AN ENGLISH-PERSIAN AUTOMATIC SPEECH TRANSLATOR: RECENT DEVELOPMENTS IN DOMAIN PORTABILITY AND USER MODELING

Panayiotis G. Georgiou, Abhinav Sethy, JongHo Shin, Shrikanth Narayanan
Speech Analysis and Interpretation Lab,
University of Southern California,
Los Angeles, CA 90089-2564
Email: georgiou@sipi.usc.edu

ABSTRACT

In this paper we describe the English-Persian speech to speech translation device, and provide insight on the lessons learned during the first phase of development of this system. We start by giving an overview of the underlying components of the device: the front end ASR, the machine translation system and the speech generation system. Challenges such as the sparseness of available spoken language data and solutions that have been employed to maximize the obtained benefits from using these limited resources are described. In addition, we present results on system evaluation, which identify that a large degree of the system errors can not be attributed to machine performance, but rather to user operation and process losses. The identification of this large margin of improvement led to the initiation of a new research field: user modeling in cross-lingual mediated communication, and we describe our first results in user type classification.

1. INTRODUCTION

Creating a speech-to-speech (S2S) translation system presents significant challenges. These arise not only due to the complex nature of the individual technologies involved in a S2S translation, but also due to the intricate interaction that these technologies have to achieve. Additionally, a great challenge for the specific S2S translation system we are presenting stems from the great discrepancy in the structure of the English and Persian (Modern Persian is called Farsi by native speakers) languages, as well as the extremely limited amount of data for the Persian language. Furthermore, for the most part the Persian writing system (employing the Arabic script) lacks the explicit inclusion of vowel sounds, thus resulting in a very large amount of one-to-many mappings from transcription to acoustic and semantic representations.

There have been major efforts in creating end-to-end S2S translation systems by several research institutions, for example, [1, 2, 3, 4]. The goal of such systems is to be truly cognizant of the interaction, and serve as a communication aid than a mere message conduit.

It should be noted that unlike other languages such as Arabic, Mandarin, and European languages, the Persian language had not been investigated by the speech community prior to this project. This resulted in two immediate problems: The lack of available data, and the lack of standardized transcription schemes.

2. SYSTEM OVERVIEW

Our system comprises several spoken language components that act in a collaborative manner, and a visual and control graphical user interface (GUI). A functional block diagram is shown in Fig. 1.

In short, all messages are received by an *audio client* (AC) acquisition interface, the audio is converted to text through the *Automatic Speech Recognition* (ASR) server, translated by the *Machine Translation* (MT) component and gets re-synthesized in the target language by the *Text-To-Speech Synthesizer* (TTS) before being played out by a *playout client* (PC) component. In our architecture all components are networked, so they can run on multiple devices. In addition all components can run under both Windows and Linux OS's. We have often for example run the system on two laptops, a Windows and a Linux one.

The introduction of the two very low requirement audio acquisition and playout (AC & PC) clients serve for allowing remote usage of the translation system through low-power devices such as handhelds.

In addition to all the transparent to the user components, the device includes a *Dialog Manager* (DM) and a *Graphical User Interface* (GUI) (including a user guide and help system). In fact, as will be discussed later, the introduction of the DM, in combination with multi-translation paths, has resulted in enabling the device to reach performance much greater than the performance of the subcomponents. To make sure our thin-client architecture is not affected by the usage of the DM/GUI, the logic is built in the DM server, while the GUI is a low-requirement separate component.

Furthermore it should be noted that some of the components are themselves comprised of multiple subcomponents. For example the TTS depending on the required synthesis may operate in utterance-level units, on word-level, or in diphone level, each of these requiring different processing.

In our architecture, all messages are broadcast with a tag that includes among other information the message's originating and proposed terminating point, and are visible to all subsystems. This allows for ease of collaboration and monitoring of the internal communication channels by the DM, which can interrupt and re-

