Risk-aware Distribution of SMT Outputs for Translation of Documents Targeting Many Anonymous Readers

Yo Ehara[†], *Masao Utiyama*[‡], *Eiichiro Sumita*[‡]

†: Tokyo Metropolitan University, Tokyo, Japan
‡: National Institute of Information and Communications Technology, Kyoto, Japan
†:ehara@tmu.ac.jp, ‡:{mutiyama,eiichiro.sumita}@nict.go.jp

Abstract

Web documents and news articles are typically written for many anonymous readers. Thus, when translating such documents, the total quality of translations distributed to the entire readers should be considered. Previous statistical machine translation studies have focused on selecting the best translation from N-best candidates. However, when dealing with many readers, it is not necessary to identify the best translation. Our key idea is to distribute all good candidate translations to the readers and improve the total quality of the translations. We simulated a case with $1,000$ news document readers and showed statistically significant gain in sentence-level BLEU scores averaged over those readers.

1. Introduction

Web documents and news articles are typically written for many anonymous readers. Unlike documents that target specific readers such as mails and letters, the number of readers of web documents and news articles cannot be determined in advance. When translating documents that target a large number of readers, our goal is to improve the total quality of all translated documents rather than improving the translation quality of a single document.

Previous statistical machine translation (SMT) studies have focused on selecting one best translation from many candidate translations and have not considered the number of readers [1, 2]. Selecting one translation frees us from considering the number of readers because a target language reader usually only reads one translation of source language material. Thus, selecting a single translation is an effective strategy if a good translation is always selected as the best translation.

However, current SMT techniques cannot always identify the *actual* best translation from candidate translations. In many cases, even when there is a good translation among the candidates, SMT systems frequently rank bad translations higher than good translations. In other words, the strategy that attempts to find a single best translation risk selecting poor translations, even when good candidate translations are available. Thus, it is preferable to select multiple candidate translations when the task setting allows us to do so.

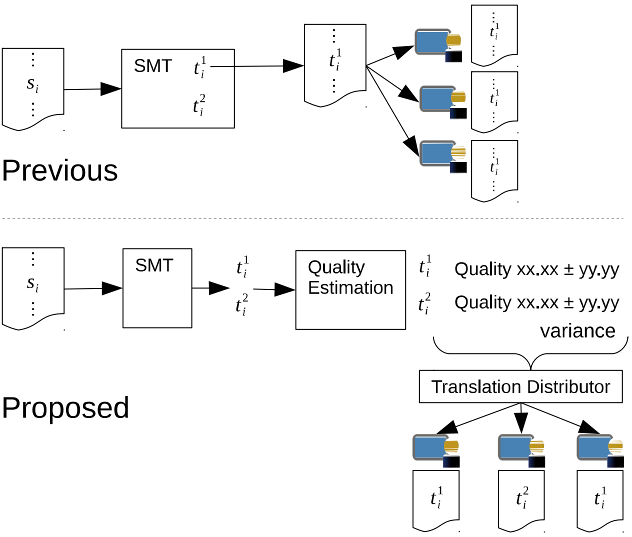

Figure 1: *Schematic Comparison of Previous and Proposed Approach*

We propose an approach for distributing multiple translation candidates when translating documents for many anonymous readers, such as web documents and news articles. Our key idea is to distribute all seemingly good candidate translations. A schematic diagram of the proposed approach is presented in Figure 1. In a previous approach [1], for source sentence s_i, an SMT system produces and ranks several candidate translations of s_i, i.e., t_i^1 and t_i^2. Only the top ranked translation, t_i^1, is used; therefore, the three readers only read t_i^1. However, it is possible that the actual quality of t_i^1 is lower than that of t_i^2. In this situation, the readers do not have access to the best translation. In the proposed approach, we perform *quality estimation* (QE) for the quality and quality variance of each candidate translation. Considering both quality and variance, we calculate *rates* that determine how many of the entire readers should read each candidate translation. Then, using these rates, we *distribute* candidate translations to all the readers. As can be seen in Figure 1, the

[†]This work was conducted when the first author was employed at NICT. We would like to thank anonymous reviewers for their insightful comments that helped us to improve this paper. This paper was proofread under the funding of NICT and JSPS KAKENHI Grant Number 15K16059.

Table 1: Motivating Example using Japanese to English Translation; "inu" means dog or dogs, and "naku" has multiple meanings. BLEU[3] is a widely used translation quality metric.

