Mutual Information and Diverse Decoding Improve Neural Machine Translation

Jiwei Li and Dan Jurafsky
Computer Science Department
Stanford University, Stanford, CA, 94305, USA
jiweil,jurafsky@stanford.edu

Abstract

Sequence-to-sequence neural translation models learn semantic and syntactic relations between sentence pairs by optimizing the likelihood of the target given the source, i.e., $p(y|x)$, an objective that ignores other potentially useful sources of information. We introduce an alternative objective function for neural MT that maximizes the mutual information between the source and target sentences, modeling the bi-directional dependency of sources and targets. We implement the model with a simple re-ranking method, and also introduce a decoding algorithm that increases diversity in the N-best list produced by the first pass. Applied to the WMT German/English and French/English tasks, both mechanisms offer a consistent performance boost on both standard LSTM and attention-based neural MT architectures. The result is the best published performance for a single (non-ensemble) neural MT system, as well as the potential application of our diverse decoding algorithm to other NLP re-ranking tasks.

1 Introduction

Sequence-to-sequence models for machine translation (SEQ2SEQ) (Sutskever et al., 2014; 3; Cho et al., 2014; Kalchbrenner and Blunsom, 2013; Li et al., 2015b) are of growing interest for their capacity to learn semantic and syntactic relations between sequence pairs, capturing contextual dependencies in a more continuous way than phrase-based SMT approaches. SEQ2SEQ models require minimal domain knowledge, can be trained end-to-end, have a much smaller memory footprint than the large phrase tables needed for phrase-based SMT, and achieve state-of-the-art performance in large-scale tasks like English to French (Luong et al., 2015b) and English to German (Luong et al., 2015a; Jean et al., 2014) translation.

SEQ2SEQ models are implemented as an encoder-decoder network, in which a source sequence input x is mapped (encoded) to a continuous vector representation from which a target output y will be generated (decoded). The framework is optimized through maximizing the log-likelihood of observing the paired output y given x:

$$\text{Loss} = -\log p(y|x) \tag{1}$$

While standard SEQ2SEQ models thus capture the unidirectional dependency from source to target, i.e., $p(y|x)$, they ignore $p(x|y)$, the dependency from the target to the source, which has long been an important feature in phrase-based translation (Och and Ney, 2002; Shen et al., 2010). Phrase based systems that combine $p(x|y)$, $p(y|x)$ and other features like sentence length yield significant performance boost.

We propose to incorporate this bi-directional dependency and model the maximum mutual information (MMI) between source and target into SEQ2SEQ models. As Li et al. (2015a) recently showed in the context of conversational response generation, the MMI based objective function is equivalent to linearly combining $p(x|y)$ and $p(y|x)$. With a tuning weight λ, such a loss function can be written as :

$$\begin{aligned}\hat{y} &= \arg\max_y \log \frac{p(x,y)}{p(x)p(y)^{\lambda}} \\ &= \arg\max_y (1-\lambda)\log p(y|x) + \lambda p(x|y)\end{aligned} \tag{2}$$

But as also discussed in Li et al. (2015a), direct decoding from (2) is infeasible because computing $p(x|y)$ can't be done until the target has been computed[1].

To avoid this enormous search space, we propose to use a reranking approach to approximate the mutual information between source and target in neural machine translation models. We separately trained two SEQ2SEQ models, one for $p(y|x)$ and one for $p(x|y)$. The $p(y|x)$ model is used to generate N-best lists from the source sentence x. The lists are followed by a reranking process using the second term of the objective function, $p(x|y)$.

Because reranking approaches are dependent on having a diverse N-best list to rerank, we also propose a diversity-promoting decoding model tailored to neural MT systems. We tested the mutual information objective function and the diversity-promoting decoding model on English→French, English→German and German→English translation tasks, using both standard LSTM settings and the more advanced Attention-model based settings that have recently shown to result in higher performance.

As we will show, each of our two models yields a consistent performance boost on neural MT, and the combined system achieves what is to our knowledge the best published BLEU score from a single (non-ensemble) neural MT system.

The next section presents related work, followed by a background section 3 introducing LSTM/Attention machine translation models. Our proposed model will be described in detail in Sections 4, with datasets and experimental results in Section 6 followed by conclusions.

2 Related Work

This paper draws on three prior lines of research: SEQ2SEQ models, modeling mutual information, and promoting translation diversity.

