From Speech Signals to Semantics – Tagging Performance at Acoustic, Phonetic and Word Levels

Yao Qian[1]*, Rutuja Ubale*[1]*, Patrick Lange*[1]*, Keelan Evanini*[1]*, Frank Soong*[2]

[1]Educational Testing Service Research, USA, [2] Microsoft Research Asia, China

{yqian, rubale, plange, kevanini}@ets.org, frankkps@microsoft.com

Abstract

Spoken language understanding (SLU) is to decode the semantic information embedded in speech input. SLU decoding can be significantly degraded by mismatched acoustic/language models between training and testing of a decoder. In this paper we investigate the semantic tagging performance of bi-directional LSTM RNN (BLSTM-RNN) with input at acoustic, phonetic and word levels. It is tested on a crowdsourced, spoken dialog speech corpus spoken by non-native speakers in a job interview task. The tagging performance is shown to be improved successively from low-level, acoustic MFCC, mid-level, stochastic senone posteriorgram, to high-level, ASR recognized word string, with the corresponding tagging accuracies at 70.6%, 82.1% and 85.1%, respectively. With a score fusion of the three individual RNNs together, the accuracy can be further improved to 87.0%.

Index Terms: spoken language understanding, MFCC, senone, posteriorgram

1. Introduction

Spoken language understanding (SLU) technology is to interpret the semantic meaning conveyed in spoken input for taking appropriate actions in human/computer interactions, e.g. a spoken dialog system (SDS). State-of-the-art SLU in SDS systems generally consists of two key components: automatic speech recognition (ASR) to convert input speech into recognized text and a natural language understanding (NLU) to transform the ASR word string into a concept of semantic labels that can drive subsequent SDS responses. The two components are statistical models trained on a large amount of data with machine learning.

SLU can be regarded as cascaded conversion from speech signal input to semantic tag output in acoustic, phonetic, lexical and semantic spaces. To make SLU robust in the successive conversion process, we need to make: acoustic model (AM) resistant to different acoustic channel conditions and ambient noises; lexical model adaptable to speaker variation in accented pronunciations and out-of-vocabulary words (OOV); language model (LM) flexible to handle syntactic and grammatical variation and NLU model insensitive to ASR errors and variance in pragmatics.

Large field training data with deep learning can address the robustness issues in SLU. However, massive matched training data is not available and difficult to collect when a new application is built from scratch. In this study, we investigate predicting semantic labels in successive stages of SLU, i.e., from speech signal by ASR-free modeling, from sub-phone search space by skipping the language model, and from ASR transcription at word level, and enhancing the performance by fusing the separate tagging results.

2. Related Work

Most state-of-the-art SLUs utilize deep learning technologies to perform semantic tagging with transcriptions or ASR hypotheses [1, 2, 3]. Recurrent neural networks (RNNs) with different architectures have been proposed for semantic slot filling and they were evaluated on the well-known Airline Travel Information System (ATIS) benchmark task. The experimental results show that the RNN-based models outperform the conditional random field (CRF) baseline [1]. Joint slot filling and intent detection based on convolutional neural networks (CNNs), in which the features are extracted through CNN layers and shared by these two tasks, also shows better performance than CRF [2].

Many researchers tried to skip ASR entirely or use only partial information extracted from its modules for semantic classification [4, 5, 6, 7]. Utterance classification is performed by unsupervised phonotactic models together with token sequence classifiers [6], which can avoid manual word-level transcription of the utterances and achieve a performance close to those of conventional methods involving word-level language models. Techniques of building call routers from scratch without any knowledge of the application vocabulary or grammar are also explored in [4].

Recently, research studies have addressed the modeling of speech signals with an end-to-end (E2E) optimization, which utilizes as little a prior knowledge as possible, e.g., using filter-bank features instead of MFCC [8] or directly using the speech waveform [9]. Multiple studies have demonstrated that features automatically extracted by DNNs are far superior to those produced by feature-engineering techniques generally used in GMM-based acoustic modeling, e.g. [10]. E2E speech recognition systems have yielded competitive performance when compared with conventional hybrid DNN-HMM systems [11, 12, 13]. E2E learning has also produced promising results on speaker verification [14], language identification [15] and keyword search [16]. Our previous work [17, 18] and work in [19] show that it is promising but there is still a gap in performance in terms of prediction accuracy between ASR-free E2E modeling of SLU and the conventional SLU, i.e., NLU with ASR hypotheses. To the best of our knowledge, there is few research work exploring SLU modeling from three different levels (acoustic, phonetic and lexical) and their fusion.

