RECENT IMPROVEMENTS AND PERFORMANCE ANALYSIS OF ASR AND MT IN A SPEECH-TO-SPEECH TRANSLATION SYSTEM

David Stallard, Chia-lin Kao, Kriste Krstovski, Daben Liu, Prem Natarajan, Rohit Prasad, Shirin Saleem, Krishna Subramanian

BBN Technologies, Cambridge MA

ABSTRACT

We report on recent ASR and MT work on our English/Iraqi Arabic speech-to-speech translation system. We present detailed results for both objective and subjective evaluations of translation quality, along with a detailed analysis and categorization of translation errors. We also present novel ideas for quantifying the relative importance of different subjective error categories, and for assigning the blame for an error to a particular phrase pair in the translation model.

Index Terms—Speech-to-Speech translation, Arabic speech recognition, dialog systems

1. INTRODUCTION

This paper describes new work on a speech-to-speech (S2S) translation system earlier reported on in [1]. The system is designed to allow an English-speaking soldier to engage in translingual dialog with a native Iraqi speaker. The domain of the system is military force protection, including checkpoints, house searches, civil affairs, etc. The system combines Automatic Speech Recognition (ASR), Machine Translation (MT) and Text to Speech (TTS) technologies. The research reported in this paper was performed under DARPA's TRANSTAC program, which conducts evaluations of systems and provides common training data to develop them.

In this paper, we report on recent work on the ASR and MT components of our speech-to-speech translation system. We present results for both objective and subjective evaluations of translation quality, along with a subjective error analysis, including a categorization of errors. We also present a new method for quantifying the relative importance of different error categories. Finally, we describe a technique for localizing errors within the phrase translation model, discuss the distribution of the errors that were found, and report on an initial experiment on blocking phrase pairs with high error rate.

2. ASR IMPROVEMENTS

The ASR component of our system is the Byblos speech recognition system [2]. A key improvement in the current system is the use of Heteroscedastic Linear Discriminant Analysis (HLDA) to estimate the feature transformations. We developed a low-latency online speaker adaptation method that applies speaker adaptation on the fly during decoding [3]. In this work, online speaker adaptation is applied in the HLDA space. A speaker-dependent transformation matrix is estimated and updated whenever a new utterance is received. HLDA features are transformed to speaker-dependent feature space with the estimated matrix before decoding

To speed up the process, we used a block-diagonal transformation matrix. For a 39-dimensional HLDA feature, two smaller matrices, one is 19x19 and the other 20x20, are estimated for the first and second half of the feature dimension, respectively. This avoids an estimate of a 39x39 matrix, which requires more compute for both estimation and transformation

Table 1 shows the summary of improvements on offline test set used in the January Transtac evaluation. The absolute gain for HLDA in the July '07 system is 1.1% for English and 0.3% for Iraqi. The relative gain of the July '07 models over the January '07 models is 9.4% for English and 15.3% for Iraqi.

Language	System	HLDA	%WER
English	January 07	No	21.1
	July 07	No	20.2
	July 07	Yes	19.1
Iraqi	January 07	No	28.1
	July 07	No	24.1
	July 07	Yes	23.8

Table 1: HLDA Improvements

3. MT IMPROVEMENTS

The baseline Iraqi-to-English (I2E) system was trained on 478K sentence pairs, consisting of 2.5M Iraqi words and 3.6M English words) The baseline English-to-Iraqi (E2I) system was trained on a combination of 71K E2I sentence pairs, consisting of 560K Iraqi and 800K English words, plus the reversed I2E data. The performance of the system was evaluated on a held-out development set and validation set (size 20K for I2E and 1.7K for E2I).

3.1. Using WordNet Synonyms in SMT

It would be desirable if an MT component could apply some knowledge of word meaning during translation, such as knowledge of when two different words mean the same

thing. For example, if "trash" frequently appears in the training data but "rubbish" does not, one would still like "rubbish" to be translatable in the same contexts as "trash".. In this section, we propose a method to use such semantic information to increase the coverage of the MT system and improve alignments of infrequent words.

We do this by harnessing WordNet [4], a lexical database of words in English. We employ WordNet to generate all possible synonyms of a word. The English side of the training data is first tagged with part-of-speech (POS) tags. We then use WordNet to generate a set of synonyms given a word and its POS tag. WordNet returns synonyms for every sense of the word. These synonyms are then used to generate new sentence pairs in which the word is replaced by its synonym.

