Using Language and Translation Models to Select the Best among Outputs from Multiple MT systems

Yasuhiro Akiba, Taro Watanabe and **Eiichiro Sumita**
ATR Spoken Language Translation Research Laboratories
2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-0288, Japan
{yasuhiro.akiba, taro.watanabe, eiichiro.sumita}@atr.co.jp

Abstract

This paper addresses the problem of automatically selecting the best among outputs from multiple machine translation (MT) systems. Existing approaches select the output assigned the highest score according to a target language model. In some cases, the existing approaches do not work well. This paper proposes two methods to improve performance. The first method is based on a multiple comparison test and checks whether a score from language and translation models is significantly higher than the others. The second method is based on probability that a translation is not inferior to the others, which is predicted from the above scores. Experimental results show that the proposed methods achieve an improvement of 2 to 6 % in performance.

1 Introduction

This paper addresses the challenging problem of automatically selecting the best among outputs from multiple machine translation (MT) systems (Figure 1). In combinations of multiple MT systems, some component MT systems can translate a source sentence well while others cannot well. In such a case, correct selection of the best can obviously boost performance.

ATR has been developing such multiple MT systems, including three Japanese-to-English (J-E) MT systems: TDMT (Furuse and Iida, 1996), D3 (Sumita, 2001), and SMT (Watanabe et al., 2002), and three English-to-Japanese (E-J) MT systems: TDMT (Furuse and Iida, 1996), HPAT (Imamura, 2002), and SMT (Watanabe et al., 2002). In order to evaluate each MT system, the MT outputs were manually assigned one of four ranks[1], A, B, C, and D, by native speakers of the target language. The ideal selection for J-E MT systems is the highest-ranked outputs from the three J-E MT systems: TDMT, D3, and SMT. The ideal selection for E-J MT systems is the highest-ranked outputs from the three E-J MT systems: TDMT, HPAT, and SMT.

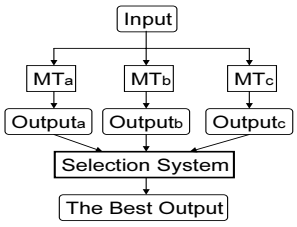

Figure 1: The selection system.

Figure 2 shows the individual performances of the three J-E MT systems and the ideal selection system derived from their combination. Figure 3 shows the individual performances of the three E-J MT systems and the ideal selection system derived from their combination. The left-hand group of bars indicates the ra-

[1]The four ranks are defined as follows: (A) Perfect: no problem in either information or grammar; (B) Fair: easy to understand, with either some unimportant information missing or flawed grammar; (C) Acceptable: broken, but understandable with effort; (D) Nonsense: important information has been translated incorrectly.

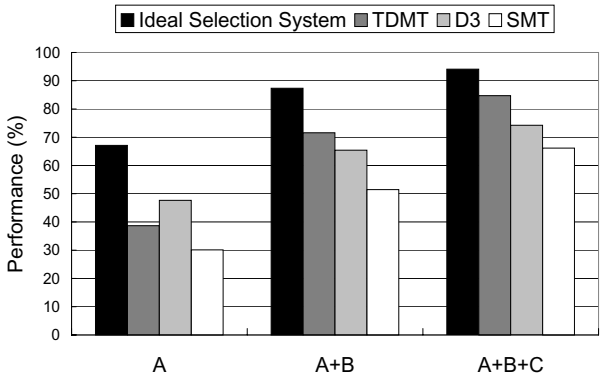

Figure 2: Performance of the ideal selection system for Japanese-to-English MT outputs.

tio of the number of sentences ranked as A to the total number of sentences translated by each MT system (hereafter, performance for Rank A). The middle group of bars indicates the ratio of the number of sentences ranked as A or B to the total number of sentences translated by each MT system (hereafter, performance for Rank A+B). The right-hand group of bars indicates the ratio of the number of sentences ranked as A, B, or C to the total number of sentences translated by each MT system (hereafter, performance for Rank A+B+C). The black bars indicate the performance of the ideal selection system. As Figures 2 and 3 show, the performance of the ideal J-E and E-J selection system is much better than that of each component MT system.