[1]As demonstrated in (Li et al., 2015a)

$$\log \frac{p(x,y)}{p(x)p(y)^{\lambda}} = \log p(y|x) - \lambda p(y) \tag{3}$$

Equ. 2 can be immediately achieved by applying bayesian rules

$$\log p(y) = \log p(y|x) + \log p(x) - \log p(x|y)$$

SEQ2SEQ Models SEQ2SEQ models map source sequences to vector space representations, from which a target sequence is then generated. They yield good performance in a variety of NLP generation tasks including conversational response generation (Vinyals and Le, 2015; Serban et al., 2015a; Li et al., 2015a), and parsing (Vinyals et al., 2015).

A neural machine translation system uses distributed representations to model the conditional probability of targets given sources, using two components, an encoder and a decoder. Kalchbrenner and Blunsom (2013) used an encoding model akin to convolutional networks for encoding and standard hidden unit recurrent nets for decoding. Similar convolutional networks are used in (Meng et al., 2015) for encoding. Sutskever et al. (2014; Luong et al. (2015a) employed a stacking LSTM model for both encoding and decoding. 3; Jean et al. (2014) adopted bi-directional recurrent nets for the encoder.

Maximum Mutual Information Maximum Mutual Information (MMI) was introduced in speech recognition (Bahl et al., 1986) as a way of measuring the mutual dependence between inputs (acoustic feature vectors) and outputs (words) and improving discriminative training (Woodland and Povey, 2002). Li et al. (2015a) showed that MMI could solve an important problem in SEQ2SEQ conversational response generation. Prior SEQ2SEQ models tended to generate highly generic, dull responses (e.g., *I don't know*) regardless of the inputs (Sordoni et al., 2015; Vinyals and Le, 2015; Serban et al., 2015b). Li et al. (2015a) shows that modeling the mutual dependency between messages and response promotes the diversity of response outputs.

Our goal, distinct from these previous uses of MMI, is to see whether the mutual information objective improves translation by bidirectionally modeling source-target dependencies. In that sense our work is designed to incorporate into SEQ2SEQ models features that have proved useful in phrase-based MT, like the reverse translation probability or sentence length (Och and Ney, 2002; Shen et al., 2010).

Generating Diverse Translations Various algorithms have been proposed for generated diverse translations in phrase-based MT, including compact representations like lattices and hypergraphs (Macherey et al., 2008; Tromble et al., 2008; Kumar

and Byrne, 2004), "traits" like translation length (Devlin and Matsoukas, 2012), bagging/boosting (Xiao et al., 2013), or multiple systems (Cer et al., 2013). Gimpel et al. (2013; Batra et al. (2012), produce diverse N-best lists by adding a dissimilarity function based on N-gram overlaps, distancing the current translation from already-generated ones by choosing translations that are highly-scoring but distinct from previous ones. While we draw on these intuitions, these existing diversity promoting algorithms are tailored to phrase-based translation frameworks and not easily transplanted to neural MT decoding. For example the (Gimpel et al., 2013) approach can only be evaluated after translations are constructed. We will propose an on-line algorithm that can promote diversity during beam search.

3 Background: LSTM & Attention Models

Neural machine translation models map source $x = \{x_1, x_2, ...x_{N_x}\}$ to a continuous vector representation, from which target output $y = \{y_1, y_2, ..., y_{N_y}\}$ is to be generated.

3.1 LSTM Models

A long-short term memory model (Hochreiter and Schmidhuber, 1997) associates each time step with an input gate, a memory gate and an output gate, denoted respectively as i_t, f_t and o_t. Let e_t denote the vector for the current word w_t, h_t the vector computed by the LSTM model at time t by combining e_t and h_{t-1}., c_t the cell state vector at time t, and σ the sigmoid function. The vector representation h_t for each time step t is given by:

$$i_t = \sigma(W_i \cdot [h_{t-1}, e_t]) \quad (4)$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, e_t]) \quad (5)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, e_t]) \quad (6)$$

$$l_t = \tanh(W_l \cdot [h_{t-1}, e_t]) \quad (7)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot l_t \quad (8)$$

$$h_t^s = o_t \cdot \tanh(c_t) \quad (9)$$

where W_i, W_f, W_o, $W_l \in \mathbb{R}^{K \times 2K}$. The LSTM defines a distribution over outputs T and sequentially predicts tokens using a softmax function:

$$p(y|x) = \prod_{t=1}^{n_T} \frac{\exp(f(h_{t-1}, e_{y_t}))}{\sum_{w'} \exp(f(h_{t-1}, e_{w'}))}$$

where $f(h_{t-1}, e_{y_t})$ denotes the activation function between h_{t-1} and e_{w_t}, where h_{t-1} is the representation output from the LSTM at time $t-1$. Each sentence concludes with a special end-of-sentence symbol *EOS*. Commonly, the input and output each use different LSTMs with separate sets of compositional parameters to capture different compositional patterns. During decoding, the algorithm terminates when an *EOS* token is predicted.