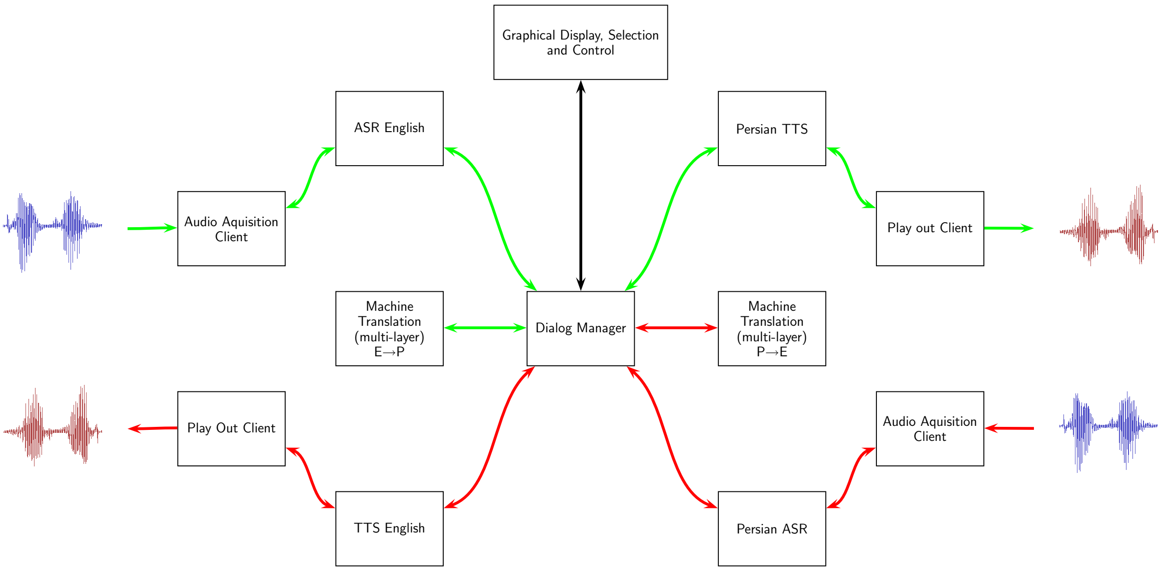

Fig. 1. Simplified block diagram of system components

quest corrective action by the user. The most probable corrective actions are requests for repeat, rephrase, confirmation, and disambiguation where the user is asked to choose the utterance from a list of options (using the speech and/or the GUI).

3. DATA REQUIREMENTS

In designing such a device, different kinds of data are required for the different components. For the ASR we require statistical pattern matching for converting sound to phonemes, phonemes to words and words to utterances. Of these the phonemes to words is assumed to be deterministic, hence usually manually constructed, while the acoustic and language models respectively require large amounts of acoustic data and large amounts of transcripts of representative, in domain dialogs.

For statistical translation, parallel transcripts are required in order to heuristically learn phrase translations using word-based and phrase-based alignments. In addition, we require a language model of the target language for rescoring these phrase translations. For the English-Persian pair, there were no available parallel data[1].

For dialog models, the requirement is of transcripts of in-domain cross-lingual interactions, as close as possible to what the system would see in the real usage environment. However it is unclear how much data are needed before the technology limitations are reached. The potential variability of constructing utterances is a super-set of the lexical variability due to a combinatorial explosion. Simply put, there is an infinite number of concept sequences that a dialog can go through.

4. DATA COLLECTION & TRANSCRIPTION

In addition to lack of data for building the S2S device, there was no appropriate existing transcription scheme for encoding orthographic and phonetic information in Persian.

Persian employs the Arabic script, which is is the second most widely used written script after Latin. Languages that use this script have posed a problem for automated language processing such as speech recognition and translation systems. Although there were many existing labeling schemes for a wide variety of languages there was still no transcription system for languages like Arabic, Persian, Dari, Urdu and many others, which use the Arabic script [5, 6]. The motivation for creating the proposed class of transcription schemes stemmed from the necessities of our system: appropriate schemes for unique encoding of phonetic and lexical information.[2]

The first issue is that of unusual prevalence of *homographs* due to the convention of ignoring most vowels. In the Arabic script the vowels are represent by diacritic marks, which however are never found in the Persian written language. This results in many to one mapping from the concept to the written form. An analogous example in English would be poetry, peter, pater, pewter, peatier, Patra etc being represented as ptr. The problem of homographs creates large ambiguity in language modeling where lexical states are merged. For example a bigram like “Hello Peter” will have equal probability in the language model to the bigram “Hello pewter”.

The second is the high occurrence of *homophones*. English examples of this include pair-pear and peer-pier. Homophones make the acoustic discrimination of these words impossible thus discrimination depends solely on the language model.

The first step in creating an encoding scheme for the Persian language was to define a romanized, (left to

[1] There exist several parallel corpora in other language pairs, some available freely. e.g. the Europarl corpus

[2] Note that following our developments in Persian, we proposed and employed similar schemes for Pashto and Arabic.