Conventional approaches to the selection problem include methods (Callison-Burch and Flournoy, 2001; Kaki et al., 1999) that automatically select the output assigned the highest probability P(t) (hereafter, LM-score), according to a language model (LM) for the translation target language. As a preliminary experiment, the authors applied this LM-score to selecting the best among the outputs from the three J-E MT systems. In order to make a comparison, the authors also used a score based on a translation model (TM) called IBM4 (Brown et al., 1993) (hereafter, TM-score) and a score based on the product of the TM-score and the LM-score (hereafter, TM∗LM-score) to select the best output. Table 1 shows the results of this preliminary experiment. The floating number indicates the difference between the performance for Rank A of each selection system and that of D3 (the best MT system, i.e., with the highest performance for Rank A). The LM-score and TM-score based selections did not

Table 1: Difference in performance for Rank A between each selection system and D3.

Scoring method	TM∗LM	TM	LM
Difference in performance	**4.1**	-1.5	**-0.5**

boost/improve the performance for Rank A, whereas the TM∗LM-score did. The preliminary experiment appears to indicate that the TM∗LM-score works better than the LM-score in selecting the best output.

As can be easily guessed, the scores from language model, translation model, or both models combined has two problems. The first problem is that TM∗LM-score, TM-score, and LM-score are statistical variables. Even in the case that TM and LM are trained on a corpus of the same size, changing the training corpus also changes the TM-score, the LM-score, and the TM∗LM-score. Figure 4 shows this phenomenon. In the figure, TRN1, TRN2, and TRN3 correspond, respectively, to (2), (3), and (4) below, which are translations of (1) by different J-E MT systems.

(1) ϕ *Konputa-no shisutemu enjinia desu*
I-SUBJ computer-of system engineer am
"I'm a computer engineer."

(2) I'm a computer systems engineer. (TRN1)

(3) I'm a computer salesman. (TRN2)

(4) It's computer. (TRN3)

LM and TM were trained in ten ways on training sets, which were subsets of ATR broad-coverage bilingual basic expression (BE) corpus (Takezawa et al., 2002), according to ten-fold cross validation (Mitchell, 1997). For each i (i =

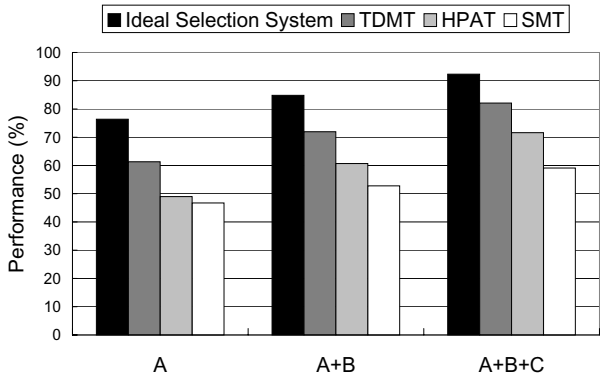

Figure 3: Performance of the ideal selection system for English-to-Japanese MT outputs.

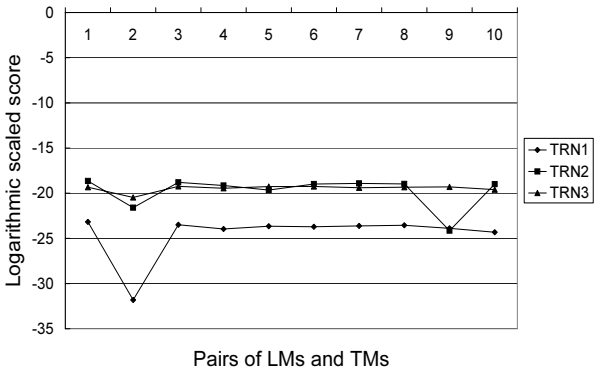

Figure 4: The same translation can be scored differently by TM∗LM-score.

