Cogn Comput
DOI 10.1007/s12559-017-9452-x

# Extreme Learning Machine for Huge Hypotheses Re-ranking in Statistical Machine Translation

**Yan Liu**[1] · **Chi Man Vong**[1] · **Pak Kin Wong**[2]

Received: 8 September 2016 / Accepted: 30 January 2017


**Abstract** In statistical machine translation (SMT), a possibly infinite number of translation hypotheses can be decoded from a source sentence, among which re-ranking is applied to sort out the best translation result. Undoubtedly, re-ranking is an essential component of SMT for effective and efficient translation. A novel re-ranking method called *Scaled Sorted Classification Re-ranking* (SSCR) based on *extreme learning machine* (ELM) classification and *minimum error rate training* (MERT) is proposed. SSCR contains four steps: (1) the input features are normalized to the range of 0 to 1; (2) an ELM classification model is constructed for hypothesis ranking; (3) each translation hypothesis is ranked using the ELM classification model; and (4) the highest ranked subset of hypotheses are selected, in which the hypothesis with best predicted score based on MERT (system score) is returned as the final translation result. Compared with the baseline score (lower bound), SSCR with ELM classification can raise the translation quality up to 6.7% in IWSLT 2014 Chinese to English corpus. Compared with the state-of-the-art *rank boosting*, SSCR has a relatively 7.8% of improvement on BLEU in a larger WMT 2015 English-to-French corpus. Moreover, the training time of the proposed method is about 160 times faster than traditional regression-based re-ranking.

**Keywords** Re-ranking · Extreme learning machine · Scaled sorted classification re-ranking · Statistical machine translation

✉ Chi Man Vong
cmvong@umac.mo

Yan Liu
senliuy@gmail.com

Pak Kin Wong
fstpkw@umac.mo

[1] Department of Computer and Information Science, University of Macau, Macau, China

[2] Department of Electromechanical Engineering, University of Macau, Macau, China

## Introduction

In statistical machine translation (SMT), a source sentence is decoded into a possibly infinite set of translation hypotheses, from which the best one is sorted out through a re-ranking model as the final translation result. Clearly, the re-ranking model is an essential component in SMT to determine the quality of translation result. The re-ranking model can be constructed through different ways over a set of translation hypotheses. However, millions of translation hypotheses can be easily generated from just thousands of source sentences, leading to heavy computation burden for constructing the re-ranking model. Practically, among these millions of translation hypotheses, only a few are of high quality while the remaining are of poor quality and unnecessary. Therefore, there is a practical demand to sort out the high quality ones from the huge set of translation hypotheses in order to reduce the computational complexity for re-ranking modeling, which raises the rationale of our proposed method.

Currently, the research of re-ranking modeling focuses on the following approaches. Linguistics knowledge approach [14, 18, 20, 34] is effective but human linguistics experts are

Published online: 17 February 2017

heavily involved. Discriminative training approach [6] constructs a re-ranking model whose target is to discriminate whether a translation hypothesis is of high quality or not, which does not require much linguistics knowledge [14, 18, 20, 34]. On the other hand, it focuses on finding the relationship between hypothesis features and the translation quality. However, either linear model [23, 30] or log-linear model [4, 17, 22] is insufficient to fit re-ranking tasks, because these tasks are highly linear. To remedy this issue, one approach is using boosting method [6, 8, 20, 21] but not very effective.

Nonlinear machine learning is another popular approach to construct re-ranking model such as Markov Random Fields [6], nonlinear regression [29], support vector machine (SVM) [28], and multilayer perceptron [19, 32]. Among these nonlinear methods, neural network (i.e., SVM or perceptron) works the best but is computationally inefficient for the construction of re-ranking model. Recently, a kind of single hidden layer neural network called *extreme learning machine* (ELM) was proposed, whose input weight and bias are randomly generated and output weights can be calculated without iteration [9–12, 27, 33]. Therefore, the model training is extremely fast for huge amount of training data while the accuracy is even outperforming.

In this work, a novel re-ranking method called *scaled sorted classification re-ranking* (SSCR) based on *extreme learning machine* (ELM) [10–12, 27, 33] and *minimum error rate training* (MERT) is proposed in order to effectively and efficiently improve the re-ranking modeling. SSCR involves two models: a nonlinear ELM classification and a log-linear regression model (MERT). Firstly, the ELM classification model ranks all hypotheses. According to the ranking, the top-ranked hypotheses are sorted out for numerical evaluation under the MERT model. As a result, the number of translation hypotheses is significantly reduced in order to relieve the stress of computation. Based on ELM, the training time of ranking model can be significantly reduced while higher translation accuracy is achieved.

The rest of this paper is organized as follows. In "SSCR", the details of SSCR for the re-ranking problem are given. Experiment results and discussion are provided in "Experiments". Finally, conclusion is drawn in "Conclusion".

## SSCR

### Problem Statement

Given a trained language model and a translation model, a source sentence is firstly decoded into a set of $N$ translation hypotheses (along with their features listed in Table 1) called $N$-best list, in which each feature may contain one more attribute values about the translation hypothesis. Note that the feature "System score" is a nonlinear function of all other features/attribute values. Although a score is given for a hypothesis translation, it is unreliable to depend on just this score. Basically, the system score is the combination of some features decoded from the source sentence, which does not directly reflexes the translation quality (an evaluation about the hypothesis and its target sentence). Therefore, by comparing the hypothesis with its target sentence, another popular metric called *Bilingual Evaluation Understudy* (BLEU) is returned. BLEU is a measure to evaluate the translation quality of one language to another. Some other evaluation methods such as *National Institute of Standards and Technology* (NIST) [7], *Metric for Evaluation of Translation with Explicit ORdering* (METEOR) [2], or *Word Error Rate* (WER) [15] are also worked in SSCR.

