Combining Quality Prediction and System Selection for Improved Automatic Translation Output

Radu Soricut
SDL Language Weaver
6060 Center Drive, Suite 150
Los Angeles, CA 90045
rsoricut@sdl.com

Sushant Narsale*
Google Inc
1600 Amphitheatre Parkway
Mountain View, CA 94043
snarsale@google.com

Abstract

This paper presents techniques for reference-free, automatic prediction of Machine Translation output quality at both sentence- and document-level. In addition to helping with document-level quality estimation, sentence-level predictions are used for system selection, improving the quality of the output translations. We present three system selection techniques and perform evaluations that quantify the gains across multiple domains and language pairs.

1 Introduction

Aside from improving the performance of core-translation models, there additionally exist two orthogonal approaches via which fully-automatic translations can achieve increased acceptance and better integration in real-world use cases. These two approaches are: improved translation accuracy via system combination (Rosti et al., 2008; Karakos et al., 2008; Hildebrand and Vogel, 2008), and automatic quality-estimation techniques used as an additional layer on top of MT systems, which present the user only with translations that are predicted as being accurate (Soricut and Echihabi, 2010; Specia, 2011).

In this paper, we describe new contributions to both these approaches. First, we present a novel and superior technique for performing quality estimation at document level. We achieve this by changing the granularity of the prediction mechanism from document-level (Soricut and Echihabi, 2010) to sentence-level, and predicting BLEU scores via directly modeling the sufficient statistics for BLEU computation. A document-level score is then recreated based on the predicted sentence-level sufficient statistics. A second contribution is related to system combination (or, to be more precise, system selection). This is an intended side-effect of the granularity change: since the sentence-level statistics allow us to make quality predictions at sentence level, we can use these predictions to perform system combination by selecting among various sentence-level translations produced by different MT systems. That is, instead of presenting the user with a document with sentences translated entirely by a single system, we can present documents for which, say, 60% of the sentences were translated by system A, and 40% were translated by system B. We contribute a novel set of features and several techniques for choosing between competing machine translation outputs. The evaluation results show better output quality, across multiple domains and language pairs.

2 Related Work

Several approaches to reference-free automatic MT quality assessment have been proposed, using classification (Kulesza and Shieber, 2004), regression (Albrecht and Hwa, 2007), and ranking (Ye et al., 2007; Duh, 2008). The focus of these approaches is on system performance evaluation, as they use a constant test set and measure various MT systems against it.

In contrast, we are interested in evaluating the quality of the translations themselves, while treat-

*Research was completed before the author started in his current role at Google Inc. The opinions stated are his own and not of Google Inc.

ing the MT components as constants. In this respect, the goal is more related to the area of confidence estimation for MT (Blatz et al., 2004). Confidence estimation is usually concerned with identifying words/phrases for which one can be confident in the quality of the translation. A sentence-level approach to quality estimation is taken on the classification-based work of Gamon et al. (2005) and regression-based work of Specia et al. (2009).

Our approach to quality estimation focuses on both sentence-level and document-level estimation. We improve on the quality estimation technique that is proposed for document-level estimation in (Soricut and Echihabi, 2010). Furthermore, we exploit the availability of multiple translation hypotheses to perform system combination. Our system combination methods are based on generic Machine Learning techniques, applied on 1-best output strings. In contrast, most of the approaches to MT system combination combine N-best lists from multiple MT systems via confusion network decoding (Karakos et al., 2008; Rosti et al., 2008). The closest system combination approach to our work is (Hildebrand and Vogel, 2008), where an ensemble of hypotheses is generated by combining N-best lists from all the participating systems, and a log-linear model is trained to select the best translation from all the possible candidates.