Table 2: The 2×2 confusion matrices for three J-E MT systems: J-E TDMT, D3 and J-E SMT: Each matrix shows agreement and disagreement between the ideal selection and the score-based selection using the TM∗LM-score. 1 and 0 indicates "selected" and "not selected", respectively.

Ideal Selection	TM∗LM-Score-Based Selection					
	TDMT		D3		SMT	
	1	0	1	0	1	0
1	22.7	**44.5**	37.3	**23.7**	31.4	5.5
0	1.0	31.8	7.1	32.0	**36.7**	26.5

1, 2, 3), TRNi is scored in ten ways by TM∗LM-score. Some TM∗LM-scores place TRNi (i = 1, 2, 3) in a different order. Even if a huge corpus is prepared to train a good TM and LM, this phenomenon remains.

In order to solve this first problem, this paper propose a statistical-test-based selection system. Here, the statistical test used is a multiple comparison test based on the Kruskal-Wallis test (Hochberg and Tamhane, 1983). The proposed method checks whether the highest score is significantly different from the others.

The second problem is that the translations with the highest TM∗LM-score tend to differ from those ranked highest by human evaluators. Table 2 shows this phenomenon. The Table consists of three 2×2 confusion matrices for three J-E MT systems: J-E TDMT, D3, and J-E SMT. Each matrix shows agreement and disagreement between the ideal selection by a human evaluator and the selection by the TM∗LM-score. The (1,1)-element and the (0,0)-element indicate the percentage of agreement, and the (1,0)-element and the (0,1)-element indicate the percentage of disagreement. In the confusion matrix for J-E SMT, the number in the (1,0)-element is larger than that in the (0,1)-element. This means that the TM∗LM-score tends to give the highest score to the translation from J-E SMT when it is not the translation assigned the best rank. On the other hand, in the confusion matrices for J-E TDMT and D3, the number in the (0,1)-element is larger than that in the (1,0)-element. This means that the TM∗LM-score tends not to give the highest score to the translation from the MT systems, except for J-E SMT, even if that translation is assigned the best rank.

To solve this second problem, this paper proposes a selection system based on the conditional probability that a translation is not inferior to the other translations when the translation encoded by using the TM∗LM-score, the TM-score, or the LM-score, satisfies some conditions. For each MT system, the conditional probability is learned as a regression tree (Chambers and Hastie, 1992; Breiman et al., 1984) from the vector-encoding of the translations labeled as "not inferior" or "inferior".

The next section presents our two proposed methods. Experimental results are shown and discussed in Section 3. Finally, our conclusions are presented in Section 4.

2 Proposed Method

2.1 Proposed Method (1)

To solve the first problem described in Section 1, this Subsection proposes a method that selects the translation according to whether the scores of outputs from each MT system significantly differ from each other.

In order to detect a significant difference, the proposed method first prepares multiple subsets of the full parallel corpus according to k-fold cross validation (Mitchell, 1997) and trains both TM and LM on each subset. For example, the full parallel corpus C is divided into ten subsets V_i ($i = 0, 1, \cdots, 9$). For each i ($i = 0, 1, \cdots, 9$), the proposed method trains a translation model TM_i on C_i ($= C - V_i$) and a language model LM_i on the target-language part of C_i (Figure 5).

Hereafter, let $L_i(t)$ denote Tri-gram statistics of a translation t by using LM_i. Also, let $T_i(s,t)$ denote $\sum_{a \in \mathcal{S}} P(s,a|t)$, where s is a translation source sentence, t is a translation target sentence, and $\mathcal{S}$ is the alignment set[2] (Brown et

[2]Note that the definition of S is changed depending

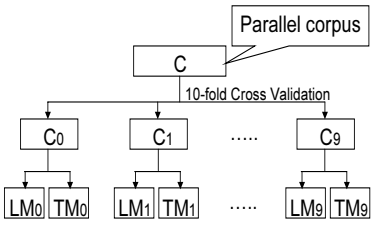

Figure 5: The method to train multiple pairs of LM and TM.

al., 1993) that includes the best alignment, the neighboring alignments, and the pegged alignments.