Using the features in $N$-best list and some additional features as input, a regression model is traditionally trained to predict the corresponding BLEU value for re-ranking as illustrated Fig. 1. However, there are two drawbacks in traditional regression: (i) re-ranking modeling usually involves millions of translation hypotheses that cannot be efficiently constructed; and (ii) among these millions of hypotheses, only a few are valuable and the remaining majority become a noise that affects the performance of re-ranking model.

From these perspectives, we propose to replace the traditional regression by an ELM classification model for its high generalization and fast training speed even for millions of hypotheses. After classifying the top-ranked subset of translation hypotheses based on BLEU, the hypothesis with highest system score (MERT) in the subset is returned as the final result. These steps constitute our proposed SSCR. In the following, the procedure of applying ELM classification in re-ranking modeling is detailed.

### Data Pre-processing

Data normalization and ranking are two main tasks in data preprocessing, in which training and testing data are prepared.

**Feature Scaling** Feature scaling is known as data normalization which is a significant procedure because model performance may be deteriorated without feature scaling [13]. For example, if one feature has a broader range of values than other features, this feature could easily dominate the model output [1]. Assume $m$ hypotheses $h_i$ are decoded from $s$ source sentences, $i = 1, 2, ..., m$. Then, a set of translation hypotheses $\mathbf{D} = \{h_i; \boldsymbol{f}_i; b_i\}$ is obtained where $\boldsymbol{f}_i = [f_{i,1}, f_{i,2}, ..., f_{i,\tau}]$ consists of $\tau$ features, and $b_i$ is the

**Table 1** Features extracted for each hypothesis from $N$-best list

| Feature name | Number of values | explanation |
|---|---|---|
| Reordering model | 6 values | Modeling the influence of semantic relationships on the order of words within the scope of a phrase |
| Distortion model | 1 value | A model measures the reordering of the input sentence, the more of reordering, the more expensive of the translation |
| Language model | 1 value | Measure the influence of this sentence. The statistical language model assigns a probability to a sequence of $m$ words $P(\omega_1, ..., \omega_m)$ by means of a probability distribution |
| Word penalty | 1 value | Measure the word is too long or not |
| Phrase penalty | 1 value | Per produced phrase generate a cost |
| Translation model | 4 values | Measure whether the translation result is accord with the target language |
| System score | 1 value | The log-linear weighted score of features above produced by the model with MERT algorithm |

BLEU value for $h_i$. Before training a classification model $\mathbf{M}$, all $f_{i,j}$ are normalized into the range [0, 1]:

$$f_{i,j}^{\text{new}} = \frac{f_{i,j} - \min(\boldsymbol{f}_i)}{\max(\boldsymbol{f}_i) - \min_(\boldsymbol{f}_i)} \tag{1}$$

where $j = 1, 2, .., \tau$, $\max(\boldsymbol{f}_i)$ and $\min_(\boldsymbol{f}_i)$ are the maximum and minimum value of $\boldsymbol{f}_i$, respectively. All new features $f_{i,j}^{\text{new}}$ form a new matrix $\mathbf{F}_C$ which is used to train $\mathbf{M}$.

**Ranking** Aside from generating $\mathbf{F}_C$ under $\mathbf{D}$, the ranking for each hypothesis in $\mathbf{D}$ is determined according to its translation quality (i.e., BLEU value). As shown in Fig. 2, given a user-defined parameter $c$ (number of classes), $c-1$ boundaries $\boldsymbol{\theta} = \{\theta_1, \theta_2, ..., \theta_{c-1}, \}$ are determined for different rankings. Assuming $m'$ hypotheses are decoded from a source sentence, the $m'$ hypotheses are sorted in descending order using $Quicksort$ based on their BLEU values $b_i$. These $m'$ sorted BLEU values constitute an array $\mathbf{A}$ for hypotheses ranking,

$$\mathbf{A} = Quicksort(b_1, b_2, ..., b_{m'}, \textit{'descending'}) \tag{2}$$

In the sorted array $\mathbf{A}$, $\theta_k$ is the $(k \times \lceil \frac{m'}{c} \rceil)^{th}$ BLEU value, $k = 1, 2, ..., c-1$, and $\theta_c$ is upper boundary and $\theta_0 = 0$ is the lower boundary. Subsequently, for $p = 1, 2, ..., m'$, the ranking $t_p$ for the $p^{th}$ hypothesis can be determined by

$$t_j = \begin{cases} 1 \text{ if } \theta_1 < b_p \leq 1 \\ j \text{ if } \theta_j < b_p \leq \theta_{j-1} \quad j = 2, 3, ..., c-1 \\ c \text{ if } 0 \leq b_p \leq \theta_{c-1} \end{cases} \tag{3}$$

The rankings of all hypotheses then constitute a vector $\boldsymbol{t}$ so that the preprocessed training data $\aleph = \{\mathbf{F}_C; \boldsymbol{t}\}$ is obtained to train the classification model $\mathbf{M}$.

## Training ELM Classification Model

Given the training data $\aleph$, the ELM classification model in SSCR can be constructed as illutrated in Fig. 3. Suppose the