Legend	Context	BLEU
Source	inu/N ga/SUBJ-marker naku/V .	-
1st best	A dog cries.	50.8
2nd best	A dog barks.	100.0
3rd best	Dog weeps.	38.5
Reference	A dog barks.	100.0

Table 2: BLEU scores of Translations Distributed to Each Reader (when "1st best" candidate is distributed to all four readers)

	Candidate to be Distributed	BLEU
Reader1	1st	50.8
Reader2	1st	50.8
Reader3	1st	50.8
Reader4	1st	50.8
Average	-	50.8

Table 3: BLUE scores of Translations Distributed to Each Reader (when "2nd best" candidate is distributed to one reader and "1st best" is distributed to the other three readers)

	Candidate to be Distributed	BLEU
Reader1	1st	50.8
Reader2	2nd	100.0
Reader3	1st	50.8
Reader4	1st	50.8
Average	-	63.1

proposed method distributes t_i^1 to two readers and t_i^2 to one reader. In this example, if the quality of t_i^1 is lower than that of t_i^2, the average quality of the three translations distributed to the three readers is improved.

We explain our motivation using the example in Table 1. In this example, we want to translate the Japanese sentence "inu ga naku" (A dog barks.) to English. Here "inu" translates as dog or dogs, and "ga" is a subject marker that does not need to be translated. Translating the verb "naku" is problematic because it is ambiguous in English; "naku" means to cry, to bark, and to weep.

Suppose an SMT system translates this Japanese source sentence to English and that the top three translations are those shown in Table 1. Moreover, suppose there are *four* readers. If we distribute the "1st best" candidate in Table 1 to all four readers, the baseline average BLEU [3] score, a widely used metric for translation quality, is 50.8 (Table 2). Because we rely only on the "1st best" candidate, if this candidate's quality is low, the translation quality will be affected.

Table 4: BLUE scores of Translations Distributed to Each Reader (when "3rd best" candidate is distributed to one reader and "1st best" is distributed to the other three readers.)

	Candidate to be Distributed	BLEU
Reader1	1st	50.8
Reader2	1st	50.8
Reader3	1st	50.8
Reader4	3rd	38.5
Average	-	47.7

In contrast, considering the risk that the SMT system may fail to identify the actual best translation, we can distribute other candidates to a small number of readers. For example, as shown in Table 3, if we distribute the "2nd best" translation to one reader randomly, we can achieve an average BLEU score of 63.1, which is a great improvement compared to distributing the "1st best" candidate to all readers.

However, avoiding the risks associated with SMT systems in this manner does not always achieve good results. For example, as can be seen in Table 4, if we distribute the "3rd best" translation to one reader randomly, the average BLEU score is 47.7, which is less than the baseline average BLEU score (50.8; Table 2).

Thus, to improve performance in averaged quality, we need to 1) estimate (predict) quality of candidates accurately without a reference translation, and 2) optimize and determine the risks associated with considering both successful and unsuccessful cases.

We conducted simulation experiments to evaluate the proposed approach. In these simulation experiments, an SMT system distributes translations to $1,000$ readers. We found that the proposed approach consistently and significantly outperform the previous approach.

The contributions of this study are summarized as follows.

- We propose an approach for distributing translation candidates to readers when documents with many readers such as web documents and news articles are translated.

- Our key idea is to use all translation candidates rather than using only the top candidate by considering the possibility that the top ranked candidate is not actually the best translation.

- Our experimental results show that the proposed approach consistently outperforms the baseline approach in which only the top candidate is distributed to all readers.

The remainder of this paper is organized as follows. Section 2 differentiates our task from previous studies. Section 3 describes how to estimate quality considering risks. Section 4 explains the key idea of the proposed approach: how to use the estimated quality and its risk to distribute translations. Section 5 describes the experimental settings. Section 6 and Section 7 present quantitative and qualitative results, respectively. A discussion is presented in Section 8, and the paper is concluded in Section 9.

2. Related Work

Our approach is closely related to a quality estimation (QE) task. In this approach, the QE task estimates the quality of a given source text and its translation without a reference translation [4, 5]. From a machine learning perspective, a QE task is generally categorized as a regression problem [6]. A regression problem differs from typical classification problems, such as those that apply support vector machine (SVM) techniques, in that it tries to predict real values while the latter tries to predict classes. Many regression algorithms have been applied to QE tasks, e.g., SVM-based regression [7] and Gaussian process (GP) regression [8, 6, 9].

In addition to predicted scores, a GP can output their variances [10]; however, SVM-based regression algorithms can only output predicted scores and cannot output their variances. More precisely, SVMs can output confidence values; however, such values cannot be interpreted as variances. Although GPs can output variances, most QE systems that use a GP only use the predicted scores.