3. Trajectory Modeling with BLSTM-RNN

We use an SDS that can leverage a variety of open source components in a framework that is cloud-based, modular and standards compliant. [20] provides further details about the SDS architecture. This study examines an interactive conversational task for English language learners designed to provide speaking practice in the context of a simulated job interview. The conversation is structured as a system-initiated dialog where a

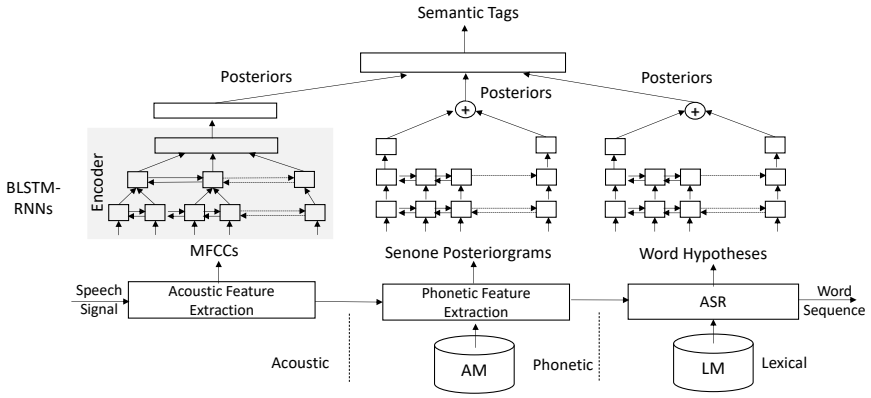

Figure 1: *BLSTM-RNNs with acoustic features, sub-phone posteriorgrams and ASR word hypotheses*

job placement agent interviews an applicant (language learner) about his/her job interests and qualifications.

Predicting the semantic labels of a spoken utterance in job interview conversations is formulated as a problem of classifying a given utterance into an appropriate semantic class. A straightforward way to do semantic utterance classification is to use a sequence-to-tag function, which maps a sequence of input feature vectors to a semantic label. The schematic diagram of learning the sequence-to-tag function by bi-directional long short-term memory (BLSTM) RNN with acoustic feature sequence, sub-phone posteriorgrams and ASR word hypotheses as inputs and concatenating three BLSTM-RNN together is shown in Figure 1.

RNNs configured to process arbitrary length input sequences have been successfully applied to solve a wide range of machine learning problems with sequence data. With BLSTM cells [21], RNN can overcome the vanishing gradient problem in training. For a sequence-to-tag function of semantic utterance classification for spoken dialog systems, the output layer of BLSTM-RNN is a softmax layer which contains semantic labels represented by a one-hot vector and the input layer contains feature vectors along the time axis. The semantic label posteriors generated from three BLSTM-RNNs are concatenated together and modeled by a multilayer perceptron (MLP) to predict the semantic labels again as final predictions, which can be regarded as a score level fusion. An alternative fusion method can be feature level fusion, in which the outputs generated from the middle layers (instead of output layers) of three BLSTM-RNNs are concatenated together to predict the semantic labels.

3.1. Acoustic Features

We have tried to use frame-level spectral information, i.e., MFCC, as input for predicting semantic labels by BLSTM-RNN directly. However, the preliminary results were not encouraging. We conjecture that the approach might be constrained by the limited training data used in our experiments and the resultant model either overfits the training data or mismatches the testing data. Speech acoustic features vary largely from various factors, e.g., age, gender, dialectic accent and personalized speaking style. Even for the same speaker, the actual spectral features change from one trial to the next, caused by different speech rates or even different articulations. Therefore, a large number of spoken utterances tagged with corresponding semantic labels is required to get a decent classifier performance.

