A Coarse-Grained Model for Optimal Coupling of ASR and SMT Systems for Speech Translation

Gaurav Kumar [1], **Graeme Blackwood** [2], **Jan Trmal** [1], **Daniel Povey** [1], **Sanjeev Khudanpur** [1]
[1] CLSP & HLTCOE, Johns Hopkins University, Baltimore, MD, USA
[2]IBM T. J. Watson Research Center, Yorktown Heights, NY, USA
{gkumar6, dpovey1, khudanpur}@jhu.edu, blackwood@us.ibm.com

Abstract

Speech translation is conventionally carried out by cascading an automatic speech recognition (ASR) and a statistical machine translation (SMT) system. The hypotheses chosen for translation are based on the ASR system's acoustic and language model scores, and typically optimized for word error rate, ignoring the intended downstream use: automatic translation. In this paper, we present a *coarse-to-fine* model that uses features from the ASR and SMT systems to optimize this coupling. We demonstrate that several standard features utilized by ASR and SMT systems can be used in such a model at the speech-translation interface, and we provide empirical results on the Fisher Spanish-English speech translation corpus.

1 Introduction

Speech translation is the process of translating speech in the source language to text or speech in the target language. This process is typically structured as a three step pipeline. Step one involves training an Automatic Speech Recognition (ASR) system to transcribe speech to text in the source language. Step two involves extracting an appropriate form of the ASR output to translate. We will refer to this step as the Speech-Translation interface. In the simplest scenario, the ASR 1-best output can be used as the source text to translate. It may be useful to consider alternative ASR hypotheses and these take the form of an N-best list or a word-lattice. An N-best list can be included easily into the tuning and the decoding process of a statistical machine translation (SMT) system (Zhang et al., 2004). Several researchers have proposed solutions to incorporating lattices and confusion networks in this process (Saleem et al., 2004; Matusov et al., 2005; Bangalore and Riccardi, 2000; Dyer et al., 2008a; Bertoldi and Federico, 2005; Quan et al., 2005; Mathias and Byrne, 2006; Bertoldi et al., 2007). Word lattice input to SMT for tuning and decoding increases the complexity of the decoding process because of the exponential number of alternatives that are present. Finally, step three involves training and tuning a Statistical Machine Translation (SMT) system and decoding the output extracted through the speech translation interface.

This paper presents a featurized model which performs the job of hypothesis selection from the outputs of the ASR system for the input to the SMT system. Our motivation is as follows:

1. **Using downstream information** : Hypothesis selection for the input to the SMT system should be done jointly by the ASR and the SMT systems. That is, there may exist hypotheses that a trained SMT system may find easier to translate and produce better translations for than the ones that are deemed best based on the ASR acoustic and language model scores. Incorporation of knowledge from the downstream process (translation) is vital to selecting translation options, and subsequently producing better translations.

2. **Coarse-to-fine grained decoding** : An intermediate model which acts as an interface and is a weak (coarse) version of the downstream process may be able to select better hypotheses. In effect, a weak translation decoder can be used as the interface to estimate the expected translation quality of an ASR hypothesis. This method of hypothesis selection should be able to incorporate features from the ASR and the SMT system.

3. **Phrase units vs. word units** : When a phrase based SMT system is used for translation,

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1902–1907,
Lisbon, Portugal, 17-21 September 2015.

optimization for hypothesis selection at the Speech-Translation interface should be conducted using phrases as the basic unit instead of words.

2 Coarse-to-Fine Speech Translation

In this section, we describe the featurized model (coarse-grain MT decoder) for hypothesis selection that uses information from the ASR and SMT systems (impedance matching). We assume the presence of ASR and SMT systems that have been trained separately. In addition to creating almost no disruption in the traditional pipeline approach, this allows us to incorporate local gains from each system. To elaborate, our methods avoid joint optimization of the ASR and the SMT system with respect to a translation metric (Vidal, 1997; Ney, 1999), which is not feasible for larger datasets. Also, considering the dearth of speech translation training datasets, this method allows independent training of the ASR and SMT systems on data created only for ASR training and parallel data for SMT. We start by introducing the formal machinery that will be used and by presenting a simple example to motivate the model. The complete featurized model follows this exposition.