QE tasks can also be categorized by the source text unit used to estimate quality: words, sentences, or documents. This study uses sentences because they are the most widely used and studied [7]. However, the proposed approach is also applicable to words or documents. To use other types of source text units, we simply switch sentences in Figure 1 to another unit type.

Our task is also related to another previous approach, i.e., system combination [11, 12]. Given single best translations from multiple SMT systems, system combination techniques attempt to output a more sophisticated single translation by combining the given translations. Like the system combination approach, the proposed approach deals with multiple translations for a given source text.

However, the proposed task clearly differs from system combination in both objective and outputs. The objective of the proposed task is to distribute given translations to readers considering the risk in translation quality. In contrast, the goal of the system combination approach is to refine translations. In the proposed task, a translation distributed to a reader is one of the input translations. In contrast, the translation output by a system combination technique can be very different from the input translations because its objective is to refine translations.

The system combination approach and the proposed approached can be aggregated to create a new system. Given a source text, suppose a system-combination system can output *multiple* sophisticated translations rather than a single best translation. Then, the proposed approach can input the sophisticated translations and distribute them to readers. Note that, for simplicity, we do not focus on this aggregated system; however, being able to create an aggregated system implies that our task is independent of the system combination tasks.

Re-ranking candidates to find the best translation candidate has been addressed in a previous study [13]. However, unlike our goal, this study does not aim to distribute translation candidates.

3. Gaussian Process-based Quality Estimation

Here we explain how to estimate the quality of given translations considering risks in quality. As described in Section 2, we use a GP to estimate quality and its risk simultaneously because a GP can output variance in addition to quality, and this variance encodes the quality's risk.

We introduce the notations used to explain the GP. Our notations are based on a previously QE study that used a GP [6]; however, this study used a GP for multitask learning, a purpose very different from ours.

We model the proposed task as a regression problem where the training data is given as M pair $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}$. Here $\mathbf{x}_i \in \mathbb{R}^d$ denotes a d-dimensional feature vector constructed from a pair of source sentences and its translation. $\mathbf{x}_i \in \mathbb{R}^d$ encodes linguistic features taken from the pair. $y_i \in \mathbb{R}$ is a response variable, which is the gold standard in regression problems. It numerically encodes the translation quality, i.e., how good the translation is for the source sentence in the i-th source sentence-translation pair. For y_i in QE, typically, a manual quality assessment such as post-editing time or a Likert score is used. However, to the best of our knowledge, no dataset with manually assessed quality for N-best output of an SMT system exists. Therefore, we have used sentence BLEU scores implemented in the Moses [2] toolkit [2].

The goal of the GP is to predict y_* for an unseen test sample $\mathbf{x}_*$ given the training data $\mathcal{D}$. The GP performs this prediction by integrating over a functional space as follows. Intuitively, this means that all possible regressor functions f within the functional space are considered in the GP.

$$p(y_*|\mathbf{x}_*, \mathcal{D}) = \int_f p(y_*|\mathbf{x}_*, f)p(f|\mathcal{D}) \tag{1}$$

In (1), function f is defined as follows.

$$f(\mathbf{x}) \sim \mathcal{GP}(\mathbf{0}, k(\mathbf{x}, \mathbf{x}')) \tag{2}$$

[2] http://www.statmt.org/moses/

(2) has two parameters. The first is the mean function $\mathbf{0}$, which simply implies that the function f is normalized to 0. The key component in (2) is k, a *covariance kernel function*, which intuitively encodes the closeness of $\mathbf{x}$ and $\mathbf{x}'$.

A typical covariance kernel function is a radial basis function (RBF), which is expressed as follows[3].

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{x}')^\top A^{-1}(\mathbf{x}-\mathbf{x}')\right) \quad (3)$$

There are two hyperparameters in (3), σ_f and A. σ_f is a scalar that determines the overall size of the variances. $A = \mathrm{diag}(\mathbf{a})$ is a diagonal matrix that determines the weight of each feature; the importance of the i-th feature increases as a_i increases. Typically, $\mathbf{a}$ is defined as $\mathbf{a} = \sigma_\ell^2 \mathbf{1}$ where $\mathbf{1}$ is a vector of appropriate size whose elements are all 1 and σ_ℓ is a hyperparameter. In this definition, the importance of all features is equal and hyper-parameter σ_ℓ tunes the kernel's sensitivity to feature values. This definition is also advantageous in that σ_ℓ can automatically be tuned only using the training data [10]. We use this definition in our experiments.