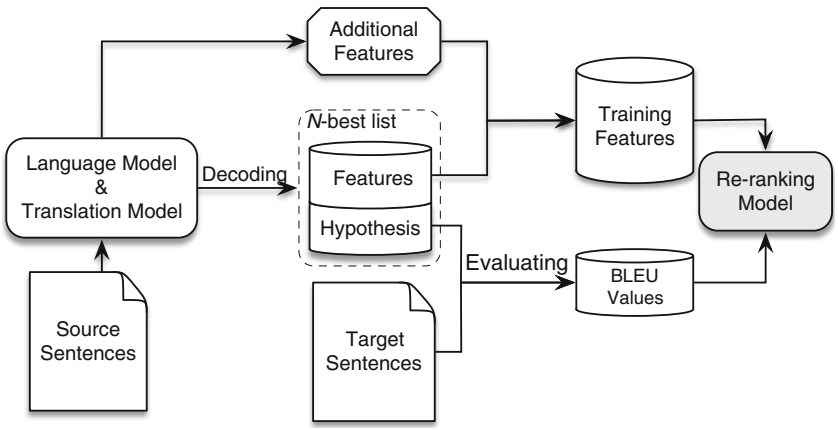


**Fig. 1** Re-ranking in SMT

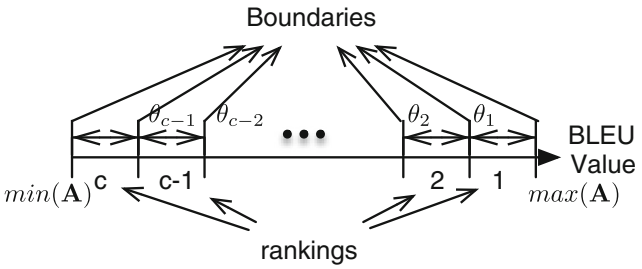


**Fig. 2** The illustration of ranks assignment

ELM model has $n$ hidden nodes, the model can be expressed as:

$$\sum_{i=1}^{m}\sum_{j=1}^{n}\boldsymbol{\beta}_j g(\boldsymbol{w}_j \boldsymbol{f}_i + b_j) \tag{4}$$

where $\boldsymbol{w}_j$ and $b_j$ are randomly generated input weights and bias, $g(\cdot)$ is the user-defined activation function and $\boldsymbol{\beta}_j$ is the output weight. If the activation function $g(\cdot)$ and the neural network can approximate the translation quality of these $m$ hypotheses, the ELM model can be constructed as follows:

$$h(\boldsymbol{f}) = \sum_{i=1}^{m}\sum_{j=1}^{n}\boldsymbol{\beta}_j g(\boldsymbol{w}_j \boldsymbol{f}_i + b_j); \tag{5}$$

In matrix form, Eq. 5 can be rewritten as:

$$\mathbf{G}\boldsymbol{\beta} = \mathbf{R} \tag{6}$$

where

$$\mathbf{G} = \begin{bmatrix} g(\boldsymbol{w}_1 \cdot \boldsymbol{f}_1 + b_1) & \dots & g(\boldsymbol{w}_n \cdot \boldsymbol{f}_1 + b_n) \\ \vdots & \dots & \vdots \\ g(\boldsymbol{w}_1 \cdot \boldsymbol{f}_m + b_1) & \dots & g(\boldsymbol{w}_n \cdot \boldsymbol{f}_m + b_n) \end{bmatrix} \tag{7}$$

and $\mathbf{R} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, ..., \boldsymbol{r}_m\}$ is the matrix format of $\boldsymbol{t}$ (e.g., if $c = 4$, $t_i = 2$, then $\boldsymbol{r}_i = \{0, 1, 0, 0\}$, $i = 1, 2, ..., m$).

In our method, the output matrix $\mathbf{G}$ is a $m \times n$ matrix where $m$ is the number of training hypotheses and $n$ is the number of hidden nodes. In our experiments, there are more than hundred thousands of training cases. However, only 200 or less hidden nodes are needed. Therefore, it means the output matrix G is a *fat* matrix ($m \gg n$). With this reason, $\boldsymbol{\beta}$ can be calculated with a faster method:

$$\boldsymbol{\beta} = \left(\frac{\mathbf{I}}{C} + \mathbf{G}^T\mathbf{G}\right)^{-1}\mathbf{G}^T\mathbf{R} \tag{8}$$

where $\mathbf{I}$ is the identity matrix and $C$ is a user-defined regularization factor. Finally, the classification model $\mathbf{M} = \{\mathbf{W}; \boldsymbol{b}; g(\cdot); \boldsymbol{\beta}\}$ is generated.

## MERT

MERT is a measure to evaluate the translation hypothesis. This is employed to calculate the system score in Table 1. Given $m'$ hypotheses $\mathbf{H} = \{h_1, h_2, ...h_{m'}\}$ decoded from a source sentence $s$, the most probable hypothesis $\hat{h}$ for $s$ is chosen as:

$$\hat{h} = \arg\max P(h|s) \tag{9}$$

The posterior probability $P(h|s)$ can be constructed with a log-linear model of other features $\mathbf{F} = \{f_1, f_2, ..., f_\tau\}$. In Table 1, the system score is assigned with:

$$f_{\text{score}}(h, s) = \mathbf{WF} = \sum_{j=1}^{\tau-1} w_j f_j \tag{10}$$

where $\mathbf{W} = \{w_1, w_2, ..., w_{\tau-1}\}$ is the weights of features. Then, the posterior probability $P(s|h)$ is defined as:

$$P(h|s) = \frac{\exp(f_{\text{score}}(h, s))}{\sum_{h' \in \mathbf{H}} \exp(f_{\text{score}}(h', s))} \tag{11}$$

Therefore, $\hat{h}$ is return as the best translation where:

$$\begin{aligned} \hat{h} = \arg\max_h P(h|s) &= \arg\max_h \frac{\exp(f_{\text{score}}(h, s))}{\sum_{h' \in \mathbf{H}} \exp(f_{\text{score}}(h', s))} \\ &= \arg\max_h f_{\text{score}}(h, s) \end{aligned} \tag{12}$$

Instead of using maximum likelihood estimation, MERT uses the translation quality (i.e., BLEU) as the optimization