Let Σ and Γ be alphabets of words and phrases respectively in the source language. Using these, we can define the following finite state machines:

1. **Word Lattice** (L) : A finite state acceptor that accepts word-sequences in the source language ($L : \Sigma^* \rightarrow \Sigma^*$). This represents the unpruned ASR word lattice output in our model (Figure 1a).

2. **Phrase segmentation Transducer** (S) : A cyclic finite state transducer that transduces a sequence of words to phrases in the source language ($S : \Sigma^* \rightarrow \Gamma^*$). This is built from the source side of the phrase table. Each path represents one source side phrase in the phrase table. Traversing a path is equivalent to consuming the words in a phrase and producing the phrase as a token (Figure 1b).

3. **Weighted word lattice** ($\tilde{L}_{ASR}$) : A weighted version of L ($\tilde{L}_{ASR} : \Sigma^* \rightarrow \Sigma^*/\mathbb{R}^+$). We use the subscript to denote the nature/source of the weights.

4. **Phrase acceptor** ($\tilde{W}_{MT}$) : A finite state acceptor that accepts source phrases in the SMT system's phrase table ($\tilde{W}_{MT} : \Gamma \rightarrow \Gamma/\mathbb{R}$). It is weighted by features derived from the SMT system.

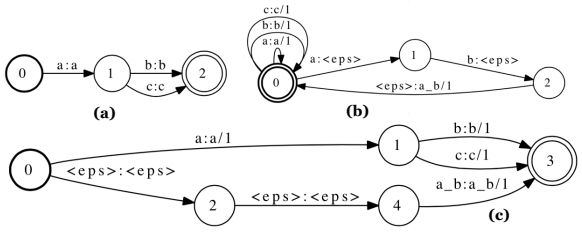

Figure 1: *A toy example for producing a phrase length weighted phrase lattice. (a) An unweighted word lattice. (b) A phrase segmentation transducer which transduces words to phrases and has a weight of one per path. Each path is a source phrase in the phrase table. (c) A phrase lattice produced by composing the word lattice and phrase segmentation transducer.*

5. **Phrase lattice** (P) : The result of the composition of a word lattice (acceptor) with the phrase segmentation transducer ($P : \Sigma^* \rightarrow \Gamma^*$). This represents all possible phrase segmentations of all the ASR hypotheses in the word lattice.

$$P = \det(\min(L \circ S))$$

We will represent weighted versions of P as $\tilde{P}_{ASR/MT}$ with subscripts to denote the origin of the weights (Figure 1c).

2.1 A simple model : Maximum Spanning Phrases

We motivate our model with this fairly simple scenario. Suppose that we believe that if our SMT input could be covered by longer source side phrases[1], we would produce better translations. This may be viewed as a tiling problem where the tiles are the source phrases in the phrase table and the goal is to select the ASR hypothesis that requires the least number of phrases to cover[2]. To achieve this using our existing machinery, we create $\tilde{S}$, a weighted version of S (Figure 1 (b)), such

[1] In phrase based translation, target translations are produced for each possible span of the input sentence allowed by the phrase table. Translation of a longer source side phrase produces fewer translation options and may be more reliable given sufficient occurrences in the training data.

[2] It may be useful to incorporate a brevity penalty here, since this approach has a strong bias towards selecting shorter hypotheses. We will use other features to counter this bias in the following sections.

that

$$w(\delta^{(\tilde{S})}) = \begin{cases} 0 & : \pi_1(\delta^{(S)}) \in \Sigma \text{ and } \pi_2(\delta^{(S)}) = \epsilon \\ 1 & : \pi_2(\delta^{(S)}) \in \Gamma \text{ and } \pi_1(\delta^{(S)}) = \epsilon \end{cases}$$

where $\delta^{(\tilde{S})}$ is an edge in $\tilde{S}$ and π_1 and π_2 are the input and output projections respectively. Using this segmentation transducer and an unweighted word lattice, L (Figure : 1 (a)), we produce a phrase lattice $\tilde{P}_{MT}$. Assuming the weights are in the log-semiring, the weight of a path $\delta^{(\tilde{P})*}$ in $\tilde{P}_{MT}$ is

$$w(\delta^{(\tilde{P})*}) = \sum_{\delta^{(\tilde{P})} \in \delta^{(\tilde{P})*}} w(\delta^{(\tilde{P})})$$

Figure 1(c) shows an example of this phrase lattice. Weights in the phrase lattice follow the same definition as the weights in the segmentation transducer. Hence, the weight of a path in the phrase lattice is simply the number of phrases used to cover this path. The shortest path [3] in the phrase lattice $\tilde{P}_{MT}$, corresponds to the hypothesis we were looking for. This simple example, demonstrates how we may be able to use SMT features (source phrase length in this case) to select hypotheses from the phrase lattice.

