SIP (2019), vol. 8, e8, page 1 of 16 © The Authors, 2019.

doi:10.1017/ATSIP.2018.31

ORIGINAL PAPER

Learning from past mistakes: improving automatic speech recognition output via noisy-clean phrase context modeling

PRASHANTH GURUNATH SHIVAKUMAR[1], HAOQI LI[1], KEVIN KNIGHT[2] AND PANAYIOTIS GEORGIOU[1]

Automatic speech recognition (ASR) systems often make unrecoverable errors due to subsystem pruning (acoustic, language and pronunciation models); for example, pruning words due to acoustics using short-term context, prior to rescoring with long-term context based on linguistics. In this work, we model ASR as a phrase-based noisy transformation channel and propose an error correction system that can learn from the aggregate errors of all the independent modules constituting the ASR and attempt to invert those. The proposed system can exploit long-term context using a neural network language model and can better choose between existing ASR output possibilities as well as re-introduce previously pruned or unseen (Out-Of-Vocabulary) phrases. It provides corrections under poorly performing ASR conditions without degrading any accurate transcriptions; such corrections are greater on top of out-of-domain and mismatched data ASR. Our system consistently provides improvements over the baseline ASR, even when baseline is further optimized through Recurrent Neural Network (RNN) language model rescoring. This demonstrates that any ASR improvements can be exploited independently and that our proposed system can potentially still provide benefits on highly optimized ASR. Finally, we present an extensive analysis of the type of errors corrected by our system.

Keywords: Error correction, Speech recognition, Phrase-based context modeling, Noise channel estimation, Neural Network Language Model

Received 6 April 2018; Revised 10 December 2018

I. INTRODUCTION

Due to the complexity of human language and quality of speech signals, improving performance of automatic speech recognition (ASR) is still a challenging task. The traditional ASR comprises of three conceptually distinct modules: acoustic modeling, dictionary, and language modeling. Three modules are fairly independent of each other in research and operation.

In terms of acoustic modeling, Gaussian Mixture Model (GMM) based Hidden Markov Model (HMM) systems [1, 2] were a standard for ASR for a long time and are still used in some of the current ASR systems. Lately, advances in Deep Neural Network (DNN) led to the advent of Deep Belief Networks (DBN) and Hybrid DNN-HMM [3, 4], which basically replaced the GMM with a DNN and employed a HMM for alignments. Deep RNN, particularly Long Short Term Memory (LSTM) Networks replaced the traditional DNN and DBN systems [5]. Connectionist Temporal Classification (CTC) [6] proved to be effective with the ability to compute the alignments implicitly under the DNN architecture, thereby eliminating the need of GMM-HMM systems for computing alignments.

The research efforts for developing efficient dictionaries or lexicons have been mainly in terms of pronunciation modeling. Pronunciation modeling was introduced to handle the intra-speaker variations [7, 8], non-native accent variations [7, 8], speaking rate variations found in conversational speech [8] and increased pronunciation variations found in children's speech [9]. Various linguistic knowledge and data-derived phonological rules were incorporated to augment the lexicon.

Research efforts in language modeling share those of the Natural Language Processing (NLP) community. By estimating the distribution of words, statistical language modeling (SLM), such as n-gram, decision tree models [10], linguistically motivated models [11] amount to calculating the probability distribution of different linguistic units, such as words, phrases [12], sentences, and whole documents [13]. Recently, DNN-based language models [14–16] have also

[1] Signal Processing for Communication Understanding and Behavior Analysis Laboratory (SCUBA), University of Southern California, Los Angeles, USA
[2] Information Sciences Institute, University of Southern California, Los Angeles, USA

Corresponding author:
Panayiotis Georgiou
Email: georgiou@sipi.usc.edu

shown success in terms of both perplexity and word error rate.

Very recently, state-of-the-art ASR systems are employing end-to-end neural network models, such as sequence-to-sequence [17] in an encoder–decoder architecture. The systems are trained end-to-end from acoustic features as input to predict the phonemes or characters [18, 19]. Such systems can be viewed as an integration of acoustic and lexicon pronunciation models. The state-of-the-art performance can be attributed toward the joint training (optimization) between the acoustic model and the lexicon models (end-to-end) enabling them to overcome the short-comings of the former independently trained models.

Several research efforts were carried out for error correction using post-processing techniques. Much of the effort involves user input used as a feedback mechanism to learn the error patterns [20, 21]. Other work employs multi-modal signals to correct the ASR errors [21, 22]. Word co-occurrence information-based error correction systems have proven quite successful [23]. In [24], a word-based error correction technique was proposed. The technique demonstrated the ability to model the ASR as a noisy channel. In [25], similar technique was applied to a syllable-to-syllable channel model along with maximum entropy-based language modeling. In [26], a phrase-based machine translation system was used to adapt a generic ASR to a domain-specific grammar and vocabulary. The system trained on words and phonemes was used to re-rank the n-best hypotheses of the ASR. In [27], a phrase-based machine translation system was used to adapt the models to the domain-specific data obtained by manual user-corrected transcriptions. In [28], an RNN was trained on various text-based features to exploit long-term context for error correction. Confusion networks from the ASR have also been used for error correction. In [29], a bi-directional LSTM-based language model was used to re-score the confusion network. In [30], a two-step process for error correction was proposed in which words in the confusion network are re-ranked. Errors present in the confusion network are detected by conditional random fields (CRF) trained on n-gram features and subsequently long-distance context scores are used to model the long contextual information and re-rank the words in the confusion network. [31, 32] also makes use of confusion networks along with semantic similarity information for training CRFs for error correction.

Our Contribution: The scope of this paper is to evaluate whether subsequent transcription corrections can take place, on top of a highly optimized ASR. We hypothesize that our system can correct the errors by (i) re-scoring lattices, (ii) recovering pruned lattices, (iii) recovering unseen phrases, (iv) providing better recovery during poor recognitions, (v) providing improvements under all acoustic conditions, (vi) handling mismatched train-test conditions, (vii) exploiting longer contextual information, and (viii) text regularization. We target to satisfy the above hypotheses by proposing a Noisy-Clean Phrase Context Model (NCPCM). We introduce context of past errors of an ASR system that consider all the automated system noisy transformations. These errors may come from any of the ASR modules or even from the noise characteristics of the signal. Using these errors we learn a noisy channel model, and apply it for error correction of the ASR output.

Compared with the above efforts, our work differs in the following aspects:

- Error corrections take place on the output of a state-of-the-art *Large Vocabulary Continuous Speech Recognition* (LVCSR) system trained on matched data. This differs from adapting to constrained domains (e.g. [26, 27]) that exploit domain mismatch. This provides additional challenges both due to the larger error-correcting space (spanning larger vocabulary) and the already highly optimized ASR output.
- We evaluate on a standard LVCSR task thus establishing the effectiveness, reproducibility and generalizability of the proposed correction system. This differs from past work where speech recognition was on a large-vocabulary task but subsequent error corrections were evaluated on a much smaller vocabulary.
- We analyze and evaluate multiple type of error corrections (including but not restricted to OOV words). Most prior work is directed toward recovery of OOV words.
- In addition to evaluating a large-vocabulary correction system on in-domain (Fisher, 42k words) we evaluate on an out-of-domain, larger vocabulary task (TED-LIUM, 150k words), thus assessing the effectiveness of our system on challenging scenarios. In this case, the adaptation is to an even bigger vocabulary, a much more challenging task to past work that only considered adaptation from large to small vocabulary tasks.
- We employ multiple hypotheses of ASR to train our noisy channel model.
- We employ state-of-the-art neural network-based language models under the noisy-channel modeling framework which enable exploitation of longer context.

Additionally, our proposed system comes with several advantages: (1) the system could potentially be trained without an ASR by creating a phonetic model of corruption and emulating an ASR decoder on generic text corpora, (2) the system can rapidly adapt to new linguistic patterns, e.g. can adapt to unseen words during training via contextual transformations of erroneous LVCSR outputs.

Further, our work is different from discriminative training of acoustic [33] models and discriminative language models (DLM) [34], which are trained directly to optimize the word error rate using the reference transcripts. DLMs in particular involve optimizing, tuning, the weights of the language model with respect to the reference transcripts and are often utilized in re-ranking n-best ASR hypotheses [34–38]. The main distinction and advantage with our method is the NCPCM can potentially re-introduce unseen or pruned-out phrases. Our method can also operate when there is no access to lattices or n-best lists. The NCPCM can also operate on the output of a DLM system.

The rest of the paper is organized as follows: Section II presents various hypotheses and discusses the different types of errors we expect to model. Section III elaborates on the proposed technique and Section IV describes the experimental setup and the databases employed in this work. Results and discussion are presented in Section V and we finally conclude and present future research directions in Section VI.

II. HYPOTHESES

In this section we analytically present cases that we hypothesize the proposed system could help with. In all of these the errors of the ASR may stem from realistic constraints of the decoding system and pruning structure, while the proposed system could exploit very long context to improve the ASR output.