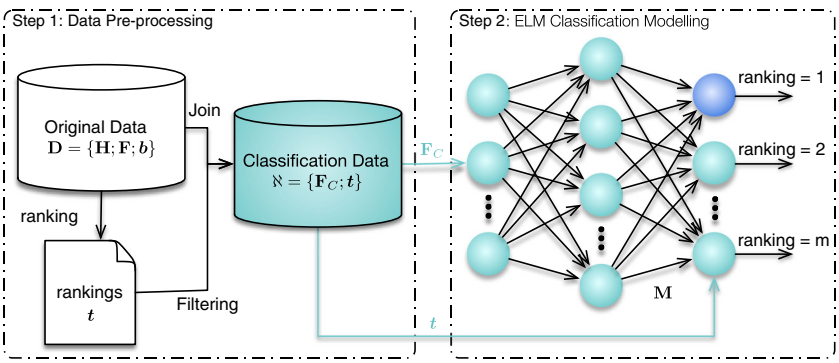


**Fig. 3** Training procedure of SSCR

target to adjust the weight $w_i$ of each feature. Powell's algorithm [26] is adopted which is designed to adjust one parameter with others fixed each time. Assume we set all weights $w_j (j \neq c)$ fixed except the $c^{th}$ one. Then, Eq. 12 is equivalent to :

$$\hat{h} = \arg\max_{h}(w_c f_c + c(f_{\bar{c}})) \tag{13}$$

where $c(f_{\bar{c}}) = \sum_{j \neq c} w_j f_j$ is a constant number.

Therefore, the system score is a linear function with *gradient* $f_c$ and *y-intercept* $c(f_{\bar{c}})$. If the lines for all hypotheses are plotted (Fig. 4), the intersection points of different lines become *threshold points* ($\Theta = \{\theta_1, ..., \theta_k\}$) where $k$ is the number of intersection points. And the upper envelops of top-most lines correspond to arg max in Eq. 12. The intersection $\theta_i$ is the point where the error changes (due to the change of 1-best hypothesis).

Therefore, these different regions can determine a piecewise constant function of $h_i$, namely *error surface*. By merging a new error surface to the existing error surface of all $m'$ hypotheses one by one, a corpus-level error surface is generated. Figure 5 shows an example of merging two hypotheses.

At each iteration of Powell's Algorithm, MERT randomly chooses a starting weight vector $\boldsymbol{w}_0$ and a search direction vector $\boldsymbol{d}$, and then determines which hypothesis is the best one for all weight vectors $\boldsymbol{w}' = \boldsymbol{w} + \theta \boldsymbol{d}$ by traversing $\theta$ from $-\infty$ to $+\infty$. The optimal $\theta$ is chosen by maximizing the total BLEU value of each region. Then, another weight vector and another search direction vector are selected. The Powell's algorithm is iteratively run until the weights convergence, or reaching the maximum number of iterations.

### Testing or Execution

Given a source sentence, a list of hypotheses and its corresponding features $\aleph_e = \{\mathbf{F}_e; \mathbf{H}_e\}$ are generated using a commercial software called Moses [16] where $\mathbf{F}_e$ is the feature matrix and $\mathbf{H}_e$ is the corresponding set of translation hypotheses. Based on the features $\mathbf{F}_e$ only (no translation quality), the ELM classification model $\mathbf{M}$ predicts the rankings for all translation hypotheses. This execution procedure is done in the following steps as illustrated in Fig. 6:

i) Feature scaling: apply feature normalization in "FeatureScaling".
ii) Ranking hypotheses: $\mathbf{M}$ is applied to predict rankings for all hypotheses. Afterwards, the subset $\aleph'_e$ of $\aleph_\mathbf{e}$ with highest rank (i.e., rank = 1) is selected.
iii) Output translation result: in the selected subset, a log-linear model trained by MERT is used to generate the final translation.

## Experiments

In this section, the experiments to validate the proposed SSCR are described, together with the property of the corpus/data. Experimental setup and the process of model parameters tuning are detailed. Finally, the performance evaluation of ELM classification against regression model in terms of model training time and accuracy are shown.

### Hardware and Software

All the experiments were carried out under macOS Sierra 10.12 with an Intel Core i5 of 3.4 GHz and 24 GB RAM. In