The proposed method scores each output in k ways. In the example, as shown in Figure 6, the proposed method scores translation t_a from translation system MT_a in ten ways, $L_0(t_a)$, $L_1(t_a)$, $\cdots$, $L_9(t_a)$, when translations are scored by using a language model of the translation target language (Figure 6). On the other hand, the proposed method scores translation t_a in ten ways, $T_0(s,t_a)$, $T_1(s,t_a),\cdots$, $T_9(s,t_a)$, when the translations are scored by using a translation model. Whereas, the proposed method scores translation t_a in ten ways, $T_0(s,t_a) * L_0(t_a)$, $T_1(s,t_a) * L_1(t_a)$, $\cdots$, $T_9(s,t_a) * L_9(t_a)$, when the translations are scored by using the products of the scores of a language model and a translation model.

Then the proposed method compares the means of the scores. In the example (Figure 6), $\sum_{i=0}^{9} L_i(t_a)/10$, $\sum_{i=0}^{9} L_i(t_b)/10$, and $\sum_{i=0}^{9} L_i(t_c)/10$ are compared when the translations are scored by using a language model of the translation target. $\sum_{i=0}^{9} T_i(s,t_a)/10$, $\sum_{i=0}^{9} T_i(s,t_b)/10$, and $\sum_{i=0}^{9} T_i(s,t_c)/10$ are compared when the translations are scored by using a translation model. $\sum_{i=0}^{9} T_i(s,t_a) * L_i(t_a)/10$, $\sum_{i=0}^{9} T_i(s,t_b) * L_i(t_b)/10$, and $\sum_{i=0}^{9} T_i(s,t_c) * L_i(t_c)/10$ are compared when the translations are scored by using the products of the scores of a language model and a translation model.

The proposed method checks whether the highest mean is significantly different from the others by using a multiple comparison test[3] with the Kruskal-Wallis test[4], which is known as a Tukey-Kramer-type modification of the Dunn test (Hochberg and Tamhane, 1983). If the highest mean is significantly different, the proposed method selects the translation with the highest score. If not, the proposed method selects from among the translations whose scores are not significantly different from the highest score the translation from the MT whose performance is the best.

2.2 Proposed Method (2)

To solve the second problem described in Section 1, this Subsection proposes a selection system based on the conditional probability that a translation is not inferior to other translations when the translation encoded by using the TM∗LM-score, the TM-score, or the LM-score satisfies some conditions. For each MT system, the conditional probability is learned as a regression tree from the vector-encoding of the translations labeled as "not inferior" or "inferior", which is the criterion variable.

In order to learn the conditional probability mentioned above, translations from the component translation systems, MT_a, MT_b, and MT_c, are ranked by human evaluators in advance (Figure 7). Let r_a denote the rank assigned to translation t_a from MT system MT_a. Also, let r_{best} denote the best rank among r_a, r_b,

on the TM-training algorithms used.

[3] It is well known that repeating a simple t-test multiple times increases the chance of incorrectly finding a significant difference. Multiple comparison is designed to avoid such a phenomenon.

[4] The Kruskal-Wallis test is a non-parametric one-way Analysis of Variance (ANOVA). This test does not assume the data distribution.