Note that the vocabulary of an ASR does not always match the one of the error correction system. Lets consider for example, an ASR that does not have lexicon entries for “Prashanth” or “Shivakumar” but it has the entries “Shiva” and “Kumar”. Lets also assume that this ASR consistently makes the error “Pression” when it hears “Prashanth”. Given training data for the NCPCM, it will learn the transformation “Pression Shiva Kumar” into “Prashanth Shivakumar”, thus it will have a larger vocabulary than the ASR and learn to recover such errors. This demonstrates the ability to learn OOV entries and to rapidly adapt to new domains.

A) Re-scoring lattices

1. *“I was born in nineteen ninety three in Iraq”*
2. *“I was born in nineteen ninety three in eye rack”*
3. *“I was born in nineteen ninety three in I rack”*

Phonetic Transcription: “*ay . w aa z . b ao r n . ih n . n ay n t iy n . n ay n t iy . th r iy . ih n . ay . r ae k*”

Example 1

In Example 1, all the three samples have the same phonetic transcription. Let us assume sample 1 is the correct transcription. Since all the three examples have the same phonetic transcription, this makes them indistinguishable by the acoustic model. The language model is likely to down-score the sample 3. It is possible that sample 2 will score higher than sample 1 by a short context LM (e.g. bigram or 3-gram), i.e. “in” might be followed by “eye” more frequently than “Iraq” in the training corpora. This will likely result in an ASR error. Thus, although the oracle WER can be zero, the output WER is likely going to be higher due to LM choices.

Hypothesis A: An ideal error correction system can select correct options from the existing lattice.

B) Recovering pruned lattices

A more severe case of Example 1 would be that the word “Iraq” was pruned out of the output lattice during decoding. This is often the case when there are memory and complexity constraints in decoding large acoustic and language models, where the decoding beam is a restricting parameter. In such cases, the word never ends up in the output lattice. Since the ASR is constrained to pick over the only existing possible paths through the decoding lattice, an error is inevitable in the final output.

Hypothesis B: An ideal error correction system can generate words or phrases that were erroneously pruned during the decoding process.

C) Recovery of unseen phrases

On the other hand, an extreme case of Example 1 would be that the word “Iraq” was never seen in the training data (or is OOV), thereby not appearing in the ASR lattice. This would mean the ASR is forced to select among the other hypotheses even with a low confidence (or output an unknown, $< unk >$, symbol) resulting in a similar error as before. This is often the case due to the constant evolution of human language or in the case of a new domain. For example, names such as “Al Qaeda” or “ISIS” were non-existent in our vocabularies a few years ago.

Hypothesis C: An ideal error correction system can generate words or phrases that are OOV and thus not in the ASR output.

D) Better recovery during poor recognitions

An ideal error correction system would provide more improvements for poor recognitions from an ASR. Such a system could potentially offset for the ASR’s low performance providing consistent performance over varying audio and recognition conditions. In real-life conditions, the ASR often has to deal with varying level of “mismatched train-test” conditions, where relatively poor recognition results are commonplace.

Hypothesis D: An ideal error correction system can provide more corrections when the ASR performs poorly, thereby offsetting ASR’s performance drop (e.g. during mismatched train-test conditions).

E) Improvements under all acoustic conditions

An error correction system which performs well during tough recognition conditions, as per *Hypothesis* D is no good if it degrades good recognizer output. Thus, in addition to our *Hypothesis* D, an ideal system would cause no degradation on good ASR output. Such a system can be hypothesized to consistently improve upon and provide benefits over any ASR system including state-of-the-art recognition systems. An ideal system would provide improvements over the entire spectrum of ASR performance (WER).

> ***Hypothesis E:*** An ideal error correction system can not only provide improvements during poor recognitions, but also preserves good speech recognition.

F) Adaptation

We hypothesize that the proposed system would help in adaptation over mismatched conditions. The mismatch could manifest in terms of acoustic conditions and lexical constructs. The adaptation can be seen as a consequence of *Hypothesis* D & E. In addition, the proposed model is capable of capturing patterns of language use manifesting in specific speaker(s) and domain(s). Such a system could eliminate the need of retraining the ASR model for mismatched environments.

> ***Hypothesis F:*** An ideal error correction system can aid in mismatched train-test conditions.

G) Exploit longer context

- *"**Eyes** melted, when he placed his hand on her shoulders."*
- *"**Ice** melted, when he placed it on the table."*

Example 2

The complex construct of human language and understanding enables recovery of lost or corrupted information over different temporal resolutions. For instance, in the above Example 2, both the phrases, "Eyes melted, when he placed" and "Ice melted, when he placed" are valid when viewed within its shorter context and have identical phonetic transcriptions. The succeeding phrases, underlined, help in discerning whether the first word is "Eyes" or "Ice". We hypothesize that an error correction model capable of utilizing such longer contexts is beneficial. As new models for phrase-based mapping, such as sequence to sequence models [17], become applicable this becomes even more possible and desirable.

> ***Hypothesis G:*** An ideal error correction system can exploit longer context than the ASR for better corrections.

H) Regularization

1. - *"I guess 'cause I went on a I went on a ..."*
   - *"I guess because I went on a I went on a ..."*
2. - *"i was born in nineteen ninety two"*
   - *"i was born in 1992"*
3. - *"i was born on nineteen twelve"*
   - *"i was born on 19/12"*

Example 3

As per the three cases shown in Example 3, although both the hypotheses for each of them are correct, there are some irregularities present in the language syntax. Normalization of such surface form representation can increase readability and usability of output. Unlike traditional ASR, where there is a need to explicitly program such regularizations, our system is expected to learn, given appropriate training data, and incorporate regularization into the model.

> ***Hypothesis H:*** An ideal error correction system can be deployed as an automated text regularizer.

III. METHODOLOGY

The overview of the proposed model is shown in Fig. 1. In our paper, the ASR is viewed as a noisy channel (with transfer function H), and we learn a model of this channel, $\widehat{H}^{-1}$ (estimate of inverse transfer function H^{-1}) by using the corrupted ASR outputs (equivalent to signal corrupted by H) and their reference transcripts. Later on, we use this model to correct the errors of the ASR.

The noisy channel modeling mainly can be divided into word-based and phrase-based channel modeling. We will first introduce previous related work, and then our proposed NCPCM.

A) Previous related work

1) Word-based noisy channel modeling

In [24], the authors adopt word-based noisy channel model borrowing ideas from a word-based statistical machine translation developed by IBM [39]. It is used as a post-processor module to correct the mistakes made by the ASR. The word-based noisy channel modeling can be presented as:

$$\begin{aligned}\hat{W} &= \underset{W_{clean}}{\operatorname{argmax}}\, P(W_{clean}|W_{noisy}) \\ &= \underset{W_{clean}}{\operatorname{argmax}}\, P(W_{noisy}|W_{clean}) P_{LM}(W_{clean}),\end{aligned}$$

where $\hat{W}$ is the corrected output word sequence, $P(W_{clean}| W_{noisy})$ is the posterior probability, $P(W_{noisy}|W_{clean})$ is the channel model and $P_{LM}(W_{clean})$ is the language model. In [24], authors hypothesized that introducing many-to-one and one-to-many word-based channel modeling (referred to as fertility model) could be more effective, but was not implemented in their work.

2) Phrase-based noisy channel modeling

Phrase-based systems were introduced in application to phrase-based statistical translation system [40] and were shown to be superior to the word-based systems. Phrase-based transformations are similar to word-based models with the exception that the fundamental unit of observation and transformation is a phrase (one or more words). It can be viewed as a super-set of the word-based [39] and the fertility [24] modeling systems.

B) Noisy-clean phrase context modeling

We extend the ideas by proposing a complete phrase-based channel modeling for error correction which incorporates the many-to-one and one-to-many as well as many-to-many words (phrase) channel modeling for error-correction. This also allows the model to better capture errors of varying resolutions made by the ASR. As an extension, it uses a distortion model to capture any re-ordering of phrases during error-correction. Even though we do not expect big benefits from the distortion model (i.e. the order of the ASR output is usually in agreement with the audio representation), we include it in our study for examination. It also uses a word penalty to control the length of the output. The phrase-based noisy channel modeling can be represented as:

$$\begin{aligned}\hat{p} &= \underset{p_{clean}}{\operatorname{argmax}}\, P(p_{clean}|p_{noisy}) \qquad (1)\\ &= \underset{p_{clean}}{\operatorname{argmax}}\, P(p_{noisy}|p_{clean}) P_{LM}(p_{clean}) w_{\text{length}}(p_{clean}),\end{aligned}$$

where $\hat{p}$ is the corrected sentence, p_{clean} and p_{noisy} are the reference and noisy sentence, respectively. $w_{\text{length}}(p_{clean})$ is the output word sequence length penalty, used to control the output sentence length, and $P(p_{noisy}|p_{clean})$ is decomposed into:

$$P(p^I_{noisy}|p^I_{clean}) = \prod_{i=1}^{I} \phi(p^i_{noisy}|p^i_{clean}) D(start_i - end_{i-1}), \quad (2)$$

where $\phi(p^i_{noisy}|p^i_{clean})$ is the phrase channel model or phrase translation table, p^I_{noisy} and p^I_{clean} are the sequences of I phrases in noisy and reference sentences, respectively, and i refers to the i^{th} phrase in the sequence. $D(start_i - end_{i-1})$ is the distortion model. $start_i$ is the start position of the noisy phrase that was corrected to the i^{th} clean phrase, and end_{i-1} is the end position of the noisy phrase corrected to be the $i - 1^{th}$ clean phrase.