Models	Data Type	Initial	Min Req.	Acquired
Acoustic Models	Persian Audio w/ Transcripts	–	100h	7h
Language Models	In-Domain Transcripts	–	1M words	300k[a]
Language Models	Spontaneous Transcripts	–	∼200M	∼[b]
Translation Models	In-Domain parallel cross-lingual Transcripts	–	1M words	300k[c]
Dialog Models	In domain cross-lingual spontaneous transcripts	–	?	?

[a] These were not of real Persian speech but translated from the English data collected

[b] These transcripts are of read speech

[c] These were not of real Persian speech but translated from the English data collected

Table 1. Data requirements, and start and final acquired data

Gloss	Arabic Script	USCPers	USCPers+	USCPron
One Hundred	صد	$d	$ad	sad
Dam	سد	sd	sad	sad
Six	شش	SS	SeS	SeS
Lung	شش	SS	SoS	SoS

Fig. 2. Examples of the limitations of both the Arabic script and USCPers (a direct romanized version of Arabic script), which share a one-to-one mapping, in representing transcribed speech. Due to the lack of vowels in the orthographic transcription, it is extremely common that two completely different acoustic and semantic representations lead to the same orthographic transcription. Our lexicalization convention called USCPers+ includes vocalic information thereby retaining the unique mapping between the acoustic input and lexical output of ASR. USCPron provides the pronunciations for the lexical items represented in USCPers+.

right) version of the Arabic script. This is a convenient step for readability and transcription purposes and being a one-to-one mapping to the original script, it does not affect the information content. This mapping leads to the USCPers transcription scheme, which employs the ASCII code and is shown on the third column of Fig. 2.

The second necessary transcription scheme is one that encodes the phonetic information as required by the ASR and TTS components in order to represent the pronunciation(s) of each word. For this purpose we have created the USCPron transcription system.

Creating a phonetic transcription scheme addresses the problem of encoding the phonemes in the ASR lexicon, but does not solve the lack of a sufficient lexical transcription. Both USCPers and USCPron are suffering from a many-to-one mapping for representing lexical entries as shown on Fig. 2, due to the homograph and homophone issues respectively. To address this we proposed a different transcription scheme that enables a one-to-one mapping between the acoustic representation (excluding user variability) and the transcription method, which we introduce as USCPers+. An in-depth analysis of our transcription schemes is given in [7], but the main idea is to borrow the phonetic information from USCPron and augment the USCPers. This ensures that homophone letters (consonants from USCPers) are preserved while phonetic information from vowels augments the USCPers+ transcription.

5. STRATEGIES FOR ADDRESSING RESOURCE CHALLENGES AND DATA SPARSENESS

As mentioned in table 1 the development of S2S systems requires spoken language data of various forms and amounts.

On the Persian side, the only available data was the Farsdat Speech database that includes 20 read sentences from 300 speakers, diverse in age, sex, education level, and dialect, for a total of 6000 utterances (available from ELDA). The transcripts were unsatisfactory for a speech recognition application and hence had to be recreated. In order to augment these transcripts, we recruited Persian speakers from the diverse Los Angeles area and recorded read and semi-spontaneous speech data. The semi-spontaneous speech was solicited in a *Wizard of Oz* (WoZ) scenario, while read speech was collected and verified by the speakers themselves with an interactive data collection tool.

These data were not suitable for language modeling: For language models, in parallel to developing data mining techniques described below, an extensive data collection effort took place [8] where 300 Standardized Patient (medical student-actor patient) sessions were run at the USC campus in collaboration with the Medical School.

Standardized Patients (SPs) are actors carefully trained to portray all of the characteristics of a real patient, in order to provide the opportunity for a student to learn, or be evaluated, on clinical skills first hand. The practice of using SP's dates to the 1960's and was introduced at University of Southern California School of Medicine. For this domain, SP testing provides us with a unique opportunity to collect doctor-patient interaction data without any privacy concerns, while maintaining a high degree of realism. SP's are trained to portray cases based on actual patients encountered by physicians and are trained to simulate, not only the signs and symptoms, but also the emotional and personality characteristics of the patient

The collected SP data was subsequently transcribed and translated thus resulting in over 300k words of in domain data in both English and Persian. These, as well as targeted smaller collections through domain publications, were also used to extract the vocabulary and handcraft the dictionary in Persian.