In our work, we show that it is possible to gain significant translation quality by taking advantage of only two participating systems. This makes the system-combination proposition much more palatable in real production deployment scenarios for Machine Translation, as opposed to pure research scenarios as the ones used in the previous NIST and DARPA/GALE MT efforts (Olive et al., 2011). As our evaluations show, the two participating systems can be at very similar performance levels, and yet a system-selection procedure using Machine Learning techniques can achieve significant translation improvements in quality. In addition, in a scenario where quality estimation needs to happen as a requirement for MT integration in large applications, having two translation systems producing translations for the same inputs is part of the deployment set-up (Soricut and Echihabi, 2010). The improvement in overall translation quality comes in these cases at near-zero cost.

3 Sentence-level Quality Predictions

The requirement for document-level quality estimation comes from the need to present a fully-automated translation solution, in which translated documents are either good enough to be directly published (or otherwise must undergo, say, a human-driven post-processing pipeline). In the proposal of Soricut and Echihabi (2010), regression models predict BLEU-like scores for each document, based on document-level features.

However, even if the predicted value is at document-level, the actual feature computation and model prediction does not necessarily need to happen at document-level. It is one of the goals of this work to determine if the models of prediction work better at a coarser granularity (such as document level) or finer granularity (such as sentence-level).

We describe here a mechanism for predicting BLEU scores at sentence level, and then combining these scores into document-level scores. To make explicit our prediction mechanism, we present here in detail the formula for computing BLEU scores (Papineni et al., 2002). First, n-gram precision scores P_n are computed as follows:

$$P_n = \frac{\sum_{C \in Candidates} \sum_{n\text{-gram} \in C} Count_{clip}(n\text{-gram})}{\sum_{C \in Candidates} \sum_{n\text{-gram} \in C} Count(n\text{-gram})} \quad (1)$$

where $Count_{clip}(n\text{-gram})$ is the maximum number of n-grams co-occurring in a candidate translation and a reference translation, and $Count(n\text{-gram})$ is the number of n-grams in the candidate translation. To prevent very short translations that try to maximize their precision scores, BLEU adds a brevity penalty, BP, to the formula:

$$BP = \begin{cases} 1 & \text{if } |c| > |r| \\ e^{(1-|r|/|c|)} & \text{if } |c| \leq |r| \end{cases} \quad (2)$$

where $|c|$ is the length of the candidate translation and $|r|$ is the length of the reference translation. The BLEU formula is then written as follows:

$$BLEU = BP \cdot exp(\sum_{n=1}^{N} w_n \log p_n) \quad (3)$$

where the weighting factors w_n are set to $1/N$, for all $1 \leq n \leq 4$.

3.1 The learning method

The results we report in this section are obtained using the freely-available Weka engine. [1] For both sentence-level and document-level quality prediction, we report all the results using Weka implementation of M5P regression trees (weka.classifiers.trees.M5P).

We use the components of the BLEU score (Equations 1 and 2) to train fine-granularity M5P models using our set of features (Section 3.2), for a total of five individual regression-tree models (four for the sentence-level precision scores $P_n, 1 \leq n \leq 4$ factors, and one for the BP factor). The numbers produced individually by our models are then combined using the BLEU equation 3 into a sentence-level BLEU score. The sentence-level predicted BLEU scores play an important role in our system combination mechanism (see Section 4).

At the same time, we sum up the sufficient statistics for the sentence-level precision scores P_n (Equation 1) over all the sentences in a document, thus obtaining document-level precision scores. A document-level BP score (Equation 2) is similarly obtained by summing over all sentences. Finally, we plug the predicted document-level P_n and BP scores in the BLEU formula (Equation 3) and arrive at a document-level predicted BLEU score.

3.2 The features

Most of the features we use in this work are not internal features of the MT system, but rather derived starting from input/output strings. Therefore, they can be applied for a large variety of MT approaches, from statistical-based to rule-based approaches. The features we use can be divided into text-based, language-model–based, pseudo-reference–based, example-based, and training-data–based feature types (these latter features assume that the engine is statistical and one has access to the training data). These feature types can be computed both on the source-side (MT input) and on the target-side (MT output).

