WHY WORD ERROR RATE IS NOT A GOOD METRIC FOR SPEECH RECOGNIZER TRAINING FOR THE SPEECH TRANSLATION TASK?

Xiaodong He, Li Deng, and Alex Acero

Microsoft Research, One Microsoft Way, Redmond, WA 98052, USA

ABSTRACT

Speech translation (ST) is an enabling technology for cross-lingual oral communication. A ST system consists of two major components: an automatic speech recognizer (ASR) and a machine translator (MT). Nowadays, most ASR systems are trained and tuned by minimizing word error rate (WER). However, WER counts word errors at the surface level. It does not consider the contextual and syntactic roles of a word, which are often critical for MT. In the end-to-end ST scenarios, whether WER is a good metric for the ASR component of the full ST system is an open issue and lacks systematic studies. In this paper, we report our recent investigation on this issue, focusing on the interactions of ASR and MT in a ST system. We show that BLEU-oriented global optimization of ASR system parameters improves the translation quality by an absolute 1.5% BLEU score, while sacrificing WER over the conventional, WER-optimized ASR system. We also conducted an in-depth study on the impact of ASR errors on the final ST output. Our findings suggest that the speech recognizer component of the full ST system should be optimized by translation metrics instead of the traditional WER.

***Index Terms*—** Speech translation, speech recognition, machine translation, translation metric, word error rate, BLEU score optimization, log-linear model.

1. INTRODUCTION

Speech translation (ST) is an important technology for cross-lingual (one-way or two-way) oral communication, whose societal role is rapidly increasing in the modern global and interconnected informational age. ST technology as key enabler of universal translation is one of the most promising and challenging future needs and wants in the coming decade [15].

A ST system consists of two major components: automatic speech recognition (ASR) and machine translation (MT). Over the past years, significant progress has been made in the integration of these two components in the end-to-end ST task [2][7][9][10][16][20]. In [10], a Bayes-rule-based integration of ASR and MT was proposed, in which the ASR output is treated as a hidden variable. In [19], a log-linear model was proposed to directly model the posterior probability of the translated output given the input speech signal, where the feature functions are derived from the overall outputs of the ASR model, the translation model, and the Part-of-Speech language model. This set of work is later extended with the use of the phrase-based MT component and a lattice/confusion-network based interface between ASR and MT [8][13].

Despite their importance, there have been relatively few studies on the impact of ASR errors on the MT quality. Unlike ASR, where the widely used metric is word error rate (WER), the translation accuracy is usually measured by the quantities including BLEU (Bi-Lingual Evaluation Understudy), NIST-score, and Translation Edit Rate (TER) [12][14]. BLEU measures the n-gram matches between the translation hypothesis and the reference(s). In [1][2], it was reported that translation accuracy degrades 8 to 10 BLEU points when the ASR output was used to replace the verbatim ASR transcript (i.e., assuming no ASR error). On the other hand, although WER is widely accepted as the de facto metric for ASR, it only measures word errors at the surface level. It takes no consideration of the contextual and syntactic roles of a word. In contrast, most modern MT systems rely on syntactic and contextual information for translation. Therefore, despite the extreme example offered in [1], it is not clear whether WER is a good metric for ASR in the scenario of ST. Since the latest ASR systems are usually trained by discriminative techniques where models are optimized by the criteria that are strongly correlated with WER (see an overview paper in [4]), the answers to the question of whether WER is a good metric for training the ASR component of a full ST system become particularly important.

The question is addressed in our recent investigation, where we use a log-linear model to integrate the ASR and MT modules for ST. In our approach, the ASR output is treated as a hidden variable, and the posterior probability of a <ASR-output, MT-output> pair given the speech signal is modeled through a regular log-linear model using the feature functions derived from hidden Markov model (HMM)-based ASR and a hierarchical phrase-based MT [3]. These “features” include acoustic model (AM) score, source and target language model (LM) scores, phrase level and lexicon level translation model (TM) scores, etc. All the parameters of the log-linear model are trained to directly optimize the quality of the final translation output measured in BLEU. On a ST task of oral lecture translation from English to Chinese, our experimental results show that the log-linear model and global optimization improve the translation quality by 1.5% in the BLEU score. Our investigation also provides insights to the relationship between the WER of the ASR output and the BLEU score of the final ST output. The experimental results show a poor correlation between the two, suggesting that WER is not a good metric for the ASR component of the ST system. In particular, using real examples extracted from the test data, we further isolate two typical situations where ASR outputs with higher WER can lead to counter-intuitively better translations. These findings suggest that the speech recognizer in a ST system should be trained directly by the translation metric of the full system such as the BLEU score, instead of the local measure of WER.

