# ASR ERROR DETECTION IN A CONVERSATIONAL SPOKEN LANGUAGE TRANSLATION SYSTEM[1]

*Wei Chen, Sankaranarayanan Ananthakrishnan, Rohit Kumar, Rohit Prasad, and Prem Natarajan*

Speech, Language, and Multimedia Processing Unit
Raytheon BBN Technologies
Cambridge, MA 02138, U.S.A
{wchen,sanantha,rkumar,rprasad,pnataraj}@bbn.com

## ABSTRACT

Detection of automatic speech recognition (ASR) errors is crucial to preventing their further propagation through statistical machine translation (SMT) in conversational spoken language translation (CSLT) systems. In this paper, we venture beyond traditional features obtained from the ASR decoder and hypothesized word sequence, and explore additional information streams provided by an error-robust CSLT system, including SMT confidence estimates and posteriors from named entity detection (NED). Another significant novelty of this work is the use of an automated *word boundary detector* based on acoustic-prosodic features to verify the existence of ASR-hypothesized word boundaries, which further improves ASR error detection. Offline evaluation on a test set designed to invoke ASR errors showed that at 10% false alarm rate, the proposed features provide 2.8% absolute (4.2% relative) improvement in detection rate over a state-of-the-art baseline error detector that uses a rich set of features traditionally employed in the existing literature.

***Index Terms***— automatic speech recognition, ASR error detection, conversational speech translation, SMT confidence estimation, word boundary detection

## 1. INTRODUCTION

Conversational spoken language translation (CSLT) systems use automatic speech recognition (ASR) to transcribe input speech into a sequence of hypothesized words, which are processed by a statistical machine translation (SMT) system. An optional text-to-speech (TTS) engine renders the translated SMT output as speech. This pipeline propagates ASR errors in the initial stage, often producing incomprehensible output in the target language [1]. Thus, identifying ASR errors and taking corrective action to prevent their downstream propagation is crucial to preserving the speaker's intended meaning and alleviating problems with the communication flow.

Spontaneous conversational speech presents numerous challenges, including out-of-vocabulary (OOV) words, speech repairs, phonetic/linguistic confusability, and related issues that cause problems for even the best ASR systems. Even so, the knowledge that errors are present can help the system take action to rectify or reduce the impact of the problem. For example, it may prompt the user to confirm or rephrase segments which it thinks may contain ASR errors.

This paper focuses on automatic detection of ASR errors in the context of an interactive, error-robust English-Iraqi Arabic CSLT system. In addition to the standard pipeline (ASR → SMT → TTS), the system contains built-in error detection modules that pinpoint regions in the input where ASR and SMT are likely to fail, including an *SMT confidence estimator* and *named-entity detector (NED)*, whose predictions are used to augment an existing set of features traditionally used for ASR error/OOV detection.

In a significant departure from the existing literature, we leverage *signal-level information* for ASR error detection by using acoustic-prosodic features to verify word boundaries hypothesized by the ASR system. This *word boundary detector* allows us to identify misplaced boundaries due to insertion, substitution, and deletion errors, and is used as an additional feature for ASR error detection.

## 2. RELATION TO PRIOR WORK

Existing ASR error detection approaches usually focus on features generated from the ASR decoder, such as word posteriors, phonetic acoustic scores, language model (LM) scores, confusion network density, and sub-word to word comparisons [e.g., 2, 3-6]. Features generated from the hypothesized word sequence, such as *n*-grams, parts of

[1] This paper is based upon work supported by the DARPA BOLT program. The views expressed here are those of the author(s) and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

speech, syntactic, semantic, and even discourse-level features [e.g., 7, 8-11], are commonly employed.

The novelty of our approach stems from the incorporation of additional feature streams provided by multiple non-ASR components in our CSLT system. Another significant contribution of this paper is the finding that word boundary verification using signal-level acoustic-prosodic features further improves ASR error detection performance over a strong baseline system.

## 3. BASELINE ERROR DETECTOR

Baseline speech recognition is based on the BBN Byblos ASR system [12]. The acoustic model (AM) was trained on approximately 200 hours of transcribed English speech (129K segmented utterances) from the DARPA TransTac two-way spoken dialogue collections covering various domains, including force protection (e.g., checkpoint, reconnaissance, and patrol), medical diagnosis and aid, maintenance, infrastructure, etc. The LM was trained on 5.8M English sentences (60M words), drawn from both in-domain and out-of-domain sources. LM and decoding parameters were tuned on a held-out development set of 3,534 utterances (45K words). We obtained 11% WER on a held-out test set of 3,138 utterances (38k words). Besides 1-best hypotheses, the decoder also generates confusion networks (word graphs) from the decoding lattice.