C) Our other enhancements

In order to effectively demonstrate our idea, we employ (i) neural language models, to introduce long-term context and justify that the longer contextual information is beneficial for error corrections; (ii) minimum error rate training (MERT) to tune and optimize the model parameters using development data.

1) Neural language models

Neural network-based language models have been shown to be able to model higher order n-grams more efficiently [14–16]. In [25], a more efficient language modeling using maximum entropy was shown to help in noisy-channel modeling of a syllable-based ASR error correction system.

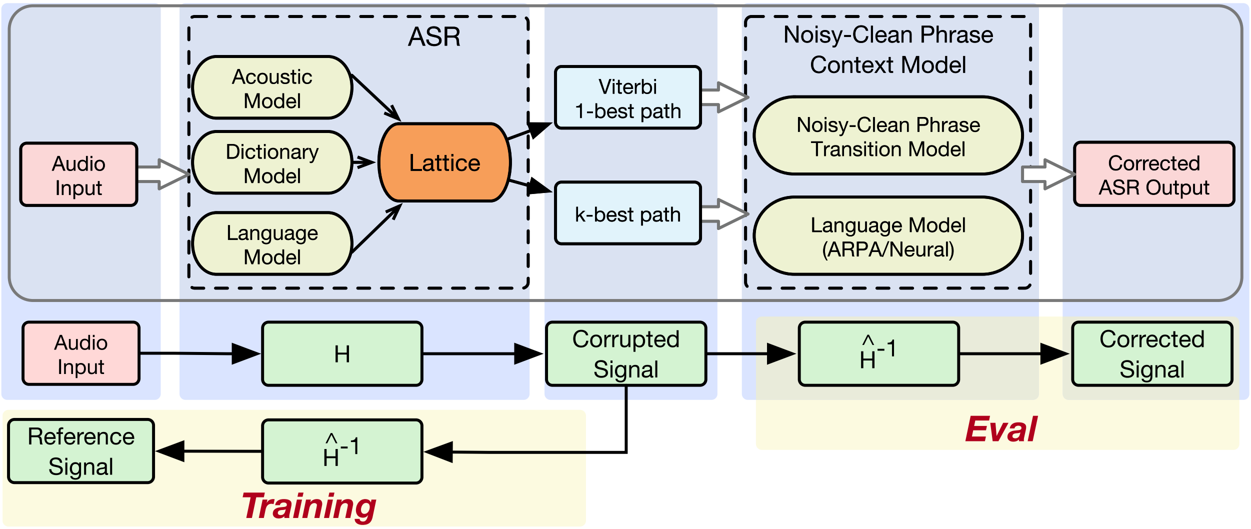


Fig. 1. Overview of NCPCM.

Incorporating such language models would aid the error-correction by exploiting the longer context information. Hence, we adopt two types of neural network language models in this work. (i) Feed-forward neural network which is trained using a sequence of one-hot word representation along with the specified context [41]. (ii) Neural Network Joint Model (NNJM) language model [42]. This is trained in a similar way as in (i), but the context is augmented with noisy ASR observations with a specified context window. Both the models employed are feed-forward neural networks since they can be incorporated directly into the noisy channel modeling. The RNN LM could potentially be used during phrase-based decoding by employing certain caching and approximation tricks [43]. Noise Contrastive Estimation was used to handle the large vocabulary size output.

2) Minimum error rate training (MERT)

One of the downsides of the noisy channel modeling is that the model is trained to maximize the likelihood of the seen data and there is no direct optimization to the end criteria of WER. MERT optimizes the model parameters (in our case weights for language, phrase, length, and distortion models) with respect to the desired end evaluation criterion. MERT was first introduced in application to statistical machine translation providing significantly better results [44]. We apply MERT to tune the model on a small set of development data.

IV. EXPERIMENTAL SETUP

A) Database

For training, development, and evaluation, we employ Fisher English Training Part 1, Speech (LDC2004S13) and Fisher English Training Part 2, Speech (LDC2005S13) corpora [45]. The Fisher English Training Part 1 is a collection of conversation telephone speech with 5850 speech samples of up to 10 min, approximately 900 hours of speech data. The Fisher English Training Part 2 contains an addition of 5849 speech samples, approximately 900 hours of telephone conversational speech. The corpora is split into training, development and test sets for experimental purposes as shown in Table 1. The splits of the data sets are consistent over both the ASR and the subsequent NCPCM. The development dataset was used for tuning the phrase-based system using MERT.

We also test the system under mismatched training-usage conditions on TED-LIUM. TED-LIUM is a dedicated ASR corpus consisting of 207 hours of TED talks [46]. The data set was chosen as it is significantly different to Fisher Corpus. Mismatch conditions include: (i) variations in channel characteristics, Fisher, being a telephone conversations corpus, is sampled at 8 kHz whereas the TED-LIUM is originally 16 kHz, (ii) noise conditions, the Fisher recordings are significantly noisier, (iii) utterance lengths, TED-LIUM has longer conversations since they are extracted from TED talks, (iv) lexicon sizes, vocabulary size of TED-LIUM is much larger with 150000 words whereas Fisher has 42150 unique words, (v) speaking intonation, Fisher being telephone conversations is spontaneous speech, whereas the TED talks are more organized and well articulated. Factors (i) and (ii) mostly affect the performance of ASR due to acoustic differences while (iii) and (iv) affect the language aspects, (v) affects both the acoustic and linguistic aspects of the ASR.

B) System setup

1) Automatic speech recognition system

We used the Kaldi Speech Recognition Toolkit [47] to train the ASR system. In this paper, the acoustic model was trained as a DNN-HMM hybrid system. A tri-gram maximum likelihood estimation (MLE) language model was trained on the transcripts of the training dataset. The CMU pronunciation dictionary [48] was adopted as the lexicon. The resulting ASR is state-of-the-art both in architecture and performance and as such additional gains on top of this ASR are challenging.

2) Pre-processing

The reference outputs of ASR corpus contain non-verbal signs, such as [laughter], [noise] etc. These event signs might corrupt the phrase context model since there is little contextual information between them. Thus, in this paper, we cleaned our data by removing all these non-verbal signs from dataset. The text data are subjected to traditional tokenization to handle special symbols. Also, to prevent data sparsity issues, we restricted all of the sample sequences to a maximum length of 100 tokens (given that the database consisted of only three sentences having more than the limit). The NCPCM has two distinct vocabularies, one associated with the ASR transcripts and the other one pertaining to the ground-truth transcripts. The ASR dictionary is often smaller than the ground-truth transcript mainly because of not having pronunciation-phonetic transcriptions for certain words, which usually is the case for names, proper-nouns, out-of-language words, broken words, etc.

3) NCPCM

We use the Moses toolkit [49] for phrase-based noisy channel modeling and MERT optimization. The first step in

Table 1. Database split and statistics

	Train			Development			Test		
Database	Hours	Utterances	Words	Hours	Utterances	Words	Hours	Utterances	Words
Fisher English	1,890.5	1,833,088	20,724,957	4.7	4906	50245	4.7	4914	51230
TED-LIUM	–	–	–	1.6	507	17792	2.6	1155	27512

the training process of NCPCM is the estimation of the word alignments. IBM models are used to obtain the word alignments in both the directions (reference-ASR and ASR-reference). The final alignments are obtained using heuristics (starting with the intersection of the two alignments and then adding the additional alignment points from the union of two alignments). For computing the alignments "mgiza", a multi-threaded version of GIZA++ toolkit [50] was employed. Once the alignments are obtained, the lexical translation table is estimated in the maximum likelihood sense. Then on, all the possible phrases along with their word alignments are generated . A max phrase length of 7 was set for this work. The generated phrases are scored to obtain a phrase translation table with estimates of phrase translation probabilities. Along with the phrase translation probabilities, word penalty scores (to control the translation length) and re-ordering/distortion costs (to account for possible re-ordering) are estimated. Finally, the NCPCM model is obtained as in the equation (2). During decoding equation (1) is utilized.

For training the MLE n-gram models, SRILM toolkit [51] was adopted. Further we employ the Neural Probabilistic Language Model Toolkit [41] to train the neural language models. The neural network was trained for 10 epochs with an input embedding dimension of 150 and output embedding dimension of 750, with a single hidden layer. The weighted average of all input embeddings was computed for padding the lower order estimates as suggested in [41].

The NCPCM is an ensemble of phrase translation model, language model, translation length penalty, re-ordering models. Thus the tuning of the weights associated with each model is crucial in the case of proposed phrase-based model. We adopt the line-search-based method of MERT [52]. We try two optimization criteria with MERT, i.e. using BLEU(B) and WER(W).

C) Baseline systems

We adopt four different baseline systems because of their relevance to this work:

Baseline-1: *ASR Output*: The raw performance of the ASR system, because of its relevance to the application of the proposed model.