2. SPEECH TRANSLATION SYSTEMS

978-1-4577-0539-7/11/$26.00 ©2011 IEEE

A general framework for ST is illustrated in Fig. 1. The input speech signal X is first fed into the ASR module. Then the ASR module generates the recognition output set $\{F\}$, which is in the source language. The recognition hypothesis set $\{F\}$ is finally passed to the MT module to obtain the translation sentence E in the target language. In our setup, an N-best list is used as the interface between ASR and MT. In the following, we use F to represent an ASR hypothesis in the N-best list.

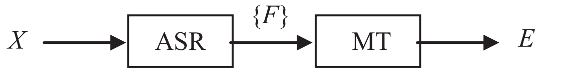

Fig. 1. Two components of a full speech translation system

2.1. The unified log-linear model for ST

The optimal translation $\hat{E}$ given the input speech signal X is obtained via the decoding process according to

$$\hat{E} = \operatorname*{argmax}_{E} P(E|X) \tag{1}$$

Based on law of total probability, we have,

$$P(E|X) = \sum_{F} P(E,F|X) \tag{2}$$

Then we model the posterior probability of the (E, F) sentence pair given X through a log-linear model:

$$P(E,F|X) = \frac{1}{Z} exp\left\{\sum_{i} \lambda_i log\varphi_i(E,F,X)\right\} \tag{3}$$

where $Z = \sum_{E,F} exp\{\sum_i \lambda_i log\varphi_i(E,F,X)\}$ is the normalization denominator to ensure that the probabilities sum to one. In the log-linear model, $\{\varphi_i(E,F,X)\}$ are the feature functions empirically constructed from E, F, and X. The only free parameters of the log-linear model are the feature weights, i.e., $\Lambda = \{\lambda_i\}$. Details of these features used in our experiments are provided in the next section.

2.2. Features in the ST model

The full set of feature functions constructed and used in our ST system are derived from both the ASR and the MT [2][3] modules as listed below:

- Acoustic model (AM) feature: $\varphi_{AM}(E,F,X) = p(X|F)$, which is the likelihood of speech signal X given a recognition hypothesis F, computed from the AM of the source language.
- Source language model (LM) feature: $\varphi_{SLM}(E,F,X) = P_{LM}(F)$, which is the probability of F computed from a N-gram LM of the source language.
- ASR hypothesis length: $\varphi_{SWC}(E,F,X) = e^{|F|}$ is the exponential of the word count in the source sentence F. (This is also referred to as word insertion penalty.)
- Forward phrase translation feature: $\varphi_{F2Eph}(E,F,X) = P_{TMph}(E|F) = \prod_k p(\tilde{e}_k|\tilde{f}_k)$, where $\tilde{e}_k$ and $\tilde{f}_k$ are the k-th phrase in E and F, respectively, and $p(\tilde{e}_k|\tilde{f}_k)$ is the probability of translating $\tilde{f}_k$ to $\tilde{e}_k$.
- Forward word translation feature: $\varphi_{F2Ewd}(E,F,X) = P_{TMwd}(E|F) = \prod_k \prod_m \sum_n p(e_{k,m}|f_{k,n})$, where $e_{k,m}$ is the m-th word of the k-th target phrase $\tilde{e}_k$, $f_{k,n}$ is the n-th word in the k-th source phrase $\tilde{f}_k$, and $p(e_{k,m}|f_{k,n})$ is the probability of translating word $f_{k,n}$ to word $e_{k,m}$. (This is also referred to as the lexical weighting feature.)
- Backward phrase translation feature: $\varphi_{E2Fph}(E,F,X) = P_{TMph}(F|E) = \prod_k p(\tilde{f}_k|\tilde{e}_k)$, where $\tilde{e}_k$ and $\tilde{f}_k$ are defined as above.
- Backward word translation feature: $\varphi_{E2Fwd}(E,F,X) = P_{TMwd}(F|E) = \prod_k \prod_n \sum_m p(f_{k,n}|e_{k,m})$, where $e_{k,m}$ and $f_{k,n}$ are defined as above.
- Count of NULL translations: $\varphi_{NC}(E,F,X) = e^{|Null(F)|}$ is the exponential of the number of the source words that are not translated (i.e., translated to NULL word in the target side).
- Count of phrases: $\varphi_{PC}(E,F,X) = e^{|\{(\tilde{e}_k,\tilde{f}_k),k=1,\dots,K\}|}$ is the exponential of the number of phrase pairs.
- Translation length: $\varphi_{TWC}(E,F,X) = e^{|E|}$ is the exponential of the word count in translation E.
- Hierarchical phrase segmentation and reordering feature: $\varphi_{Hiero}(E,F,X) = P_{hr}(S|E,F)$ is the probability of particular phrase segmentation and reordering S, given the source and target sentence E and F [3].
- Target language model (LM) feature: $\varphi_{TLM}(E,F,X) = P_{LM}(E)$, which is the probability of E computed from an N-gram LM of the target language.