6. COMPONENT TECHNOLOGIES OVERVIEW

6.1. Acoustic modeling

For both English and Persian systems we used front-ends with 16 kHz sampling rate, 10 ms frame advance rate, 12 mel frequency cepstral coefficients plus normalized energy and first- and second-order differences (39 features). The phonetic set for Persian has 34 units (29 phonemes, silence, breath, lipsmack, garbage, and laughter), while the English one contains 55 units (50 phonemes). Three-state triphone hidden Markov models (HMMS) were trained with state clustering. In the real-time system we employed the SONIC [9] speech recognition engine while we used HTK [10] and internally developed software for research purposes.

The Persian recognizer was bootstrapped with an English phoneme mapping into the target language as an initialization step for the models, which were subsequently adapted or re-trained using the limited amount of data available. We investigated three different phonetic mapping methods: a knowledge based one, a data driven phoneme mapping, and a data driven state mapping method. The data driven techniques employed the Earth Movers Distance (EMD) method, which tries to minimize the amount of work needed to change one GMM into another. The EMD based techniques gave gains at the initial stages of development when the acoustic data was limited, but the advantage was insignificant once the available data size increased. In that case the models derived from cross-lingual phonetic alignment were used only for alignment purposes.

6.2. Language modeling

The Language Modeling technique used was a class based trigram. Classes were handcrafted, although significant information was extracted from publicly available resources (e.g. medication names, people names etc). Due to the significant man-hours that it takes to collect and transcribe domain data, at the beginning of the project we initiated research in domain specific datamining techniques. A more detailed discussion of these will follow later.

6.3. Machine Translation

The approach we employ for the Machine Translation unit is twofold. A concept classifier is applied as the main translator unit of the system because of its faster and more accurate performance[3], while a statistical machine translator (SMT) is kept as the backup unit for the cases when the classifier response is not within an acceptable confidence margin. These cases should be relatively infrequent if significantly large number of classes are chosen for the classifier based MT.

The classifier has been implemented using n-gram language models and competitive scoring, by identifying appropriate concepts through medical phrase books. The concept clusters were enriched through paraphrasing using interactive online games, expansions through INTEX, manual paraphrasing etc.

The second method, to be used in the case of poor classification confidence, is the Stochastic MT method. SMT is based on n-gram to n-gram translation and can generate a translation for every input sentence (within vocabulary). However, accuracy of the SMT, although a function of the training corpus, is in general lower than the accuracy of the classifier. The best performance for SMT can be achieved by employing a large amount of bilingual parallel text for training. This training corpus can be used to build a language model for the target language along with a statistical translation table, which relates words in the source language and their counterparts in the target language. For training our SMT system we employed the SP data we collected after transcription and translation.

6.4. Text to Speech

We rely on a hybrid unit selection based speech synthesis. In the best case, when the output is chosen from a classifier based MT, the generated phrases are known a priori. Hence, our first system release enabled us to use a prompt based system for spoken output. The other end of the unit selection possibility is through diphone concatenation. We have implemented such a synthesizer based on Festival for Persian, and used default models for English.

6.5. Dialogue Manager and User Interface

The dialogue manager component is closely bundled with the user interface and has the main task of using the other components to promote effective communication between the participants. Together the two enable selection of selection mode (automatic translation, confirmation, selection, or dynamic combinations of the above based on confidence thresholds), present the user with visualization of the dialog, including history and real time ASR and translation results, redirects all the messages to the other components, etc.

The infinite concepts that users can present to the device have hampered the deployment of dialog management in large domains, such as this. However there are potentially significant advantages to dialog modeling as demonstrated in smaller tasks such as under the DARPA Communicator project, which aimed in providing intelligent conversational interfaces and was evaluated for the task of air travel reservations.

Our preliminary results [11] demonstrate that the Dialog Manager can aid in rescoring concepts based on the conversation history. This is obvious for human communication, where a greeting for example is very likely to be followed by a greeting response. To achieve prediction we clustered the commonly seen concepts of the dialog (over 1200 of them) and we assumed a representation of the concept space as the states representing these states plus a null state where the concept can not be identified.

[3] Note that in addition to being more accurate the concept based translator provides oracle back-translations, thus allowing the user to make an extremely accurate decision on accepting for translation or repairing

7. EVALUATION

The S2S device has been tested in various settings. It has been evaluated by the funding agency, by the U.S. military, and we additionally run various types of experiments internally using the SP scenario in a cross-lingual communication scenario.