Baseline-2: *Re-scoring lattices using RNN-LM*: In order to evaluate the performance of the system with more recent re-scoring techniques, we train a recurrent-neural network with an embedding dimension of 400 and sigmoid activation units. Noise contrastive estimation is used for training the network and is optimized on the development data set which is used as a stop criterion. Faster-RNNLM [1] toolkit is used to train the recurrent-neural network. For re-scoring, 1000-best ASR hypotheses are decoded and the old LM (MLE) scores are removed. The RNN-LM scores are computed from the trained model and interpolated with the old LM. Finally, the 1000-best hypotheses are re-constructed into lattices, scored with new interpolated LM and decoded to get the new best path hypothesis.

Baseline-3: *Word-based noisy channel model*: In order to compare to a prior work described in Section 1 which is based on [24]. The word-based noisy channel model is created in a similar way as the NCPCM model with three specific exceptions: (i) the max-phrase length is set to 1, which essentially converts the phrase-based model into word based, (ii) a bi-gram LM is used instead of a tri-gram or neural language model, as suggested in [24], (iii) no re-ordering/distortion model and word penalties are used.

Baseline-4: *Discriminative Language Modeling (DLM)*: Similar to the proposed work, DLM makes use of the reference transcripts to tune language model weights based on specified feature sets in order to re-rank the n-best hypothesis. Specifically, we employ the perceptron algorithm [34] for training DLMs. The baseline system is trained using unigrams, bigrams and trigrams (as in [35–37]) for a fair comparison with the proposed NCPCM model. We also provide results with an extended feature set comprising of rank-based features and ASR LM and AM scores. Refr (Reranker framework) is used for training the DLMs [53] following most recommendations from [37]. The 100-best ASR hypotheses are used for training and re-ranking purposes.

D) Evaluation criteria

The final goal of our work is to show improvements in terms of the transcription accuracy of the overall system. Thus, we provide word error rate as it is a standard in the ASR community. Moreover, Bilingual Evaluation Understudy (BLEU) score [54] is used for evaluating our work, since our model can be also treated as a transfer-function ("translation") system from ASR output to NCPCM output.

V. RESULTS AND DISCUSSION

In this section we demonstrate the ability of our proposed NCPCM in validating our hypotheses A-H from Section II along with the experimental results. The experimental results are presented in three different tasks: (i) overall WER experiments, highlighting the improvements of the proposed system, presented in Tables 3, 4 & 5, (ii) detailed analysis of WERs over subsets of data, presented in Figs 2 & 3, and (iii) analysis of the error corrections, presented in Table 2. The assessment and discussions of each task is structured similar to Section II to support their respective claims.

A) Re-scoring lattices

Table 2 shows selected samples through the process of the proposed error correction system. In addition to the reference, ASR output and the proposed system output, we provide the ORACLE transcripts to assess the presence of the correct phrase in the lattice. Cases 4-6 from Table 2 have the correct phrase in the lattice, but get down-scored in the ASR final output which is then recovered by our system as hypothesized in *Hypothesis* A.

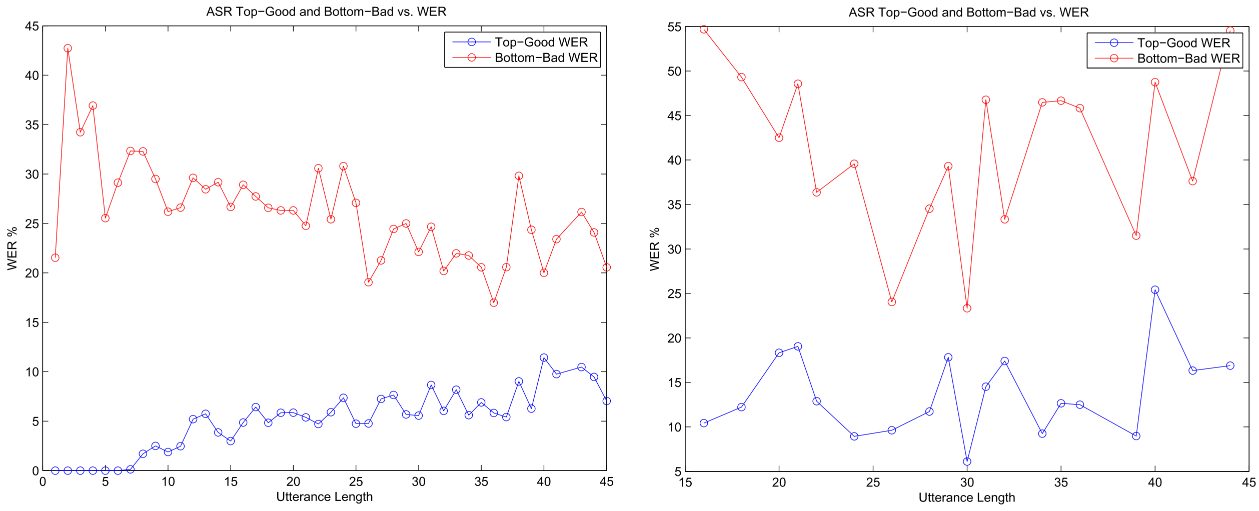
[1] https://github.com/yandex/faster-rnnlm

Table 2. Analysis of selected sentences. REF: Reference ground-truth transcripts; ASR: Output ASR transcriptions; ORACLE: Best path through output lattice given the ground-truth transcript; NCPCM: Transcripts after NCPCM error-correction Green color highlights correct phrases. Orange color highlights incorrect phrases.

1.		
	REF:	oysters clams and mushrooms i think
	ASR:	wasters clams and mushrooms they think
	ORACLE:	wasters clams and mushrooms i think
	NCPCM:	oysters clams and mushrooms they think
		Example of hypothesis B
2.		
	REF:	yeah we had this awful month this winter where it was like a good day if it got up to thirty it was ridiculously cold
	ASR:	yeah we had this awful month uh this winter where it was like a good day if i got up to thirty was ridiculous lee cold
	ORACLE:	yeah we had this awful month this winter where it was like a good day if it got up to thirty it was ridiculous the cold
	NCPCM:	yeah we had this awful month uh this winter where it was like a good day if i got up to thirty it was ridiculously cold
		Example of hypotheses A, B, G
3.		
	REF:	oh well it depends on whether you agree that al qaeda came right out of afghanistan
	ASR:	oh well it depends on whether you agree that al <unk> to came right out of afghanistan
	ORACLE:	oh well it depends on whether you agree that al <unk> to came right out of afghanistan
	NCPCM:	oh well it depends on whether you agree that al qaeda to came right out of afghanistan
		Example of hypothesis C
4.		
	REF:	they laugh because everybody else is laughing and not because it's really funny
	ASR:	they laughed because everybody else is laughing and not because it's really funny
	ORACLE:	they laugh because everybody else is laughing and not because it's really funny
	NCPCM:	they laugh because everybody else is laughing and not because it's really funny
		Example of hypotheses A, G
5.		
	REF:	yeah especially like if you go out for ice cream or something
	ASR:	yeah it specially like if you go out for ice cream or something
	ORACLE:	yeah it's especially like if you go out for ice cream or something
	NCPCM:	yeah especially like if you go out for ice cream or something
		Example of hypothesis A
6.		
	REF:	we don't have a lot of that around we kind of live in a nicer area
	ASR:	we don't have a lot of that around we kinda live in a nicer area
	ORACLE:	we don't have a lot of that around we kind of live in a nicer area
	NCPCM:	we don't have a lot of that around we kind of live in a nicer area
		Example of hypotheses A, H

B) Recovering pruned lattices

In the cases 1 and 2 from Table 2, we see the correct phrases are not present in the ASR lattice, although they were seen in the training and are present in the vocabulary. However, the proposed system manages to recover the phrases as discussed in *Hypothesis* B. Moreover, Case 2 also demonstrates an instance where the confusion occurs due to same phonetic transcriptions ("ridiculously" versus "ridiculous lee") again supporting *Hypothesis* A.



(a) In Domain (Fisher)

(b) Out of Domain (TEDLIUM)

Fig. 2. Top-Good, Bottom-Bad WER Splits. As we can see the WER for top-good is often 0%, which leaves no margin for improvement. We will see the impact of this later, as in Fig. 3.

C) Recovery of unseen phrases

Case 3 of Table 2, demonstrates an instance where the word "qaeda" is absent from the ASR lexicon (vocabulary) and hence absent in the decoding lattice. This forces the ASR to output an unknown-word token ($< unk >$). We see that the system recovers an OOV word "qaeda" successfully as claimed in *Hypothesis* C.

D) Better recovery during poor recognitions

To justify the claim that our system can offset for the performance deficit of the ASR at tougher conditions (as per *Hypothesis* D), we formulate a sub-problem as follows:

Problem Formulation: We divide equally, per sentence length, our development and test datasets into good recognition results (top-good) and poor recognition results (bottom-bad) subsets based on the WER of the ASR and analyze the improvements and any degradation caused by our system.