In our experiments, we only consider the approximately 200 most frequent nouns and the two most frequent senses of each of these words. The list of synonyms was filtered manually to remove synonyms which were completely out of context. A total of 66K and 5K new sentence pairs for I2E and E2I respectively were generated, and added to the training data. Table 2 shows the results on the held out development set for BLEU, METEOR, TER [5], and STER [6]. As can be seen, this technique achieved gains for BLEU and TER on I2E.

Iraqi-to-English				
WNSyn	**BLEU**	**MET**	**100-TER**	**100-STER**
No	40.2	71.1	54.8	62.8
Yes	40.8	70.7	55.4	63.3
English-to-Iraqi				
WNSyn	**BLEU**	**MET**	**100-TER**	**100-STER**
No	14.3	39.9	35.2	35.2
Yes	14.4	39.9	35.1	35.1

Table 2: MT Results for Synonomy

3. 2 Disfluency Cleaning

Spontaneous dialogs are often characterized by ill-formed and disfluent speech. Disfluencies are often not translated from source language to target language in parallel corpora, resulting in spurious word alignments. Removing such disfluencies from the corpus would help the improve the quality of the training data. Previous work on cleaning disfluencies for machine translation [7] has focused on repetitions, false starts, filled pauses, etc. In our work, we target only repeated words/phrases.

The frequency of occurrence of repeated phrases was first measured on the training data. In the I2E training data, 0.9% of the English and 1.5% of the Iraqi source sentences had repeated phrases which did not have corresponding repetitions on the translated target side. This problem was more severe for the E2I training data where the numbers are 2.7% for English and 1.2% for Iraqi. The training data was sanitized by replacing all occurrences of a repeated word/phrase with a single entry of that word/phrase. The cleaned and original versions of the data were then pooled to train the translation models, and only the cleaned version of the data was used to train the language model.

Table 3 shows the effect of cleaning repetitions on the development set. On I2E, the BLEU and STER gained by 1.5% and 1% relative respectively. However, a drop was observed in METEOR. Note that repetitions were not cleaned from the references while measuring performance. We believe this is a possible reason for the drop in METEOR since the metric would favor longer translations if repeated phrases occurred in the references. This was verified by measuring performance on the subset of the development set which had no repetitions on the source or target side. All metrics showed an improvement suggesting that the quality of the alignments and hence the phrase table used for translation had improved with disfluency cleaning. On E2I, the translation performance degraded across all metrics.

Iraqi -> English				
Disflu	**BLEU**	**MET**	**100-TER**	**100-STER**
No	40.2	71.1	54.8	62.8
Yes	40.8	70.2	55.5	63.4
English-to-Iraqi				
Disflu	**BLEU**	**MET**	**100-TER**	**100-STER**
No	14.3	39.9	35.2	35.2
Yes	13.8	39.1	35.2	35.1

Table 3: Disfluency Results

4. SUBJECTIVE ANALYSIS OF MT OUTPUT

The MT component of our system was evaluated subjectively on a test set consisting of 419 Iraqi and 429 English utterances. A bilingual judge rated each MT output on a 1 – 5 Likert scale, where a score of 5 denoted perfect translation, 4 adequate translation, 3 semi-adequate, and so on. Approximately one week of effort was required to do this. The results of this evaluation are shown in Table 4. The large difference in performance between E2I and I2E is due to the higher perplexity of the Iraqi set (586 vs. 54).

Translation	Type	Likert
E2I	T2T	4.28
	S2T	4.05
I2E	T2T	3.85
	S2T	3.35

Table 4: Likert Scores

As part of the subjective evaluation, the bilingual judge categorized and labeled the specific translation errors made by the MT. The set of error categories was created based on an initial review of the MT output. There were approximately 15 categories, which included major errors, such as dropping a concept or using the wrong sense of a word, and minor errors, such as using the singular form of a

word instead of the plural. A principle goal of this effort was to quantify the relative importance of each error category in terms of the "damage" it did to the overall translation performance, so as to better direct our efforts towards improving the system.

To quantify our notion of "damage", we first define the "Likert Error" (LER) for a translation as 5 minus its Likert score. We then define the "Total Likert Error" (TLE) of a set of translations as the sum of the LER's of the translations. Table 5 gives TLE statistics for the utterances in I2E and E2I that contain errors. As can be seen, the average TLE per error and per utterance with error is higher for E2I, but I2E has many more utterances with an error. This is consistent with the lower average Likert score for I2E above.