The component performance in the DARPA final evaluation testing is as depicted on Table 2. This performance by it self would suggest an acceptable quality of the device for real use. In addition the concept transfer rate in the DARPA experiments was quite acceptable at 78%.

DARPA Evaluation results		
	English	Persian
ASR WER	11.5%	13.4%
	E to P	P to E
IBM BLEU (text)	0.31	0.29
IBM BLEU (ASR)	0.27	0.24
Overall concept transfer		78%

Table 2. DARPA evaluation on medical domain. Component and Concept measures as: ASR word error rate (lower is better), SMT BLEU score (higher is better) with the clean text transcript input or with the ASR output as an input.

However, during the final evaluation we saw that communication between users often broke down despite the very high accuracy of the subcomponents on those occasions. This observation prompted us to design the sequence of experimental interactions under the cross-lingual SP setting to further clarify the origin of these breakdowns. Identifying the reason for these breakdowns allows us to improve the system beyond only technological advances by introducing a socio-technical design process where human experiments drive technological advances. This iterative process of usability experimentation and system redesign revealed a key finding for framing future improvements to our translation system. System functionality, and communicative success between users, are not always strongly correlated. Often times the system components will ”function” but there will be no concept communication between users and vice-versa.

8. LESSONS LEARNED

In developing the traslation device we faced several expected and unexpected hurdles. The first one is the limited data availability in new languages and new domains that limit *rapid porting* and in the absence of large amounts of data result in lack of *robustness*. The second lesson we learned was that *evaluation* was not as straight forward as originally believed. We could simply not assume that having good component technology results in the cognitive synchrony of the participants. This led finally to the realization that careful *user modeling* and the design of a communication system – and not just a device – is required along good component performance.

These led to several research directions. *Data Mining*, *Evaluation and Socio-Technical Design* and *User Modeling*, are some of the resulting research directions currently under investigation, briefly described below.

8.1. Data Mining

The lack of data at the start of the project, led us immediately to consider data mining techniques for enriching our language models using large resources such as the WWW. These efforts were geared initially towards collecting background model data for the Persian language. Mining, cleaning and converting to USCPers was done by first mining specific Persian newspaper sites. On the English side, we had sufficient background model data so we decided to mine using a bag-of-words (BOW) and n-grams approach for in domain-data, with the small amounts of data we had as a seed.

With the evolution of our work, we realized that rapid development in new domains, and new languages, as well as robust language modeling are two very significant factors in the creation of S2S translation systems. This led us to significant research on data mining for language modeling.

Both our BOW approach and other algorithms in recent literature are based around a scoring function that measures the similarity of each observed sentence in the web-data to the in-domain set and assign an appropriate score. Subsequently there is a certain kind of thresholding function on the rank ordered list, that eliminates low-rank data based on a held out set. Rank ordering schemes, however, do not address the issue of distributional similarity and select many sentences which already have a high probability in the in-domain text. Adapting models on such data has the tendency to skew the distribution even further towards the center. For example, in our doctor-patient interaction task short sentences containing the word ‘okay’ such as ‘okay’,‘yes okay’, ‘okay okay’ were very frequent in the in-domain data. Perplexity and other similarity measures assign a high score to all such examples in the web-data, boosting the probability of these words even further while other pertinent concepts seen little in the domain data, can receive low rank and be rejected.

We address the issue of distributional similarity through incremental algorithm which compares the distribution of the selected set and the in-domain examples by using a relative entropy (RE) criterion.

To address distributional similarity we developed an incremental greedy selection scheme based on relative entropy. Our method selects a sentence if adding it to the already selected set of sentences reduces the relative entropy with respect to the in-domain data distribution.

A short description is given here that utilizes a unigram for explanatory purposes. The algorithm is initialized with a language model from a subset of the existing in-domain data P_{init} while the language model of the whole existing in-domain data is P.

We convert the model P_{init} into a set of counts $W(i)$ for words i in the vocabulary V. Our selection algorithm considers every sentence in the corpus sequentially and evaluates it's relative entropy – using a fast implementation – to the initial model P.