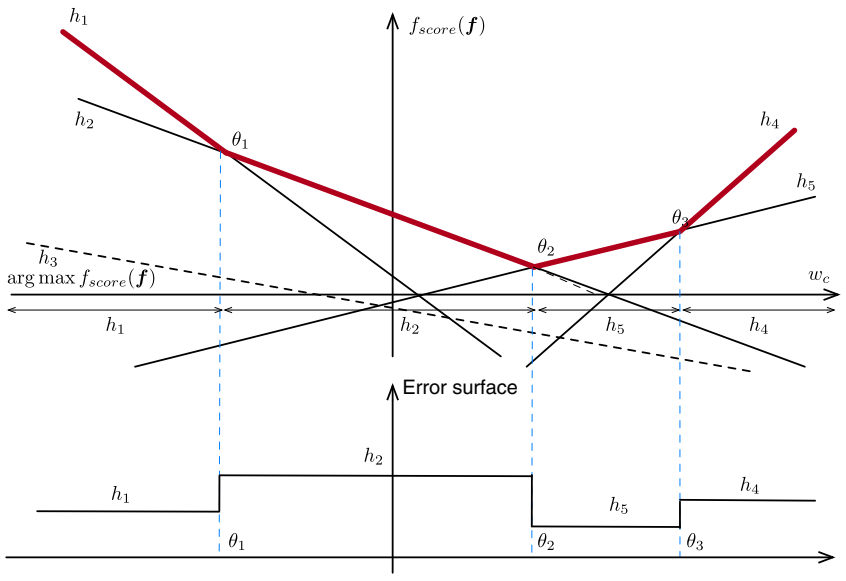


**Fig. 4** An example to illustrate the definition of $f_{\text{score}}$, $\Theta$, error surface, etc

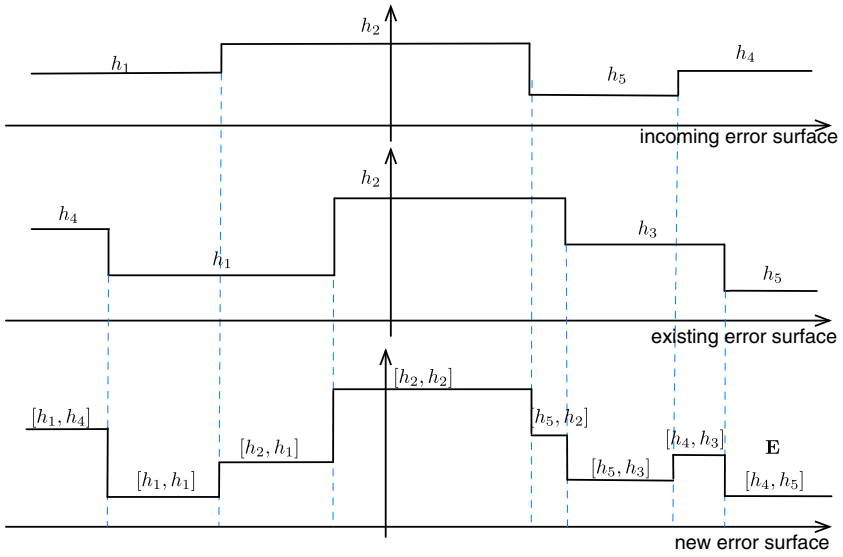


**Fig. 5** Illustration of merging error surface of two hypotheses

addition, the following software packages were employed in our work.

**GIZA++ v2** [25]: an implementation of IBM Models and Hidden Markov Model (HMM) based alignment algorithm for computing the word alignments.

**Moses** [16]: (1) The decoder to generate the $N$-best list for each input source sentence from the test data. "$N$" is the number of hypotheses for each source sentence. (2) The toolkit to train the log-linear model of MERT.

**SRILM** [31]: A 5-gram language model with Kneser-Ney smoothing was trained with SRILM on monolingual target side of the parallel corpus for each languages pair.

## Data

The experiments were conducted on two most recent benchmarking corpora, namely, IWSLT 2014 English-Chinese [5] and WMT 2015 French-English parallel corpora [3]. Hypotheses are firstly decoded and generated from the test sentences in the corpora using Moses, out of which the $N$-best list of hypotheses was obtained as training data for the re-ranking model, while the remaining hypotheses as test data. We used SRILM to train the language model and then used Moses to train the translation model and language model. The translation model is used to decode the $N$-best list. For Chinese-English translation of IWSLT 2014, $N = 100$-best hypotheses was employed for constructing the re-ranking model. For translation of French-English pair (WMT 2015), as the size of the parallel corpus is much larger than IWSLT 2014, $N = 2{,}000$-best hypotheses were considered. The features for each hypothesis is presented in Table 1 and the statistics of the data is presented in Table 2.

## Settings

Using toolkits mentioned in "Hardware and Software" and corpora in "Data", we generated a 100-best file for IWSLT 2014 and a larger 2,000-best file for WMT 2015. With feature scaling introduced in "Feature Scaling", a matrix with scaled features was generated. Then, the BLEU value of each hypothesis is calculated. According to its BLEU value, the ranking of each hypothesis is determined as mentioned in "Data Pre-processing". Finally, the dataset is divided into training set and testing set. With the scaled features and the rankings of training data, SSCR with ELM is applied

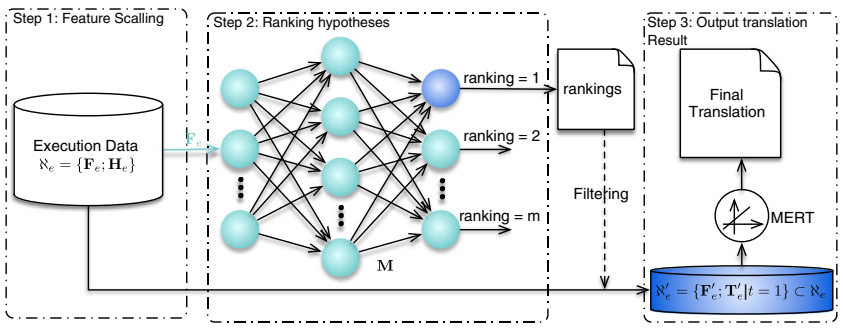


**Fig. 6** Testing procedure of SSCR

**Table 2** Statistics of datasets

| Corpus | IWSLT 2014 | WMT 2015 |
|---|---|---|
| Language pair | English-Chinese | French-English |
| Test sentences | 1,093 | 3,000 |
| $N$-best hypotheses | 100 | 2,000 |
| Collected hypotheses | 109,040 | 3,213,906 |

to generate the re-ranking model. In SSCR, the following parameters were empirically adjusted to obtain an optimal result:

- class number $c$;
- activation function $g(\cdot)$;
- number of hidden nodes $n$;
- regularization factor $C$.

## Baseline

In SMT tasks, baseline is always taken as the lower bound of translation quality. MERT [22] is adopted as the baseline with its high accuracy in our experiments. Detailed procedure of MERT can be found in "MERT".

## Results and Discussion

In order to fairly evaluate the performance of SSCR, different numbers of training data were employed for re-ranking modeling. Each training dataset was run for ten times in order to observe the stability of SSCR, and the results of each experiment are shown in Fig. 7. In general, SSCR is stable as illustrated Fig. 7. Therefore, once the model parameters are optimized, a stable and effective nonlinear re-ranking model can be obtained.