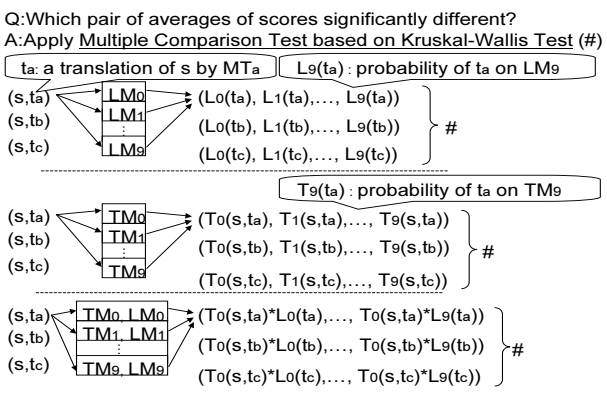

Figure 6: Preparation for multiple comparison test based on Kruskal-Wallis test.

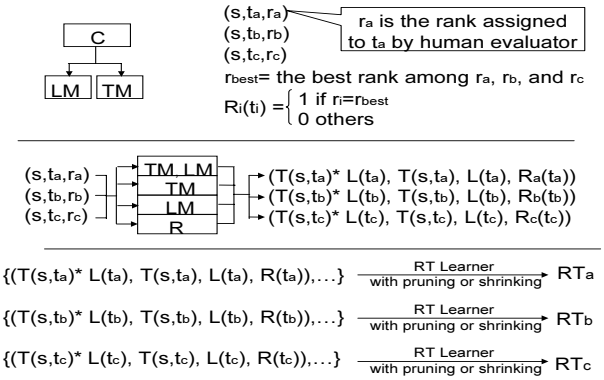

Figure 7: Preparation for regression tree learning.

and r_c. Let $R_i(t_i)$ be defined as follows: $R_i(t_i)$ is equal to 1 if $r_i = r_{best}$; otherwise $R_i(t_i)$ is equal to 0. Therefore when $R_a(t_a)$ is equal to 1, the transaction t_a is superior to or as good as the other translations.

The proposed method trains a translation model TM on the full parallel corpus C and a language model LM on the translation-target-language part of C (Figure 7).

Hereafter, let $L(t)$ denote Tri-gram statistics of a translation t by using LM. Also, let $T(s,t)$ denote $\sum_{a \in \mathcal{S}} P(s,a|t)$, where s is a translation source sentence, t is a translation target sentence, and $\mathcal{S}$ is the alignment set (Brown et al., 1993).

Next, the proposed method encodes three vectors $(s,\ t_a, r_a)$, $(s,\ t_b,\ r_b)$, and $(s,\ t_c,\ r_c)$ (Figure 7) into three score-vectors with non-inferiority or inferiority, respectively: $(T(s,t_a) * L(t_a),\ T(s,t_a),\ L(t_a), R_a(t_a))$, $(T(s,t_b) * L(t_b),\ T(s,t_b),\ L(t_b), R_a(t_b))$, and $(T(s,t_c) * L(t_c),\ T(s,t_c),\ L(t_c), R_c(t_c))$.

For each MT_i $(i = a,\ b,\ or\ c)$, the proposed method learns from $\{(T(s,t_i) * L(t_i),\ T(t_i,s),\ L(t_i),\ R_a(t_i))$ s.t. t_i is a translation of s by MT system $MT_i\}$ the conditional probability, which is expressed by the regression tree (Chambers and Hastie, 1992; Breiman et al., 1984) RT_i (Figure 7). Regression tree (RT) learner is known as recursive binary partitioning. In growing a tree, an RT learner recursively splits the training data in each node so as to reduce variance within partitions as much as possible.