3.2 Attention Models

Attention models adopt a look-back strategy that links the current decoding stage with input time steps to represent which portions of the input are most responsible for the current decoding state (Xu et al., 2015; Luong et al., 2015b; 3).

Let $H = \{\hat{h}_1, \hat{h}_2, ..., \hat{h}_{N_x}\}$ be the collection of hidden vectors outputted from LSTMs during encoding. Each element in H contains information about the input sequences, focusing on the parts surrounding each specific token. Let h_{t-1} be the LSTM outputs for decoding at time $t-1$. Attention models link the current-step decoding information, i.e., h_t with each of the representations at decoding step $\hat{h}_{t'}$ using a weight variable a_t. a_t can be constructed from different scoring functions such as the *dot product* between the two vectors, i.e., $h_{t-1}^T \cdot \hat{h}_t$, a *general* model akin to tensor operation i.e., $h_{t-1}^T \cdot W \cdot \hat{h}_t$, and the *concatenation* model by concatenating the two vectors i.e., $U^T \cdot \tanh(W \cdot [h_{t-1}, \hat{h}_t])$. The behavior of different attention scoring functions have been extensively studied in (Luong et al., 2015a). For all experiments in this paper, we adopt the *general* strategy where the relevance score between the current step of the decoding representation and the encoding representation is given by:

$$v_{t'} = h_{t-1}^T \cdot W \cdot \hat{h}_t$$
$$a_i = \frac{\exp(v_{t^*})}{\sum_{t*} \exp(v_{t^*})} \quad (10)$$

The attention vector is created by averaging weights over all input time-steps:

$$m_t = \sum_{t' \in [1, N_S]} a_i \hat{h}_{t'} \quad (11)$$

Attention models predict subsequent tokens based on the combination of the last step outputted LSTM

vectors h_{t-1} and attention vectors m_t:

$$\begin{aligned}\vec{h}_{t-1} &= \tanh(W_c \cdot [h_{t-1}, m_t]) \\ p(y_t|y_<, x) &= \text{softmax}(W_s \cdot \vec{h}_{t-1})\end{aligned} \tag{12}$$

where $W_c \in \mathbb{R}^{K\times 2K}$, $W_s \in \mathbb{R}^{V\times K}$ with V denoting vocabulary size. (Luong et al., 2015a) reported a significant performance boost by integrating $\vec{h}_{t-1}$ into the next step LSTM hidden state computation (referred to as the *input-feeding* model), making LSTM compositions in decoding as follows:

$$\begin{aligned}i_t &= \sigma(W_i \cdot [h_{t-1}, e_t, \vec{h}_{t-1}]) \\ f_t &= \sigma(W_f \cdot [h_{t-1}, e_t, \vec{h}_{t-1}]) \\ o_t &= \sigma(W_o \cdot [h_{t-1}, e_t, \vec{h}_{t-1}]) \\ l_t &= \tanh(W_l \cdot [h_{t-1}, e_t, \vec{h}_{t-1}])\end{aligned} \tag{13}$$

where $W_i, W_f, W_o, W_l \in \mathbb{R}^{K\times 3K}$. For the attention models implemented in this work, we adopt the *input-feeding* strategy.

3.3 Unknown Word Replacements

One of the major issues in neural MT models is the computational complexity of the softmax function for target word prediction, which requires summing over all tokens in the vocabulary. Neural models tend to keep a shortlist of 50,00-80,000 most frequent words and use an unknown (UNK) token to represent all infrequent tokens, which significantly impairs BLEU scores. Recent work has proposed to deal with this issue: (Luong et al., 2015b) adopt a post-processing strategy based on aligner from IBM models, while (Jean et al., 2014) approximates softmax functions by selecting a small subset of target vocabulary.