### 3.1. Training and Evaluation Data for Error Detection

We employed the jack-knifing technique to generate a large training corpus for the ASR error detector. We divided the 200 hours of English speech with reference transcriptions into ten equal partitions. Each partition was decoded with an LM that left out transcriptions for that partition (ten different LMs were trained, one for each partition). The global baseline AM was used for decoding all partitions. ASR errors were elicited in each partition by excluding singleton words in that partition from the corresponding decoding lexicon and LM. Word-level reference error labels were generated for the entire training set (combining all jack-knife partitions) by automatically aligning the ASR hypotheses with corresponding reference transcriptions. Since ASR errors are often "bursty", we use the so-called BI encoding for the reference labels – the first mis-recognized word is labeled "BERROR" for "beginning of error", and all the following consecutive errors are labeled "ERROR".

Besides the jack-knifed training set, we created held-out development and test sets designed to cause various types of ASR errors. For instance, the two sets are rich in previously unseen OOV names and non-names. They also contained phonetically confusable words (homophones), as well as mispronunciations and fragments. We emphasize that our ASR error-detection evaluation framework mirrors a real-world scenario where OOV words and other ASR errors are not simulated (e.g., by holding them out of the ASR lexicon/LM), but are completely unknown to and previously unobserved by the system. The overall OOV and ASR error rates for our training, development, and test sets are summarized in Table 1.

**Table 1.** OOV rates and WER for training & evaluation data

| Corpus | Size (words) | OOV Rate | WER |
|---|---|---|---|
| *Training* | 1.5M | 1.4% | 24.3% |
| *Development* | 21.7K | 3.0% | 26.5% |
| *Test* | 4.6K | 8.9% | 41.9% |

### 3.2. Baseline Features

We built a strong baseline ASR error detector using a variety of features commonly employed in existing literature [e.g., 2, 3-11], described below. These features were evaluated for every hypothesized candidate word.

***ASR confidence*:** Word posterior probabilities (WPP) computed from acoustic and language model scores using the forward-backward algorithm on a word lattice. Higher WPP implies greater "confidence" in the candidate word.

***LM perplexity*:** We used the negative LM log-likelihood of the candidate word given its *n*-gram context applied during ASR decoding. High perplexity implies mismatched linguistic context.

***Confusion network density*:** The number of competing words in the confusion network slot corresponding to the candidate word. Higher density suggests potential ASR error.

***Phonetic acoustic model score deviation*:** We measured the average, maximum, and minimum normalized deviation of phoneme acoustic scores across a hypothesized word with respect to true acoustic scores for those phonemes (z-scores). To facilitate this, we pre-computed the mean and standard deviation of acoustic scores for each phoneme from force-aligned reference transcriptions in the training corpus.

***Word vs. graphone decoding disagreement*:** We computed the phonetic edit distance between hypotheses generated by word and sub-word ASR systems. We used *graphones* [13] as sub-word units due to their robustness (lower error rate) vis-à-vis phoneme decoding. Graphones were automatically learned and inferred from letter-to-phoneme alignment obtained using standard SMT phrase extraction techniques [14]. Maximum length of graphemes was set to three letters.

***Parts of speech*:** We automatically tagged the ASR hypotheses with their parts-of-speech using the Stanford tagger [15] and used these as categorical features.

***Homophone indicator*:** We employed a binary feature for each word indicating whether it is a homophone, i.e. shares lexicon pronunciation with other words (e.g. *cell* and *sell*). These acoustically confusable words tend to generate errors.

### 3.3. Modeling and Predicting Error Labels

The baseline features in Section 3.2 were used in conjunction with a conditional random field (CRF) [16] using CRF++ [17] to model the relationship between the features and labels, as well as sequential dependencies between the labels. Because CRF++ is designed to work with categorical features, we discretized all real-valued features into 100 bins (this number was set empirically, which may need to be adjusted for a different task), each containing roughly the same number of instances.

We tuned various model parameters, including the CRF order (first and second), context of neighboring words, and feature cutoff, on the held-out development set. Best development set performance was achieved using a second order CRF to model dependency between successive labels, and a context of five words (i.e., features from the two prior, current, and two successive hypothesized words).