3.1. Prediction of a single unseen datum

An advantage of the GP is that we do not need to perform numerical integration to calculate (1). Given the characteristics of Gaussian functions, y_* in (1) can be obtained analytically as follows where $\mathcal{N}$ denotes the *Gaussian (Normal) probability distribution.*

$$y_* \sim \mathcal{N}\left(\mathbf{k}_*^\top (K+\sigma_n^2 I)^{-1}\mathbf{y}, k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^\top (K+\sigma_n^2 I)^{-1}\mathbf{k}_*\right) \quad (4)$$

In (4), $\mathbf{y} = (y_1, \ldots, y_M)$, $\mathbf{k}_* = (k(\mathbf{x}_*, \mathbf{x}_1), k(\mathbf{x}_*, \mathbf{x}_2), \ldots, k(\mathbf{x}_*, \mathbf{x}_M))^\top$, and K is an $M \times M$ matrix whose i, j element is defined as $K_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$.

In summary, given an unseen test sample $\mathbf{x}_*$, we can obtain its prediction using (4).

The GP is also advantageous in that hyperparameter optimization is computationally easy because of the use of the Gaussian function. To this point, we have the following hyperparameters: σ_f, σ_n, and $\mathbf{a}$. These hyperparameters can be tuned automatically so that the likelihood of $\mathcal{D}$ can be maximized.

3.2. Prediction of multiple unseen data

Section 3.1 discussed the prediction of a single unseen data $\mathbf{x}_*$. When n multiple unseen data, e.g., $\mathbf{x}_{*1}, \mathbf{x}_{*2}, \ldots, \mathbf{x}_{*n}$, the GP considers not only the closeness between each unseen data point and the training data but also the closeness between each unseen data point. In this case, the prediction can be written as follows.

$$\mathbf{y}_* \sim \mathcal{N}(\mu, \Sigma) \quad (5)$$

Here $\mu = (\mu_1, \ldots, \mu_n)^\top$ and Σ are the quality prediction and its covariance matrix, respectively. These play a key role and are used in the subsequent distribution process. They can be calculated analytically as follows.

$$\begin{aligned} \mu &= K_*(K+\sigma_n^2 I)^{-1}\mathbf{y} && (6) \\ \Sigma &= (K_{**}+\sigma_n^2 I) - K_*(K+\sigma_n^2 I)^{-1}K_*^\top && (7) \end{aligned}$$

Here K_* is an $n \times M$ matrix whose i, j-th element is defined as $(K_*)_{i,j} = k(\mathbf{x}_{*i}, \mathbf{x}_j)$, and K_{**} is an $n \times n$ matrix whose i, j-th element is defined as $(K_{**})_{i,j} = k(\mathbf{x}_{*i}, \mathbf{x}_{*j})$.

In summary, given multiple unseen data points $\mathbf{x}_{*1}, \ldots, \mathbf{x}_{*n}$ as input, the GP outputs quality predictions in the form of a vector, $\mu = (\mu_1, \ldots, \mu_n)^\top$, and the (co-)variance matrix between the predicted values, Σ. Intuitively, the diagonal element of Σ, i.e., i, i-th element, encodes the risk or uncertainty of the prediction of the i-th unseen input. In addition, the nondiagonal element of Σ, i.e., the i, j-th element where $i \neq j$, encodes how uncertain the i-th prediction is when the j-th prediction is uncertain (and vice versa).

The theoretical background of the GP has been addressed in [10] . For implementation, we used the GPy toolkit [4], a GP library for the Python language.

4. Risk-aware Distribution of Translation Candidates

This section explains the key idea of the proposed approach: how the proposed system distributes translation candidates to readers considering the risk in translation quality. Assume that an SMT system outputs n-best translations for a source sentence. Here let $\mathbf{x}_{*1}, \mathbf{x}_{*2}, \ldots, \mathbf{x}_{*n}$ be the feature vectors constructed from the source sentence and the n-best translations. As explained in (3.2), given $\mathbf{x}_{*1}, \ldots, \mathbf{x}_{*n}$ as input, the GP outputs the predicted quality $\mu = (\mu_1, \ldots, \mu_n)^\top$ and the covariance matrix of the prediction Σ, which can be interpreted as the risk encoding how inaccurate the predicted quality might be.

Given μ and Σ, our goal is to calculate the *rate vector* $\lambda = (\lambda_1, \ldots, \lambda_n)^\top$ where each λ_i is the probability that i-th best translation is selected and distributed to a reader. In other words, λ_i determines what percentage of the entire readers should read the i-th best translation. This can be formally expressed as $\sum_{i=1}^n \lambda_i = 1$, and for each $i \in \{1, \ldots, n\}$, $\lambda_i \geq 0$.

