Modeling and Automating Detection of Errors in Arabic Language Learner Speech

Abhinav Sethy, Shrikanth Narayanan
USC/Department of Electrical Engineering
Speech Analysis and Interpretation Laboratory
Los Angeles, CA 90089
{sethy, shri}@sipi.usc.edu

Nicolaus Mote, W. Lewis Johnson
USC / Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292
{mote,johnson}@isi.edu

Abstract

Human tutors, in dealing with non-native speakers, draw from their knowledge of common learner mistakes to understand learner speech and offer effective corrective advice. In this paper we present our work towards embedding some of this knowledge in the speech recognition and learner speech error detection subsystems of the Tactical Language Training System (TLTS). We discuss the implementation and effectiveness of our methodology which uses a combination of rule based and probabilistic models derived from linguistic knowledge about the target language and annotated speech to identify potential learner errors, detect them using ASR and to provide the user with corrective feedback based on error severity and factors such as the learner history. Evaluation results show that our system can provide effective feedback to the learner with good accuracy.

1. Introduction

The Tactical Language Training System uses an Automated Speech Recognition (ASR)-driven pedagogical environment to guide the learner in rapid acquisition of basic language and cultural skills[1]. We currently have systems built for Levantine and Iraqi Arabic, and have a Pashto system under development. Language learners perform a variety of interactive exercises, speaking to the computer and getting feedback on their choice of responses and pronunciation. They also practice in an interactive game in which they must communicate with non-player characters, giving learners a task-based method of learning the language. This paper focuses on learner error modeling for Levantine Arabic, as the Levantine system is the most mature, and has the most learner data to analyze system functionality.

The interaction between the learner and the system is mediated by the Pedagogical Agent subsystem, which is modeled after one-on-one tutoring conventions. Interaction in this subsystem typically consists of a perception-action loop in which the agent listens to the learner's speech, diagnoses the characteristics of the learner speech that most demand pedagogical reply, and selects appropriate feedback to give the learner. Without this kind of pedagogical feedback loop, the learner is limited to self-supervised practice and game interaction, and thus runs the risk of fossilization of language errors. Moreover, evaluations of the system indicate that speech interaction has an important role in motivating learners to persevere with using the system[2].

In order to function properly, the Pedagogical Agent needs to process—both understand and analyze--learner speech. This is no easy feat. Learner speech is infamously hard to predict using traditional language processing and ASR techniques[3]. Learner speech varies significantly from native speech, to the extent that naïve detection of non-native speech using native-trained ASR models yields very poor results[4]. Further compounding the problem, while corpora for native speech are typically plentiful, corpora of non-native speech are much more rare.

The systematic nature of user errors, governed by the linguistic structure of their native and target learning languages, can be used to predict and identify learner errors. This idea has been well supported by psycholinguistic theories of language learning[5]. As we show in this paper, the systematic nature of errors can help in building ASR grammars for learner error detection, identification and ranking in terms of their severity, and generation of appropriate feedback.

In the next section we describe our approach toward modeling learner language errors—analyzing and contextualizing learner errors, in order to extract the most pedagogically salient errors of learner speech. In section 3 we describe our ASR setup. Results are presented in section 4. We conclude with a summary of our important findings and directions for future research.

2. Analysis of Learner Speech Errors

The TLTS incorporates two speech-enabled learning environments: an interactive game called the Mission Practice Environment (MPE) that simulates conversations with native speakers, and an intelligent tutoring system called the Mission Skill Builder (MSB) for acquiring and practicing communicative skills. The speech error analysis module is responsible for providing users with feedback on their performance in both these environments.

The task of the error analysis module is to categorize the ASR-processed learner speech signal. This error analysis process is composed of two parts. First, it generates the appropriate grammars for detecting learner speech errors using knowledge about the target language and learning from annotated data. Second, it processes the ASR output to rank these hypothesized errors in the context of their pedagogical salience—that is, in a language pedagogy context, the measure of how feedback-worthy an error is.

2.1. Grammar Generation for Error Detection

Identifying possible learner speech errors is actually an off-line predictive process. A good prior model is essential for ASR to successfully identify learner speech errors. In order to populate speech recognition grammars, therefore, our system must model errors before they are detected. Examining native-speaker annotations of learner speech, we find that there is a wide gamut of factors that contribute to errors in learner speech. Learners, in speaking, make mistakes in pronunciation, grammar, and morphology. They will confuse words that sound or mean similar things, and they will leave out words in sentences that are too complex[1].