2.2 A general featurized model for hypothesis selection

We now present a general framework in which hypothesis selection can be carried out using knowledge (features) from the ASR and the SMT system. As described earlier, this form of 'impedance' matching allows us to select hypotheses from an unpruned ASR word lattice for which the SMT system is more likely to find good translations. Incorporating ASR weights also ensures that we take into account what the ASR system considers to be good hypotheses. We start with the previously discussed idea of a phrase lattice, using weights from the ASR system only. That is,

$$\tilde{P}_{ASR} = \det(\min(\tilde{L}_{ASR} \circ S))$$

Now, we use the weighted phrase acceptor $\tilde{W}_{MT}$ to bring in the SMT features [4]. Composing this with the weighted phrase lattice, we get

$$\tilde{P}_{ASR,MT} = \det(\min(\tilde{P}_{ASR} \circ (\tilde{W}_{MT})^*)$$

where $(\tilde{W}_{MT})^*$ is the Kleene closure of $(\tilde{W}_{MT})$. We assume that the edge weights are in the log-semiring. Hence, after these two compositions, the edge weights in $\tilde{P}_{ASR,MT}$ can be represented as

$$\begin{aligned} w(\delta^{(\tilde{P}_{ASR,MT})}) &= \sum_j \beta_j f_{j,ASR} + \sum_k \gamma_k f_{k,MT} \\ &= \sum_i \lambda_i f_i \end{aligned}$$

where $\delta^{(\tilde{P}_{ASR,MT})}$ is an edge in $\tilde{P}_{ASR,MT}$, β, γ are feature weights, f_{ASR} and f_{MT} are features from the ASR and SMT system respectively. This form represents a log-linear model (our features are already assumed to be in log-space). where f_i is any feature and λ_i is the corresponding feature weight. We may now extract the one-best, N-best or lattice input for the SMT system from $\tilde{P}_{ASR,MT}$.

2.2.1 A discussion about related techniques

1. Decoding (Translation) : Our model closely resembles a featurized finite-state transducer based translation model. If we replace the output alphabet of the acceptor $(\tilde{W}_{MT})^*$ with the target side phrases, we will actually get output in the target language. Even though this model does not explicity include reordering, the coarse-grained decoder has access to information that can enable better decisions about which hypotheses are better for the downstream process (translation).

2. Lattice Decoding : (Dyer et al., 2008b) suggests passing the entire word lattice to the SMT system. However, even if these lattices are not pruned, a beam based decoder might not consider hypotheses that our model may produce through coarse-grained decoding.

3. Language model re-scoring : One may use a bigger source language model to re-score the ASR lattice (or an N-best list). This however, does not consider any SMT features in re-scoring. With our model, we can simply use this as an additional feature.

2.2.2 Training

Training the hypothesis selection model can be carried out using standard methods for log linear models on a held-out set. This also requires decoding (translation) of a deep N-best list derived from the held-out set. The objective of training then simply becomes maximization of the translation

[3] To compute the shortest path, we switch from the log to the tropical semiring (A semiring with ordinary addition as the multiplication operator and $\max$ as the addition operator).

[4] Alternatively, we may have introduced the weights in the segmentation transducer itself. This separate machine is introduced for efficient training of this model.

quality given any metric that provides sentence level scores. Each time our model produces a hypothesis, its score can be looked up from the pre-translated N-best list. Also, whenever the weights are updated, the only structures that need to be rebuilt are $\tilde{W}^*_{MT}$ and $\tilde{P}_{ASR,MT}$ [5].

2.2.3 Features

We use the following features in our implementation of this model. However, any relevant ASR and SMT feature may be readily added to this model.

1. **ASR scores** : We incorporate the ASR acoustic (AM) and language (LM) model scores as one combined feature.

   $$f_{ASR} = LM + \alpha * AM$$

   Here, LM, AM are negative log-probabilities and α is the acoustic scaling parameter chosen to minimize ASR word error rate.

2. **Source phrase count** : As described in section 2.1, this feature may be used to capture the intuition that using a fewer number of phrases to cover the input sentence may produce better translations.