We applied these settings to predict error labels for ASR hypotheses of the independent high-error test set. We obtained the receiver operating characteristic (ROC) curve for the system by sweeping the CRF marginal probabilities. Detailed results are presented in Section 6, but to summarize, the baseline system achieved an error detection rate of 66.3% at 10% false alarm rate. As a comparison, sweeping the ASR confidence score (WPP) gave an error detection rate of just 44.2% at the same false alarm rate.

## 4. CSLT-BASED FEATURES

We augmented the strong baseline error detector from Section 3 with novel features derived from the CSLT system. These include estimated SMT confidence projected to source words, and posteriors from the NED subsystem.

### 4.1. SMT Confidence Estimates

Our English-Iraqi CSLT system incorporates a phrase-based SMT confidence estimation framework that predicts the probability of error for each hypothesized target word in the SMT output. This was achieved using a set of SMT-derived features (e.g., forward translation probability, lexical smoothing score, target LM likelihood, etc.), as well as bilingual indicator features that capture word co-occurrences in the generating source phrase and candidate target word. A maximum-entropy (maxent) classifier predicted error probability of target words in conjunction with automatically derived reference labels from TER alignment. We projected target (Iraqi) error probabilities back on to source (English) words using the SMT decoding phrase derivations to obtain a measure of *translation success* over English words. Details of this system are available in our previous work [18].

Our rationale for incorporating projected translation success is that English ASR errors usually produce poor English-Iraqi translations, which are likely to be identified by the SMT confidence estimator. In other words, poor translation quality could be indicative of source ASR error.

### 4.2. Named-Entity Detection (NED) Posteriors

Our CSLT system frequently encounters named entities such as person and location names, a significant fraction of which tend to be OOV, causing ASR errors. Thus, knowing whether an ASR-hypothesized word might be a named entity could be useful in identifying a possible error. Prior work (e.g., [19]) has used OOV detection features to help NED, but not the other way round.

We built a maxent-based NED subsystem using contextual lexical and part-of-speech features trained on name-annotated corpora [18]. This classifier estimates the posterior probability of each ASR-hypothesized word being part of a named entity. We incorporated a discretized version of the NED posterior within the ASR error detector.

## 5. WORD BOUNDARY FEATURES

OOV words are often broken up by ASR into multiple in-vocabulary words. As a result, word insertions are frequent, constituting around 33%, 32%, and 47% of word errors in our training, development, and test sets, respectively. The test set in particular exhibits a high insertion rate due to its high OOV word rate. The high insertion rate resulted in falsely-hypothesized word boundaries in speech. For instance, the ASR hypothesis "WHILE BEING" in Figure 1 suggests a word boundary right in the middle of the OOV word "WELDING". Thus, automatic speech-based word boundary detection can be used to verify ASR-hypothesized boundaries, with false boundaries indicative of ASR error.

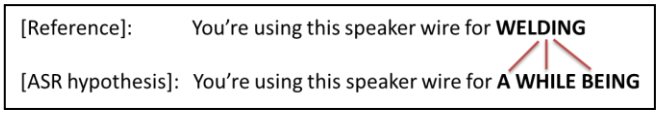


**Figure 1.** OOV word in development set split in three by ASR

Acoustic-prosodic features have been shown to be useful for word boundary detection. Rao and Srichand [20] report that pitch variations between and within words are indicative of word boundaries. Chi [21] suggests that consonants at word boundaries have longer durations than word-medial consonants. Based on these findings, we used fundamental frequency (F0), voicing probability, loudness, and phoneme duration features for word boundary detection.

For each hypothesized word boundary time point, we calculated the absolute difference of the maximum, minimum, and average values of F0, voicing probability, and loudness between hypothesized words immediately preceding and following the boundary. We computed duration z-scores for the phonemes immediately preceding and following the hypothesized boundary point. To facilitate this, mean and standard deviation of phoneme durations

were pre-computed from force-aligned reference transcriptions of the training set. Finally, we included the left and right phoneme identities for a total of 13 features for word boundary detection.

Reference labels for word boundary detection were obtained automatically using the jack-knife partitions of Section 3.1 by comparing the boundaries hypothesized by the ASR to true boundaries obtained by force-aligning reference transcriptions. A hypothesized word boundary is "correct" only when it aligns exactly with a true word boundary, otherwise it is "incorrect".