Text-based features

These features compute the length of the input in terms of (tokenized) number of words. The source-side text feature is computed on the input string, while the target-side text feature is computed to the output translation string. These two features are useful in modeling the relationship between the number of words in the input and output and the expected BLEU score for these sizes.

Language-model–based features

These features are among the ones that were first proposed as possible differentiators between good and bad translations (Gamon et al., 2005). They are a measure of how likely a collection of strings is under a language model trained on monolingual data (either on the source or target side).

The language-model–based feature values we use here are computed as perplexity numbers using a 5-gram language model trained on the MT training set. This can be achieved, for instance, by using the publicly-available SRILM toolkit [2]. These two features are useful in modeling the relationship between the likelihood of a string (or set of strings) under an n-gram language model and the expected BLEU score for that input/output pair.

Pseudo-reference–based features

Previous work has shown that, in the absence of human-produced references, automatically-produced ones are still helpful in differentiating between good and bad translations (Albrecht and Hwa, 2008). When computed on the target side, this type of features requires one (or possibly more) secondary MT system(s), used to generate translations starting from the same input. These pseudo-references are useful in gauging translation convergence, using BLEU scores as feature values. In intuitive terms, their usefulness can be summarized as follows: "if system X produced a translation A and system Y produced a translation B starting from the same input, and A and B are similar, then A is probably a good translation".

An important property here is that systems X and Y need to be as different as possible from each other. This property ensures that a convergence on similar translations is not just an artifact of the systems sharing the same translation model/resources, but a true indication that the translations converge. The secondary systems we use in this work are

[1] Weka software at http://www.cs.waikato.ac.nz/ml/weka/.

[2] Available at www-speech.sri.com/projects/srilm.

still phrase-based, but equipped with linguistically-oriented modules similar with the ones proposed in (Collins et al., 2005; Xu et al., 2009). Our experiments indicate that this single feature is one of the most powerful ones in terms of its predictive power.

Example-based features

For example-based features, we use a development set of parallel sentences, for which we produce translations and compute sentence-level BLEU scores. We set aside the top BLEU scoring sentences and bottom BLEU scoring sentences. These sets are used as positive examples (with better-than-average BLEU) and negative examples (with worse-than-average BLEU), respectively. We define a positive-example–based feature function as a geometric mean of 1-to-4–gram precision scores (i.e., the BLEU equation 3 with the BP term set to 1) between a string (on either source or target side) and the positive examples used as references. That is, we compute precision scores against all the positive examples at the same time, similar with how multiple references are used to increase the precision of the BLEU metric. (The negative-example–based features are defined in an analogous way.) The set of positive and negative examples is a fixed set that is used in the same manner both at training-time (to compute the example-based feature values for the training examples) and at test-time (to compute the example-based feature values for the test examples).

The intuition behind these features can be summarized as follows: "if system X translated A well/poorly, and A and B are similar, then system X probably translates B well/poorly". The total number of features on this type is 4 (2 for positive examples against source/target strings, 2 for negative examples against source/target strings).

Training-data–based features

If the system for which we make the predictions is trained on a parallel corpus, the data in this corpus can be exploited towards assessing translation quality (Specia et al., 2009; Soricut and Echihabi, 2010; Specia, 2011). In our context, the documents that make up this corpus can be used in a fashion similar with the positive examples. One type of training-data–based features operates by computing the number of out-of-vocabulary (OOV) tokens with respect to the training data (on source side).

A more powerful type of training-data–based features operates by computing a geometric mean of 1-to-4–gram precision score between a string (source or target side) and the training-data strings used as references. Intuitively, these features assess the coverage of the candidate strings with respect to the training data: "if the n-grams of input string A are well covered by the source-side of the training data, then the translation of A is probably good" (on the source side); "if the n-grams in the output translation B are well covered by the target-side of the parallel training data, then B is probably a good translation" (on the target side). The total number of features on this type is 3 (1 for the OOV counts, and 2 for the source/target-side n-gram coverage).