	# Utts	# Errs	Errs/ Utt	Tot TLE	TLE/ Err	TLE/ Utt
E2I	184	228	1.24	305	1.34	1.7
I2E	273	383	1.40	484	1.26	1.3

Table 5: Total Likert Error Stats

To determine the damage done by each category of error, we make the simplifying assumption that the damage done by an individual error is at least approximately separable from and additive to the damage done by others. The relative importance of an error category *C* is then the fraction of the TLE that can be ascribed to its instances, or:

$$TLE(C) = Count(C)*LER(C),$$

where *LER(C)* is the average damage done by instances of *C*, and quantifies the "seriousness" of the error.

Estimating LER(C) is not wholly straightforward, because many sentences have both multiple errors. For example, the same sentence might have both a "Word Sense" and an "Incorrect Pronoun" error. So we cannot determine the LER simply by averaging over instances of C. The key question is how to apportion the blame between these errors.

One might imagine various heuristic or hill-climbing approaches to this problem. Our approach instead views each annotated utterance as an equation, in which the annotator has asserted that the sum of the error labels equals the given Likert error value. The variables of this equation are the error labels, whose unknown values are the LER weights of the categories. The complete set of annotated utterances can then be viewed as a set of simultaneous equations over the LER's. That is, we seek $\boldsymbol{x}$ such that $\boldsymbol{Ax=k}$, where $\boldsymbol{A}$ is a matrix of coefficients for each equation, $\boldsymbol{x}$ is the vector of unknown LER weights, and $\boldsymbol{k}$ is the vector of annotator-assigned LER values.

Due to the variability inherent in subjective analysis, one cannot in general expect this system of equations to be consistent. To take just one example, a "Missing Concept" error might legitimately result in a higher Likert error in one sentence than in another, depending upon the missing concept itself. We must instead settle for an approximation $\boldsymbol{Ax=k+e}$, where $\boldsymbol{e}$ is the difference between the predicted and actual LER values, and seek the $\boldsymbol{x}$ that minimizes $|\boldsymbol{e}|$. This is a least-squares linear regression problem, to which an exact solution can be found by solving the equation:

$$\boldsymbol{A^TAx = A^Tk}$$

Once we have estimated the LER value for a category, we multiply it by the frequency of the category to estimate the category's TLE. Table 6 gives the solved-for LER weights and the estimated TLE's for each language direction. Note that the categories "Word Sense", "Wrong Concept", "Missing Concept", and "Pronoun Error" account for the lion's share of the TLE. ("Wrong Concept" is a word or phrase translation that is wrong in all contexts, regardless of word sense). All have high frequencies and, except for "Pronoun Error", also high weights. "Pronoun Error" has a smaller weight (approximately 1.0), reflecting its lesser importance. The error "Wrong Polarity", (e.g. "I am *not* sick" instead of "I am sick") is given a high weight as it should, but because its frequency is low, it contributes only a small share to the TLE.

Interestingly, the weights for some minor errors, such as "Word Order", are driven below 1.0, even though 1 was the lowest Likert error the annotator could give a sentence that contained an error, as fractional scores were not allowed.. Thus, the algorithm mitigates somewhat the rather severe quantization of the scoring system, which forces all imperfect but still adequate translations to have the same score. Of course, the advantage of the integer scale is that it is easier for annotators to use than real numbers. A useful future compromise would be to allow half-point scores.

Iraqi-to-English			
	%Count	LER	%TLE
Word Sense	16.2	1.73	21.3
Wrong Concept	13.3	1.96	19.9
Missing Concept	13.1	1.73	17.2
Pronoun Error	21.4	0.94	15.3
Function Words	9.7	0.87	6.4
Word Order	8.6	0.83	5.5
Wrong Polarity	2.6	1.80	3.6
Other	15.1	--	10.8
English-to-Iraqi			
	%Count	LER	%TLE
Word Sense	17.1	1.88	23.5
Wrong Concept	14.5	2.00	21.4
Missing Concept	10.1	1.94	14.3
Pronoun Error	25.9	1.01	19.1
Function Words	10.5	1.07	8.2
Word Order	8.8	0.81	5.2
Wrong Polarity	0.4	2.0	0.6
Other	12.7		7.7

Table 6: Estimated Likert Error values

5. LOCALIZING ERRORS TO PHRASE PAIRS

Automated metrics such as BLEU, METEOR, or TER only tell us how wrong a set of translations are, not *what* is wrong with those translations. Human analysis, on the other hand, can tell us what is wrong, but is very labor intensive. We would like to find a way to use automated metrics to tell us where the errors are in the sentence, and even where the errors came from