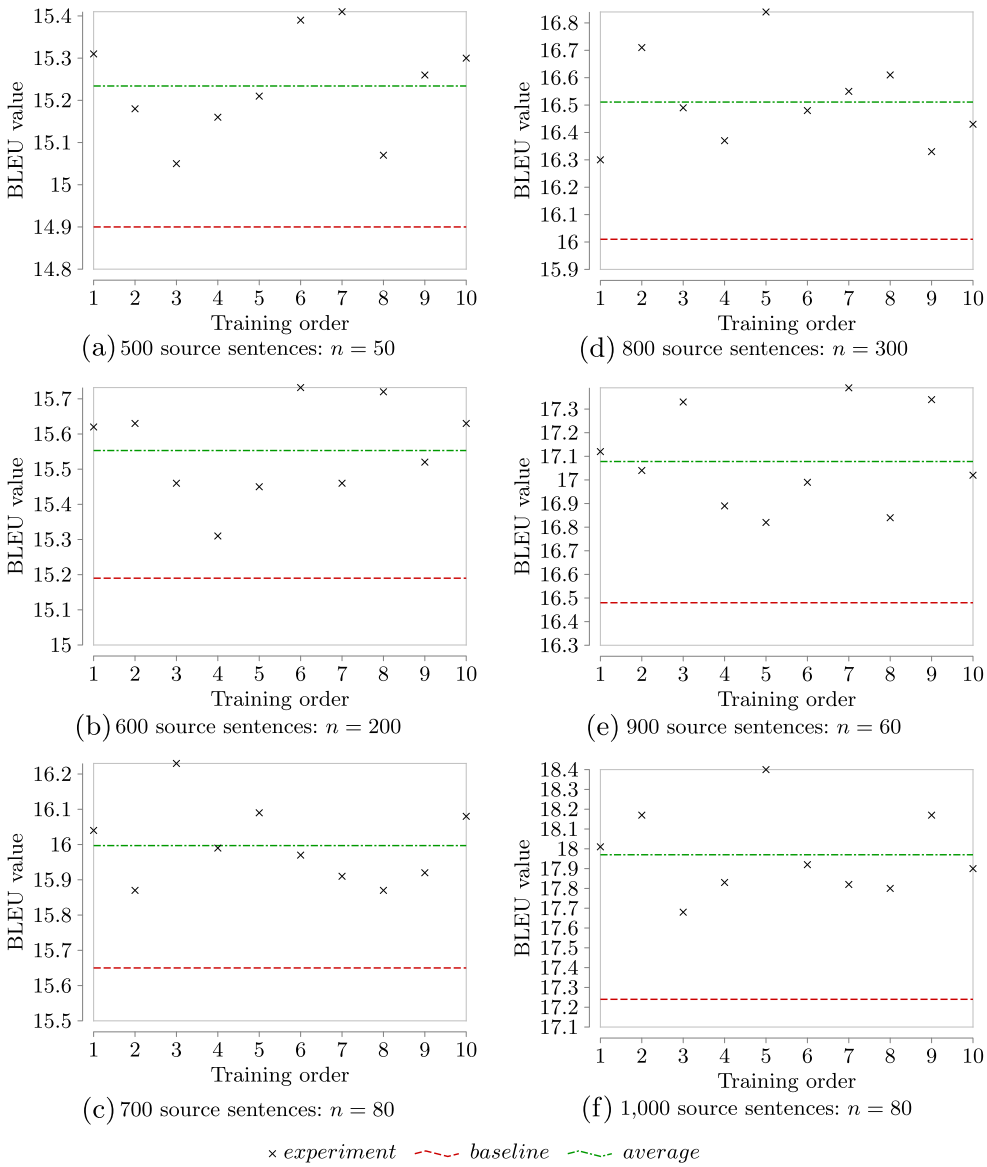


**Fig. 7** Re-ranking results. Parameters: the number of class $c = 10$, activation function $g(\cdot) = sigmoid$ and regularization factor $C = 2$. For the number of hidden nodes, according to different corpora, different numbers of nodes were used. Mostly, 60–100 hidden nodes could yield the best results

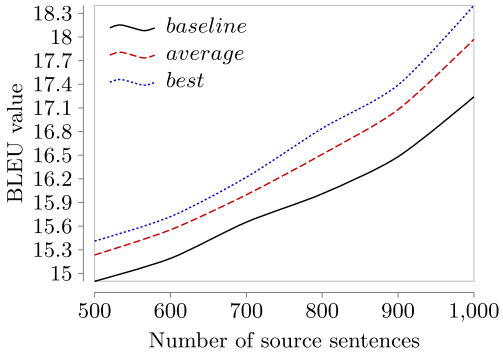


**Fig. 8** Result of IWSLT 2014

For a better illustration, the experimental results are respectively shown in curve form (Fig. 8) and in numerical form (Table 3). The relative improvement in percentage are also presented in Table 4, in which SSCR can raise on average 3.5 % of BLEU value over baseline after optimization of parameters. Another observation is that as training dataset gets larger, SSCR tends to obtain a better result. Note when 1,000 source sentences are used, the best re-ranking model achieves a relative improvement of 6.7 % over baseline.

Another experiment is about scale-up issue for which the experiment was carried out on a bigger corpus WMT 2015, containing totally 3,009 test sentences. As mentioned, a 2,000-best list was generated from WMT 2015. After experiments, the performance was found reaching the peak when the number of source sentences goes beyond 2,000. A nonlinear discriminative training method [24] was compared with SSCR under WMT 2015 because both of these methods has similar features. 2,000 source sentences were randomly selected as training dataset. The results can be seen in Fig. 9. With the same corpus and same features, the method proposed in [24] only has an improvement of 0.346 in BLEU while SSCR obtains 0.373, which is an improvement of about 1.5 %.