In this paper, we use a strategy similar to that of Jean et al. (2014), thus avoiding the reliance on external IBM model word aligner. From the attention models, we obtain word alignments from the training dataset, from which a bilingual dictionary is extracted. At test time, we first generate target sequences. Once a translation is generated, we link the generated UNK tokens back to positions in the source inputs, and replace each UNK token with the translation word of its correspondent source token using the pre-constructed dictionary.

As the unknown word replacement mechanism relies on automatic word alignment extraction which is not explicitly modeled in vanilla SEQ2SEQ models, it can not be immediately applied to vanilla SEQ2SEQ models. However, since unknown word replacement can be viewed as a post-processing technique, we can apply a pre-trained attention-model to any given translation. For SEQ2SEQ models, we first generate translations and replace UNK tokens within the translations using the pre-trained attention models to post-process the translations.

4 Mutual Information via Reranking

As discussed in Li et al. (2015a), direct decoding from (2) is infeasible since the second part, $p(x|y)$, requires completely generating the target before it can be computed. We therefore use an approximation approach:

1. Train $p(y|x)$ and $p(x|y)$ separately using vanilla SEQ2SEQ models or Attention models.
2. Generate N-best lists from $p(y|x)$.
3. Rerank the N-best list by linearly adding $p(x|y)$.

4.1 Standard Beam Search for N-best lists

N-best lists are generated using a beam search decoder with beam size set to 200 from $p(y|x)$ models. As illustrated in Figure 1, at time step $t-1$ in decoding, we keep record of K hypotheses based on score $S(Y_{t-1}|x) = \log p(y_1, y_2, ..., y_{t-1}|x)$. As we move on to time step t, we expand each of the K hypotheses (denoted as $Y^k_{t-1} = \{y^k_1, y^k_2, ..., y^k_{t-1}\}$, $k \in [1, K]$), by selecting top K of the translations, denoted as $y^{k,k'}_t$, $k' \in [1, K]$, leading to the construction of $K \times K$ new hypotheses:

$$[Y^k_{t-1}, y^{k,k'}_t], k \in [1, K], k' \in [1, K]$$

The score for each of the $K \times K$ hypotheses is computed as follows:

$$S(Y^k_{t-1}, y^{k,k'}_t|x) = S(Y^k_{t-1}|x) + \log p(y^{k,k'}_t|x, Y^k_{t-1}) \tag{14}$$

In a standard beam search model, the top K hypotheses are selected (from the $K \times K$ hypotheses computed in the last step) based on the score $S(Y^k_{t-1}, y^{k,k'}_t|x)$. The remaining hypotheses are ignored as we proceed to the next time step.

We set a maximum length to 1.5 times the length of sources. As decoding proceeds, sentences that are generated with a predicted *EOS* token are stored for later reranking.

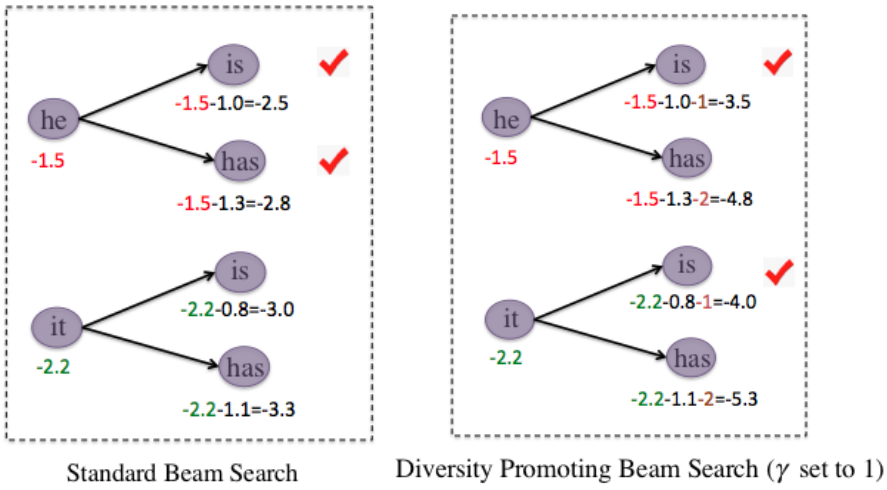

Figure 1: Illustration of Standard Beam Search and proposed diversity promoting Beam Search.

