y_p, y);
13           q ← Q(N_e, ·);
14           if q_e > q then
15             Q(N_e, y_e) ← q_e;
16   return ŷ, Q̂;
17 end
```

is given by a combination of its score so far, and an estimate of the rest score to complete the translation. Similarly as done in [20], we perform a light decoding process (considering at each step only the single best expansion) to estimate the complete translation that can be obtained from each hypothesis. The score of these complete translations are then used as the complete scores $\overline{\mathrm{R}}(\mathbf{y} \mid \mathbf{x})$ of the partial hypotheses.

Algorithm 2 shows the proposed beam search algorithm with pruning. It takes as input a source sentence $\mathbf{x}$, the number of hypotheses to keep after pruning (M), and the maximum translation length under consideration (I). We use some auxiliary functions: $\Theta(\mathbf{y}, y)$ returns the set of new n-grams generated in the expansion of hypothesis $\mathbf{y}$ with word y, $\Delta(\mathbf{y})$ returns the valid expansion words for $\mathbf{y}$, $\overline{\mathrm{R}}(\mathbf{y} \mid \mathbf{x})$ returns the complete score of $\mathbf{y}$, and $\Pi(i, M)$ denotes the M best states of size i; lower-scoring states are pruned out.

To avoid repeated computations, the first loop in Algorithm 2 performs a breadth-first exploration of the search graph. Additionally, this loop introduces an upper bound to the maximum translation size under consideration, and thus, to the number of iterations of the algorithm. At each iteration, line 4 loops over the non-pruned states that remain from the previous iteration. For each of these predecessor states, line 5 loops over the corresponding expansion words. Given a predecessor state $(\mathcal{N}_p, \mathbf{y}_p)$ and a valid expansion word y, we compute the complete score q_e of the expanded hypothesis $\mathbf{y}_e = \mathbf{y}_p\, y$ (line 6). If the expanded hypothesis is a complete translation ($y == \$$) and it improves the score $\widehat{Q}$ of the current best consensus translation, we then update it (lines 7–10). If not, we first compute the bag of n-grams $\mathcal{N}_e$ of the expanded hypothesis (line 12). Then, if the score q_e of the expanded hypothesis improves the score stored in the corresponding successor state $(\mathcal{N}_e, \cdot)$ (line 14), we update the state.

The proposed beam search algorithm with pruning has a computational complexity in $O(I^2 \cdot M \cdot D)$, where M denotes the pruning parameter that controls the number of predecessor states in line 4, D denotes the maximum number of expansion words in line 5, and I is the maximum translation size in line 3. The extra $O(I)$ factor is given by the score computation in line 6.

4. MBRSC DP Search for Linear BLEU

A potential drawback of decoding Algorithm 2 is that it cannot exploit the full potential of the DP framework. The problem stems in the BLEU based risk proposed in [3]: the n-gram count clippings in its formulation, see Equation (4), make impossible to compute it incrementally. To address this problem, we import the linear approximation to the logarithm of the BLEU scores proposed in [13]:

$$\log(\mathrm{B}(\mathbf{y}, \mathbf{y}')) \approx \lambda_0 |\, \mathbf{y} \,| + \sum_{\mathbf{w} \in \mathcal{W}(\mathbf{y})} \lambda_{\mathbf{w}} \#_{\mathbf{w}}(\mathbf{y}) \delta_{\mathbf{w}}(\mathbf{y}') \qquad (8)$$

where $\mathcal{W}(\mathbf{y})$ is the complete set of n-grams (up to size four) in $\mathbf{y}$, λ_0 and $\lambda_{\mathbf{w}}$ are free parameters, and $\delta_{\mathbf{w}}(\mathbf{y}')$ is an indicator feature whose value is equal to one if n-gram $\mathbf{w}$ is present in $\mathbf{y}'$ and zero otherwise. Given this BLEU approximation, the risk of a candidate translation $\mathbf{y}$ is given by:

$$\mathrm{R}(\mathbf{y} \mid \mathbf{x}) = \lambda_0 |\, \mathbf{y} \,| + \sum_{\mathbf{w} \in \mathcal{W}(\mathbf{y})} \lambda_{\mathbf{w}} \#_{\mathbf{w}}(\mathbf{y}) \mathbb{E}_{P(\mathbf{y}' \mid \mathbf{x})}[\delta_{\mathbf{w}}(\mathbf{y}')] \qquad (9)$$

where $\mathbb{E}_{P(\mathbf{y}' \mid \mathbf{x})}[\delta_{\mathbf{w}}(\mathbf{y}')]$ denotes the expected probability of n-gram $\mathbf{w}$ to be present. Values $\lambda_0, \lambda_{\mathbf{w}}$ can be computed from the n-gram precision statistics of a separate development set [13]. Gradient ascent decoding

can also implement this risk formulation by using Equation (9) as risk function $\mathrm{R}(\mathbf{y} \mid \mathbf{x})$ in Algorithm 1.