**Table 3** Re-ranking result of IWSLT 2014

| Number of source sentences | BLEU baseline | Experiment average | Experiment best |
|---|---|---|---|
| 500 | 14.90 | 15.234 | 15.41 |
| 600 | 15.19 | 15.553 | 15.72 |
| 700 | 15.65 | 15.997 | 16.22 |
| 800 | 16.01 | 16.511 | 16.84 |
| 900 | 16.48 | 17.078 | 17.39 |
| 1,000 | 17.24 | 17.970 | 18.40 |

**Table 4** Improvement over baseline for IWSLT 2014

| Number of source sentences | Average over baseline | Average improvement (%) | Best over baseline | Best improvement (%) |
|---|---|---|---|---|
| 500 | 0.334 | 2.2 | 0.51 | 3.4 |
| 600 | 0.363 | 2.4 | 0.53 | 3.5 |
| 700 | 0.347 | 2.2 | 0.56 | 3.6 |
| 800 | 0.50 | 3.1 | 0.91 | 5.2 |
| 900 | 0.598 | 3.6 | 0.89 | 5.5 |
| 1,000 | 0.730 | 4.2 | 1.16 | 6.7 |

## Training

A significant contribution of SSCR is its extremely fast training time, it dues to two folds: (1) the advantage of plain ELM classification and (2) the result of MERT is stored in features **D** (which is system score) during generating the $N$-best list, which means the MERT need not running again when constructing the re-ranking model. Regression was proposed in re-ranking [29] which has been verified as a significant method. In our work, a very fast regression based on kernel ELM [11] was employed as a comparison. For IWSLT 2014 corpus with 100-best list, the training times of ELM kernel regression and SSCR. The results are shown in Fig. 10. As shown, the training time of SSCR is almost constant with increasing number of hypotheses. However, the training time of ELM kernel regression increases exponentially. For WMT 2015 with 3,000 source sentences and 2,000-best hypotheses, SSCR only takes less than one hour to train the re-ranking model, the results are in Fig. 10.

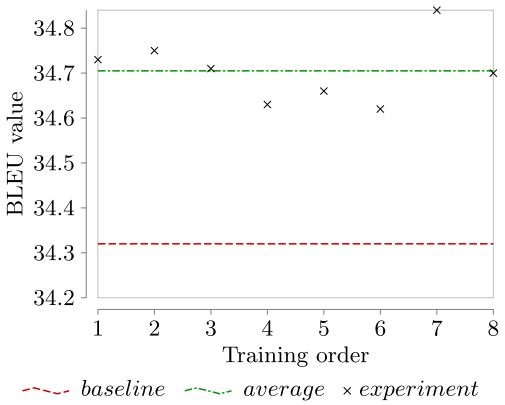


**Fig. 9** Result of WMT 2015 under 2,000 source sentences, 400 hidden nodes, binary classes, activation function $g(\cdot) = \text{sigmoid}$

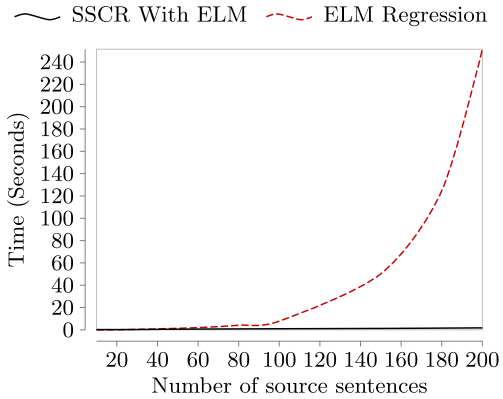


**Fig. 10** Comparison of training times for SSCR and traditional regression using ELM

## Conclusion

Re-ranking is a significant component for effectiveness and efficiency in SMT. With consideration of cognitive computation, a novel nonlinear re-ranking model called SSCR based on ELM and MERT is proposed. The significance of SSCR is twofold:

- The effectiveness of SSCR was evaluated over two popular benchmark corpora, IWSLT 2014 Chinese-English and WMT 2015 French-English. By the innovative mechanism of SSCR, the effectiveness on IWSLT 2014 is raised by 6.7 % of BLEU value over baseline, and 1.5 % on WMT 2015 in our experiments which is better than a state-of-the-art method [30].
- Another significance of SSCR is the adoption of classification rather than regression in traditional re-ranking model. Under SSCR, the training time of re-ranking model is significantly reduced from hours to seconds. In current experiments, for a corpus of 200 source sentences with 2,000-best hypotheses, SSCR based on ELM classification can run 160 times faster than traditional re-ranking model based on ELM regression.

Therefore, it can be concluded that the proposed SSCR can significantly improve both effectiveness and efficiency for SMT which possesses a large amount of training data.

#### Compliance with Ethical Standards

**Funding** The work is financially supported by funding from University of Macau, project number MYRG2014-00083-FST, MYRG2016-00134 and from FDCT Macau, project number 050/2015/A.

**Conflict of Interests** Yan Liu, Chi Man Vong, and Pak Kin Wong declare that they have no conflict of interest.

**Ethical Approval** Informed consent was not required as no human or animals were involved.

**Human and Animal Rights** This article does not contain any studies with human or animal subjects performed by any of the authors.