Unlike the previous work [8][19], we used a hierarchical phrase-based MT module [3]. It is based on probabilistic synchronous context-free grammar (PSCFG) models that define a set of weighted transduction rules. These rules describe the translation and reordering operations between source and target languages. In training, our MT module is learned from parallel training data; and in runtime, the decoder will choose the most likely rules to parse the source language sentence while synchronously generating the target language output. Compared with the simple phrase-based MT [6], the hierarchical MT supports the translation of non-contiguous phrases with more complex segmentation and re-ordering, and it also gives better translation performance [3].

2.3. Training of feature weights

The free parameters of the log-linear model, i.e., the weights (denoted by Λ) of these features, are trained by maximizing the BLEU score of the final translation on a dev set, i.e.,

$$\hat{\Lambda} = \operatorname*{argmax}_{\Lambda} BLEU(E^*, \hat{E}(\Lambda, X)) \tag{4}$$

where E^* is the translation reference(s), and $\hat{E}(\Lambda, X)$ is the translation output, which is obtained through the decoding process according to (1) given input speech X and feature weights Λ. In the experiments, we adopt *Powell*'s search [11] to optimize the feature weights in our experiments.

3. EXPERIMENTAL RESULTS AND ANALYSIS

3.1. Experimental conditions

In our experiments, the data for acoustic model training come from the switchboard telephony speech data set. A Gaussian mixture model (GMM) based continuous density HMM is used for acoustic modeling. The MT component is trained on the English-Chinese parallel corpus used in NIST MT08 open evaluation, which is available from LDC. It includes a total of seven million parallel sentence pairs. A hierarchical phrase-based translation system is trained from these parallel data [3].

We conducted experiments on a Microsoft-internal English-to-Chinese lecture translation task. The data are from a segment of a recorded lecture. The speaker delivered the talk in English, and our ST task is to translate it into Chinese. It includes about 31 minutes of speech. This segment of speech is manually transcribed in English, and then translated to Chinese by two human translators. The English transcription includes about 313 sentences and 5585 words in total.

Since the spoken utterance data that have both transcription and translation references are very limited, we perform a two-fold cross-validation in the following evaluation: We first split the 313 sentences of the data into two equal parts. We then train the log-linear model using the first half of the sentences and apply the trained model to decode the second half of the sentences, and vice versa. Finally, we merge the two parts of testing outputs and measure the overall results.

3.2. Experimental results

3.2.1. End-to-end ST results

We first evaluate the end-to-end ST performance using the log-linear model. The baseline is a simple cascading ST model, i.e., the ASR module generates the recognition output, and then it is fed into the MT module to generate the final translation output. In the baseline, the ASR module is tuned for WER and the MT module is tuned for BLEU with clean English text as the input. Since ASR and MT operate independently in the cascading model, only the top best ASR output is fed into MT. In the log-linear model based approach, we used a 20-best list as the interface between ASR and MT, and BLEU is used for optimizing the log-linear model's parameters. The evaluation results are tabulated in table 1.

Table 1. Performance comparison of three ST systems

ST model/system	BLEU
Simple cascading model	33.77%
Log-linear – all features	35.22% *
Truncated log-linear – ASR features	34.77% *

* These improvements are both with a statistical significance level greater than 99%, computed based on the paired bootstrap re-sampling method [5]

As shown in Table 1, global optimization of all feature weights in the log-linear model gives a 1.45% BLEU score improvement compared with the cascading baseline.

In order to study the effect of the training metric of ASR in our ST task, we test a "truncated log-linear model", which gives a controlled setting designed to be the same as the cascading baseline, except that the weights of only the three ASR-related features are trained, but all remaining MT features' weights fixed. This is equivalent to the cascading baseline except that the speech recognizer is tuned by the final BLEU score according to (4). This gives a significant and somewhat surprising 1.0% BLEU score improvement. By checking the value of the trained ASR feature weights, we found that a relatively large LM scale is obtained. Our hypothesis is that, by putting more weights on the LM, the ASR module is encouraged to generate more grammatically fluent English output. Intuitively, this is preferred as the input for the MT module, hence higher BLEU scores, despite the fact that this may cause an increase of WER. We have conducted more detailed analysis to verify our intuition, which we report next.