Note that the BLEU risk over expected counts in Equation (7) yields a decoding alternative to MBR using BLEU, while the linear BLEU risk in Equation (9) results in a MBR decoding for an alternative to BLEU.

Using the linear BLEU risk in Equation (9), two partial hypotheses that share their last three words are indistinguishable. Hence, the states in the corresponding DP search graph can be represented by a particular three-word history $\boldsymbol{\sigma}$. To distinguish between hypotheses of different size, we also index the search states by the size of the best hypothesis that arrives to the state. We define $Q(i, \boldsymbol{\sigma})$ as the maximum score of a path leading from the initial state to the state $(i, \boldsymbol{\sigma})$, and $\widehat{Q}$ as the score of the optimal translation $\hat{\mathbf{y}}$. Finally, we obtain the following DP recursion equations:

$$Q(0, "") = 0$$

$$Q(i, \boldsymbol{\sigma}_e) = \max_{\substack{y \in \Sigma: \\ q_p = Q(i-1, \boldsymbol{\sigma}_p) \\ \mathbf{y}_e = \boldsymbol{\sigma}_p\, y \\ \boldsymbol{\sigma}_e = \text{tail}(\boldsymbol{\sigma}_p\, y)}} q_p + \lambda_0 + \sum_{\mathbf{w} \in \Theta(\boldsymbol{\sigma}_p, y)} \lambda_{\mathbf{w}} \mathbb{E}_{P(\mathbf{y}' \mid \mathbf{x})}[\delta_{\mathbf{w}}(\mathbf{y}')]$$

$$\widehat{Q} = \max_{\substack{q_p = Q(\cdot, \boldsymbol{\sigma}_p) \\ \boldsymbol{\sigma}_e = \text{tail}(\boldsymbol{\sigma}_p\, \$)}} q_p + \lambda_0 + \sum_{\mathbf{w} \in \Theta(\boldsymbol{\sigma}_p, \$)} \lambda_{\mathbf{w}} \mathbb{E}_{P(\mathbf{y}' \mid \mathbf{x})}[\delta_{\mathbf{w}}(\mathbf{y}')]$$

where $\text{tail}(\boldsymbol{\sigma}\, y)$ returns the last three words of word sequence $\boldsymbol{\sigma}\, y$, and $\Theta(\boldsymbol{\sigma}, y)$ returns the new n-grams generated when extending history $\boldsymbol{\sigma}$ with word y.

Since the number of states is at most cubical with the target vocabulary, these recursive equations can be implemented exactly. Algorithm 3 depicts DP decoding using linear BLEU risk. It takes as input the indicator feature expectations ($\mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\delta_{\mathbf{w}}(\mathbf{y}')]$), the values for the free parameters of linear BLEU ($\lambda_0, \lambda_{\mathbf{w}}$), and the maximum translation length under consideration (I). At each iteration the algorithm loops over the predecessor states (line 4) and the corresponding expansion words (line 5). Given a predecessor state $(i, \boldsymbol{\sigma}_p)$, we compute the score q_e of the expanded hypothesis (line 6), and if q_e improves the score in the corresponding successor state $(i+1, \boldsymbol{\sigma}_e)$ (line 8), we update it and the corresponding backpointer $B(i+1, \boldsymbol{\sigma}_e)$. Finally, backpointer variables allow us to retrieve the highest-scoring consensus translation.