4.2 Generating a Diverse N-best List

Unfortunately, the N-best lists outputted from standard beam search are a poor surrogate for the entire search space (Finkel et al., 2006; Huang, 2008). The beam search algorithm can only keep a small proportion of candidates in the search space and most of the generated translations in N-best list are similar, differing only by punctuation or minor morphological variations, with most of the words overlapping. Because this lack of diversity in the N-best list will significantly decrease the impact of our reranking process, it is important to find a way to generate a more diverse N-best list.

We propose to change the way $S(Y_{t-1}^k, y_t^{k,k'}|x)$ is computed in an attempt to promote diversity, as shown in Figure 1. For each of the hypotheses Y_{t-1}^k (*he* and *it*), we generate the top K translations, $y_t^{k,k'}$, $k' \in [1, K]$ as in the standard beam search model. Next we rank the K translated tokens generated from the same parental hypothesis based on $p(y_t^{k,k'}|x, Y_{t-1}^k)$ in descending order: *he is* ranks the first among *he is* and *he has*, and *he has* ranks second; similarly for *it is* and *it has*.

Next we rewrite the score for $[Y_{t-1}^k, y_t^{k,k'}]$ by adding an additional part $\gamma k'$, where k' denotes the ranking of the current hypothesis among its siblings, which is first for *he is* and *it is*, second for *he has* and *it has*.

$$\hat{S}(Y_{t-1}^k, y_t^{k,k'}|x) = S(Y_{t-1}^k, y_t^{k,k'}|x) - \gamma k' \quad (15)$$

The top K hypothesis are selected based on $\hat{S}(Y_{t-1}^k, y_t^{k,k'}|x)$ as we move on to the next time step. By adding the additional term $\gamma k'$, the model punishes bottom ranked hypotheses among siblings (hypotheses descended from the same parent). When we compare newly generated hypotheses descended from different ancestors, the model gives more credit to top hypotheses from each of different ancestors. For instance, even though the original score for *it is* is lower than *he is*, the model favors the former as the latter is more severely punished by the intra-sibling ranking part $\gamma k'$. The model thus generally favors choosing hypotheses from diverse parents, leading to a more diverse N-best list.

The proposed model is straightforwardly implemented with minor adjustment to the standard beam search model[2].

We employ the diversity evaluation metrics in (Li et al., 2015a) to evaluate the degree of diversity of the N-best lists: calculating the average number of distinct unigrams *distinct-1* and bigrams *distinct-2* in the N-best list given each source sentence, scaled by the total number of tokens. By employing the diversity promoting model with γ tuned from the development set based on BLEU score, the value of *distinct-1* increases from 0.54% to 0.95%, and *distinct-2* in-

[2]Decoding for neural based MT model using large batch-size can be expensive resulted from softmax word prediction function. The proposed model is tailored to decoding in chunk using GPU, significantly speed up decoding process than other diversity fostering models tailored to phrase based MT systems.

creases from 1.55% to 2.84% for English-German translation. Similar phenomenon are observed from English-French translation tasks and details are omitted for brevity.

4.3 Reranking

The generated N-best list is then reranked by linearly combining $p(y|x)$ with $p(x|y)$. The score of the source given each generated translation can be immediately computed from the previously trained $p(x|y)$. We also consider an additional term that takes into account the length of targets (denotes as L_T) in decoding. We thus linearly combine the three parts, making the final ranking score for a given target candidate y as follows:

$$Score(y) = (1-\lambda)p(y|x) + \lambda p(x|y) + \eta L_T \quad (16)$$

We applied grid search to achieve the combination value for η, λ and γ. Hyperparameters are tuned via BLEU score (Papineni et al., 2002) on the development set.

5 Experiments

Our models are trained on the WMT'14 training dataset containing 4.5 million pairs for English-German and German-English translation, and 12 million pairs for English-French translation. For English-German translation, we limit our vocabularies to the top 50K most frequent words for both languages. For English-French translation, we keep the top 200K most frequent words for the source language and 80K for the target language. Words that are not in the vocabulary list are noted as the universal unknown token.

For the English-German and English-German translation, we use newstest2013 (3000 sentence pairs) as the development set and translation performances are reported in BLEU (Papineni et al., 2002) on newstest2014 (2737) sentences. For English-French translation, we concatenate news-test-2012 and news-test-2014 to make a development set (6,003 pairs in total) and evaluate the models on news-test-2014 with 3,003 pairs[3].