Iraqi -> English				
PPBlk	**BLEU**	**METOR**	**100-TER**	**100-STER**
No	40.2	71.1	54.8	62.8
Yes	40.3	71.0	55.1	63.1

Table 7: Results for Phrase Pair Blocking

It would be desirable to have some way of aligning the MT output and reference translation such that errors are revealed. The key problem, of course, is that the MT output translation can vary in word order and word choice from the reference, yet still be correct, so simple edit-distance is not useful. The TER metric is an improvement in this regard, since it adds a sub-string shift operation, and so can accommodate variation in word order. But TER has no concept of word meaning, and so cannot take variation in word choice into account. An alternative is the recently developed metric Semantic Translation Error Rate (STER) [6], which is a version of TER modified to use the same Porter stemming and WordNet synonymy matching as METEOR. STER also differs from TER in that it disallows the alignment of concept and stop words. Thus a substitution error identified by STER is more likely to be meaningful.

Note that the SMT decoder also outputs an alignment; specifically the alignment between source and target phrases given by the phrase pairs used in the decoding result. By composing the decoder and STER alignments, one can obtain an alignment between the phrase pairs and the reference translation, from which one can directly read off the insertion and substitution errors that can be hypothesized as coming from that phrase pair.

A test run on our development set generates approximately 17K insertion and substitution STER errors. Of these, almost half are generated by a phrase pair that generates only one error, and which was usually used only once in the whole dev set. By manually sampling the head and the tail of this distribution, we found that approximately 60% of the phrase pair "indictments" are spurious, in that they involve either a still-unrecognized equivalency of meaning between words or phrases, an error in the reference translation, or a noise phrase (e.g. "you know", "I mean"). At the head of the distribution, about 15% of the indicted phrase pairs were found "guilty", with another 13% being context-dependent (the remainder had errors of mixed type). At the tail of the distribution, the guilty rate doubles to 30%.

As a very initial experiment, we manually filtered the high-error phrases for I2E, and identified 408 phrase pairs as definitely. We modified our MT decoder to block these phrases during decoding. The effect of phrase pair blocking on the translation performance on an independent validation set is shown in Table 7. A very slight improvement in STER and BLEU is observed, along with a small drop in METEOR. We plan to continue this work with more sophisticated weighting schemes in the future.

6. CONCLUSIONS AND FUTURE WORK

We have presented a set of improvements and attempted improvements to our system, including HLDA for ASR, and synonymy and disfluency cleaning for MT. We have also presented an algorithm for weighting subjective MT error categories, which can be used to target work on improving the system's performance. Finally, we have presented an automatic method for locating possible errors in the phrase pairs of the translation model, and given results of an initial experiment in using this algorithm.

In the future work, we plan to run our Likert error-weighting method on larger test sets, multiple annotators, and if possible, on the outputs of multiple MT systems. It would also be worthwhile to investigate the use of half-point intermediate Likert score values, so as to relieve the current severe quantization of this metric. Finally, we plan to continue work with the phrase-pair error identification scheme, especially with more sophisticated weighting schemes for down-weighting phrase pairs..

7. REFERENCES

[1] D. Stallard, F. Choi, C. Kao, K. Krstovski, P. Natarajan, R. Prasad, S. Saleem, and K. Subramanian, "The BBN 2007 Displayless English/Iraqi Speech-to-Speech Translation System," Proc. INTERSPEECH 2007, pp. 2817-2820, Antwerp, Belgium, August 27-31 2007

[2] L. Nguyen, and R. Schwartz, "Efficient 2-pass N-best Decoder," Proc. EUROSPEECH, ISCA, Rhodes, Greece, Sep. 1997

[3] D. Liu, D. Kiecza, A. Srivastava, F. Kubala, "Online Speaker Adaptation and Tracking for Real-Time Speech Recognition," Eurospeech'05, pp 281-284, Lisbon, Portugal, September, 2005

[4] http://wordnet.princeton.edu/

[5] M. Snover, B. Dorr, R. Schwartz, L. Micciulla, and J. Makhoul, "A Study of Translation Edit Rate with Targeted Human Annotation," Proc. of Association for Machine Translation in the Americas, 2006.

[6] K. Subramanian, D. Stallard, R. Prasad, S. Saleem, P. Natarajan, "Semantic Translation Error Rate for Evaluating Translation Systems", to appear in Proc. of ASRU, Dec 2007

[7] M. Honal and T. Schultz, "Correction of Disfluencies in Spontaneous Speech using a Noisy-Channel Approach", Proc. of Eurospeech 2003.