This DP algorithm has a computational complexity in $O(I \cdot |\Sigma|^3 \cdot D)$, where I is the maximum translation length in line 3, $|\Sigma|$ denotes the size of the target vocabulary that controls the number of predecessor states in line 4, and D denotes the maximum number of expansion words in line 5.

Algorithm 3: MBRSC DP search for linear BLEU.

input : $\mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\delta_{\mathbf{w}}(\mathbf{y}')]$ (indicator feature expectations),
$\lambda_0, \lambda_{\mathbf{w}}$ (free parameters of linear BLEU),
I (maximum translation length)
output : $Q(\cdot, \cdot)$ (search graph),
$B(\cdot, \cdot)$ (backpointer variables)
auxiliary : $\text{tail}(\mathbf{y})$ (returns the last three words of $\mathbf{y}$),
$\Theta(\mathbf{y}, y)$ (new n-grams after expanding hypothesis $\mathbf{y}$ with word y),
$\Delta(\mathbf{y})$ (set of expansion words for $\mathbf{y}$)

1 **begin**
2 $\quad Q(\cdot, \cdot) \leftarrow 0$;
3 $\quad$ **for** $i = 0$ **to** I **do**
4 $\quad\quad$ **forall** $\boldsymbol{\sigma}_p \in Q(i, \cdot)$ **do**
5 $\quad\quad\quad$ **forall** $y \in \Delta(\boldsymbol{\sigma}_p)$ **do**
6 $\quad\quad\quad\quad q_e \leftarrow Q(i, \boldsymbol{\sigma}_p) + \lambda_0 + \sum_{\mathbf{w} \in \Theta(\boldsymbol{\sigma}_p, y)} \lambda_{\mathbf{w}} \mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\delta_{\mathbf{w}}(\mathbf{y}')]$;
7 $\quad\quad\quad\quad \boldsymbol{\sigma}_e \leftarrow \text{tail}(\boldsymbol{\sigma}_p\, y)$;
8 $\quad\quad\quad\quad$ **if** $q_e > Q(i+1, \boldsymbol{\sigma}_e)$ **then**
9 $\quad\quad\quad\quad\quad Q(i+1, \boldsymbol{\sigma}_e) \leftarrow q_e$;
10 $\quad\quad\quad\quad\quad B(i+1, \boldsymbol{\sigma}_e) \leftarrow (i, \boldsymbol{\sigma}_p)$;
11 **end**

5. Experiments

5.1. Experimental Setup

We now describe the experimentation carried out to evaluate the proposed decoding algorithms. Experiments were performed on the French–English corpus from the translation task of the 2009 workshop on statistical MT [21]. The corpus contains a development and a test partition with 502 and 2525 sentences respectively. We combined the outputs of the five MT systems that submitted lists of N-best translations. The next table displays the average number of translations for each source sentence, and BLEU scores for the single best translations of each system.

System	#avg_trans_N	BLEU [%]
A	13	24.8
B	9	25.2
C	41	25.8
D	263	25.8
E	126	26.4

Translations were tokenized and lower-cased before combination. We report case-insensitive results to factor out the effect of true-casing from the effect of computing the consensus translation.

The separate development set was used to compute the values of the parameters $(\lambda_0, \lambda_{\mathbf{w}})$ of linear BLEU. The maximum translation length I was always set equal to the length of the longest provided translation; a more sophisticated length model could be devised, but this is a research direction beyond the scope of this article. Except stated otherwise, all experiments were carried out using uniform ensemble weights (α_k in Equation (1)). This approach defines a controlled environment that assures a fair comparison between the different decoding algorithms. For each source sentence, we combined all the translation provided by the five individual systems, on average, about 450 translations. We used these translations to compute the expected n-gram counts $\mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\#_{\mathbf{w}}(\mathbf{y}')]$, and the n-grams expectations $\mathbb{E}_{P(\mathbf{y}'|\mathbf{x})}[\delta_{\mathbf{w}}(\mathbf{y}')]$ for each source sentence.