[3]As in (Luong et al., 2015a). All texts are tokenized with tokenizer.perl and BLEU scores are computed with multi-bleu.perl

5.1 Training Details for $p(x|y)$ and $p(y|x)$

We trained neural models on Standard SEQ2SEQ Models and Attention Models. We trained $p(y|x)$ following the standard training protocols described in (Sutskever et al., 2014). $p(x|y)$ is trained identically but with sources and targets swapped.

We adopt a deep structure with four LSTM layers for encoding and four LSTM layers for decoding, each of which consists of a different set of parameters. We followed the detailed protocols from Luong et al. (2015a): each LSTM layer consists of 1,000 hidden neurons, and the dimensionality of word embeddings is set to 1,000. Other training details include: LSTM parameters and word embeddings are initialized from a uniform distribution between [-0.1,0.1]; For English-German translation, we run 12 epochs in total. After 8 epochs, we start halving the learning rate after each epoch; for English-French translation, the total number of epochs is set to 8, and we start halving the learning rate after 5 iterations. Batch size is set to 128; gradient clipping is adopted by scaling gradients when the norm exceeded a threshold of 5. Inputs are reversed.

Our implementation on a single GPU[4] processes approximately 800-1200 tokens per second. Training for the English-German dataset (4.5 million pairs) takes roughly 12-15 days. For the French-English dataset, comprised of 12 million pairs, training takes roughly 4-6 weeks.

5.2 English-German Results

Results for different models on WMT2014 are shown in Figure 1. As can be seen, the mutual information reranking models result in improved performance: +1.4 and +1.3 for standard SEQ2SEQ models without and with unknown word replacement, +0.9 for attention models. We see the benefit from our diverse N-best list by comparing *mutual+diversity* models with *diversity* models. On top of the improvements from standard beam search due to mutual information reranking, the *diversity* models introduce additional gains of +0.7, +0.9 and +0.6, leading the total gains roughly up to +2.0. The unknown token replacement technique yields significant gains, in line with observations from Jean et al. (2014; Luong et al. (2015a).

We compare our English-German system with var-

[4]Tesla K40m, 1 Kepler GK110B, 2880 Cuda cores.

Language Pairs	Model	BLEU scores
English→German	Standard	13.6
English→German	Standard (*mutual*)	15.0 (+1.4)
English→German	Standard (*mutual*+*diversity*)	15.7 (+2.1)
English→German	Standard+*UnkRep*	14.6
English→German	Standard (*mutual*)+*UnkRep*	15.7 (+1.1)
English→German	Standard (*mutual*+*diversity*)+*UnkRep*	16.6 (+2.0)
English→German	Attention+*UnkRep*	20.4
English→German	Attention (*mutual*)+*UnkRep*	21.5 (+1.1)
English→German	Attention (*mutual*+*diversity*)+*UnkRep*	22.1 (+1.7)
English→German	Buck et al., 2014	20.7
English→German	Jean et al., 2015 (without *ensemble*)	19.4
English→German	Jean et al., 2015 (with *ensemble*)	21.6
English→German	Thang et al., 2015 (with *UnkRep*, without *ensemble*)	20.9
English→German	Thang et al., 2015 (with *UnkRep*, with *ensemble*)	23.0

Table 1: BLEU scores from different models for on WMT14 English-German results. *UnkRep* denotes applying unknown word replacement strategy. *diversity* indicates diversity-promoting model for decoding being adopted. Baselines performances are reprinted from Buck et al. (2014; Luong et al. (2015a; Jean et al. (2014).

Language Pairs	Model	BLEU scores
German→English	Standard+*UnkRep*	15.2
German→English	Standard (*mutual*+*UnkRep*)	16.5 (+1.3)
German→English	Standard (*mutual*+*diversity*+*UnkRep*)	18.0 (+1.8)
German→English	Attention+*UnkRep*	19.6
German→English	Attention (*mutual*)+*UnkRep*	20.8 (+1.2)
German→English	Attention (*mutual*+*diversity*+*UnkRep*)	21.2 (+1.6)

Table 2: BLEU scores from different models for on WMT'14 German-English results.

ious others: (1) The winning system in WMT2014 (Buck et al., 2014) where language models were trained on a huge monolingual dataset. (2) The end-to-end neural MT system from Jean et al. (2014) using a large vocabulary size. (3) Models from Luong et al. (2015a) that combines different attention models. For the models described in (Jean et al., 2014) and (Luong et al., 2015a), we reprint their results from both the single model setting and the *ensemble* setting, which a set of (usually 8) neural models that differ in random initializations and the order of minibatches are trained, the combination of which jointly contributes in the decoding process. The *ensemble* procedure is known to result in improved performance (Luong et al., 2015a; Jean et al., 2014; Sutskever et al., 2014).