Suppose we are at the j^{th} sentence s_j. We denote the count of word i in s_j with m_{ij}. Let $n_j = \sum_i m_{ij}$ be the number of words in the sentence and $N = \sum_i W(i)$ be the total number of words already selected. The relative entropy of the maximum likelihood estimate of the language model of the selected sentences to the initial model P is given by

$$H(j-1) = -\sum_i P(i) \ln \frac{P(i)}{W(i)/Nj} \tag{1}$$

The model parameters and the RE remain unchanged if sentence s_j is not selected. If we select s_j , the updated r.e is given by

$$H(j-1) = -\sum_i P(i) \ln \frac{P(i)}{(W(i)+m_{ij})/(N+n_j)} \tag{2}$$

Our method will accept inclusion of s_j if including it decreases the RE with the in-domain distribution, i.e $H(j) < H(j-1)$.

However given the fact that m_{ij} is sparse, we can split the summation $H^+(j)$ into

$$\begin{aligned} H^+(j) &= -\sum_i P(i) \ln P(i) + \\ &\quad + \sum_i P(i) \ln \frac{W(i)+m_{ij}}{N+n_j} \\ &= H(j) + \underbrace{\ln \frac{N+n_j}{N}}_{T1} \\ &\quad \underbrace{- \sum_{i, m_{ij} \neq 0} P(i) \ln \frac{(W(i)+m_{ij})}{W(i)}}_{T2} \end{aligned}$$

Thus s_j is selected if $T1 < T2$. For more refined selection, the threshold can be offset by $thr(j)$, a function of j, resulting in $T1 + thr(j) < T2$.

8.1.1. Experiments

We run experiments using the transonics data we currently have available: 50K in-domain sentences (300K words). The first step is mining data from the web (5GB) using automatically generated queries, which after filtering and normalization amount to 150M words. The test set for perplexity evaluations consists of 5000 sentences(35K words) and the heldout set had 2000 sentences (12K words). The test set for word error rate evaluation consisted of 520 utterances. A generic conversational speech language model was built from the WSJ, Fisher and SWB corpora interpolated with the CMU LM. All language models built from webdata and in-domain data were interpolated with this language model with the interpolation weight determined on the heldout set.

	10K	20K	30K	40K
No Web	19.8	18.9	18.3	17.9
AllWeb	19.5	19.1	18.7	17.9
PPL	19.2	18.8	18.5	17.9
BLEU	19.3	18.8	18.5	17.9
LPU	19.2	18.8	18.5	17.8
Proposed	**18.3**	**18.2**	**18.2**	**17.3**

Table 3. Word Error Rate (WER) with web adapted models for different number of initial sentences.

The WER results in Table 3 show that adding data from the web without proper filtering can actually harm the performance of the speech recognition system when the initial in-domain data size increases. This can be attributed to the large increase in vocabulary size which increases the acoustic decoder perplexity. The average reduction in WER is close to 3% relative. For comparison reasons, the proposed method accepts only 12% of the 150M words, while PPL, BLEU, and LPU include 91%, 89% and 87% of the mined data, thus increasing unnecessarily and in a damaging fashion.

8.2. Evaluation – Socio-Technical Design

Observing the high rate of communication breakdowns that were not attributed to device errors, we set out to investigate how human needs should drive technological developments, especially when these human needs affect the way we communicate.

Human-centered design is the process by which the development of the device should be guided by predicted improvements in human communication, and constantly refined with the user's needs, It is this cycle of refinement that maintains the link between the functionality of a product and its usability and usefulness.

Typically device development has been approached in two distinct ways, the technical approach – what can technology do – and the social approach – what are the human needs. The perfect merger between social requirement and technological capability is difficult to reach in settings that involve human interaction. The trade-off between what people want and what technology can deliver is referred to as the "social-technical gap" [12].

We assert that identifying and minimizing this gap in our S2S translation device should be an important part of the development process, that can not take place unless careful analysis of test cases takes place. To do this we designed a set of experiments to identify the importance of this gap, and to analyze potential reasons behind it.

Structured analysis of the fifteen videotaped cross-lingual, machine-mediated interactions (15 minutes each) revealed three broad categories of errors. These errors were all identified as factors which reduced the performance of the system relative to the communication goal provided to the users (a medical diagnosis and treatment recommendation).

The first category, machine errors, included difficulties related to system performance, such as poor ASR or bad translation class. These were primarily "known

Machine related errors	157	36%
User operational errors	109	25%
Process losses	170	39%
Total errors	436	

Table 4. Results from Transonics user studies revealing the causes of communication breakdown.

errors” and accounted for roughly one third of the total cause of communication breakdowns.