In general, the learned RT over-fits the training data. As post-processing, the learned RT is simplified by using two procedures: pruning and shrinking (Chambers and Hastie, 1992; Breiman et al., 1984). Pruning successively snips off the least important splits. The Importance of a rooted subtree is determined by the cost-complexity measure, $D_k(T') = D(T') + k * size(T')$, where $D(T')$ denotes the deviance of the subtree T', $size(T')$ is the number of terminal nodes of T', and k is a cost-complexity parameter. Shrinking reduces the number of effective nodes by shrinking the fitted value of each node towards its parent node. Shrunken fitted values, for a shrinking parameter k, are computed according to the recursion, $\hat{y}(node) = k * \tilde{y}(node) + (1-k) * \hat{y}(parent)$, where $\tilde{y}(node)$ denotes the usual fitted value for a node, $\hat{y}(parent)$ is the shrunken fitted value for the node's parent, and k is a shrinking parameter such that $0 < k < 1$. The parameter k in each of the two procedures is fixed so as to minimize cross-validation estimates of the deviance. Therefore, after growing each RT_i $(i = a,\ b,\ or\ c)$, the proposed method performs one of the simplified procedures of pruning and shrinking.

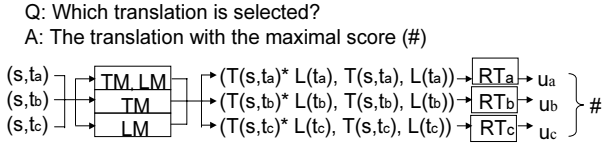

Figure 8: The method to select a translation by using learned regression tree.

In the selection phase, the proposed method encodes three pair of a source sentence and its translation, (s,t_a), (s,t_b), and (s,t_c) into three vectors $(T(s,t_a) * L(t_a),\ T(s,t_a),\ L(t_a))$, $(T(s,t_b) * L(t_b),\ T(s,t_b),\ L(t_b))$, and $(T(s,t_c) * L(t_c),\ T(s,t_c),\ L(t_c))$, respectively (Figure 8).

The proposed method predicts the conditional probability that each t_i (i=a, b, or c) is not inferior to the others by using RT_i and selects[5] the translation with the highest conditional probability.

3 Experimental Comparison

3.1 Experimental Method

The authors evaluated the proposed methods in order to answer the following question: Which selection system improves performance best in comparison with that of the best MT system i.e. the MT systems with the highest performance as shown in Figures 2 and 3?

In order to answer the above question, the authors used a set of three J-E component MT systems (TDMT, D3, and SMT) and a set of three E-J component MT systems (TDMT, HPAT, and SMT). Bilingual English and Japanese data were from ATR broad-coverage bilingual basic expression (BE) corpus (Takezawa et al., 2002), which is split into three parts: a training set of 125,537 sentence pairs, a verification set of 9,872 pairs, and a test set of 10,023 pairs.

The full corpus C in training translation target language model and translation model is the training set. Ten subsets of the full corpus were

[5] Preparing multiple RTs for each component MT system enables the second method to be extended so as to select the best output according to the multiple comparison.

used for the first proposed method. The translation model and language model are learned by using GIZA++ (Och and Ney, 2000) and the CMU-Cambridge Toolkit (Clarkson and Rosenfeld, 1997), respectively. The translation model is learned from IBM 1 to 4, including the HMM model, as suggested by Och and Ney (2000), and its training loop was terminated when the perplexity for the validation set indicated the lowest scores. The word classes used in TM learning are the part-of-speech (POS) classes in TDMT. The P-value used for the multiple comparison test is 0.05.

Four sets of about five hundred pairs of English and Japanese sentences were randomly selected from the test set. The English sentences in the four sets were translated by the E-J component MT systems and ranked by a native speaker of Japanese; likewise the Japanese sentences in the four sets were translated by the J-E component MT systems and ranked by a native speaker of English. Each performance was calculated as the average of the performance over the four sets. In particular, the performance of the second proposed method is calculated according to four-fold cross validation (Mitchell, 1997).

3.2 Experimental Results

In order to evaluate the point mentioned at the beginning of Section 3.1, the authors compared the performance of each selection system with that of the best MT system. As shown in Figure 2, among the J-E component MT systems, D3 had the best performance for Rank A, and TDMT had the best performance for both Rank A+B (equal to or better than B) and Rank A+B+C (equal to or better than C). As shown in Figure 3, among the E-J component MT systems, TDMT had the best performance for Rank A, Rank A+B, and Rank A+B+C. Figures 9, 10, and 11 show the results of the comparisons. The vertical axis in each figure shows the difference in the performances.