Note that the reported results from the standard SEQ2SEQ models and attention models in Table 1 (those without considering mutual information) are from models identical in structure to the corresponding models described in (Luong et al., 2015a), and achieve similar performances (13.6 vs 14.0 for standard SEQ2SEQ models and 20.4 vs 20.7 for attention models).

To the best of our knowledge, our proposed mutual information model achieves the best published performance from a single neural model: +2.1 when compared with Jean et al. (2014) and +1.2 when compared with Luong et al. (2015a). (Due to time and computational constraints, we did not implement an ensemble mechanism, making our results incomparable to the ensemble mechanisms in these papers.)

Language Pairs	Model	BLEU scores
French→English	Standard	29.4
French→English	Standard (*mutual*)	31.0 (+1.6)
French→English	Standard (*mutual*+*diversity*)	32.2 (+2.8)
French→English	Standard+*UnkRep*	31.2
French→English	Standard (*mutual*)+*UnkRep*	32.7 (+1.5)
French→English	Standard (*mutual*+*diversity*)+*UnkRep*	33.7 (+2.5)
French→English	Attention+*UnkRep*	33.6
French→English	Attention (*mutual*)+*UnkRep*	34.8 (+1.2)
French→English	Attention (*mutual*+*diversity*)+*UnkRep*	35.8 (+2.2)
French→English	LSTM (Google) (without ensemble)	30.6
French→English	LSTM (Google) (with ensemble)	33.0
French→English	Luong *et al.* (2015), *UnkRep* (without ensemble)	32.7
French→English	Luong *et al.* (2015), *UnkRep* (with ensemble)	37.5

Table 3: BLEU scores from different models for on WMT'14 English-French results. Google is the LSTM-based model proposed in Sutskever *et al.* (2014). Luong *et al.* (2015) is the extension of Google models with unknown token replacements.

5.3 German-English Results

We carried out similar set of experiments for the WMT'15 German to English translation task. Mutual information reranking again results in gains in BLEU, as demonstrated in Table 2. The mutual information model gives +1.3 and +0.9 performance gains, on top of which we obtain another boost of up to +0.5-0.7 form the *diversity* decoding mechanism.

5.4 French-English Results

Results from the WMT'14 French-English datasets are shown in Table 3, along with results reprinted from Sutskever et al. (2014; Luong et al. (2015b). We again observe that applying mutual information yields better performance than the corresponding standard neural MT models.

Relative to the English-German dataset, the English-French translation task shows a larger gap between our new model and vanilla models where mutual information is not considered; our models respectively yield up to +2.8, +2.5, +2.2 boost in BLEU compared to standard neural models without and with unknown word replacement, and Attention models.

6 Discussion

In this paper, we introduce a new objective for neural MT based on the mutual dependency between the source and target sentences, inspired by recent work in neural conversation generation (Li et al., 2015a). We build an approximate implementation of our model using reranking, and then to make reranking more powerful we introduce a new decoding method that promotes diversity in the first-pass N-best list.

On English→French and English→German translation tasks, we show that the neural machine translation models trained using the proposed method perform better than corresponding standard models, and that both the mutual information objective and the diversity-increasing decoding methods contribute to the performance boost..

The new models come with the advantages of easy implementation with sources and targets interchanged, and of offering a general solution that can be integrated into any neural generation models with minor adjustments. Indeed, our diversity-enhancing decoder can be applied to generate more diverse N-best lists for any NLP reranking task.

Finding a way to introduce mutual information based decoding directly into a first-pass decoder without reranking naturally constitutes our future work.