Given the described 12 feature functions, the training for our five M5P prediction models is done using the feature-function values at sentence-level, and associating these values with reference labels that are automatically-produced from parallel-text using the sufficient-statistics of the BLEU score (Equations 1 and 2).

3.3 Metrics for Quality Prediction Performance

The metrics we use here are designed to answer the following question: how well can we automatically separate better translations from worse translations (in the absence of human-produced references)?

A first metric we use is Ranking Accuracy (rAcc), see (Gunawardana and Shani, 2009; Soricut and Echihabi, 2010). In the general case, it measures how well N elements are assigned into n quantiles as a result of a ranking procedure. The formula is:

$$\text{rAcc}[n] = \text{Avg}_{i=1}^{n} \frac{\text{TP}_i}{\frac{N}{n}} = \frac{1}{N} \times \sum_{i=1}^{n} \text{TP}_i$$

where TP_i (True-Positive$_i$) is the number of correctly-assigned documents in quantile i. Intuitively, this formula is an average of the ratio of elements correctly assigned in each quantile. For simplicity, we present here results using only 2 quantiles ($n = 2$), which effectively makes the rAcc[2] metric equivalent with binary classification accuracy when the two sets are required to have equal size. That is, we measure the accuracy of placing the 50%

	Training Size	BLEU		Ranking Test Size	rAcc[2]		DeltaAvg[2]	
		Sys1	Sys2		Doc	Sent	Doc	Sent
WMT09 Hungarian-English	26 Mw	26.9	26.9	510 Kw	88%	89%	+8.3	+8.4
Travel English-French	30 Mw	32.3	34.6	282 Kw	77%	80%	+9.1	+10.1
Travel English-German	44 Mw	40.6	43.4	186 Kw	74%	79%	+9.8	+11.7
HiTech English-French	0.4 Mw	44.1	44.7	69 Kw	75%	77%	+4.4	+6.0
HiTech English-Korean	16 Mw	37.4	36.1	80 Kw	78%	79%	+9.3	+10.0

Table 1: MT system performance and ranking performance using BLEU prediction at Doc- and Sent-level.

best-translated documents (as measured by BLEU against human reference) in the top 50% of ranked documents. Note that a random assignment gives a performance lower bound of 50% accuracy.

A second metric we use here is the DeltaAvg metric (Callison-Burch et al., 2012). The goal of the DeltaAvg metric is to measure how valuable a proposed ranking (hypothesis) is from the perspective of an extrinsic metric associated with the test entries (in our case, the BLEU scores). The following notations are used: for a given entry sentence s, $V(s)$ represents the function that associates an extrinsic value to that entry; we extend this notation to a set S, with $V(S)$ representing the average of all $V(s), s \in S$. Intuitively, $V(S)$ is a quantitative measure of the "quality" of the set S, as induced by the extrinsic values associated with the entries in S. For a set of ranked entries S and a parameter n, we denote by S_1 the first quantile of set S (the highest-ranked entries), S_2 the second quantile, and so on, for n quantiles of equal sizes.[3] We also use the notation $S_{i,j} = \bigcup_{k=i}^{j} S_k$. Using these notations, the metric is defined as:

$$\text{DeltaAvg}_V[n] = \frac{\sum_{k=1}^{n-1} V(S_{1,k})}{n-1} - V(S) \qquad (4)$$

When the valuation function V is clear from the context, we write DeltaAvg$[n]$ for DeltaAvg$_V[n]$. The parameter n represents the number of quantiles we want to split the set S into. For simplicity, we consider there only the case for $n = 2$, which gives DeltaAvg[2] = $V(S_1) - V(S)$. This measures the difference between the quality of the top quantile (top half) S_1 and the overall quality (represented by $V(S)$). For the results presented here, the valuation function V is taken to be the BLEU function (Equation 3).

[3]If the size $|S|$ is not divisible by n, then the last quantile S_n is assumed to contain the rest of the entries.