Each bar corresponds to a selection system. The first three bars in left-to-right order correspond to TM∗LM-score-based selection, TM-score-based selection, and LM-score-based selection, which were used in the preliminary experiment described in Section 1. The next three bars correspond to the first proposed method based on TM∗LM-score, TM-score, and LM-score. The next three bars correspond to the second proposed method in which predictor variables are restricted to TM∗LM-score, to both TM∗LM-score and TM-score, to all scores, LM∗TM-score, TM-score, and TM∗LM-score. In these selection methods, the regression trees are simplified by using the shrinking procedure. The last three bars also correspond to the second proposed method, but in these selection methods, the regression trees are simplified by using the pruning procedure. Accuracy means the percentage of correctly selecting the output assigned the highest rank in all trials.

Figure 9 shows that the first proposed system based on TM∗LM-score achieved the greatest improvement of a little under 6%, in the performance for Rank A. On the other hand, the existing selection system simply using the LM-score (language model of the translation target) could not improve and even degraded performance for Rank A.

Figure 10 shows that the second proposed system with the pruning procedure based on both TM∗LM-score and TM-score (marked RT12-PRN in the graph) achieved the greatest improvement of about 5%, for Rank A+B (equal to or higher than B). On the other hand, Figure 10 shows that the existing selection system

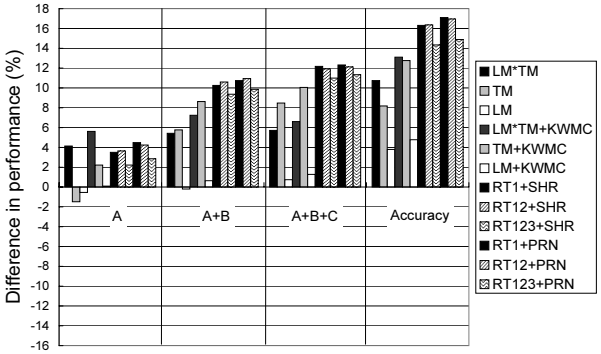

Figure 9: Difference in performance between each selection system and D3.

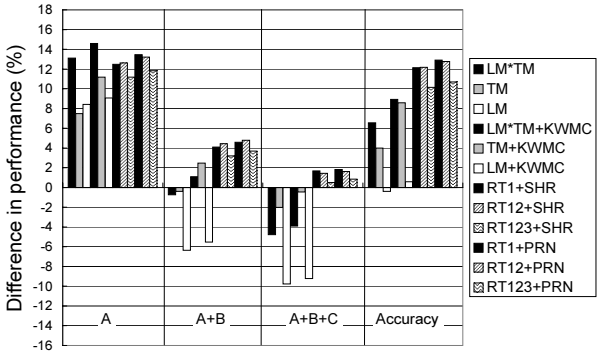

Figure 10: Difference in performance between each selection system and J-E TDMT.

simply using the LM-score had the worst performance for Rank A+B, with a degradation of 6%.

Figure 11 shows that the second proposed systems, with either the pruning procedure or the shrinking procedure, based on only TM*LM-score, on both TM*LM-score and TM-score, or on all scores achieved an improvement of about 2% for rank A in all cases. Figure 11 also shows that the second proposed system with the shrinking procedure based on all scores (marked RT123-SHR in the graph) achieved an improvement of a little more than 2% for Rank A+B.

4 Conclusions

This paper addressed the challenging problem of automatically selecting the best among outputs from multiple MT systems to improve translation quality. This paper proposed two methods. The first method is based on a multiple comparison test based on the Kruskal-Wallis test and checks whether the highest score from the language model, the translation model, or both models combined is significantly different from the others. The second method is based on conditional probability that a translation is not inferior to the others when the translation satisfies some conditions. The conditional probability is predicted by a regression tree learned from the above scores. The proposed methods were evaluated using an ATR travel corpus. Experimental results showed that the performance of the proposed methods is much better than that of the existing methods and achieved the improvement of 2 to 6 % in performance.