Table 1 shows the breakdown of learner behavior for vocabulary exercises in MSB (Mission Skill Builder). Annotators (Native speakers of Levantine Arabic) examined 1252 utterances by beginner Arabic learners on a case-by-case basis, singling out the most "pedagogically salient" feature of that utterance. The chart shows that learners correctly perform vocabulary exercises in the MSB environment approximately 15% of the time. Of the remaining incorrect times, the overwhelming majority of the mistakes that language learners make are pronunciation-related. Speech was rated as G(-) if it was unintelligible or uncategorizably bad, and a G(+) rating was given if the speech sample had no major errors.

Table 1: Breakdown of learner performance in MSB(1252 Annotated Samples)

Type of Utterance	Percent
Good -- 5 rating or G(+)	14
Phonetic Error -- P(*)	69
Lexical/Syntactic/Other Error	6
Bad -- 1 rating or G(-)	10

Phonological errors (i.e, mistakes in pronunciation) arise due to interference between native- and second-language phoneme sets. Such transfers of linguistic knowledge from the native language have been well documented[5]. To model these errors, instead of a rule-based system, the TLTS uses a Naive Bayesian classifier that was trained on native speaker annotation of learner speech. To train the phonological error model, we used a corpus of 1893 non-native speaker pronunciations, gathered from 7 male non-native speakers with no prior Levantine experience saying 188 distinct words. From this corpus, we apply techniques similar to those used in machine translation to find "translations" from phonemes in native speech to phonemes in learner speech. More specifically, we use co-occurrence statistics to populate a Bayesian model of phoneme-to-phoneme (and phoneme-to-nil) n-gram mappings, and use these to generate a noisy-channel system[6] modeling learner mistakes. This noisy-channel model is applied in a way so that, given a canonical utterance, it generates a list of probabilistically ranked expected mispronounced utterances.

The probability values arrived upon by our Bayesian model are close to what a linguistically-inspired rule-based system would compute. The grammars that we generated for error detection were found to correlate with findings in Second Language Acquisition (SLA) theory[7]. In particular we found close agreement with the SLA hypothesis that learners experience more problems with sounds in the second language that are not present in their mother tongue (native English speakers learning Arabic, in our case, will have trouble with /□/, /Φ/, /ξ/, and /αʁ/). In addition, some of our phone mappings can be attributed to the SLA theory that tells us to expect higher confusion between sounds that are allophones in the L1 but distinct phonemes in the L2, especially over phonological features that carry low functional load in English-gemination, sound-lengthening, and pharyngealization (our native speakers often fail to reproduce the distinction between lengthened vowel sounds /a⌉/ and non-lengthened /a/, or pharyngealized /□/ and non-pharyngealized /ħ/). A statistically-driven process allows us to rapidly approximate expert linguistic knowledge without the necessity or expense of gathering linguistic expert knowledge.

In addition to pronunciation error modeling, we have also implemented a number of other systems to model other types of learner errors. These include Lexical, Syntactic, Cognitive-load-based word drop, and Morphological systems. More information on subsystems is discussed in [9].

2.2. Disambiguation of Learner Speech Errors

A number of factors such as ASR confidence in error detection, derived confidence as inferred from past learner history, and intrinsic characteristics of the error committed are taken into account to contextualize learner errors and re-rank them.

First, raw ASR confidence (Section 3.2) gives us an idea of error severity. We boost raw ASR confidence by considering results in the context of the learner's performance history. Positive evidence of the learner having made a specific mistake in the past boosts our confidence of that error in current detection, and likewise trends of performing a problematic speech unit correctly lowers our confidence. Subsequently the characteristics of the errors committed, as taken in cultural and listener context are judged for their severity. Evaluating annotator data shows that some errors affect utterance intelligibility more than others. Native listeners have high tolerance to (and will overlook) learners mistakes on "hard" sounds like /αʁ/ and /Φ/. They have low tolerance for missing sounds that they generally deem "easy" to say. Errors are considered more severe by native speakers when they allow the intended utterance to be confused with other words in the target language. Finally, errors are considered extremely severe when collision with other words causes the learner to break social taboos (e.g. mixing up the very phonetically similar

[1] We note that the distribution of learner errors is largely affected by the pedagogical interface. While the vocabulary exercises for the Mission Skill Builder elicit mainly phonological errors in learner speech, we have found that the Mission Practice Environment, as well as dialog exercises in the Mission Skill Builder draw out more morphosyntactic errors. In this paper we focus on the learner speech errors in the Mission Skill Builder environment.

words /raaxid/ ["terrible"] and /raa'id/ ["major"]). Our system, imitating native-speaker listener behavior, adjusts error severity based on the ramifications of that error upon listener understanding and on violation of cultural norms.