Therefore, we employed a pyramid structure, as proposed in [22] and shown in Figure 1, in the encoder layers. The pyramid structure makes the model training converge quickly. The final encoder layer is a fixed-dimensional vector, which can be regarded as spoken sentence embedding. The encoder layers are initialed (pre-trained) by RNN-based acoustic autoencoder [16, 23], in which the acoustic feature vector sequence is mapped unto a fixed-dimensional vector with the Encoder RNN, and the Decoder RNN reconstructs another sequence from the fixed-dimensional vector to minimize the reconstruction error. It is a feature compression based approach with unsupervised learning. Additional speech data without transcriptions can be used to enhance the pre-training performance.

3.2. Sub-phone Posteriorgrams

DNN-based acoustic models for LVCSR use senones (which are tied tri-phone states of the HMM) as the output nodes of the DNN. The senones and the corresponding aligned speech frames by the GMM-HMM are used to train the DNN. In decoding, given the frame-level feature vector, the posteriors of senones are generated as the DNN output. A senone (sub-phone) posteriorgram is the posterior distribution across the whole senone set over all frames in an utterance. It is a matrix where the horizontal axis is time or the frame index, while the vertical axis is marked with senone indices, and the cell value is the posterior of the senone x at time t.

3.3. ASR Hypotheses

Each recognized word is represented by a vector with a Word2Vec model. ASR recognition results, in terms of a sequence of recognized words, can then be represented by a matrix and fed into the BLSTM-RNN for semantic label tagging.

4. Experiments and Results

The performance of the three input features extracted in acoustic, phonetic and lexical spaces, mentioned in Section 3, for semantic utterance classification is evaluated in a spoken-dialog-based language learning application.

4.1. Corpus

Spoken dialog data was collected via crowdsourcing by interacting with non-native interlocutors in a job interview task. The dialog states and the corresponding semantic labels are shown

in Table 1. Dialog-state dependent semantic labels are used in this corpus. The semantic labels with nomatch, which means no response (silence) or unrelated responses, require to be modeled separately for different dialog states. The detailed semantic labels and an example was shown in [17]. The collected dialog corpus consists of 4,778 utterances spoken by 1,179 speakers. Among them, 4,192 utterances are used as the training set and the remaining 586 utterances as the testing set. The out of vocabulary (OOV) rate of the testing set is 8.5%.

4.2. ASR System

ASR system is trained on two corpora with the tools in Kaldi [24]. One corpus is drawn from a large-scale, speech database collected globally for assessing the English proficiency of non-native speakers' ability on speaking and understanding English as a university student. It consists of over 800 hours of non-native, spontaneous speech covering over 100 L1 native languages recorded by 8,700 speakers. Another corpus was collected by the SDS via crowdsourcing for different spoken dialog-based applications. The corpus was collected under realistic application scenarios. The acoustic environments and speaking styles were matched with the data of the job interview task. It consists of 41,185 utterances (roughly 50 hours).

A GMM-HMM is first trained to obtain senones and the corresponding aligned frames for DNN training. Concatenated MFCCs and i-vector features [25], a promising approach to adapting the acoustic model for better speech recognition performance, are used for DNN training. The input features stacked over a 15 frame window (7 frames on either side of the current frame where the output prediction is made) are used as the input layer of the DNN. The output layer of the DNN consists of 3,686 senones. The DNN has 5 hidden layers, and each layer has 1,024 nodes. The Sigmoid activation function is used for all hidden layers. All the parameters of the DNN are firstly initialized by pre-training, then trained by optimizing the cross-entropy function through backpropagation (BP), and finally refined by sequence-discriminative training, state-level minimum Bayes risk (sMBR).

4.3. The Configurations for BLSTM-RNN

BLSTM-RNNs with acoustic features, senone posteriorgrams or ASR hypotheses as input features and semantic labels as output nodes are constructed using the Keras Python package[1]. 15% of the training data is randomly selected to tune the parameters of BLSTM-RNN and avoid overfitting by using early stopping. The merge mode for two directional LSTM-RNN is average. The structures of BLSTM-RNNs are configured for different input features as follows.