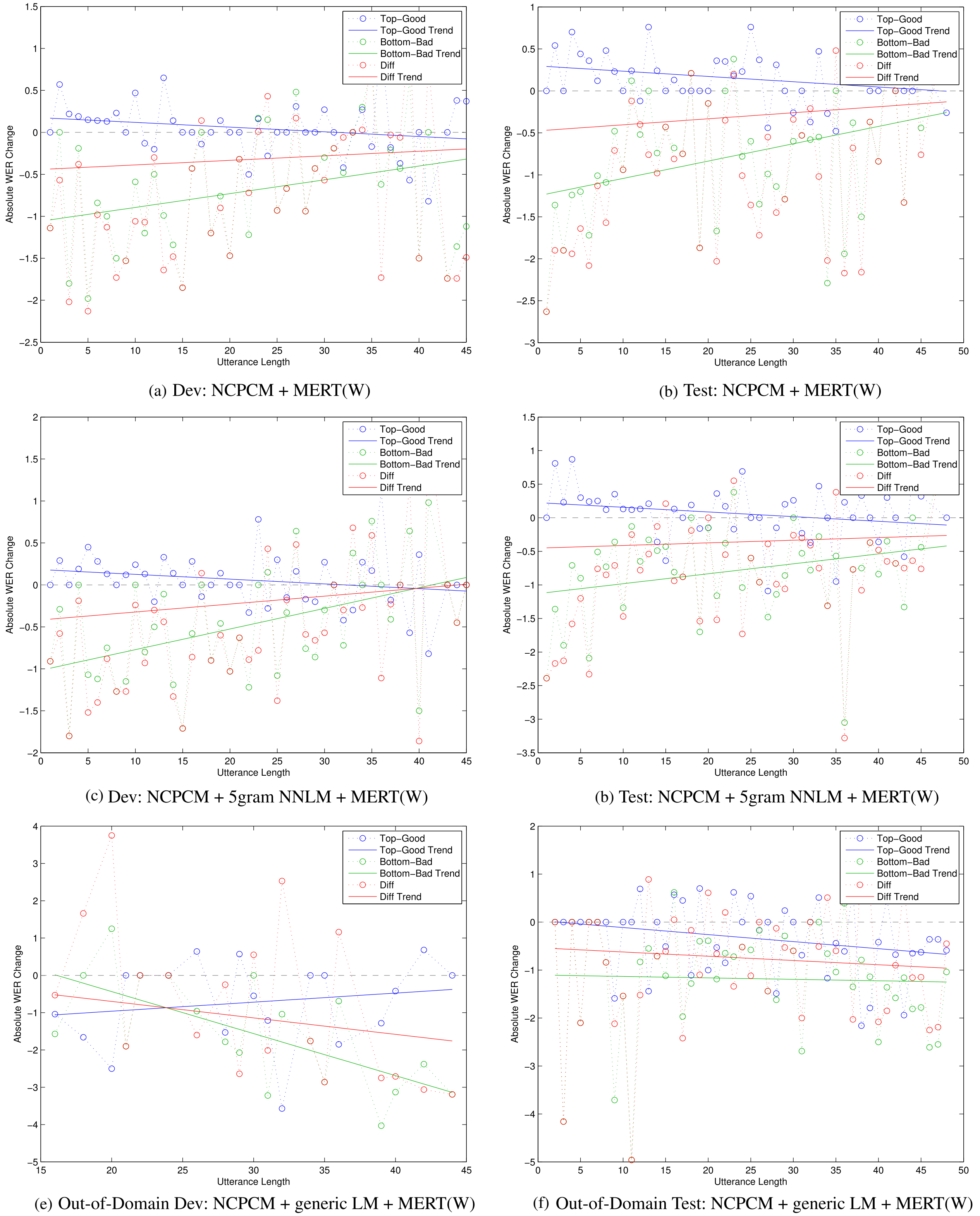
Figure 3 shows the plots of the above mentioned analysis for different systems as captioned. The blue lines are representative of the improvements provided by our system for top-good subset over different utterance lengths, i.e. it indicates the difference between our system and the original WER of the ASR (negative values indicate improvement and positive values indicate degradation resulting from our system). The green lines indicate the same for bottom-bad subset of the database. The red indicates the difference between the bottom-bad WERs and the top-good WERs, i.e. negative values of red indicate that the system provides more improvements to the bottom-bad subset relative to the top-good subset. The solid lines represent their respective trends which is obtained by a simple linear regression (line-fitting).

For poor recognitions, we are concerned about the bottom-bad subset, i.e. the green lines in Fig. 3. Firstly, we see that the solid green line is always below zero, which indicates there is always improvements for bottom-bad, i.e. poor recognition results. Second, we observe that the solid red line usually stays below zero, indicating that the performance gains made by the system add more for the bottom-bad poor recognition results compared with the top-good subset (good recognitions). Further, more justifications are provided later in the context of out-of-domain task (Section V F) where high mismatch results in tougher recognition task are discussed.

E) Improvements under all acoustic conditions

To justify the claim that our system can consistently provide benefits over any ASR system (*Hypothesis* E), we need to show that the proposed system: (i) does not degrade the performance of the good recognition, (ii) provides improvements to poor recognition instances, of the ASR. The latter has been discussed and confirmed in the previous Section V D. For the former, we provide evaluations from two point of views: (1) assessment of WER trends of top-good and bottom-bad subsets (as in the previous Section V D), and (2) overall absolute WER of the proposed systems.

Firstly, examining Fig. 3, we are mainly concerned about the top-good subset pertaining to degradation/improvement of good recognition instances. We observe that the solid blue line is close to zero in all the cases, which implies that the degradation of good recognition is extremely minimal. Moreover, we observe that the slope of the line is almost zero in all the cases, which indicates that the degradation is minimal and mostly consistent over different utterance lengths. Moreover, assessing the degradation from the absolute WER perspective, Fig. 2a shows the WER over utterance lengths for the top-good and bottom-bad subsets for the in-domain case. The top-good WER is small, at times even 0% (perfect recognition) thereby allowing very small



(a) Dev: NCPCM + MERT(W)

(b) Test: NCPCM + MERT(W)

(c) Dev: NCPCM + 5gram NNLM + MERT(W)

(b) Test: NCPCM + 5gram NNLM + MERT(W)

(e) Out-of-Domain Dev: NCPCM + generic LM + MERT(W)

(f) Out-of-Domain Test: NCPCM + generic LM + MERT(W)

Fig. 3. Length of hypotheses through our NCPCM models versus absolute WER change.
Blue and green lines represent difference between WER of our system and the baseline ASR, for top-good and bottom-bad hypotheses, respectively. In an ideal scenario, all these lines would be below 0, thus all providing a change in WER toward improving the system. However, we see in some cases that the WER increases, especially when the hypotheses length is short and when the performance is good. This is as expected since from Fig. 2 some cases are at 0% WER due to the already highly optimized nature of our ASR.
The red line represents the aggregate error over all data for each word length and as we can see in **all** cases the trend is one of **improving** the WER. This matches Hypotheses D, E, F, G.

Table 3. Noisy-Clean Phrase Context Model (NCPCM) results (uses exactly same LM as ASR)

In domain testing on Fisher Data				
	Dev		Test	
Method	WER (%)	BLEU	WER (%)	BLEU
ASR output (Baseline-1)	15.46	75.71	17.41	72.99
ASR + RNNLM re-scoring (Baseline-2)	16.17	74.39	18.39	71.24
Word based + bigram LM (Baseline-3)	16.23	74.28	18.10	71.76
Word based + bigram LM + MERT(B)	15.46	75.70	**17.40**	72.99
Word based + bigram LM + MERT(W)	**15.39**	75.65	**17.40**	72.77
Word based + trigram LM + MERT(B)	15.48	75.59	17.47	72.81
Word based + trigram LM + MERT(W)	15.46	75.46	17.52	72.46
DLM (Baseline-4)	23.65	63.35	25.36	61.19
DLM w/ extended feats	24.48	62.92	26.12	60.98
Proposed NCPCM	20.33	66.70	22.32	63.81
NCPCM + MERT(B)	**15.11**	**76.06**	**17.18**	**73.00**
NCPCM + MERT(W)	**15.10**	**76.08**	**17.15**	**73.05**
NCPCM + MERT(B) w/o re-ordering	**15.27**	**76.02**	**17.11**	**73.33**
NCPCM + MERT(W) w/o re-ordering	**15.19**	**75.90**	**17.18**	**73.04**
NCPCM + 10best + MERT(B)	**15.19**	**76.12**	**17.17**	**73.22**
NCPCM + 10best + MERT(W)	**15.16**	**75.91**	**17.21**	**73.03**

margin for improvement. In such a case, we see minimal degradation. Although we lose a bit on very good recognitions which is extremely minimal, we gain significantly in the case of 'bad' recognitions. Thus to summarize, the damage that this system can make, under the best ASR conditions, is minimal and offset by the potential significant gains present when the ASR hits some tough recognition conditions.

WER experiments: Secondly, examining the overall WER, Table 3 gives the results of the baseline systems and the proposed technique. Note that we use the same language model as the ASR. This helps us evaluate a system that does not include additional information. We provide the performance measures on both the development and held out test data. The development data are used for MERT tuning.