Acknowledgements We would especially want to thank Thang Luong Minh for insightful discussions and releasing relevant code, as well as Will Mon-

roe, Sida Wang, Chris Manning and other members from Stanford NLP group for helpful comments and suggestions. The authors also want to thank Michel Galley, Bill Dolan, Chris Brockett, Jianfeng Gao and other members of the NLP group at Microsoft research for helpful discussions. Jiwei Li is very grateful to be supported by a Facebook Fellowship.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

LR Bahl, Peter F Brown, Peter V De Souza, and Robert L Mercer. 1986. Maximum mutual information estimation of hidden markov model parameters for speech recognition. In *Proc. Icassp*, volume 86, pages 49–52.

Dhruv Batra, Payman Yadollahpour, Abner Guzman-Rivera, and Gregory Shakhnarovich. 2012. Diverse m-best solutions in markov random fields. In *Computer Vision–Eccv 2012*, pages 1–16. Springer.

Christian Buck, Kenneth Heafield, and Bas van Ooyen. 2014. N-gram counts and language models from the common crawl. In *Proceedings of the Language Resources and Evaluation Conference*.

Daniel Cer, Christopher D Manning, and Daniel Jurafsky. 2013. Positive diversity tuning for machine translation system combination. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 320–328.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Jacob Devlin and Spyros Matsoukas. 2012. Trait-based hypothesis selection for machine translation. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 528–532. Association for Computational Linguistics.

Jenny Rose Finkel, Christopher D Manning, and Andrew Y Ng. 2006. Solving the problem of cascading errors: Approximate bayesian inference for linguistic annotation pipelines. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 618–626. Association for Computational Linguistics.

Kevin Gimpel, Dhruv Batra, Chris Dyer, Gregory Shakhnarovich, and Virginia Tech. 2013. A systematic exploration of diversity in machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, October*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Liang Huang. 2008. Forest reranking: Discriminative parsing with non-local features. In *Acl*, pages 586–594.

Sébastien Jean, Kyunghyun Cho, Roland Memisevic, and Yoshua Bengio. 2014. On using very large target vocabulary for neural machine translation. *arXiv preprint arXiv:1412.2007*.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Emnlp*, pages 1700–1709.

Shankar Kumar and William Byrne. 2004. Minimum bayes-risk decoding for statistical machine translation. Technical report, DTIC Document.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2015a. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*.

Jiwei Li, Minh-Thang Luong, and Dan Jurafsky. 2015b. A hierarchical neural autoencoder for paragraphs and documents. *arXiv preprint arXiv:1506.01057*.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015a. Effective approaches to attention-based neural machine translation. *EMNLP*.

Minh-Thang Luong, Ilya Sutskever, Quoc V Le, Oriol Vinyals, and Wojciech Zaremba. 2015b. Addressing the rare word problem in neural machine translation. In *Proceedings of ACL*.

Wolfgang Macherey, Franz Josef Och, Ignacio Thayer, and Jakob Uszkoreit. 2008. Lattice-based minimum error rate training for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 725–734. Association for Computational Linguistics.

Fandong Meng, Zhengdong Lu, Mingxuan Wang, Hang Li, Wenbin Jiang, and Qun Liu. 2015. Encoding source language with convolutional neural network for machine translation. *arXiv preprint arXiv:1503.01838*.

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of ACL 2002*, pages 295–302.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proc. Of ACL*, pages 311–318.

Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2015a. Building end-to-end dialogue systems using generative hierarchical neural network models. *arXiv preprint arXiv:1507.04808*.

Iulian Vlad Serban, Ryan Lowe, Laurent Charlin, and Joelle Pineau. 2015b. A survey of available corpora for building data-driven dialogue systems. *arXiv preprint arXiv:1512.05742*.

Libin Shen, Jinxi Xu, and Ralph Weischedel. 2010. String-to-dependency statistical machine translation. *Computational Linguistics*, 36(4):649–671.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. *arXiv preprint arXiv:1506.06714*.

Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112.

Roy W Tromble, Shankar Kumar, Franz Och, and Wolfgang Macherey. 2008. Lattice minimum bayes-risk decoding for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 620–629. Association for Computational Linguistics.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. In *Proc. Of ICML Deep Learning Workshop*.

Oriol Vinyals, Lukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. 2015. Grammar as a foreign language. In *Proc. Of NIPS*.

P. C. Woodland and D. Povey. 2002. Large scale discriminative training of hidden Markov models for speech recognition. *Computer Speech and Language*, 16:25–47.

Tong Xiao, Jingbo Zhu, and Tongran Liu. 2013. Bagging and boosting statistical machine translation systems. *Artificial Intelligence*, 195:496–527.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*.