3.2.2. Analysis on the WER vs. the BLEU score

Given the observation above, it is hypothesized that in a ST task, better (lower) WER does not necessarily lead to better translation quality or higher BLEU score. In our experiments involving the truncated log-linear model as described above, we measured the WER of the ASR's output vs. the BLEU score of the final translation by varying the weights of the source LM features over a relevant range, while the word insertion penalty is adjusted accordingly so that the same insertion/deletion ratio is maintained. The results are presented in Figure 2.

As shown in fig. 2, the WER of the ASR output reaches the lowest point (highest accuracy) with a LM scale = 12. However, the BLEU score of the translation at that setting is 33.77%, far from optimal yet. When increasing the LM scale gradually, the WER starts to get worse (higher), but the BLEU score gets improved. This trend keeps until reaching LM Scale = 18, at where the BLEU score peaks. Compared with the setting corresponding to LM scale = 12, the BLEU score improved by 1.2%. However, the WER increased from 19.6% to 24.5%. This clearly shows that the optimal setting tuned by WER does not necessarily lead to the optimal translation result. Therefore, WER is not a good metric for recognizer training for the ST task in our experiments.

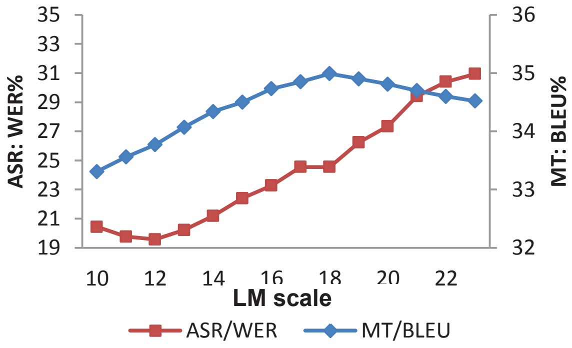

Fig. 2. WER of the ASR's output and BLEU score of the final translation as a function of the sweeping LM scale

3.2.3. The impact of ASR errors on MT

Going beyond the quantitative analysis as described in the last two sections, we also carried out case studies on the impact of ASR errors on the MT quality. In Fig. 3, we present two representative examples to explain how some ASR outputs with more "word errors" can correlate with better translation.

In the first example, speech recognition output B contains one more insertion error than output A. However, the inserted function word "to" makes the whole sentence grammatical correct. It also provides a critical context that helps determine the meaning of "great" as well as the correct word order of the translation.

In the second example, speech recognition output B contains two more errors than output A. However, output B chooses the phrase "want to", which causes two ASR "errors", instead of the colloquial word “wanna”, which is correctly recognized. The mis-recognized phrase “want to”, however, is plentifully represented in the formal text that is used for MT training and hence leads to correct translation. (The remaining recognition errors in both outputs A and B for this sentence do not change the meaning, and therefore do not cause translation errors.)

These and numerous other examples from the analysis lead to the conclusion that the words that signify syntactic categories should be recognized correctly by the ASR component, a task that can be accomplished in the BLEU-optimized approach, while the use of the conventional WER-optimized approach to train ASR does not accomplish this goal.

Example 1:

Transcript	it is great seeing you all here today
Translation ref.	今天很高兴在这里见到你们
Reco A.	let's great see you all here today
Translation A.	今天在这里看到你们让我们好
Reco B.	let's great to see you all here today
Translation B.	我们今天很高兴在这里见到你们

Example 2:

Transcript	i didn't ever really wanna do this
Translation ref.	我从来没有真的想要这么做
Reco A.	i can never really wanna do this
Translation A.	我永远不能真的想
Reco B.	i ve never really want to do this
Translation B.	我从来没有真的要这么做

Fig. 3. Two examples showing the typical cases that sometimes ASR output with more “word errors” can lead to even better translation.

4. SUMMARY AND FUTURE WORK

In this work, we develop a BLEU-optimized approach for training the scale parameters of a log-linear based speech translation system. Our experimental results demonstrate the effectiveness of this approach in terms of the translation quality, although the ASR errors as the intermediate result are found to be higher than the cascaded ASR and ML approach where the ASR system is trained using the conventional WER criterion. Analysis of the errors shows the importance of correct recognition of the key words by ASR that are associated with syntactic categories. Missing such words in ASR often lead to disastrous MT results, which we have observed to occur more frequently with the conventional WER-optimized approach than with the new BLEU-optimized approach as reported in this paper.