## References

1. Aksoy S, Haralick RM. Feature normalization and likelihood-based similarity measures for image retrieval. Pattern Recogn Lett. 2001;22(5):563–82.
2. Banerjee S, Lavie A. Meteor: an automatic metric for MT evaluation with improved correlation with human judgments. Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization; 2005. p. 65–72.
3. Bojar O, Chatterjee R, Federmann C, Haddow B, Hokamp C, Huck M, Logacheva V, Pecina P, (eds). 2015. Proceedings of the tenth workshop on statistical machine translation. Lisbon, Portugal: Association for Computational Linguistics. http://aclweb.org/anthology/W15-30.
4. Brown PF, Pietra VJD, Pietra SAD, Mercer RL. The mathematics of statistical machine translation: parameter estimation. Comput Linguist. 1993;19(2):263–311.
5. Cettolo M, Niehues J, Stüker S, Bentivogli L, Federico M. Report on the 11th iwslt evaluation campaign, iwslt 2014. In: Proceedings of the eleventh international workshop on spoken language translation (IWSLT), Lake Tahoe, CA; 2014. p. 2–17.
6. Collins M, Koo T. Discriminative reranking for natural language parsing. Comput Linguist. 2005;31(1):25–70.
7. Doddington G. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. Proceedings of the second international conference on human language technology research. Morgan Kaufmann Publishers Inc; 2002. p. 138–45.
8. Duh K, Kirchhoff K. Beyond log-linear models: boosted minimum error rate training for n-best re-ranking. In: Proceedings of the 46th annual meeting of the association for computational linguistics on human language technologies: short papers. Association for Computational Linguistics; 2008. p. 37–40.
9. Huang GB. What are extreme learning machines? Filling the gap between Frank Rosenblatt's dream and John von Neumann's puzzle. Cogn. Comput. 2015;7(3):263–78. doi:10.1007/s12559-015-9333-0.
10. Huang GB, Bai Z, Kasun LLC, Vong CM. Local receptive fields based extreme learning machine. IEEE Comput Intell Mag. 2015;10(2):18–29.
11. Huang GB, Zhou H, Ding X, Zhang R. Extreme learning machine for regression and multiclass classification. IEEE Trans. Syst. Man Cybern. Part B Cybern. 2012;42(2):513–29.
12. Huang GB, Zhu QY, Siew CK. Extreme learning machine: a new learning scheme of feedforward neural networks. IEEE international joint conference on neural networks, 2004. Proceedings. 2004. IEEE; 2004. p. 985–90.
13. Juszczak P, Tax D, Duin R. Feature scaling in support vector data description. In: Proceedings of the ASCI. Citeseer; 2002. p. 95–102.
14. Kirchhoff K, Yang M. Improved language modeling for statistical machine translation. In: Proceedings of the ACL workshop on building and using parallel texts. Association for Computational Linguistics; 2005. p. 125–28.
15. Klakow D, Peters J. Testing the correlation of word error rate and perplexity. Speech Comm. 2002;38(1–2):19–28.

doi:10.1016/S0167-6393(01)00041-3. http://www.sciencedirect.com/science/article/pii/S0167639301000413.
16. Koehn P, Hoang H, Birch A, Callison-Burch C, Federico M, Bertoldi N, Cowan B, Shen W, Moran C, Zens R, Dyer C, Bojar O, Constantin A, Herbst E. Moses: open source toolkit for statistical machine translation. In: Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions, ACL '07. Stroudsburg, PA, USA: Association for Computational Linguistics; 2007. p. 177–80. http://dl.acm.org/citation.cfm?id=1557769.1557821.
17. Koehn P, Och FJ, Marcu D. Statistical phrase-based translation. In: Proceedings of the 2003 conference of the north american chapter of the association for computational linguistics on human language technology. Association for Computational Linguistics; 2003. p. 48–54.
18. Lagarda AL, Casacuberta F. Applying boosting to statistical machine translation. In annual meeting of european association for machine translation (EAMT); 2008. p. 88–96.
19. Li Z, Khudanpur S. Forest reranking for machine translation with the perceptron algorithm. GALE book chapter on MT from text. 2009.
20. Luong NQ, Besacier L, Lecouteux B. Word confidence estimation for smt n-best list re-ranking. In: Proceedings of the workshop on humans and computer-assisted translation (HaCaT) during EACL; 2014.
21. Nguyen P, Mahajan M, He X. Training non-parametric features for statistical machine translation. In: Proceedings of the second workshop on statistical machine translation. Association for Computational Linguistics; 2007. p. 72–79.
22. Och FJ. Minimum error rate training in statistical machine translation. Proceedings of the 41st annual meeting on association for computational linguistics. Association for Computational Linguistics; 2003. p. 160–67.
23. Och FJ, Gildea D, Khudanpur S, Sarkar A, Yamada K, Fraser A, Kumar S, Shen L, Smith D, Eng K, et al. A smorgasbord of features for statistical machine translation. HLT-NAACL; 2004. p. 161–68.
24. Och FJ, Ney H. Discriminative training and maximum entropy models for statistical machine translation. In: Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics; 2002. p. 295–302.
25. Och FJ, Ney H. A systematic comparison of various statistical alignment models. Comput Linguist. 2003;29(1):19–51.
26. Powell MJ. An efficient method for finding the minimum of a function of several variables without calculating derivatives. Comput J. 1964;7(2):155–62.
27. Savitha R, Suresh S, Kim HJ. A meta-cognitive learning algorithm for an extreme learning machine classifier. Cogn Comput. 2014;6(2):253–63. doi:10.1007/s12559-013-9223-2.
28. Shen L, Joshi AK. An svm based voting algorithm with application to parse reranking. In: Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003. Association for Computational Linguistics; 2003. p. 9–16.
29. Shen L, Sarkar A, Och FJ. Discriminative reranking for machine translation. HLT-NAACL; 2004. p. 177–84.
30. Sokolov A, Wisniewski G, Yvon F. Non-linear n-best list reranking with few features. In: Proceedings of the conference of the association for machine translation in the Americas (AMTA). San Diego (CA); 2012.
31. Stolcke A et al. Srilm-an extensible language modeling toolkit. INTERSPEECH; 2002.
32. Sun H, Zhao T. Neural network-based reranking model for statistical machine translation. 11th international conference on fuzzy systems and knowledge discovery (FSKD), 2014. IEEE; 2014. p. 460–65.
33. Wong PK, Gao XH, Wong KI, Vong CM. An analytical study on reasoning of extreme learning machine for classification from its inductive bias. Cogn Comput. 2016;8(4):746–56. doi:10.1007/s12559-016-9414-8.
34. Zhang Y, Hildebrand AS, Vogel S. Distributed language modeling for n-best list re-ranking. In: Proceedings of the 2006 conference on empirical methods in natural language processing. Association for Computational Linguistics; 2006. p. 216–23.