Baseline results: The output of the ASR (Baseline-1) suggests that the development data are less complex compared with the held out test set. In our case, the RNN-LM-based lattice re-scoring (Baseline-2) does not help. This result shows that even with a higher order context, the RNN-LM is unable to recover the errors present in the lattice, suggesting that the errors stem from pruning during decoding. We note that the word-based system (Baseline-3) does not provide any improvements. Even when we increase context (trigram LM) and use MERT optimization, the performance is just on par with the original ASR output. Further, DLM re-ranking (Baseline-4) fails to provide any improvements in our case. This result is in conjunction with the finding in [37], where the DLM provides improvements only when used in combination with ASR baseline scores. However, we believe introduction of ASR scores into NCPCM can be beneficial as would be in the case of DLMs. Thus, to demonstrate the independent contribution of NCPCM vs DLM's, rather than investigate fusion methods, we do not utilize baseline ASR scores for either of the two methods. We plan to investigate the benefits of multi-method fusion in our future work. When using the extended feature set for training the DLM, we do not observe improvements. With our setup, none of the baseline systems provide noticeable significant improvements over the ASR output. We believe this is due to the highly optimized ASR setup, and the nature of the database itself being noisy telephone conversational speech. Overall, the results of baseline highlights: (i) the difficulty of the problem for our setup, (ii) re-scoring is insufficient and emphasizes the need for recovering pruned out words in the output lattice.

NCPCM results: The NCPCM is an ensemble of phrase translation model, language model, word penalty model, and re-ordering models. Thus the tuning of the weights associated with each model is crucial in the case of the phrase-based models [55]. The NCPCM without tuning, i.e. assigning random weights to the various models, performs very poorly as expected. The word-based model lacks re-ordering/distortion modeling and word penalty models and hence are less sensitive to weight tuning. Thus it is unfair to compare the un-tuned phrase-based models with the baseline or word-based counterpart. Hence, for all our subsequent experiments, we only include results with MERT. When employing MERT, all of the proposed NCPCM systems significantly outperform the baseline (statistically significant with $p < 0.001$ for both word error and sentence error rates [56] with 51230 word tokens and 4914 sentences as part of the test data). We find that MERT optimized for WER consistently outperforms that with optimization criteria of BLEU score. We also perform trials by disabling the distortion modeling and see that results remain relatively unchanged. This is as expected since the ASR preserves the sequence of words with respect to the audio and there is no reordering effect over the errors. The phrase-based

Table 4. Results for out-of-domain adaptation using Noisy-Clean Phrase Context Models (NCPCM) Δ_1:Relative % improvement w.r.t baseline-1; Δ_2:Relative % improvement w.r.t baseline-2;

Cross domain testing on TED-LIUM Data						
	Dev		Test			
Method	WER (%)	BLEU	WER (%)	Δ_1 (%)	Δ_2 (%)	BLEU
Baseline-1 (ASR)	26.92	62.00	23.04	0	-10.9	65.71
ASR + RNNLM re-scoring (Baseline-2)	**24.05**	**64.74**	**20.78**	**9.8**	0	**67.93**
Baseline-3 (Word-based)	29.86	57.55	25.51	-10.7	-22.8	61.79
Baseline-4 (DLM)	33.34	53.12	28.02	-21.6	-34.8	58.50
DLM w/ extended feats	30.51	57.14	29.33	-27.3	-41.1	57.60
NCPCM + MERT(B)	**26.06**	**63.30**	**22.51**	**2.3**	-8.3	**66.67**
NCPCM + MERT(W)	**26.15**	**63.10**	**22.74**	**1.3**	-9.4	**66.36**
NCPCM + generic LM + MERT(B)	**25.57**	**63.98**	**22.38**	**2.9**	-7.7	**66.97**
NCPCM + generic LM + MERT(W)	**25.56**	**63.83**	**22.33**	**3.1**	-7.5	**66.96**
RNNLM re-scoring + NCPCM + MERT(B)	**23.36**	**65.88**	**20.40**	**11.5**	**1.8**	**68.39**
RNNLM re-scoring + NCPCM + MERT(W)	**23.32**	**65.76**	**20.57**	**10.7**	**1**	**68.07**
RNNLM re-scoring + NCPCM + generic LM + MERT(B)	**23.00**	**66.48**	**20.31**	**11.8**	**2.3**	**68.52**
RNNLM re-scoring + NCPCM + generic LM + MERT(W)	**22.80**	**66.19**	**20.23**	**12.2**	**2.6**	**68.49**

context modeling provides a relative improvement of 1.72% (See Table 3) over the baseline-3 and the ASR output. Using multiple hypotheses (10-best) from the ASR, we hope to capture more relevant error patterns of the ASR model, thereby enriching the noisy channel modeling capabilities. However, we find that the 10-best gives about the same performance as the 1-best. In this case we considered 10 best as 10 separate training pairs for training the system. In the future we want to exploit the inter-dependency of this ambiguity (the fact that all the 10-best hypotheses represent a single utterance) for training and error correction at test time.

F) Adaptation

WER experiments: To assess the adaptation capabilities, we evaluate the performance of the proposed NCPCM on an out-of-domain task, TED-LIUM database, shown in Table 4.

Baseline Results: The baseline-1 (ASR performance) confirms of the heightened mismatched conditions between the training Fisher Corpus and the TED-LIUM database. Unlike in matched in-domain evaluation, the RNNLM re-scoring provides drastic improvements (9.8% relative improvement with WER) when tuned with out-of-domain development data set. The mismatch in cross-domain evaluation reflects in considerably worse performance for the word-based and DLM baselines (compared with matched conditions).

NCPCM Results: However, we see that the phrase context modeling provides modest improvements over the baseline-1 of approximately 2.3% (see Table 4) relative on the held-out test set. We note that the improvements are consistent compared with the earlier in-domain experiments in Table 3. Moreover, since the previous LM was trained on Fisher Corpus, we adopt a more generic English LM which provides further improvements of up to 3.1% (see Table 4).

We also experiment with NCPCM over the re-scored RNNLM output. We find the NCPCM to always yield consistent improvements over the RNNLM output (see Δ_1 and Δ_2 in Table 4). An overall gains of 2.6% relative is obtained over the RNNLM re-scored output (baseline-2) i.e., 12.2% over ASR (baseline-1) is observed. This confirms that the NCPCM is able to provide improvements parallel, in conjunction to the RNNLM or any other system that may improve ASR performance and therefore supports the *Hypothesis* E in yielding improvements in the highly optimized ASR environments. This also confirms the robustness of the proposed approach and its application to the out-of-domain data. More importantly, the result confirms *Hypothesis* F, i.e. our claim of rapid adaptability of the system to varying mismatched acoustic and linguistic conditions. The extreme mismatched conditions involved in our experiments supports the possibility of going one step further and training our system on artificially generated data of noisy transformations of phrases as in [35, 36, 38, 57–59]. Thus possibly eliminating the need for an ASR for training purposes.

Further, comparing the WER trends from the in-domain task (Fig. 3b) to the out-of-domain task (Fig. 3f), we firstly find that the improvements in the out-of-domain task are obtained for both top-good (good recognition) and bottom-bad (bad recognition), i.e. both the solid blue line and the solid green line are always below zero. Secondly, we observe that the improvements are more consistent throughout all the utterance lengths, i.e. all the lines have near zero slopes compared with the in-domain task results. Third, comparing Fig. 2a with Fig. 2b, we observe more room for improvement, both for top-good portion as well as the bottom-bad WER subset of data set. The three findings are fairly meaningful considering the high mismatch of the out-of-domain data.

G) Exploit longer context

Firstly, inspecting the error correction results from Table 2, cases 2 and 4 hint at the ability of the system to select appropriate word-suffixes using long-term context information.

Table 5. Results for Noisy-Clean Phrase Context Models (NCPCM) with Neural Network Language Models (NNLM) and Neural Network Joint Models (NNJM)

In domain testing on Fisher Data

	Dev		Test	
Method	WER (%)	BLEU	WER (%)	BLEU
Baseline-1 (ASR output)	15.46	75.71	17.41	72.99
Baseline-2 (ASR + RNNLM re-scoring)	16.17	74.39	18.39	71.24
Baseline-3 (Word based + 5gram NNLM)	15.47	75.63	17.41	72.92
Word based + 5gram NNLM + MERT(B)	15.46	75.69	**17.40**	72.99
Word based + 5gram NNLM + MERT(W)	**15.42**	75.58	**17.38**	72.75
NCPCM + 3gram NNLM + MERT(B)	**15.46**	**75.91**	**17.37**	**73.24**
NCPCM + 3gram NNLM + MERT(W)	**15.28**	**75.94**	**17.11**	**73.31**
NCPCM + 5gram NNLM + MERT(B)	**15.35**	**75.99**	**17.20**	**73.34**
NCPCM + 5gram NNLM + MERT(W)	**15.20**	**75.96**	**17.08**	**73.25**
NCPCM + NNJM-LM (5,4) + MERT(B)	**15.29**	**75.93**	**17.13**	**73.26**
NCPCM + NNJM-LM (5,4) + MERT(W)	**15.28**	**75.94**	**17.13**	**73.29**

Secondly, from detailed WER analysis in Fig. 3, we see that the bottom-bad (solid green line) improvements decrease with increase in length in most cases, hinting at potential improvements to be found by using higher contextual information for error correction system as future research directions. Moreover, closer inspection across different models, comparing the trigram MLE model (Fig. 3b) with the 5gram NNLM (Fig. 3d), we find that the NNLM provides minimal degradation and better improvements especially for longer utterances by exploiting more context (the blue solid line for NNLM has smaller intercept value as well as higher negative slope). We also find that for the bottom-bad poor recognition results (green solid-line), the NNLM gives consistent (smaller positive slope) and better improvements especially for the higher length utterances (smaller intercept value). Thus emphasizing the gains provided by higher context NNLM.