3.4 Experimental Results

We measure the impact in ranking accuracy using a variety of European and Asian language pairs, using parallel data from various domains. One domain we use is the publicly available WMT09 data (Koehn and Haddow, 2009), a combination of European parliament and news data. Another domain, called Travel, consists of user-generated reviews and descriptions; and a third domain, called HiTech, consists of parallel data from customer support for the high-tech industry. Using these parallel data sets, we train statistical phrase-based MT system similar to (Och and Ney, 2004) as primary systems (Sys1). As secondary systems (Sys2) we use phrase-based systems equipped with linguistically-oriented modules similar with the ones proposed in (Collins et al., 2005; Xu et al., 2009). Table 1 lists the size of the parallel training data on which the MT systems were trained in the first column, and BLEU scores for the primary and secondary systems on held-out 1000-sentence test sets in the next two columns.

The training material for the regression-tree models consists of 1000-document held-out sets. (For parallel data for which we do not have document boundaries, we simply simulate document boundaries after every 10 consecutive sentences.) Similarly, the Ranking test sets we use consist of 1000-document held-out sets (see column 4 in Table 1 for size). In the last four columns of Table 1, we show the results for ranking the translations produced by the primary MT system (Sys1). We measure the ranking performance for the two granularity cases. The one labeled as "Doc" is an implementation of

the work described in (Soricut and Echihabi, 2010), where the BLEU prediction is done using document-level feature values and models. The one labeled as "Sent" is the novel one proposed in this paper, where the BLEU prediction is done using sentence-level feature values and models, which are then aggregated into document-level BLEU scores.

Both rAcc[2] and DeltaAvg[2] numbers support the choice of making document-level BLEU prediction at a finer, sentence-based granularity level. For Travel English-French, for instance, the accuracy of the ranking improves from 77% to 80%. To put some intuition behind these numbers, it means that 4 out of every 5 sentences that the ranker places in the top 50% do belong there. At the same time, the DeltaAvg[2] numbers for Travel English-French indicate that the translation quality of the top 50% of the 1000 Ranking Test documents exceeds by 10.1 BLEU points the overall quality of the translations (up from 9.1 BLEU points for the document-level prediction). This large gap in the BLEU score of the top 50% ranked sentences and the overall-corpus BLEU indicates that these top-ranked translations are indeed of much better quality (closer to the human-produced references). The same large numbers are measured on the WMT09 data for Hungarian-English. This is a set for which it is hard to obtain significant improvements via core-model translation improvements. Our quality-estimation method allows one to automatically identify the top 50% of the sentences with 89% accuracy. This set of top 50% sentences also has an overall BLEU score of 35.3, which is better by +8.4 BLEU-points compared to the overall BLEU score of 26.9 (we only show the base overall BLEU score and the BLEU-point gain in Table 2 to avoid displaying redundant information).

4 System Combination at Sentence Level

Since we produce two translations for every input sentence for the purpose of quality estimation, we exploit the availability of these competing hypotheses in order to choose the best one. In this section we describe three system combination schemes that choose between the output of the primary and secondary MT systems.

4.1 System Combination using Regression

This combination scheme makes use of the regression-based sentence-level BLEU prediction mechanism described in Section 3. It requires that we also train and use an additional BLEU prediction mechanism for which the secondary MT system is now considered primary, and vice-versa. As a consequence, we can predict a sentence-level BLEU score for each of the two competing hypotheses. We then simply choose the hypothesis with the highest predicted BLEU score.

4.2 System Combination using Ranking

This approach is based on ranking the candidate translations and then selecting the highest-ranked translation as the final output. To this end we use SVM-rank (Joachims, 1999), a ranking algorithm built on SVM. We use SVM-rank with a linear kernel and the same feature set as the regression-based method (we make the observation here that only the target-based features have discriminative power in this context).

4.3 System Combination using Classification

In this approach, we model the problem of selecting the best output from the two candidate translations into a binary classification problem. We use the same feature set as before for each candidate translation (again, only the target-based features have discriminative power in this context).