The rate vector can be calculated by optimization using the following formula.

$$\begin{aligned} \text{maximize}_{\lambda_1,\ldots,\lambda_n} \quad & \sum_{i=1}^n \lambda_i \mu_i - \frac{1}{2}\alpha \sum_{i=1}^n \sum_{j=1}^n \lambda_i \lambda_j (\Sigma)_{i,j} && (8) \\ \text{subject to} \quad & \sum_{i=1}^n \lambda_i = 1 && (9) \\ & \forall i \in \{1, \ldots, n\}, \lambda_i \geq 0 && (10) \end{aligned}$$

[3] $\top$ denotes the transpose of a vector or a matrix.

[4] http://sheffieldml.github.io/GPy/

In (8), the objective function, i.e., the first term, attempts to maximize the predicted quality averaged over n candidates. In contrast, the second term penalizes the first term when the risk of the quality is large. Thus, (8) can be intuitively interpreted as maximizing the averaged predicted quality while penalizing the candidate whose risk is large. (8) has a hyperparameter, i.e., α, that tunes the strength of the risk penalization.

As explained, the constraints (9) and (10) guarantee that λ is always a probability vector whose elements can be interpreted as probability mass.

Notably, (8) includes the case wherein the predicted best translation is distributed to all readers. This case arises when we set α to 0. In this case, only the first term remains in (8). Because of the constraints (9) and (10), λ remains a probability vector in this case. Because of the first term that maximizes the quality, λ becomes a unit vector such that the i-th element with the highest μ_i value is set to 1 and all other elements are set to 0.

The solution of (8) can be obtained in practical time. Theoretically, (8) can be solved using linear-constrained convex optimization techniques, which obtain a global optimum. Moreover, through preliminary experiments, we found that we could find the solution in practical time. We were able to achieve good performance in average translation quality with small n, e.g., 5 and 3. In contrast, large n values degrade performance. This is presumably because n is the number of n-best translations output from SMT systems, and we rerank these outputs. Thus, n values that are too large increase the number of low-quality candidates and makes it difficult to determine good candidates.

After calculating λ, according to this vector, the proposed system distributes candidate translation to the readers.

5. Experiment Setup

We performed our experiments under two settings, i.e., a system selection setting and an n-best output setting. The n-best output setting is identical to what we have explained so far. Under the system selection setting, we use the n single-best outputs from n SMT systems as input rather than the n-best outputs of an SMT system. The system is required to distribute the n single best outputs to readers.

In both system selection and n-best output settings, we have simulated a case wherein translations are distributed to $1,000$ readers. In both settings, five-fold cross validation was performed. To extract features from the source text and translations, we used a standard QE system, *QuEST* [5].

For features, we used the basic 17 feature set defined in the literature [5]. Here, LM denotes a language model.

- Number of tokens in the source sentence
- Number of tokens in the target sentence
- Average source token length
- LM probability of source sentence
- LM probability of target sentence
- Number of occurrences of the target word within the target hypothesis
- Average number of translations per source word in the sentence
- Average number of translations per source word in the sentence weighted by the inverse frequency of each word in the source corpus
- Percentage of unigrams in quartile 1 of frequency, i.e., lower frequency words, in a corpus of the source language
- Percentage of unigrams in quartile 4 of frequency, i.e., higher frequency words, in a corpus of the source sentence
- Percentage of bigrams in quartile 1 of frequency of source words in a corpus of the source language
- Percentage of bigrams in quartile 4 of frequency of source words in a corpus of the source language
- Percentage of trigrams in quartile 1 of frequency of source words in a corpus of the source language
- Percentage of trigrams in quartile 4 of frequency of source words in a corpus of the source language
- Percentage of unigrams in the source sentence seen in a corpus
- Number of punctuation marks in the source sentence
- Number of punctuation marks in the target sentence

6. Quantitative Evaluation

6.1. Evaluation under system selection setting

For the system selection setting, we used the dataset from the system selection competition provided by WMT-13 quality estimation shared tasks [6]. This dataset uses an English-to-Spanish translation setting. Here we have five single best Spanish translations from five systems for an English source sentence. The proposed systems distribute these five Spanish translations to the readers.

In this dataset, through manual evaluation, it is known that the "online-B" system achieves the best translation quality. Thus, as a baseline, we considered a case wherein translation by the "online-B" system is given to all readers. In this dataset, 39.51% of translation by the "online-B" system were the actual best.