The contextualization subsystem re-ranks error severities based on these factors, and passes them on to a feedback generation system that chooses from sets of pre-recorded messages. Higher confidence allows us to be more specific in our feedback, while the most general of feedbacks can be used to cover up low confidences in the ASR. This is important because in an environment so closely resembling real life, the learner's expectations for pedagogical feedback are heightened—they expect something closer to real life as some of our usability tests have revealed[2].

In the next section we describe the speech recognition setup and our approach to measure confidence in learner error detection, which is used in the error disambiguation process.

3. Automatic Speech Recognition for TLTS

The speech recognition system was implemented using the Cambridge HTK toolkit. The feature set comprised of 12 Mel Frequency Cepstral Coefficients (MFCC) extracted at a frame rate of 10ms using a 16 ms Hamming window. First and second order differentials plus an energy component were also included. Monophonic models were built for the 37 phones in our Levantine Arabic lexicon. A skip state silence model was also trained. The phone models had three states with eight mixture components. Context dependent models for the lexicon were generated using tree based clustering.

The system was trained on a modern standard Arabic dataset with around 10 hours of native speech. A mapping from modern standard Arabic phone set to Levantine Arabic phone set was used for this purpose. To support learner speech recognition in the TLTS our initial efforts focused on acoustic modeling for robust speech recognition especially in light of limited domain data availability [5].

The ASR training and decoding process differs for the two TLTS learning environments in tune with their distinct objectives.

3.1. ASR for Mission Practice Environment (MPE)

The goal of the ASR in MPE is to enable the user to communicate with virtual agents using speech. Thus, the primary objective for this task is to do robust recognition of learner speech to enable free form, naturalistic interaction. The MPE grammar thus includes utterances and their common morpho-syntactic variants in a finite state grammar. Native speakers were asked to generate the correct variants and possible erroneous variants that they expect an English speaker to make. More variants were added after the logs of initial trial runs were analyzed.

3.2. ASR for Mission Skill Builder (MSB)

The MSB grammar is tuned for detecting pronunciation variants and was generated using the error analysis grammar and generation method described in Section 2.

Error disambiguation (Section 2.2) requires a measure of ASR confidence and hypothesis verification for error identification. This is carried out using a two-step thresholding procedure. In the first step we decode the utterance with a grammar containing the correct canonical pronunciation of all the utterances in the system and a reference single path grammar containing just the intended utterance. If the output of the recognizer is the same as the intended utterance we proceed to the next step. In case the two differ we compare the acoustic score of the utterance grammar with the reference. If the difference between the two scores is higher than a leniency threshold, the utterance is judged to be out of the intended vocabulary of the learner, and the system thus generates a generic incorrect utterance feedback to the user. If this is not the case, the system diagnoses specific learner errors, as described in the next step.

To detect speech errors at the phonological level we decode the utterance with its corresponding phone confusion grammar whose generation is described in Section 2.1. If the recognizer selects the canonical output from the error analysis grammar or the difference in acoustic score between the canonical and the best path is less than a threshold, we provide the user with a generic positive feedback. Otherwise feedback is generated using the error disambiguation procedure described in Section 2.2. ASR confidence for disambiguation is taken to be the difference in system confidence between the recognized variant, the output from the utterance recognizer, and the canonical utterance.

The thresholds were determined using 500 annotated utterances as a heldout set. An analysis of the recognizer errors indicates that words with lengthened vowel English transliteration (ii, aa) needed significantly higher leniency threshold than other words. The thresholds are adjusted in the system based on user expertise/difficulty level and the pedagogical history.

4. Results

We evaluated the performance of our learner speech error detection on a set of 1000 annotated beginner Levantine utterances.

4.1. Error Grammar Generation

To evaluate the effectiveness of our error grammar generation scheme, we compared our phoneme-level noisy-channel model based system to a word-level baseline system.

Table 2: Analysis of Error Grammar Generation

System Type (Grammar size = 5)	Error Coverage
Baseline	55.3
Bayesian	59.8

The baseline uses frequency statistics generated over word-level mispronunciations, and selects the most common mispronunciations to populate the grammar.