WER experiments: Third, Table 5 shows the results obtained using a neural network language model of higher orders (also trained only on the in-domain data). For a fair comparison, we adopt a higher order (5gram) NNLM for the baseline-3 word-based noise channel modeling system. Even with a higher order NNLM, the baseline-3 fails to improve upon the ASR. We do not include the baseline-4 results under this section, since DLM does not include a neural network model. Comparing results from Table 3 with Table 5, we note the benefits of higher order LMs, with the 5-gram neural network language model giving the best results (a relative improvement of 1.9% over the baseline-1), outperforming the earlier MLE n-gram models as per *Hypothesis* G.

Moreover, experimental comparisons with baseline-3 (word-based) and NCPCM models, both incorporating identical 5-gram neural network language models confirms the advantages of NCPCM (a relative improvement of 1.7%). However, the NNJM LM with target context of 5 and source context of 4 did not show significant improvements over the traditional neural LMs. We expect the neural network models to provide further improvements with more training data.

H) Regularization

Finally, the last case in Table 2 is of text regularization as described in Section II, *Hypothesis* H. Overall, in our experiments, we found that approximately 20% were cases of text regularization and the rest were a case of the former hypotheses.

VI. CONCLUSIONS AND FUTURE WORK

In this work, we proposed a noisy channel model for error correction based on phrases. The system post-processes the output of an automated speech recognition system and as such any contributions in improving ASR are in conjunction of NCPCM. We presented and validated a range of hypotheses. Later on, we supported our claims with apt problem formulation and their respective results. We showed that our system can improve the performance of the ASR by (i) re-scoring the lattices (*Hypothesis* A), (ii) recovering words pruned from the lattices (*Hypothesis* B), (iii) recovering words never seen in the vocabulary and training data (*Hypothesis* C), (iv) exploiting longer context information (*Hypothesis* G), and (v) by regularization of language syntax (*Hypothesis* H).

Moreover, we also claimed and justified that our system can provide more improvement in low-performing ASR cases (*Hypothesis* D), while keeping the degradation to minimum in cases when the ASR performs well (*Hypothesis* E). In doing so, our system could effectively adapt (*Hypothesis* F) to changing recognition environments and provide improvements over any ASR systems.

In our future work, the output of the NCPCM will be fused with the ASR beliefs to obtain a new hypothesis. We also intend to introduce ASR confidence scores and signal SNR estimates, to improve the channel model. We are investigating introducing the probabilistic ambiguity of the ASR in the form of lattice or confusion networks as inputs to the channel-inversion model.

Further, we will utilize sequence-to-sequence (Seq2seq) translation modeling [17] to map ASR outputs to reference transcripts. The Seq2seq model has been shown to have benefits especially in cases where training sequences are of variable length [60]. We intend to employ Seq2seq model to encode ASR output to a fixed-size embedding and decode this embedding to generate the corrected transcripts.

FINANCIAL SUPPORT

The US Army Medical Research Acquisition Activity, 820 Chandler Street, Fort Detrick MD 21702- 5014 is the awarding and administering acquisition office. This work was supported by the Office of the Assistant Secretary of Defense for Health Affairs through the Psychological Health and Traumatic Brain Injury Research Program under Award No. W81XWH-15-1-0632. Opinions, interpretations, conclusions and recommendations are those of the author and are not necessarily endorsed by the Department of Defense.

REFERENCES

[1] Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition, in *Proc. of the IEEE*, vol. 77 (2), 1989, 257–286.

[2] Rabiner, L.; Juang, B.-H.: *Fundamentals of Speech Recognition*, Prentice-Hall, Inc., Upper Saddle River NJ, USA, 1993.

[3] Hinton, G. *et al.*: Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Process. Mag.*, **29** (6) (2012), 82–97.

[4] Dahl, G.E.; Yu, D.; Deng, L.; Acero, A.: Context-dependent pretrained deep neural networks for large-vocabulary speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20 (1), 2012, 30–42.

[5] Graves, A.; Mohamed, A.-r; Hinton, G.: Speech recognition with deep recurrent neural networks, in *Proc. of Acoustics, speech and signal processing (ICASSP), 2013 IEEE Int. Conf. on*. IEEE, 2013, 6645–6649.

[6] Graves, A.; Fernández, S.; Gomez, F.; Schmidhuber, J.: Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks, in *Proc. of the 23rd Int. Conf. on Machine learning*. ACM, 2006, 369–376.

[7] Strik, H.; Cucchiarini, C.: Modeling pronunciation variation for ASR: A survey of the literature. *Speech Commun*, **29** (2) (1999), 225–246.

[8] Wester, M.: Pronunciation modeling for ASR–knowledge-based and data-derived methods. *Comput. Speech Lang.*, **17** (1) (2003), 69–85.

[9] Shivakumar, P.G.; Potamianos, A.; Lee, S.; Narayanan, S.: Improving speech recognition for children using acoustic adaptation and pronunciation modeling, in *WOCCI*, 2014, 15–19.

[10] Bahl, L.R.; Brown, P.F.; P.V., de Souza; Mercer, R.L.: A tree-based statistical language model for natural language speech recognition. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37 (7), 1989, 1001–1008.

[11] Moore, R.; Appelt, D.; Dowding, J.; Gawron, J. M.; Moran, D.: Combining linguistic and statistical knowledge sources in natural-language processing for ATIS, in *Proc. ARPA Spoken Language Systems Technology Workshop*, 1995.

[12] Jeff Kuo, H.-K.; Reichl, W.: Phrase-based language models for speech recognition, in *Sixth European Conf. on Speech Communication and Technology*, 1999.

[13] Rosenfeld, R.: Two decades of statistical language modeling: Where do we go from here?, in *Proc. of the IEEE*, vol. 88 (8), 1270–1278, 2000.

[14] Arisoy, E.; Sainath, T.N.; Kingsbury, B., Ramabhadran, B.: Deep neural network language models, in *Proceedings of the NAACL-HLT 2012 Workshop: Will We Ever Really Replace the N-gram Model? On the Future of Language Modeling for HLT*. Association for Computational Linguistics, 2012, 20–28.

[15] Mikolov, T.; Karafiát, M.; Burget, L.; Cernockỳ, J.; Khudanpur, S.: Recurrent neural network based language model., in *Proc. of Interspeech*, vol. 22010, 3.

[16] Sundermeyer, M.; Schlüter, R.; Ney, H.: LSTM neural networks for language modeling, in *Proc. of Interspeech*, 2012, 194–197.

[17] Sutskever, I.; Vinyals, O., Le, Q.V.: Sequence to sequence learning with neural networks, in *Advances in neural information processing systems*, 2014, 3104–3112.

[18] Bahdanau, D.; Chorowski, J.; Serdyuk, D.; Brakel, P., Bengio, Y.: End-to-end attention-based large vocabulary speech recognition, in *Proceedings of Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE Int. Confe. on*. IEEE, 2016, 4945–4949.

[19] Chan, W.; Jaitly, N.; Le, Q., Vinyals, O.: Listen, attend and spell: A neural network for large vocabulary conversational speech recognition, in *Proc. of Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE Int. Conf. on*. IEEE, 2016, 4960–4964.

[20] Ainsworth, W.A.; Pratt, S.R.: Feedback strategies for error correction in speech recognition systems. *Int. J. Man. Mach. Stud.*, **36** (6) (1992), 833–842.

[21] Noyes, J.M., Frankish, C.R.: Errors and error correction in automatic speech recognition systems. *Ergonomics*, **37** (11) (1994), 1943–1957.

[22] Suhm, B.; Myers, B., Waibel, A.: Multimodal error correction for speech user interfaces. *ACM T. Comput-Hum Int.(TOCHI)*, **8** (1) (2001), 60–98.

[23] Sarma, A., Palmer, D.D.: Context-based speech recognition error detection and correction, in *Proc. of HLT-NAACL 2004: Short Papers*. Association for Computational Linguistics, 2004, 85–88.

[24] Ringger, E.K., Allen, J.F.: Error correction via a post-processor for continuous speech recognition, in *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conf. Proc.., 1996 IEEE Int. Conf. on*. IEEE, vol. 1, 1996, 427–430.

[25] Jeong, M.; Jung, S.; Lee, G.G.: Speech recognition error correction using maximum entropy language model, in *Proc. of INTERSPEECH*, 2004, 2137–2140.

[26] DHaro, L.F.; Banchs, R.E.: Automatic correction of ASR outputs by using machine translation. Interspeech, 2016.

[27] Cucu, H.; Buzo, A.; Besacier, L.; Burileanu, C.: Statistical error correction methods for domain-specific ASR systems, in *Int. Conf. on Statistical Language and Speech Processing*. Springer, 2013, 83–92.

[28] Tam, Y.-C.; Lei, Y.; Zheng, J.; Wang, W.: Asr error detection using recurrent neural network language model and complementary ASR, in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE Int. Conf. on*. IEEE, 2014, 2312–2316.