6.2. Compared methods

We also experimented with other methods for comparison. QE-max is a case where the best candidate with regard to QE score is given to all readers. Support vector regression-radial-basis function (SVR-RBF) is identical to QE-max, except that the quality prediction is calculated using SVR, a regression method based on a SVM, with an RBF kernel.

[5] http://staffwww.dcs.shef.ac.uk/people/L.Specia/projects/quest.html

[6] http://statmt.org/wmt13/quality-estimation-task.html

Table 5: Evaluation under System Selection Setting (values are sentence-level BLEU scores

Proposed	35.52
QE-max	35.43
SVR-RBF	34.98
Baseline	34.88

6.3. Hyperparameter tuning

Essentially we chose hyperparameters from nine points ranging in the log-space from 10^{-3} to 10^{3}: $10^{-12/4}, 10^{-9/4}, 10^{-6/4}, 10^{-3/4}, 10^{0}, 10^{3/4}, 10^{6/4}, 10^{9/4}$, and $10^{12/4}$.

For the GP, we used automatic tuning of hyperparameters with the training data [10], which is implemented in the GPy toolkit. Thus, the only hyperparameter that we tuned was α (Section 4), which tunes the strength of the risk penalization.

SVM-based regression with an RBF kernel has hyperparameters, i.e., C and γ. We chose C from these parameters. We fixed γ to 1 in this experiment.

6.4. Evaluation metric

Unlike previous studies, our objective is to improve the total quality of translations distributed to readers rather than improve the quality of the single best translation. Since previous studies did not focus on the number of readers, to the best of our knowledge, no previous evaluation metric specific to this situation has been proposed. This is problematic because previous evaluation metrics were not designed to take multiple translations as input although they are designed to handle multiple references.

For the evaluation, we simply interpreted the average of the quality scores passed to each reader as the metric for our evaluation. Even though no metric has been previously proposed for many readers, evaluation metrics for a single best translation have been studied extensively. We can evaluate the quality of the translation passed to one reader using an evaluation metric for a single best translation. By considering previously proposed metrics for a single best translation as metrics for a reader, it is natural to define the total quality of all readers as the quality score averaged over all readers. Moreover, these metrics for a single best translation have been tested extensively [3]. Therefore, we can leverage previous knowledge about these measures when analyzing our results.

For the actual evaluation metric for a reader, we have used sentence-level BLEU [3], because it is widely used for automatic evaluation when reference translations are available. For the implementation of sentence-BLEU, we used the "sentence-bleu" command bundled with the Moses toolkit.

6.5. Results

Table 5 shows our results. As can be seen, **Proposed** achieved the best results. We have also confirmed that **Proposed** significantly outperforms **Baseline**.

We also performed a Wilcoxon significance test for these results. As a result, **Proposed** was statistically significant against the **Baseline** ($p < 0.01$) and **QE-max** ($p < 0.01$).

Table 6: Evaluation under n-best Setting

Proposed	26.24
QE-max	26.06
Baseline	26.06

6.6. Evaluation under n-best setting

Here we evaluate the proposed approach in the n-best setting where n-best outputs from one SMT system are distributed to readers. For the SMT system, we used the English-to-Spanish translation setting so that we could use the same feature set as the system selection setting.

In this evaluation, we used the News Commentary corpus [7] so that the choice of corpus matches our task's target, i.e., web documents and news. The News Commentary corpus is a parallel corpus that comprises "news text and commentaries from the Project Syndicate." This corpus is provided as a part of the corpora for the series of WMT translation shared tasks.

We used the Moses toolkit trained with the News Commentary corpus as the SMT translator in our task. As usual for SMT evaluation, Minimum Error Rate Training (MERT) [14] was used to train the SMT translator. We used the same language pair, i.e., English-to-Spanish, for this evaluation, because a well-studied feature extractor for QE is provided for this language pair.

We set $n = 5$ in this experiment because, through a preliminary experiment, we found that it is quite rare for candidates ranked below fifth to be the actual best candidate. Indeed, in this experiment, only 34.23% of the first-ranked candidate was the actual best. The values for the second, third, fourth, and fifth ranked candidates were 21.31%, 17.05%, 13.92%, and 13.49%, respectively. The definitions of Baseline, QE-max, and Proposed are the same as those in Section 6.2.

Table 6 shows the results. Again, the proposed method clearly outperforms the other three methods. We also found statistical significance between **Baseline** and **Proposed** ($p < 0.01$).

7. Qualitative Evaluation by Examples

This section explains how the proposed method works successfully by demonstrating examples taken from **Proposed** in Section 6.6.