3. **Length normalized phrase unigram probability** : We may use a phrase LM feature by incorporating phrase n-gram probabilities (normalized) by length.

   $$f_{\text{uni}}(f_j) = \left[\frac{\text{freq}(f_j)}{\sum_k \text{freq}(f_k)} \right]^{\text{len}(f_j)}$$

   where f_j is a source side phrase in the phrase table.

4. **Phrase translation entropy** : For each source side phrase p_j, we may have multiple translations (e_i) in the phrase table with different translation probabilities ($p(e_i|f_j)$). A simple entropy measure can be used as a feature to estimate the confidence that the SMT system has in translating f_j.

   $$\begin{aligned} f_{\text{tr}}(p_j) &= H_{\text{tr}}(E|p_j) \\ &= -\sum_i p_{\text{tr}}(e_i|f_j) \log p_{\text{tr}}(e_i|f_j) \end{aligned}$$

5. **Lexical translation entropy** : Similarly, we can use an entropy measure based on the lexical translation probability as a feature.

   $$\begin{aligned} f_{\text{lex}}(p_j) &= H_{\text{lex}}(E|p_j) \\ &= -\sum_i p_{\text{lex}}(e_i|f_j) \log p_{\text{lex}}(e_i|f_j) \end{aligned}$$

3 Results

We use the Fisher and Callhome Spanish-English Speech Translation Corpus (Post et al., 2013) for our experiments. This Fisher dataset consists of 819 transcribed and translated telephone conversations. The corpus is split into a training, dev and two test sets (dev-2 and test). We use the dev set for training the feature weights of the proposed model.

We use the Kaldi speech recognition tools (Povey et al., 2011) to build our Spanish ASR systems. Our state-of-the-art ASR system is the p-norm DNN system of (Zhang et al., 2014). The word-error-rates on the dev and test sets of the Fisher dataset (dev, dev-2, test) are 29.80%, 29.79% and 25.30% respectively.

For the SMT system, we use the phrase based translation system of Moses (Koehn et al., 2007) with sparse features. The system is trained and tuned on the train and dev partitions of the Fisher dataset respectively. The BLEU scores of the MT output for the the dev-2 and the test partitions are 65.38% and 62.91% respectively. While decoding the ASR output, we tune on the 1-best ASR output for the dev partition. With this modified system, the BLEU scores for the ASR 1-best output of the dev2 and the test partitions are 40.06% and 40.4% respectively. We use this system as the baseline for our experiments (Table 1).

We note that if we were to use the lattice oracle[6] from our ASR system as input to the SMT system, we get a BLEU score of 46.59% for the dev2 partition of the Fisher dataset. This indicates that the best gain (+BLEU) that an oracle lattice reranker could get is only 6.53%.

To tune the weights of the coarse decoder, we decode 500-best ASR outputs for the tuning set with the SMT system. This maps each ASR hypothesis to a target language translation. An OOV feature was added to handle words that were not seen by the SMT system. The tuning process was then carried out so as to maximize the BLEU with

[5] This requires the use of one ASR feature, addressed in the "Features" section

[6] Path in the lattice that has the least word error rate.

Experiment	BLEU (dev2)	BLEU (test)
Transcripts	65.4%	62.9%
Lattice Oracle	46.59%	46.17%
ASR 1-best	40.06%	40.4%
Coarse decoder	**40.26%**	**40.46%**

Table 1: Performance when using the coarse decoder interface compared to the the decoding the human transcripts, the ASR 1-best or the lattice oracle (the path in the ASR lattice with the least WER : not available during test time.)

respect to the reference translation of the ASR hypothesis selected by the coarse grained decoder. We used ZMERT (Zaidan, 2009) for tuning which was configured to expect a 300-best list from the decoder at every iteration using the Fisher dev set. 15 iterations of tuning were carried out for each experiment. We then use the tuned weight vector to decode the Fisher-dev2 and the Fisher-test set using our coarse grained decoder. We extract the one-best output and use it as input to the pretrained SMT system (description in the preceding section). Table 1 reports the results achieved the featurized coarse grained decoder.