The final feature vectors are obtained by subtracting the values of the primary-system feature vector from the values of the secondary-system feature vector. The binary classifier is trained to predict "0" if the primary-system is better, and "1" if the secondary-system is better.

4.4 Experimental Results

In Table 2, we summarize the results for the three system combination techniques discussed before across our domains (WMT09, Travel, and Hi-Tech). To get an upper bound on the performance of these system combination techniques, we also compute an oracle function which selects the translation with highest BLEU score computed against human-produced references.

The results in Table 2 indicate that the BLEU improvements obtained by our system combina-

	BLEU		Oracle	Regression	Rank	Classify
	Sys1	Sys2				
WMT09 Hungarian-English	26.9	26.9	30.7(+3.8)	29.0(**+2.1**)	29.0(**+2.1**)	28.9(**+2.0**)
Travel English-French	32.3	34.6	38.7(+3.9)	36.2(**+1.6**)	36.0(**+1.4**)	35.7(+1.1)
Travel English-German	40.6	43.4	47.2(+3.8)	44.5(+1.1)	44.0(+0.6)	44.9(**+1.5**)
HiTech English-French	44.1	44.7	49.8(+5.1)	46.1(+1.4)	46.3(**+1.7**)	45.3(+0.6)
HiTech English-Korean	37.4	36.1	42.2(+4.8)	39.4(**+2.0**)	39.1(+1.7)	38.8(+1.4)

Table 2: BLEU scores for the proposed system combination techniques across domains and language pairs.

	Travel Eng-Fra			Hi-Tech Eng-Fra		
	Sys1	Sys2	KL	Sys1	Sy2	KL
BLEU score	32.3	34.6	-	44.1	44.7	-
Oracle distr.	34.9%	65.1%	0.00	34.5%	65.5%	0.00
Regression distr.	**31.2%**	**68.9%**	**0.68**	**32.3%**	**67.7%**	**0.11**
Rank distr.	43.4%	56.6%	1.92	47.0%	53.0%	3.31
Classify distr.	47.4%	52.7%	3.78	63.9%	36.1%	17.88

Table 3: Distribution of sentences selected from the participating system for Eng-Fra, across domains (Travel and Hi-Tech).

tion techniques are significant. For instance, both the Regression-based system combination and the Ranking-based system combination achieve a BLEU score of 29.0 on the WMT09 Hungarian-English test set, an increase of +2.1 BLEU points. In the case of Travel English-French, an increase of +1.6 BLEU points is obtained by the Regression-based system combination, in spite of the fact that one of the systems is measured to be 2.3 BLEU points lower in translation accuracy. Increases in the range of +1.5-2.0 BLEU points are obtained across all the experimental conditions that we tried: three different domains, various language pairs (both in and out of English), and various training data sizes (from 0.4Mw to 40Mw).

Since our system-combination methods chose one system translation over another system translation, we can also measure the distribution of choices made between the two participating systems. These bimodal distributions can help us gauge the performance of various methods, when compared against the BLEU Oracle distribution.

In Table 3, we report the percentages of sentences selected from each system in the oracle combination and each of the described system combination methods. We also report the Kullback-Liebler divergence (KL) between the BLEU Oracle distribution and the distribution induced by each of the system combination methods. The results indicate that, for both English-French cases that we considered (in the Travel and HiTech domains), the choice distribution of the Regression-based system combination method is much closer to the oracle distribution (KL of 0.68 and 0.11, respectively), compared to the other two methods. Note that this does not necessarily correlate with the evaluation based on overall BLEU score of the system-combination methods (Table 2). For instance, for HiTech English-French the best BLEU improvement is obtained by the Rank-based method with +1.7 BLEU points, but the KL divergence score of 3.31 is higher than the one for the Regression-based method (KL score of 0.11). Nevertheless, the choice distributions are an important factor in judging the performance of a given system selection method.