[7] http://www.statmt.org/wmt13/translation-task.html#download

Table 7: Two-top Example (the first two among the 5-best outputs are significantly better than latter cases)

Legend	Content	Actual BLEU	Predicted BLEU	Rate
Source text	Damascus, however , also brushed off this proposal .	-	-	-
1st best	Damasco , sin embargo , tambin desdeñó los esta propuesta .	23.46	27.74	0.45
2nd best	Damasco , sin embargo , tambin descartaron de esta propuesta .	23.46	27.74	0.55
3rd best	Damasco , sin embargo , tambin desdeñó los esa propuesta .	17.03	27.43	$< 10^{-6}$
4th best	Damasco , sin embargo , tambin desdeñó los de esta propuesta .	21.40	27.27	$< 10^{-6}$
5th best	Damasco , sin embargo , tambin los desdeñó los esta propuesta .	21.40	27.16	$< 10^{-6}$
Reference	entretanto , Damaskus critica tambin esta propuesta .	-	-	-

As mentioned previously, Table 7 shows the first example, which we call the "Two-top example."

By focusing on the first two elements in the **Actual BLEU** scores column, we can see that the actual BLEU scores of these elements are equal and are the highest among the five output translations. Since we cannot know the actual BLEU scores in advance, distributing only the "1st best" translation to all readers is risky because the "2nd best" might have a higher BLEU score. Thus, correctly recognizing these equal scores is crucial for handling this example.

The **Predicted BLEU** column shows the predicted BLEU scores obtained by GP-based quality estimation, i.e., the elements of the vector μ (Section 3). Comparing the predicted and actual BLEU scores, we find that the predicted values are not particularly accurate. The actual BLEU scores for all five examples are < 24; however, all of the predicted scores are > 27. The reason for this is presumably because the reference translation in this example is structurally different from the source text and its translation candidates, i.e., "however" in the source sentence is placed in the middle of the sentence as an adverb, and in the reference translation, the conjunction "entretanto" (meanwhile) is used instead and is placed at the beginning of the sentence. This result clearly demonstrates the difficulty of accurately estimating an exact value for the BLEU scores. Although actual BLEU scores depend on the reference translations, in QE, we must estimate the scores without reference translations.

Although the **Predicted BLEU** scores in Table 7 are not accurate as a regression problem, these scores successfully capture the overall characteristics in the order of the candidates with regard to their quality in this example. The first two are significantly better than the rest. Thus, we can see that the **Predicted BLEU** scores can be leveraged if we use the scores intelligently.

In the fifth column, the **Rate** vector, which we define in Section 4, successfully captures the basic characteristics of the five candidates because of the use of the (co-)variance matrix. The first two candidates consume nearly all of the weights that are to be sum up to 1.0. The rates for the latter three candidates are $< 10^{-6}$, which indicates that these candidates are almost ignored and are essentially never distributed to readers. This reflects the fact that the two top candidates are by far better than the latter candidates. We also find that the probability allocated to the first two candidates is close to 0.5. This implies that our risk-aware distribution system successfully recognizes that the first two candidates are scored equally, and this decision is reflected in the rate vector.

In summary, these experimental results show that our distribution system correctly recognizes that the first two candidates are significantly better than the latter cadidates and that they are scored equally. Thus, our system distributes the first two translations considering the case in which the second best translation would actually be better than the first. In this example, since the actual BLEU scores of the first two candidates are equal, the quality is not improved compared to the case wherein the "1st best" is distributed to all readers. However, if the actual BLEU score of "1st best" was even slightly less than that of "2nd best," our approach would have successfully outperformed the baseline.

8. Discussion

The optimization problem used to determine the rate of distribution introduced in Section 4 is a type of *multi-objective optimization*. In multi-objective optimization, there are multiple objective functions to optimize, and the goal is to optimize the functions simultaneously. In our application, we simultaneously maximize the predicted quality of the translations distributed to readers while minimizing risks. This use of multi-objective optimization is based on modern-portfolio theory, where the goal is to maximize financial profit rather than translation quality [15]. However, our task is more than a simple application of modern-portfolio theory in that we cannot directly measure the objective function and its variances, whereas these are assumed to be directly observable in modern portfolio theory. This unavailability of direct measurement of the objective function and its variances is the reason why we predict it from the training data using GP-based QE (Section 3).

Unlike our task, previous use of multi-objective optimization in machine translation studies appears limited to simultaneously optimizing multiple evaluation metrics. A previous study [16] used multi-objective optimization to optimize multiple automatic evaluation metrics simultaneously, i.e., BLEU and RIBES [17]. Another study used multi-objective optimization to optimize document-level evaluation metrics and sentence-level evaluation metrics [18]. In computational linguistics, other than machine translation tasks, multi-objective optimization was recently used in joint disambiguation of nouns and named entities [19].