However the stunning observation was that about two thirds of the errors could be attributed to the other two categories:

Operational errors encompass typical user error actions such as pressing the wrong button, recording one’s voice at the wrong time and generally using equipment in unintended ways.

Process losses [13, 14] refer to goal specific errors which, in the case of S2S translation often lead to communication breakdowns. Some examples of process loss behaviors include, forcing translation of proper names through the device even though they sound the same in both languages (e.g. Aspirin or Excedrin), lack of flexibility in accepting ASR results that are not word-for-word, attempts to employ culturally specific humor or over reliance on prosodic information for transferring meaning. In addition, many users did not adequately take advantage of the multiple other communication modalities that are not subject to the lexical barriers. The direct human-human communication channel is a very high bandwidth one and can convey information such as gestures and emotions. However, few users for example pointed to the appropriate body part when describing where they feel pain.

The high degree of errors attributed to factors other than the component performance led us to invest significant effort in training material, including the creation of reference manuals, paper, web-based and video tutorials, and quizzes. It additionally led us to initiate research in user modeling techniques that will hopefully enable narrowing the socio-technical gap.

8.3. User modeling

Acknowledging the high error due to the socio-technical gap led us to a research direction of identifying user characteristics so that we can design the device to appropriately respond to such behavior. We had noticed during our evaluation experiments that some users tend to be very picky in their expectations of the device. For example, we have observed that certain users are more accepting of minor errors in translation and recognition (e.g., AND DO YOU HAVE FEVER when they actually spoke DO YOU HAVE FEVER) while others completely reject the machine’s hypothesis as not their intended utterance, despite the fact that it conveys for all practical purposes the identical meaning.

Close analysis of the data confirmed the that for the same average WER, one user retried 95% of the time while another user only 65%. We therefore propose modeling users in one of three categories(Accommodating, Normal and Picky) based on the analysis of the active participant, the doctor, and to train a system that can detect in which category the user belongs based on their behavior patterns. We hope this will enable the future research in building agents that can appropriately adapt the system according to user behaviors as similarly demonstrated in previous studies [15, 16, 17].

Due to the small amounts of data, we decided to quantize the quality of the system’s recognition in two regions a High Quality (HQ) and Low Quality (LQ) region. We established experimentally the threshold in the WER separating the two regions by assuming that a user retries if the system performance falls below this threshold. The separating point of the accepted versus retried utterance clusters gave us a WER threshold of 56%.

Similarly to deciding the HQ/LQ boundaries, we decided to investigate the clustering of user types based on their retry rate. The system error was found to be similar for all users (all were American English speakers and used the device within domain) so we clustered them according to their overall retry rate. Although the average WER is relatively constant from user to user, the error that users consider acceptable is not as demonstrated by the variable degree of retries.

Figure 3 shows the large degree of variability of the users in the LQ region and further enforces the theory that some users are more accommodating than others to device errors.

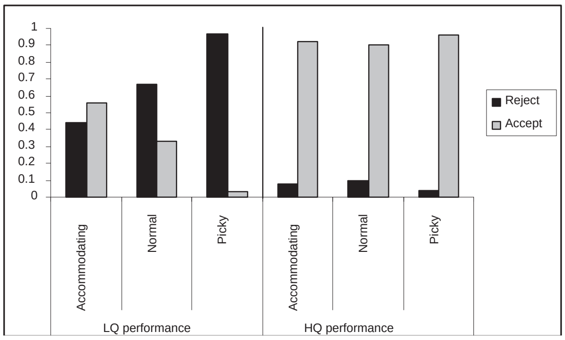

Fig. 3. Conditional Probability Table(CPT) over user behaviors(discrete) – Retry and Accept. We can see the large impact of user accommodation by observing the LQ region

Based on the analysis of this user trait, we designed a Dynamic Bayesian Network with parent random variables the user type and ASR quality (HQ/LQ) and a child the user response (accept or retry). In addition we assume a Markov chain where the user type at time t depends on the one at time $t-1$. We ran 15 experiments to test the validity of the model using 14 interactions for training and the 15th for testing. Although we do not expect this to be a large enough set for evaluating the accuracy of the algorithm, the obtained result – correctly identifying 13 of the 15 interactions – strengthens our belief of a valid and necessary model.