5.2. Assessment Measures

We present translation quality results in terms of BLEU [4] (see Equation (3)), and TER [22]. TER measures the number of words that must be edited[4] to convert the candidate translation into the reference translation. Since MBRSC is designed to optimize BLEU, we expect improvements in BLEU to be particularly important. TER scores are reported to independently assess BLEU results. We also measure the statistical significance of the results by bootstrap re-sampling [23].

5.3. Preliminary Experiments

We carried out a preliminary series of experiments to study how the number of hypotheses kept after pruning (M) affects the performance of Algorithm 2 in terms of translation quality and decoding time[5]. Figure 1 displays the quality of the generated consensus translations (on the left vertical axis) and the total decoding time (on the right vertical axis) as functions of M. We observed that decoding time increased linearly with M (note that M is log-scaled in Figure 1) while the quality of the consensus translations stayed approximately constant with slight improvements for larger M values.

Given these results, we considered that a value $M = 10$ provided the optimal trade-off between translation quality and decoding time. Thus, this is the value used in the following experiments.

[4] Valid edit operations are: deletion, insertion and substitution of single words, and shift of word sequences

[5] In a PC with an Intel Core© i5-3570K processor (3.40 GHz.).

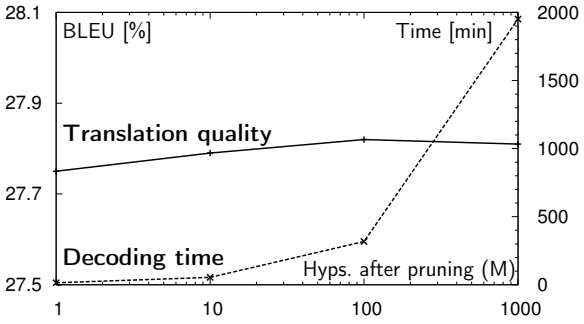

Figure 1: BLEU score (on the left vertical axis) and decoding time (on the right vertical axis) obtained by the beam search using BLEU risk on expected n-gram counts (Algorithm 2) as a function of the number of hypotheses kept after pruning (M).

5.4. Results

Table 1 displays BLEU and TER scores for the consensus translations generated by MBRSC using different decoding algorithms and risk formulations. We also report results for the best and worst single systems.

We first present results for sentence-selection decoding [18]. The risk of each candidate translation was computed by exhaustively calculating its BLEU-based risk with respect to the rest of the provided translations as in Equation (6). Results for both risk functions showed a substantial improvement over the best system: $\sim +0.9$ BLEU. Then, we used these sentence-selection consensus translations as initial solutions for the gradient ascent decoding proposed in [3] (Algorithm 1). Results for BLEU risk on expected n-gram counts slightly improved results for sentence-selection decoding: $+0.3$ BLEU and -0.1 TER. In contrast, results for linear BLEU risk showed an important degradation in performance: -0.9 BLEU and $+3.4$ TER. Finally, we generated consensus translations using BLEU risk over expected n-gram counts (Algorithm 2), and linear BLEU risk (Algorithm 3). Results for BLEU risk on expected counts slightly improved the results of the gradient ascent decoding: $+0.1$ BLEU and -0.3 TER. Regarding linear BLEU risk, it again exhibited the same poor performance observed for gradient ascent decoding.

Despite being scarce, the difference in translation quality between the proposed decoding algorithms and the original gradient ascent algorithm were statistically significant: 85% confidence for BLEU risk over expected counts, and 99% confidence for linear BLEU risk. Moreover, when we measured the risk scores of the generated consensus translations, we found that for

System setup		BLEU[%]	TER[%]
worst single system		24.8	60.4
best single system		26.4	56.0
Sentence-selection [18]	EC	27.4	55.5
	LB	27.2	56.2
Gradient ascent (Algorithm 1)	EC	27.7	55.4
	LB	26.3	59.6
BS (Algorithm 2)	EC	**27.8**	**55.1**
DP (Algorithm 3)	LB	26.8	57.8

Table 1: Quality of the consensus translations generated by different MBRSC setups. BS stands for beam search, EC for BLEU risk over expected counts (Equation (7)), and LB for linear BLEU risk (Equation (9)).