4.3.1. Acoustic Features

The input acoustic features to the BLSTM-RNN are 13-dim static MFCCs without delta features or stacked frame window since RNN architecture already captures the long-term temporal dependencies among all sequential events. The silence at the beginning and ending of utterances is deleted with an energy-based, voice activity detection (VAD). A two-layer BLSTM with 256 nodes for the first layer and 128 nodes for the second layer is employed. A layer with 400 nodes is used to compute the embedding from encoder layers. We unfolded encoder RNNs for 10 seconds or 1,000 time steps (frames) where 10 seconds is the median length of utterances in our corpus. Depending upon the length of the utterance, features are either padded with zeros at the end or down-sampled to 1,000 frames. A back-propagation through time (BPTT) learning algorithm is used to train the BLSTM-RNN parameters. A 400-dim embedding vector is then fed into a feed-forward NN with two hidden layers (each layer with 128 nodes) to predict semantic labels. All parameters of NN are trained by optimizing the cross-entropy function through BP.

[1] https://keras.io

Table 1: *Dialog state and semantic labels*

Dialog State	Semantic Labels
Mistake (MT)	coworker, depends, manager, nomatch
Part or Full (PF)	either, full time, nomatch, part time
Self or Group (SG)	both, group, nomatch,self
Work Experience (WE)	yes, no, nomatch

4.3.2. Senone Posteriorgrams

A senone posteriorgram of an utterance is a time sequence of vectors, or equivalently, a 2D tensor (with a shape of # frames $\times$ # senones). We linearly resample each spoken utterance into a fixed number of frames, i.e., 100 in our case, and construct a tensor (with a shape of 100 $\times$ 3,686 as input for BLSTM-RNN training. The structure of the BLSTM-RNN is configured as 32 LSTM cells, a rectified linear unit (ReLU) activation function and a one-half drop-out rate (p=0.5); a categorical cross-entropy loss function and Adadelta optimizer is used in training.

4.3.3. ASR Hypotheses

The input ASR hypothesis sequence is similarly converted to a 2D tensor and fed into a stacked BLSTM-RNN, and formalized as a vector to predict the semantic labels by the softmax output layer. The structure of the BLSTM-RNN is the same as that of the senone posteriorgram except the input is a tensor with a shape (50 $\times$ 300), in which the maximum number of recognized words in an utterance is 50 and the dimension of word embedding vectors is 300, as trained from Google news [2].

4.4. Concatenation of Three BLSTM-RNNs

Three BLSTM-RNNs, each with different inputs, i.e., low-level acoustic features, mid-level sub-phone posteriorgrams and high-level word hypotheses, are concatenated together to predict the semantic labels by two-layer MLP with 32 nodes for each layer. The three networks, based on input information in different granularities, may compensate for each other in predicting the semantic tags jointly. Besides MLP, we also tried other classifiers including Support Vector Machine, Random Forest, Logistic Regression, AdaBoost Decision Tree and etc, provided by SKLL toolkit[3] to do score-level fusion, i.e., use the semantic label posteriors generated from the three BLSTM-RNNs as the input to a classifier to predict the semantic labels again. The hyper-parameters of these classifiers were op-

[2] https://code.google.com/archive/p/word2vec

[3] https://github.com/EducationalTestingService/skll

timized by the SKLL internally with cross validation on the train data. Among all these classifiers, the SVM based classifier achieves the highest accuracy on semantic label prediction. It also slightly outperforms the MLP with the concatenation of three BLSTM-RNNs as inputs. But there is no significant difference between MLP and SVM for the performance of score-level fusion in this task. The feature-level fusion and a single, hierarchical BLSTM-RNN with all the inputs are also adopted as comparison approaches. But neither of them outperforms the score-level fusion with SVM based classifier.