We used the LogitBoost binary classifier in Weka [22] for word boundary detection. LogitBoost incrementally boosts the weight of training instances that are mis-classified, combining a set of weak classifiers (decision stumps). The top ranked features are the right and left phoneme identities, as well as duration features. The overall accuracy of the word boundary classifier on the test set was 76.3% vs. 75.8% for the majority classifier, which always predicts "boundary". The area under the ROC curve (AUC) was 0.61 vs. random classification AUC of 0.5.

Our objective is not to maximize word boundary detection accuracy, but to exploit it for ASR error detection. To this end, we incorporated discretized word boundary posteriors within our ASR error detection feature set. For each hypothesized word, we included word boundary posteriors corresponding to its start- and end-points.

## 6. EVALUATION RESULTS

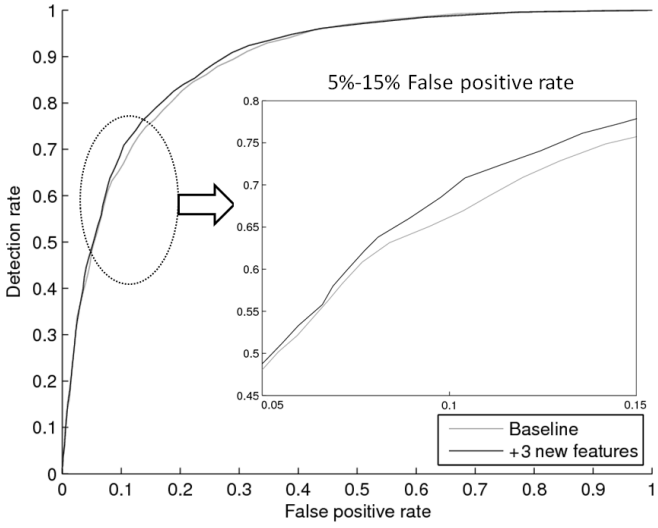


**Figure 2.** ROC curves for baseline and proposed detector with inset view focusing on 5%-15% false alarm rate.

We augmented the strong baseline system of Section 3 with CLST-based features from Section 4 and word boundary features described in Section 5. Figure 2 shows the ROC curves for the baseline system and the improved detector with the proposed features. We particularly focus on the region of the ROC curve at 5%-15% false alarm rate, where we typically operate the ASR error detector. Figure 3 shows the improvement in detection rate at 10% false alarm rate when adding these features incrementally to the baseline system. The baseline system has a detection rate of 66.3%. The proposed features increase detection rate to 69.1%, an absolute improvement of 2.8% (4.2% relative).

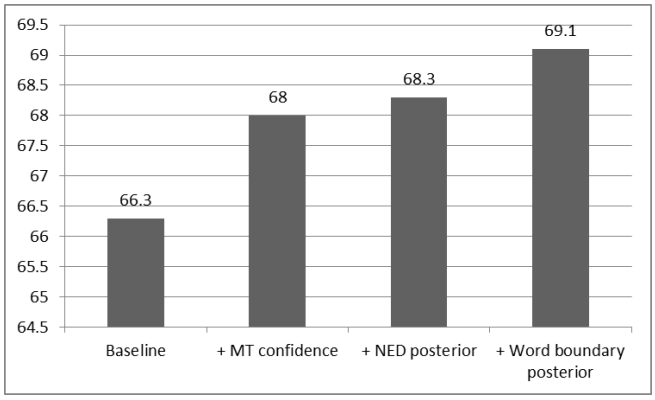


**Figure 3.** Detection rates at 10% false alarm for incremental addition of proposed features to the baseline system.

## 7. CONCLUSIONS AND FUTURE DIRECTIONS

Automated ASR error detection is crucial for CSLT systems because it can help prevent downstream propagation of errors, and give interactive systems the chance to engage the user and resolve detected errors. A CSLT system provides additional information streams for detecting ASR errors, above and beyond traditional features employed in ASR/OOV error detection. We proposed novel features derived from CSLT system components, including projected SMT confidence estimates and NED posteriors. At 10% false alarm rate, SMT confidence provided the single biggest gain in error detection rate (1.7% absolute) over the baseline.

We additionally incorporated word boundary detection to verify ASR-hypothesized word boundaries and used the estimated boundary posteriors as additional features for ASR error detection. At 10% false alarm rate, this provided a gain of 0.8% detection rate over the combination of all baseline and CSLT-based features, bringing the total absolute error detection rate improvement to 2.8% (4.2% relative) over a strong baseline system.