53% of the sentences DP-based search found a better-scoring output than gradient ascent decoding (47%).

We performed additional experiments where the values of the ensemble weights (α_k in Equation (1)) were trained to optimize BLEU in the development corpus. Results were similar to those in Table 1. For instance, beam search with risk over expected counts scored 28.1 BLEU while gradient ascent scored 27.8 BLEU. However, now DP-based search generated better-scoring consensus translation for 93% of the sentences. The scarce improvement with respect to the use of uniform values can be explained by the similar quality of the systems being combined, see §5.1.

We also compared DP search and gradient ascent search in terms of decoding time. We estimate decoding time by the number of times each algorithm calls the risk-computation function $\mathrm{R}(\mathbf{y} \mid \mathbf{x})$ during the generation of consensus translations for the whole corpus. We report this count instead of the actual decoding time to filter out the potential effects of the particular implementation of each algorithm. We observed that gradient ascent made ~ 23 millions calls to the risk function, while DP decoding made ~ 15 million calls including those involved in the estimation of the rest score. For instance, total decoding time for DP using BLEU risk over expected counts was about 55 minutes (~ 1.3 seconds per sentence).

Finally, we conclude that the proposed DP decoding is both more effective and efficient than the original gradient ascent decoding proposed in [3].

Regarding the low performance of linear BLEU risk, we consider that it was due to the the lack of n-gram count clippings in the linear BLEU risk formulation. Consensus translations obtained with linear BLEU risk tend to contain repeated instances of highly-probable n-grams which resulted in longer consensus translations (27.8 words on average) than the ones generated using BLEU risk over expected counts (26.4 words), and also longer than the average length (26.0 words) of the reference translations. Table 2 shows various examples of these erroneous consensus translations generated by Algorithm 3. Given the adequate performance of linear BLEU risk in our sentence-selection experiments and in previous works [13, 7], we conclude that linear BLEU is an effective loss function to be used in sentence-selection methods, but due to the lack of n-gram count clippings, it fails at scoring the new translations explored though decoding by subsequence-combination algorithms. The inclusion of more features, such as a language model, in the formulation of linear BLEU risk may mitigate this effect.

Alg. 2: we have made great progress . **Alg. 3:** we have made great progress . *we have made*
Alg. 2: it seems to be clear that it is better to buy only a phone . **Alg. 3:** *to be clear that* it seems to be clear that it is better to buy only a phone .
Alg. 2: i am curious to know if i could see here . **Alg. 3:** *am curious to know if* i am curious to know if i could see here .

Table 2: Consensus translations generated using BLEU risk over expected counts (Alg. 2), and using linear BLEU risk (Alg. 3). The use of linear BLEU risk in Algorithm 3 results in ill-formed consensus translations.

6. Summary

We have investigated different approaches to improve the MBRSC method described in [3]. First, we have proposed a new DP decoding algorithm to obtain the optimal consensus translation according to the original BLEU-based risk formulation. Then, we have studied a more efficient risk formulation based on the linear BLEU approximation proposed in [13]. Empirical results showed that the proposed DP decoding was able to obtain better-scoring higher-quality hypotheses than original gradient ascent search proposed in [3], and to do that with less temporal complexity. We have also shown that linear BLEU is not an adequate risk function for subsequence-combination methods due to the lack of n-gram count clippings in its formulation.

7. Acknowledgments

Work supported by the European Union Seventh Framework Program (FP7/2007-2013) under the CasMaCat project (grants agreement n^{o} 287576), by the Generalitat Valenciana under grant ALMPR (Prometeo/2009/014), and by the Spanish government under grant TIN2012-31723.

8. References

[1] O. Bojar, C. Buck, C. Callison-Burch, C. Federmann, B. Haddow, P. Koehn, C. Monz, M. Post, R. Soricut, and L. Specia, "Findings of the 2013 Workshop on Statistical Machine Translation," in *Proc. of the* 8^{th} *Workshop on SMT*, 2013, pp. 1–44.