[29] Zheng, D.; Chen, Z.; Wu, Y.; Yu, K.: Directed automatic speech transcription error correction using bidirectional LSTM, in *2016 10th Int. Symp. on Chinese Spoken Language Processing (ISCSLP)*, October 2016, 1–5.

[30] Nakatani, R.; Takiguchi, T.; Ariki, Y.: Two-step correction of speech recognition errors based on n-gram and long contextual information, in *INTERSPEECH*, 2013, 3747–3750.

[31] Byambakhishig, E.; Tanaka, K.; Aihara, R.; Nakashika, T.; Takiguchi, T.; Ariki, Y.: Error correction of automatic speech recognition based on normalized web distance, in *Fifteenth Annual Conf. of the Int. Speech Communication Association*, 2014.

[32] Fusayasu, Y.; Tanaka, K.; Takiguchi, T.; Ariki, Y.: Word-error correction of continuous speech recognition based on normalized relevance distance, in *IJCAI*, 2015, 1257–1262.

[33] Woodland, P.C.; Povey, D.: Large scale discriminative training of hidden Markov models for speech recognition. *Comput. Speech Lang.*, **16** (1) (2002), 25–47.

[34] Roark, B.; Saraclar, M.; Collins, M.: Discriminative n-gram language modeling. *Comput. Speech Lang.*, **21** (2) (2007), 373–392.

[35] Sagae, K. *et al.*: Hallucinated n-best lists for discriminative language modeling, in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE Int. Conf. on*. IEEE, 2012, 5001–5004.

[36] Xu, P.; Roark, B.; Khudanpur, S.: Phrasal cohort based unsupervised discriminative language modeling, in *Thirteenth Annual Conference of the Int. Speech Communication Association*, 2012.

[37] Bikelf, D. *et al.*: Confusion-based statistical language modeling for machine translation and speech recognition, 2012.

[38] Celebi, A. *et al.*: Semi-supervised discriminative language modeling for Turkish ASR, in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE Int. Conf. on*. IEEE, 2012, 5025–5028.

[39] Brown, P.F. *et al.*: A statistical approach to machine translation. *Comput. Linguist.*, **16** (2) (1990), 79–85.

[40] Koehn, P.; Och, F.J.; Marcu, D.: Statistical phrase-based translation, in *Proc. of the 2003 Conf. of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*. ACL, 2003, 48–54.

[41] Vaswani, A.; Zhao, Y.; Fossum, V., Chiang, D.: Decoding with large-scale neural language models improves translation., in *Proc. of the 2013 Conf. on Empirical Methods in Natural Language Processing*. EMNLP, 2013, 1387–1392.

[42] Devlin, J.; Zbib, R.; Huang, Z.; Lamar, T.; Schwartz, R.M.; Makhoul, J.: Fast and robust neural network joint models for statistical machine translation., in *Proc. of the 52nd Annual Meeting of the Association for Computational Linguistics,*. ACL, 2014, 1370–1380.

[43] Alkhouli, T.; Rietig, F.; Ney, H.: Investigations on phrase-based decoding with recurrent neural network language and translation models, in *Proc. of the Tenth Workshop on Statistical Machine Translation*, 2015, 294–303.

[44] Och, F.J.: Minimum error rate training in statistical machine translation, in *Proc. of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*. ACL, 2003, 160–167.

[45] Cieri, C.; Miller, D.; Walker, K.: The Fisher Corpus: a resource for the next generations of speech-to-text, in *Int. Conf. on Language Resources and Evaluation*. LREC, vol. 4, 2004, 69–71.

[46] Rousseau, A.; Deléglise, P.; Estève, Y.: Enhancing the TED-LIUM corpus with selected data for language modeling and more TED talks, in *Int. Conf. on Language Resources and Evaluation*. LREC, 2014, 3935–3939.

[47] Povey, D. *et al.*: The Kaldi speech recognition toolkit, in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011, number EPFL-CONF-192584.

[48] Weide, R.L.: The CMU pronouncing dictionary. *URL: http://www.speech. cs. cmu. edu/cgibin/cmudict*, 1998.

[49] Koehn, P. *et al.*: Moses: Open source toolkit for statistical machine translation, in *Proc. of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*. Association for Computational Linguistics, 2007, 177–180.

[50] Och, F.J.; Ney, H.: A systematic comparison of various statistical alignment models. *Comput. Linguist.*, **29** (1) (2003), 19–51.

[51] Stolcke, A.: SRILM-an extensible language modeling toolkit, in *Seventh Int. Conf. on spoken language processing*, 2002.

[52] Bertoldi, N.; Haddow, B., Fouet, J.-B.: Improved minimum error rate training in Moses. *The Prague Bulletin of Mathematical Linguistics*, **91** (2009), 7–16.

[53] Bikel, D.M.; Hall, K.B.: Refr: an open-source reranker framework, in *INTERSPEECH*, 2013, 756–758.

[54] Papineni, K.; Roukos, S.; Ward, T.; Zhu, W.-J.: BLEU: a method for automatic evaluation of machine translation, in *Proc. of the 40th annual meeting on association for computational linguistics*. ACL, 2002, 311–318.

[55] Neubig, G.; Watanabe, T.: Optimization for statistical machine translation: a survey. *Comput. Linguist.*, **42** (1) (2016), 1–54.

[56] Gillick, L.; Cox, S.J.: Some statistical issues in the comparison of speech recognition algorithms, in *Proc. of Acoustics, Speech, and Signal Processing,. ICASSP-89., 1989 Int. Conf. on*. IEEE, 1989, 532–535.

[57] Tan, Q.F.; Audhkhasi, K.; Georgiou, P.G.; Ettelaie, E.; Narayanan, S.S.: Automatic speech recognition system channel modeling, in *Eleventh Annual Conf. of the Int. Speech Communication Association*, 2010.

[58] Kurata, G.; Itoh, N.; Nishimura, M.: Training of error-corrective model for ASR without using audio data, in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE Int. Conf. on*. IEEE, 2011, 5576–5579.

[59] Dikici, E.; Celebi, A.; Saraçlar, M.: Performance comparison of training algorithms for semi-supervised discriminative language modeling, in *Thirteenth Annual Conf. of the Int. Speech Communication Association*, 2012.

[60] Cho, K.; Van Merriënboer, B.; Bahdanau, D.; Bengio, Y.: On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

Prashanth Gurunath Shivakumar received a B.E degree in Electronics and Communication Engineering from NMAMIT, India (2011). He received the M.S. degree in Electrical Engineering in 2014 and a M.S. degree in Computer Science in 2018 from University of Southern California. He is currently pursuing his Ph.D. within Signal Processing for Communication Understanding and Behavior Analysis (SCUBA) Laboratory at the University of Southern California. He is also a student IEEE member since 2018. His research interests include Automatic Speech Recognition, Natural Language Processing, Speech Processing and Machine Learning.

Haoqi Li received the B.S. degree in electronic information engineering from Xidian University, China, in 2011 and M.S. degree of signal and information processing from University of the Chinese Academy of Sciences in 2014. He is currently working toward the Ph.D. degree at University of Southern California. His research interests relate to machine learning, behavioral signal processing, and speech processing. His publications include work on multimodal behavior learning, supervised and unsupervised learning of behaviors from audio.

Kevin Knight is Chief Scientist for Natural Language Processing (NLP) at DiDi Chuxing and Dean's Professor of Computer Science at the University of Southern California (USC). Dr. Knight received a PhD in computer science from Carnegie Mellon University and a bachelor's degree from

Harvard University. His research interests include human-machine communication, machine translation, language generation, automata theory, and decipherment. Dr. Knight co-authored the widely-adopted textbook "Artificial Intelligence" (McGraw-Hill). In 2001, he co-founded Language Weaver, Inc., a machine translation company acquired by SDL plc in 2010. He served as President of the Association for Computational Linguistics (ACL) in 2011, as General Chair for the Annual Conference of the ACL in 2005, and as General Chair for the North American ACL conference in 2016. He is a Fellow of the ACL, a Fellow of USC's Information Sciences Institute (ISI), and a Fellow of the Association for the Advancement of Artificial Intelligence (AAAI).

Panayiotis (Panos) Georgiou is an Assistant Professor in Electrical Engineering and Computer Science at the University of Southern California (USC), the director of the Signal Processing for Communication Understanding and Behavior Analysis (SCUBA), co-director of the USC Behavioral Informatics Center, and integral member of SAIL (Signal Analysis and Interpretation Lab). His current research interests include behavioral signal processing, speech processing, NLP, and machine learning. He is currently a member of IEEE-SLTC, editor of IEEE Signal Processing Letters, IEEE Signal Processing Magazine, EURASIP Journal on Audio, Speech, and Music Processing, Advances in Artificial Intelligence and served as a guest editor of Computer Speech And Language. He has also served or serves as Technical Chair of InterSpeech 2016; General Chair of ICMI 2018, Area Chair of InterSpeech 2015, '17, '18; and a range of other speech and signal processing conferences. Panos' work has also been featured in over 100 national and international media outlets such as Washington Post, Telegraph U.K., US News and World report, etc. He published over 200 papers and his co-authored papers won 3 best paper awards and an interspeech paralinguistics challenge award and has 6 patents.