Acknowledgment

This research was supported in part by the Telecommunications Advancement Organization of Japan.

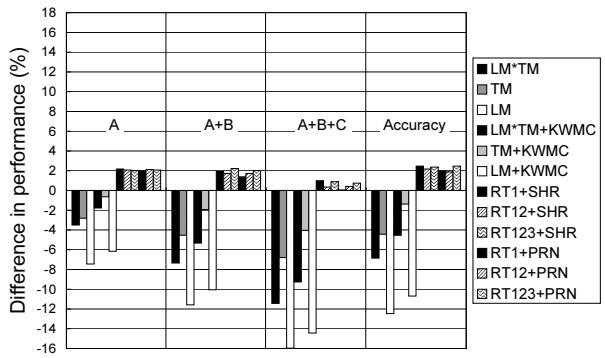

Figure 11: Difference in performance between each selection system and E-J TDMT.

References

Leo Breiman, Jerome H. Friedman, Richard A. Olshen, and Charles J. Stone. 1984. *Classification and Regression Trees.* Chapman and Hall, California, USA.

Peter F. Brown, Stephen Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Chris Callison-Burch and Raymond S. Flournoy. 2001. A program for automatically selecting the best output from multiple machine translation engines. In *Proceedings of MT Summit VIII*, pages 63–66.

John M. Chambers and Trevor J. Hastie. 1992. *Statistical Models in S.* Wadsworth & Brooks/Cole Advanced Books & Software, A Division of Wadsworth, Inc., California, USA.

Philip Clarkson and Ronald Rosenfeld. 1997. Statistical language modeling using the CMU-Cambridge toolkit. In *Proceedings of the European Conference on Speech Communication and Technology: EUROSPEECH-97*, pages 2707–2710.

Osamu Furuse and Hitoshi Iida. 1996. Incremental translation utilizing constituent boundary patterns. In *Proceedings of the 16th International Conference on Computational Linguistics: COLING-96*, pages 412–417.

Y. Hochberg and A. C. Tamhane. 1983. *Multiple Comparison Procedures.* Wiley, New York, USA.

Kenji Imamura. 2002. Application of translation knowledge acquired by hierarchical phrase alignment for pattern-based mt. In *Proceedings of the 9th Conference on Theoretical and Methodological Issues in Machine Translation*, pages 74–84.

Satoshi Kaki, Setsuo Yamada, and Eiichiro Sumita. 1999. Scoring multiple translations using character n-gram. In *Proceedings of the 5th Natural Language Processing Pacific Rim Symposium: NLPRS-99*, pages 298–302.

Tom M. Mitchell. 1997. *Machine Learning.* The McGraw-Hill Companies Inc., New York, USA.

Franz Josef Och and Hermann Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics: ACL00*, pages 440–447.

Eiichiro Sumita. 2001. Example-based machine translation using DP-matching between work sequences. In *Proceedings of the ACL 2001 Workshop on Data-Driven Methods in Machine Translation: DDMT-2001*, pages 1–8.

Toshiyuki Takezawa, Eiichiro Sumita, Fumiaki Sugaya, Hirofumi Yamamoto, and Seiichi Yamamoto. 2002. Toward a broad-coverage bilingual corpus for speech translation of travel conversations in the real world. In *Proceedings of Third International Conference on Language Resources and Evaluation: LREC2002*, pages 147–152.

Taro Watanabe, Kenji Imamura, and Eiichiro Sumita. 2002. Statistical machine translation system based on hierarchical phrase alignment. In *Proceedings of the 9th Conference on Theoretical and Methodological Issues in Machine Translation*, pages 188–198.