[2] T. G. Dietterich, "Ensemble methods in machine learning," in *Proc. of the* 1^{st} *Workshop on Multiple Classifier Systems*, 2000, pp. 1–15.

[3] J. González-Rubio, A. Juan, and F. Casacuberta, "Minimum bayes-risk system combination," in *Proc. of the Association for Computational Linguistics*, 2011, pp. 1268–1277.

[4] K. Papineni, S. Roukos, T. Ward, and W. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proc. of the Association for Computational Linguistics*, 2002, pp. 311–318.

[5] T. Nomoto, "Multi-engine machine translation with voted language model," in *Proc. of the Association for Computational Linguistics*, 2004, pp. 494–501.

[6] J. DeNero, S. Kumar, C. Chelba, and F. Och, "Model combination for machine translation," in *Proc. of the North American chapter of the Association for Computational Linguistics*, 2010, pp. 975–983.

[7] N. Duan, M. Li, D. Zhang, and M. Zhou, "Mixture model-based minimum bayes risk decoding using multiple machine translation systems," in *Proc. of the conference Computational Linguistics*, 2010, pp. 313–321.

[8] J. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER)," in *Proc. of the IEEE Workshop on Automatic Speech Recognition and Understanding*, 1997, pp. 347–354.

[9] S. Bangalore, "Computing consensus translation from multiple machine translation systems," in *Proc. of the IEEE workshop on Automatic Speech Recognition and Understanding*, 2001, pp. 351–354.

[10] A. Rosti, N. F. Ayan, B. Xiang, S. Matsoukas, R. Schwartz, and B. Dorr, "Combining outputs from multiple machine translation systems," in *Proc. of the North American Chapter of the Association for Computational Linguistics*, 2007, pp. 228–235.

[11] R. Bellman, *Dynamic Programming*. Princeton, NJ, USA: Princeton University Press, 1957.

[12] F. Jelinek, *Statistical methods for speech recognition*. Cambridge, MA, USA: MIT Press, 1997.

[13] R. Tromble, S. Kumar, F. Och, and W. Macherey, "Lattice minimum bayes-risk decoding for statistical machine translation," in *Proc. of the Empirical Methods in Natural Language Processing conference*, 2008, pp. 620–629.

[14] J. Kittler, M. Hatef, R. P. Duin, and J. Matas, "On combining classifiers," *IEEE Transact. on Pattern Analisis and Machine Intelligence*, vol. 20, pp. 226–239, 1998.

[15] R. Duda, P. Hart, and D. Stork, *Pattern classification*. Wiley, 2001.

[16] F. Och, "Minimum error rate training in statistical machine translation," in *Proc. of the Association for Computational Linguistics*, 2003, pp. 160–167.

[17] J. DeNero, D. Chiang, and K. Knight, "Fast consensus decoding over translation forests," in *Proc. of the Association for Computational Linguistics*, 2009, pp. 567–575.

[18] S. Kumar and W. Byrne, "Minimum bayes-risk decoding for statistical machine translation," in *Proc. of the North American Chapter of the Association for Computational Linguistics*, 2004, pp. 169–176.

[19] R. Stanley, *Enumerative combinatorics*. Cambridge University Press, 2002.

[20] X. He and K. Toutanova, "Joint optimization for machine translation system combination," in *Proc. of the Empirical Methods in Natural Language Processing conference*, 2009, pp. 1202–1211.

[21] C. Callison-Burch, P. Koehn, C. Monz, and J. Schroeder, "Findings of the 2009 Workshop on Statistical Machine Translation," in *Proc. of the* 4^{th} *Workshop on SMT*, 2009, pp. 1–28.

[22] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A study of translation edit rate with targeted human annotation," in *Proc. of the Association for MT in the Americas*, 2006, pp. 223–231.

[23] Y. Zhang and S. Vogel, "Measuring confidence intervals for the machine translation evaluation metrics," in *Proc. of the* 10^{t} *conference on Theoretical and Methodological Issues in MT*, 2004.