4.5. Experimental Results

The word error rate (WER) of the ASR system on the test set of the corpus mentioned in Section 4.1 is 43.5%. The poor WER is largely due to the poor audio quality of the speech data collected in the field, and also due to there being many non-native speakers. The poor audio quality could be caused by waveform distortions, e.g., clipping when A/D conversion is overdriven, packet losses in unstable internet transmissions, or low signal-to-noise ratio (SNR) caused by a high level of background noise. The speech of non-native speakers in SDS-based language learning applications may contain pronunciation errors, disfluencies, ungrammatical phrases, loan words, etc., which further degrade the ASR performance. In some cases, even the human labelers had difficulties in transcribing the speech. For example, the average inter-transcriber WER is high at 38.3%. The ASR system with i-Vector based speaker adaptation technology can deliver a WER of 18.5% on matched testing data and 23.3% on dialogue data sets by interpolating LMs to compensate for the speaking style difference across different tasks [26]. The high WERs of the spoken dialog data collected via crowdsourcing also motivated us to explore using acoustic features and senone posteriorgrams to predict the semantic labels.

Table 2 shows the performance of SLU in terms of semantic prediction accuracy obtained by different BLSTM-RNNs. The BLSTM-RNN with acoustic features as input performs much better than the majority vote baseline, i.e., from 59.8% to 70.6%, and there is no degradation of each dialog state-dependent performance. These results show an ASR-free SLU is promising in situations with low ASR accuracy. We conjecture that the performance of an ASR-free SLU will be further improved if more training data is available. Using the senone posteriorgrams instead of MFCC as the input to the BLSTM-RNN results in an improvement in tagging accuracy (ALL) of 11.5%, from 70.6% to 82.1%, when compared with the acoustic features. The conventional SLU approach, i.e., using ASR hypotheses as input for predicting the semantic label, can achieve 85.1% accuracy, which is only 3.0% better than that predicted from senone posteriorgrams. In other words, the senone posteriorgrams based SLU model without using an LM trained by manual transcriptions for the new application data can perform almost as well as the traditional ASR-based baseline in its semantic classification performance.

Table 2 also shows the best score-level fusion results achieved by SVM classifer. It indicates that acoustic features, senone posteriorgrams and ASR word hypotheses have non-overlapping information and can compensate for each other in semantic tagging. The score-level fusion by using the posteriors output from the 3 RNNs can further improve the semantic tagging accuracy from 85.1% to 87.0%. It is enlightening to find that the performance can be improved by different combinations of BLSTM-RNNs. For example, both fused systems, i.e., MFCC and ASR words (a+c) and MFCC, posteriorgram and ASR words (a+b+c), can achieve a better performance than that of each individual system.

Table 2: *Performance (tagging accuracy) of BLSTM-RNNs with different inputs*

Dialog State	PF	WE	SG	MT	ALL
Majority Vote	53.6	79.4	45.7	70.3	59.8
MFCC (a)	66.3	83.3	62.2	76.8	70.6
Posteriorgram (b)	88.4	88.2	70.7	82.6	82.1
ASR Hypotheses (c)	89.5	89.2	77.4	85.5	85.1
Fusion (a+c)	90.6	90.2	78.0	85.5	85.8
Fusion (a+b+c)	90.6	93.1	79.3	87.0	87.0

Table 3: *Performance (average tagging accuracy) of the utterances w/ and w/o OOV*

	MFCC	Posterior-gram	ASR Hypotheses	fusion (a+b+c)
w/ OOV	69.1	80.4	78.2	83.0
w/o OOV	71.0	82.6	87.1	88.2

4.6. Analysis of Results on OOV Utterances

One of motivations for using acoustic MFCC features and senone posteriorgrams is to investigate their possible advantages for the case of OOV words which cannot be recognized by ASR. Out of the testing set of 586 utterances, 131 utterances contain OOV words. The performance of average tagging accuracy for the utterances with/without OOVs is shown in Table 3. The senone posteriorgram outperforms ASR hypothesis for OOV utterances, i.e., tagging accuracy is improved from 78.2% to 80.4%. However, for the utterances without OOV, the tagging accuracies are reversed, i.e., 87.1% for using ASR hypothesis vs 82.6% for using senone posteriors. The performance gap between the utterances with OOV and without OOV is much larger for using ASR hypothesis than for using MFCC and posteriorgram. The fusion results also show that tagging performance can be boosted more for the utterances with OOV, from 78.2% (ASR hypothesis) to 83.0% (a+b+c fusion), than for the utterances without OOV, where only a marginal improvement of 1.1% is obtained, i.e., from 87.1% (ASR hypothesis) to 88.2% (a+b+c fusion). In a free conversation-based dialog system where OOV can be common, incorporating the lower-level features for tagging can be advantageous to improving the final SLU performance.