We submit that gains in error coverage over the baseline are due to the speaker-dependent nature of pronunciation errors, which are not hard to generalize using a word-level error model.

4.2. Effect of error grammar size

We experimented with different number of error grammar sizes generated using the phone mapping procedure (Section 2.1). Increasing grammar size improves the coverage of phonetic errors that our system can potentially detect but it also increases ASR confusability (Table 3). In Table 3 the error coverage is the number of phonetic errors that are covered in the grammar. ASR accuracy is evaluated as the number of phonetic variants that were correctly detected without any post-processing/filtering (Section 3.2).

Table 3: System performance as a function of grammar size

Number of variants	Error coverage (%)	Raw ASR accuracy (%)
5	59.8	55.3
10	63.8	52.7
15	66.3	52.7
20	68.1	53

The ideal grammar size was found to be 15 variants per utterance. For larger grammars the increase in ASR confusability offset the advantage we were getting in terms of higher error coverage.

4.3. Error disambiguation accuracy

The first stage of error disambiguation is classification of the learner utterance between good (G(+)), pronunciation errors which are potential candidates for specific feedback messages (P(*)) and bad (G(-)). Our results show that the three-way thresholding procedure (section 3.2) is beneficial in improving the classification accuracy. Table 4 shows the confusion matrix of our system for this task without thresholding and Table 5 shows the post thresholding confusion matrix. A comparison of the two tables shows a 6% improvement in classification accuracy, which can be measured as the sum of the diagonal entries. The system accuracy in detecting the correct pronunciation variant for the P(*) class of errors improved dramatically from 16% to 55% (partly due to the overgeneralization of P(*) errors to G(-) errors).

Table 4: Error Type confusion matrix (no thresholding)

	G(+)	P(*)	G(-)
G(+)	51	173	17
P(*)	76	461	56
G(-)	2	118	34

Table 5 Error Type confusion matrix (with thresholding)

	G(+)	P(*)	G(-)
G(+)	51	192	17
P(*)	78	551	86
G(-)	0	9	4

5. Conclusions

In this paper we presented a system for modeling learner errors for a task-based language tutoring system. Our error model incorporates a number of different factors a native listener takes into account in judging the pronunciation of non-native speech. We used a machine translation like approach to generate grammars for detecting phonetic errors, which are pedagogically salient. ASR confidence in errors was generated using a three- way thresholding procedure.

Our results show that it is possible to provide the learner with relatively accurate diagnosis and feedback despite the inherent difficulties in modeling learner speech. We intend to extend this work by incorporating suprasegmental speech information as a part of our error disambiguation process. Furthermore, we plan to extend error model to environments that elicit more syntactic and morphological speech errors.

6. Acknowledgements

This project is part of the DARWARS initiative sponsored by the US Defense Advanced Research Projects Agency.

7. References

[1] L. Johnson, S. Marsella, N. Mote, M. Si, H. Vilhjalmsson, S. Wu. “Balanced Perception and Action in the Tactical Language Training System”, In InSTIL, 2004.

[2] L. Johnson, C. Beal. “Iterative Evaluation of an Intelligent Game for Language Learning”. In AIEd, 2005.

[3] J. Mostow, G. Aist. “Giving help and praise in a reading tutor with imperfect listening …”. In CALICO Journal (1999),16:3, 407-424.

[4] S. LaRocca, J. Morgan, S. Bellinger. “On the Path to 2X Learning: Exploring the Possibilities of Advanced Speech Recognition”. In CALICO Journal (1999),16:3, 295-310.

[5] J. Jenkins, "The learning theory approach". In "Psycholinguistics: A Survey of Theory and Research", Charles E. Osgood, e.d. page 20-35, 1954.

[6] F. Jelinek. "Statistical Methods for Speech Recognition" MIT Press, 1997.

[7] F. Eckman, A. Elreyes, G. Iverson. “Some Principles of Second Language Phonology”. In Second Language Research 19:3, 2003.

[8] S. Ganjavi, P. Georgiou, and S. Narayanan. “ASCII based transcription systems with the Arabic script: The case of Persian”. In Proc. IEEE ASRU, 2003.

[9] N. Mote, L. Johnson, A. Sethy, J. Silva, S. Narayanan, "Tactical language detection and modeling of learner speech errors”. In InSTIL, 2004.

[10] L. Neumeyer, H. Franco, V. Digalakis, and M.Weintraub. “Automatic Scoring of Pronunciation Quality’’. Speech Communications, 30(2–3):83–93, 2000.