4 Conclusions

We present a *coarse-to-fine* featurized model which acts as the interface between ASR and SMT systems. By utilizing information from the upstream (ASR) and the downstream (SMT) systems, this model makes more informed decisions about which hypotheses from the ASR word lattice may result in better translation results. Moreover, the model takes the form of a coarse finite state transducer based translation decoder which imitates the downstream system. This enables it to estimate translation quality even before the complete SMT system is used for decoding. Finally, the proposed model is featurized and may accept any weight from the ASR and SMT system that are deemed useful for optimizing translation quality.

The Spanish Fisher corpus is one of a few conversational speech translation datasets available, and we start with a strong baseline system. We therefore persevere with the experimental setup described above, even though the maximum (oracle) improvement by any rescoring method is only 6.5% BLEU, as noted above. This partially explains the small gains reported here, and suggests that this method should be evaluated further on another corpus, e.g. the Egyptian Arabic translation dataset, with greater headroom for improvement.

Acknowledgments

This work was partially supported by NSF award No IIS 0963898 and DARPA contracts No HR0011-12-C-0015 and HR0011-51-6285. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NSF, DARPA or the U.S. Government.

References

Srinivas Bangalore and Giuseppe Riccardi. 2000. Finite-state models for lexical reordering in spoken language translation. In *Proceedings of the Sixth International Conference on Spoken Language Processing*, pages 422–425.

N. Bertoldi and Marcello Federico. 2005. A new decoder for spoken language translation based on confusion networks. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 86–91.

N. Bertoldi, R. Zens, and Marcello Federico. 2007. Speech translation by confusion network decoding. In *Proceedings of the IEEE International Conference in Acoustics, Speech and Signal Processing*, volume IV, pages 1297–1300.

Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008a. Generalizing word lattice translation. In *Proceedings of ACL-08: HLT*, pages 1012–1020, Columbus, Ohio, June. Association for Computational Linguistics.

Christopher Dyer, Smaranda Muresan, and Philip Resnik. 2008b. Generalizing word lattice translation. 2008/02//.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 177–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

L. Mathias and W. Byrne. 2006. Statistical phrase-based speech translation. In *Proceedings of the IEEE International Conference in Acoustics, Speech and Signal Processing*, volume I, pages 561–564.

Evgeny Matusov, Stephan Kanthak, and Hermann Ney. 2005. On the integration of speech recognition and statistical machine translation. In *Proceedings of the 9th European Conference on Speech Communication and Technology*, pages 3177–3180.

Hermann Ney. 1999. Speech translation: coupling of recognition and translation. In *Proceedings of the IEEE International Conference in Acoustics, Speech and Signal Processing*, volume 1, pages 517–520, March.

Matt Post, Gaurav Kumar, Adam Lopez, Damianos Karakos, Chris Callison-Burch, and Sanjeev Khudanpur. 2013. General lattice decoding for improved speech-to-text translation with the Fisher and Callhome Spanish-English speech translation corpus. *Proceedings of the International Workshop on Spoken Language Translation*.

Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely. 2011. The Kaldi speech recognition toolkit. In *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, December.

Vu H Quan, Marcello Federico, and Mauro Cettolo. 2005. Integrated n-best re-ranking for spoken language translation. In *Proceedings of the 9th European Conference on Speech Communication and Technology*, pages 3181–3184.

Shirin Saleem, Szu-Chen Jou, Stephan Vogel, and Tanja Schultz. 2004. Using word lattice information for a tighter coupling in speech translation systems. In *Proceedings of the International Conference on Spoken Language Processing*, pages 41–44.

Enrique Vidal. 1997. Finite-state speech-to-speech translation. In *Proceedings of the IEEE International Conference in Acoustics, Speech and Signal Processing*, volume 1, pages 111–114, April.

Omar F. Zaidan. 2009. Z-MERT: A fully configurable open source tool for minimum error rate training of machine translation systems. *The Prague Bulletin of Mathematical Linguistics*, 91:79–88.

Ruiqiang Zhang, Genichiro Kikui, Hirofumi Yamamoto, Taro Watanabe, Frank Soong, and Wai Kit Lo. 2004. A unified approach in speech-to-speech translation: integrating features of speech recognition and machine translation. In *Proceedings of the 20th international conference on Computational Linguistics*.

Xiaohui Zhang, Jan Trmal, Daniel Povey, and Sanjeev Khudanpur. 2014. Improving deep neural network acoustic models using generalized maxout networks. In *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014, Florence, Italy, May 4-9, 2014*, pages 215–219.