The technique presented in this paper is a simplistic implementation of the more general end-to-end learning framework for the ST system design. We only adjust a very small number of the system parameters, i.e., the feature scale parameters in the log-linear model, to maximize the translation quality. It has been shown that in other tasks such as speech understanding, joint optimization of feature functions can further improve the performance [17]. In the parallel work reported in [18], we adopt a more aggressive approach where the parameters “inside” the feature functions, e.g., all the individual N-gram probability values of both source and target languages, as well as the phrase table probability values, are subject to adjustment so as to maximize the end-to-end ST quality. Our future work will push this even further into each of the free parameters in the full ST system including the HMM and possibly the feature extraction parameters in ASR. More advanced modeling and optimization techniques than presented in this paper will be needed in order to accomplish the full-scale end-to-end learning and design for ST.

5. REFERENCES

[1] N. Bertoldi, R. Zens, and M. Federico, “Speech Translation by Confusion Network Decoding,” in *Proc. ICASSP* 2007

[2] F. Casacuberta, M. Federico, H. Ney, and E. Vidal, “Recent Efforts in Spoken Language Translation,” *IEEE Signal Processing Magazine*, May 2008.

[3] D. Chiang, “A hierarchical phrase-based model for statistical machine translation,” in *Proc. ACL*, 2005

[4] X. He, L. Deng, and C. Wu, “Discriminative Learning in Sequential Pattern Recognition,” *IEEE Signal Processing Magazine*, Sept. 2008

[5] P. Koehn, "Statistical Significance Tests for Machine Translation Evaluation," In *Proc. EMNLP*, 2004

[6] P. Koehn, F. J. Och, and D. Marcu. “Statistical Phrase-Based Translation," In Proc. HLT-NAACL, 2003

[7] S. Matsoukas, I. Bulyko, B. Xiang, K. Nguyen, R. Schwartz, and J. Makhoul, “Integrating Speech Recognition and Machine Translation,” in *Proc. ICASSP*, 2007

[8] E. Matusov, H. Ney, R. Schluter, “Phrase-Based Translation of Speech Recognizer Word Lattices Using Loglinear Model Combination,” in *Proc. ASRU*, 2005

[9] E. Matusov, S. Kanthak, and H. Ney, “Integrating Speech Recognition and Machine Translation: Where Do We Stand?” in *Proc. ICASSP*, 2006

[10] H. Ney, “Speech Translation: Coupling of Recognition and Translation," in *Proc. ICASSP*, 1999

[11] R. Brent, "Algorithms for Minimization without Derivatives," Prentice-Hall, Chapter 7, 1973

[12] K. Papineni, S. Roukos, T.Ward, and W.-J. Zhu, “Bleu: A Method for Automatic Evaluation of Machine Translation,” in *Proc. ACL*, 2002

[13] S. Saleem, S-C. Lou, S. Vogel, and T. Schultz, “Using Word Lattice Information for A Tighter Coupling in Speech Translation Systems,” in *Proc. ICSLP*, 2004

[14] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A Study of Translation Edit Rate with Targeted Human Annotation," in *Proc. AMTA*, 2006

[15] J. Treichler, “Signal Processing: A View of the Future, Part 2,” *IEEE Signal Processing Magazine*, May 2009

[16] W. Wang, G. Tur, J. Zheng, N. Fazil Ayan, “Automatic Disfluency Removal For Improving Spoken Language Translation,” in *Proc. ICASSP*, 2010

[17] S.Yaman, L.Deng, D.Yu, Y.-Y. Wang, and A. Acero, “An Integrative and Discriminative Technique for Spoken Utterance Classification,” in *IEEE Trans. ASLP*, 2008

[18] Y. Zhang, L. Deng, X. He, and A. Acero. “A Novel Decision Function and the Associated Decision-Feedback Learning for Speech Translation,” in *Proc. ICASSP*, 2011

[19] R. Zhang, G. Kikui, H. Yamamoto, T. Watanbe, F. Soong, and W-K., Lo, “A Unified Approach in Speech-To-Speech Translation: Integrating Features of Speech Recognition and Machine Translation”, in *Proc. COLING*, 2004

[20] B. Zhou, L. Besacier, and Y. Gao, “On efficient coupling of ASR and SMT for speech Translation,” in Proc. ICASSP, 2007