5 Conclusions

Document-level quality estimation is an important component for building fully-automated translation solutions where the translated documents are directly published, without the need for human intervention. Such approaches are the only possible solu-

tion to mitigate the imperfection of current MT technology and the need to translate large volumes of data on a continuous basis.

We show in this paper that sentence-level predictions, when aggregated to document-level predictions, outperform previously-proposed document-level quality estimation algorithms. In addition to that, these finer-granularity, sentence-level predictions can be used as part of a system selection scheme. The three alternative system selection techniques we describe here are intuitive, computationally cheap, and bring significant BLEU gains across multiple domains and language pairs. The finding that the regression-based system selection technique performs as well (or sometimes better) compared to the discriminative methods fits well with the overall theme of using two systems for both improved quality estimation and improved MT performance.

References

Joshua Albrecht and Rebecca Hwa. 2007. Regression for sentence-level MT evaluation with pseudo references. In *Proceedings of ACL*.

Joshua Albrecht and Rebecca Hwa. 2008. The role of pseudo references in MT evaluation. In *Proceedings of ACL*.

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Gouette, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *Proceedings of COLING*.

Chris Callison-Burch, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. 2012. Findings of the 2012 workshop on statistical machine translation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, Montreal, Canada, June. Association for Computational Linguistics.

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of ACL*.

Kevin Duh. 2008. Ranking vs. regression in machine translation evaluation. In *Proceedings of the ACL Third Workshop on Statistical Machine Translation*.

Michael Gamon, Anthony Aue, and Martine Smets. 2005. Sentence-level MT evaluation without reference translations: Beyond language modeling. In *Proceedings of EAMT*.

Asela Gunawardana and Guy Shani. 2009. A survey of accuracy evaluation metrics of recommendation tasks. *Journal of Machine Learning Research*, 10:2935–2962.

Almut Silja Hildebrand and Stephan Vogel. 2008. Combination of machine translation systems via hypothesis selection. In *Proceedings of AMTA*.

T. Joachims. 1999. *Making large-Scale SVM Learning Practical*. M.I.T. Press.

Damianos Karakos, Jason Eisner, Sanjeev Khudanpur, and Markus Dreyer. 2008. Machine translation system combination using ITG-based alignments. In *Proceedings of ACL*.

Philipp Koehn and Barry Haddow. 2009. Edinburgh's submission to all tracks of the WMT2009 shared task with reordering and speed improvements to Moses. In *Proceedings of EACL Workshop on Statistical Machine Translation*.

Alex Kulesza and Stuart M. Shieber. 2004. A learning approach to improving sentence-level MT evaluation. In *Proceedings of the 10th International Conference on Theoretical and Methodological Issues in Machine Translation*.

Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449.

Joseph Olive, Caitlin Christianson, and John McCary, editors. 2011. *Handbook of Natural Language Processing and Machine Translation: DARPA Global Autonomous Language Exploitation*. Springer.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*.

Antti-Veikko Rosti, Bing Zhang, Spyros Matsoukas, and Richard Schwartz. 2008. Incremental hypothesis alignment for building confusion networks with application to machine translation system combination. In *Proceedings of Third Workshop on Statistical Machine Translation*.

Radu Soricut and Abdessamad Echihabi. 2010. Trustrank: Inducing trust in automatic translations via ranking. In *Proceedings of ACL*.

Lucia Specia, Nicola Cancedda, Marc Dymetman, Marcho Turchi, and Nello Cristianini. 2009. Estimating the sentence-level quality of machine translation. In *Proceedings of EAMT*.

Lucia Specia. 2011. Exploiting objective annotations for measuring translation post-editing effort. In *Proceedings of EAMT*.

Peng Xu, Jaeho Kang, Michael Ringaard, and Franz Och. 2009. Using a dependency parser to improve SMT for Subject-Object-Verb languages. In *Proceedings of ACL*.

Yang Ye, Ming Zhou, and Chin-Yew Lin. 2007. Sentence level machine translation evaluation as a ranking. In *Proceedings of the ACL Second Workshop on Statistical Machine Translation*.