9. Conclusion

In this paper, we have proposed an approach for distributing translation candidates to readers for translated documents with many anonymous readers, such as web documents and news articles. Our key idea is to use all translation candidates rather than the top candidate in consideration of the risk that the top candidate actually has lower quality than other candidates. Our experimental results show that the proposed approach consistently outperforms the baseline approach wherein the top candidate is distributed to all readers.

In future, we would like to test the proposed approach with other language pairs.

10. References

[1] P. Koehn, *Statistical machine translation*. Cambridge University Press, 2009.

[2] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, *et al.*, "Moses: Open source toolkit for statistical machine translation," in *Proc. of ACL interactive poster and demonstration sessions*, 2007, pp. 177–180.

[3] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proc. of ACL*, Philadelphia, Pennsylvania, USA, July 2002, pp. 311–318.

[4] J. Blatz, E. Fitzgerald, G. Foster, S. Gandrabur, C. Goutte, A. Kulesza, A. Sanchis, and N. Ueffing, "Confidence estimation for machine translation," in *Proc. of COLING*, Geneva, Switzerland, Aug 23–Aug 27 2004, pp. 315–321.

[5] L. Specia, N. Cancedda, M. Dymetman, M. Turchi, and N. Cristianini, "Estimating the sentence-level quality of machine translation systems," in *Proc. of EAMT*, 2009, pp. 28–37.

[6] T. Cohn and L. Specia, "Modelling annotator bias with multi-task gaussian processes: An application to machine translation quality estimation," in *Proc. of ACL*, Sofia, Bulgaria, August 2013, pp. 32–42.

[7] O. Bojar, R. Chatterjee, C. Federmann, B. Haddow, M. Huck, C. Hokamp, P. Koehn, V. Logacheva, C. Monz, M. Negri, M. Post, C. Scarton, L. Specia, and M. Turchi, "Findings of the 2015 workshop on statistical machine translation," in *Proc. of WMT*, Lisbon, Portugal, September 2015, pp. 1–46.

[8] D. Beck, K. Shah, T. Cohn, and L. Specia, "SHEF-Lite: When less is more for translation quality estimation," in *Proc. of WMT*, Sofia, Bulgaria, August 2013, pp. 337–342.

[9] D. Beck, K. Shah, and L. Specia, "Shef-lite 2.0: Sparse multi-task gaussian processes for translation quality estimation," in *Proc. of WMT*, Baltimore, Maryland, USA, June 2014, pp. 307–312.

[10] C. Williams and C. Rasmussen, *Gaussian processes for machine learning*. MIT Press, 2006.

[11] O. Bojar, M. Ercegovčević, M. Popel, and O. Zaidan, "A grain of salt for the wmt manual evaluation," in *Proc. of WMT*, Edinburgh, Scotland, July 2011, pp. 1–11.

[12] K. Heafield and A. Lavie, "Cmu system combination in wmt 2011," in *Proc. of WMT*, Edinburgh, Scotland, July 2011, pp. 145–151.

[13] S. Kumar and W. Byrne, "Minimum bayes-risk decoding for statistical machine translation," in *Proc. of HLT-NAACL*, D. M. Susan Dumais and S. Roukos, Eds. Boston, Massachusetts, USA: Association for Computational Linguistics, May 2 - May 7 2004, pp. 169–176.

[14] F. J. Och, "Minimum error rate training in statistical machine translation," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, 2003, pp. 160–167.

[15] H. Markowitz, "Portfolio selection*," *The journal of finance*, vol. 7, no. 1, pp. 77–91, 1952.

[16] K. Duh, K. Sudoh, X. Wu, H. Tsukada, and M. Nagata, "Learning to translate with multiple objectives," in *Proc. of ACL*, Jeju Island, Korea, July 2012, pp. 1–10.

[17] H. Isozaki, T. Hirao, K. Duh, K. Sudoh, and H. Tsukada, "Automatic evaluation of translation quality for distant language pairs," in *Proc. of EMNLP*, Cambridge, MA, October 2010, pp. 944–952.

[18] C. Ding, M. Utiyama, and E. Sumita, "Document-level re-ranking with soft lexical and semantic features for statistical machine translation," in *Proc. of AMTA*, 2014.

[19] D. Weissenborn, L. Hennig, F. Xu, and H. Uszkoreit, "Multi-objective optimization for the joint disambiguation of nouns and named entities," in *Proc. of ACL-IJCNLP*, Beijing, China, July 2015, pp. 596–605.