9. CONCLUSIONS

The development of a speech to speech translation system requires multifaceted research. While significant research has taken place over the years focusing on performance of sub-components, the specific paradigm of human involvement raises a whole new breed of issues, and hence research directions, that need to be explored. In addition, the variability in the potential end-use of these devices further increases the desire to be able to rapidly prototype and deploy in different domains, languages, dialects, form-factors (and hence model sizes), and as we have found out through potential users[4] different levels of acceptable accuracy or confidence in quality, speed, robustness, interface etc.

We believe that the field of machine mediated speech to speech translation is a very exciting one, and it provides a plethora of possible research questions in a range of research fields.

10. ACKNOWLEDGEMENTS

This work was supported by the DARPA Babylon/CAST program, contract N66001-02-C-6023 and by the DARPA TransTac program contract number NBCH1050027.

11. REFERENCES

[1] S. Narayanan, P. G. Georgiou, and et al., “Transonics: A speech to speech system for english-persian interactions,” in *IEEE Automatic Speech Recognition and Understanding(ASRU) Workshop*, 2003.

[2] B. Zhou and Y. Je, “A hand-held speech-to-speech translation system,” in *IEEE Automatic Speech Recognition and Understanding(ASRU) Workshop*, 2003.

[3] K. Precoda and R. J. Podesva, “What will people say? speech system design and language/cultural differences,” in *IEEE Automatic Speech Recognition and Understanding Workshop(ASRU)*, 2003.

[4] Alan W Black, Ralf D. Brown, Robert Frederking, Rita Singh, John Moody, and Eric Steinbrecher, “TONGUES: rapid development of a speech-to-speech translation system,” in *Proceedings of HLT-2002: Second International Conference on Human Language Technology Research*, March 2002.

[5] Alan S. Kaye, *“Arabic” The World’s Major Languages*, Oxford University Press, 1987.

[6] T. Lander, “The cslu labeling guide,,” http://cslu.cse.ogi.edu/corpora/corpPublications.html.

[7] Shadi Ganjavi, Panayiotis G. Georgiou, and Shrikanth Narayanan, “A transcription scheme for languages employing the arabic script motivated by speech processing application,” in *Proceedings of the Workshop Computational Approaches to Arabic Script-based Languages. 20th International Conference on Computational Linguistics (Coling 2004)*, 2004.

[8] Robert Belvin, Win May, Shrikanth Narayanan, Panayiotis Georgiou, and Shadi Ganjavi, “Creation of a doctor-patient dialogue corpus using standardized patients,” in *Proc. LREC*, Lisbon, Portugal, 2004.

[9] Bryan Pellom, “Sonic: The university of colorado continuous speech recognizer, technical report TR-CSLR-2001-01,” Tech. Rep., University of Colorado, March 2001.

[10] Steve Young et al, “The htk book,” http://htk.eng.cam.ac.uk/docs/docs.shtml.

[11] Emil Ettaile, Panayiotis G. Georgiou, and Shrikanth Narayanan, “Cross-lingual dialog model for speech to speech translation,” in *Proc. Intl. Conf. on Spoken Language Processing*, 2006.

[12] M. S. Ackerman, “The intellectual challenge of cscw: The gap between social requirements and technical feasibility,” *Human Computer Interaction*, vol. 15, pp. 179–204, 200.

[13] I. D. Steiner, *Group processes and productivity*, Academic Press, New York, 1972.

[14] J. Nunamaker, A. Dennis, J. Valacich, D. Vogel, and J. George, “Issues in the design, development, use and management of group support systems.,” in *Group Support Systems: New Perspectives*, L. M. Jessup and J. S. Valacich, Eds., pp. 123–145. Macmillan, New York, 1993.

[15] K. Jokinen and K. Kanto, “User expertise modelling and adaptativity in a speech-based e-mail system,” in *Proc. of the 42nd Annual Meeting of the Association for Computational Linguistics ACL-04*, 2004.

[16] C. A. Kamm, D. J. Litman, and M. A. Walker, “From novice to expert: The effect of tutorials on user expertise with spoken dialog systems,” in *Proc. of the 5th International Conference on Spoken Language Processing (ICSLP)*, 1998.

[17] K. Komatani and et al., “Flexible guidance generation using user model in spoken dialogue systems,” *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pp. 256–263, 2003.

[4] military and medical,in both office and field environments, under noisy or quiet conditions, with time constraints or not, with or without freedom of using eyes or hands for navigating interface etc