We are concurrently exploring the utility of ASR error detection in improving SMT performance. Initial results have shown that high ASR error detection accuracy can help improve translation scores in an error-robust SMT decoding framework [23]. Future work includes exploring other features to improve detection rate, as well as evaluating the overall impact on the usability and effectiveness of interactive CSLT systems.

## 8. REFERENCES


[1] L. Mathias, "Statistical Machine Translation and Automatic Speech Recognition under Uncertainty," Ph.D. Dissertation, the Johns Hopkins University, Baltimore, MA, 2007.

[2] A. Allauzen, "Error Detection in Confusion Network," In *proceedings of the Interspeech*, Antwerp, Belgium, 2007.

[3] C. White, G. Zweig, L. Burget, P. Schwarz, and H. Hermansky, "Confidence Estimation, OOV Detection and Language ID using Phone-toWord Transduction and Phone-level Alignments," In *proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2008.

[4] C. Parada, M. Dredze, D. Filimonov, and F. Jelinek, "Contextual Information Improves OOV Detection in Speech," In *proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2010.

[5] T. Pellegrini and I. Trancoso, "Error Detection in Broadcast News ASR using Markov Chains," In *proceedings of the the 4th Conference on Human Language Technology: Challenges for Computer Science and Linguistics* 2009.

[6] T. Kemp and T. Schaaf, "Estimating Confidence using Word Lattices," In *proceedings of the European Conference on Speech Communication and Technology*, 1997.

[7] T. Pellegrini and I. Trancoso, "Improving ASR Error Detection with Non-Decoder Based Features," in *Interspeech*, Makuhari, Chiba, Japan, 2010.

[8] S. R. Young, "Detecting Misrecognition and Out-of-Vocabulary Words," In *proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1994.

[9] Y. Bassil and B. Semaan, "ASR Context-Sensitive Error Correction Based on Microsoft N-Gram Dataset," *Journal of Computing,* vol. 4, January 2012 2012.

[10] A. Sarma and D. D. Palmer, "Context-based Speech Recognition Error Detection and Correction," In *proceedings of the HLT-NAACL*, 2004. Short Papers.

[11] G. Skantze and J. Edlund, "Early Error Detection on Word Level," In *proceedings of the ISCA Tutorial and Research Workshop on Robustness*, 2004.

[12] L. Nguyen and R. Schwartz, "Efficient 2-pass N-best Decoder," In *proceedings of the Eurospeech*, Rhodes, Greece, 1997.

[13] M. Bisani and H. Ney, "Investigations on Joint-Multigram Models for Grapheme-to-Phoneme Conversion," In *proceedings of the Int. Conf. on Spoken Language Processing*, Denver, CO, USA, 2002.

[14] P. Koehn, F. J. Och, and D. Marcu, "Statistical Phrase-Based Translation," In *proceedings of the NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, Edmonton, Canada, 2003.

[15] K. Toutanova, D. Klein, C. Manning, and Y. Singer, "Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network," In *proceedings of the Human Language Technologies: The 4th Annual Conference of the North American Chapter of the Association for Computational Linguistics* 2003.

[16] J. Lafferty, A. McCallum, and F. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data " in *ICML*, 2001, pp. 282-289.

[17] T. Kudo. Available: http://crfpp.googlecode.com/svn/trunk/doc/index.html

[18] R. Prasad, R. Kumar, S. Ananthakrishnan, W. Chen, S. Hewavitharana, M. Roy, F. Choi, A. Challenner, E. Kan, A. Neelakantan, and P. Natarajan, "Active Error Detection and Resolution for Speech to Speech Translation," In *proceedings of the International Workshop on Spoken Language Translation*, 2012.

[19] C. Parada, M. Dredze, and F. Jelinek, "OOV Sensitive Named Entity Detection in Speech," In *proceedings of the International Speech Communication Association (INTERSPEECH)*, 2011.

[20] R. Rao and J. Srichand, "Word Boundary Detection using Pitch Variations," In *proceedings of the ICSLP*, 1996.

[21] X. Chi, "Word Boundary Detection using Landmarks: a Survey of Consonants," Ph.D. Dissertation, MIT, 2008.

[22] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "The WEKA Data Mining Software: An Update," *SIGKDD Explorations,* vol. 11, 2009.

[23] S. Ananthakrishnan, W. Chen, R. Kumar, R. Prasad, and P. Natarajan, "Source-Error Aware Phrase-Based Decoding for Robust Conversational Spoken Language Translation," submitted to *the International Speech Communication Association (INTERSPEECH)*, 2013. Submitted.