5. Conclusions

In this paper, we have studied the sematic tagging performance on spoken data in a job interview dialog application by using the extracted information with different granularities as input to BLSTM-RNNs, including: MFCC at the acoustic level, posteriorgram at the subphonemic, senone level and recognized word string by ASR at the lexical level. Experimental results show that the multi-level BLSTM-RNNs can utilize information at different levels to improve the semantic labeling performance. By fusing the three RNNs scores together, we can further improve the tagging accuracy to 87%. We will use more speech data for evaluating the learning and interpolation capability of BLSTM-RNNs to further improve SLU performance.

6. References

[1] G. Mesnil, Y. Dauphin, K. Yao, Y. Bengio, L. Deng, D. Hakkani-Tur, X. He, L. Heck, D. Y. G. Tur, and G. Zweig, "Using recurrent neural networks for slot filling in spoken language understanding," *IEEE/ACM Transactions on Audio, Speech and Language Processing, Volume 23 (3), pp. 530-539*, 2015.

[2] P. Xu and R. Sarikaya, "Convolutional neural network based triangular CRF for joint intent detection and slot filling," *In Proc. of ASRU, pp. 78-83*, 2013.

[3] G. Tur, D. Hakkani-Tur, L. Heck, and S. Parthasarathy, "Sentence simplification for spoken language understanding," *In Proc. of ICASSP, pp. 5628-5631*, 2011.

[4] Q. Huang and S. Cox, "Task-independent call-routing," *Speech Communication, Volume 48 (3), pp. 374-389*, 2006.

[5] A. L. Gorin, D. Petrovska-Delacretaz, G. Riccardi, and J. H. Wright, "Learning spoken language without transcriptions," *In Proc. of ASRU, Volume 99*, 1999.

[6] H. Alshawi, "Effective utterance classification with unsupervised phonotactic models," *In Proc. of NAACL HLT, Volume 1, pp. 1-7*, 2003.

[7] Y. Y. Wang, J. Lee, and A. Acero, "Speech utterance classification model training without manual transcriptions," *In Proc. of ICASSP, Volume 1, pp. 553-556*, 2006.

[8] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," *in Proc. of ICML, volume 14, pp. 1764-1772*, 2015.

[9] N. Jaitly and G. Hinton, "Learning a better representation of speech sound waves using restricted boltzmann machines," *in Proc. of ICASSP, pp. 5884-5887*, 2011.

[10] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," *in Proc. of ASRU, pp. 24-29*, 2011.

[11] Y. Miao, M. Gowayyed, and F. Metze, "Eesen: End-to-end speech recognition using deep RNN models and wfst-based decoding," *in Proc. of ASRU, pp. 167-174*, 2015.

[12] D. Bahdanau, J. Chorowski, D. Serdyuk, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," *in Proc. of ICASSP, pp. 4945-4949*, 2016.

[13] M. Bhargava and R. Rose, "Architectures for deep neural network based acoustic models defined over windowed speech waveforms," *in Proc. of INTERSPEECH, pp. 6-10*, 2015.

[14] G. Heigold, I. Mereno, S. Bengio, and N. Shazeer, "End-to-end text-dependent speaker verification," *In Proc. of ICASSP, pp. 5115-5119*, 2016.

[15] W. Geng, W. Wang, Y. Zhao, X. Cai, and B. Xu, "End-to-end language identification using attention-based recurrent neural networks," *In Proc. of Interpseech, pp. 2944-2948*, 2016.

[16] K. Audhkhasi, A. Rosenberg, A. Sethy, B. Ramabhadran, and B. Kingsbury, "End-to-end ASR-free keyword search from speech," *in Proc. of ICASSP*, 2017.

[17] Y. Qian, R. Ubale, V. Ramanarayanan, P. Lange, D. Suendermann-Oeft, K. Evanini, and E. Tsuprun, "Exploring ASR-free end-to-end modeling to improve spoken language understanding in a cloud-based dialog system," *In Proc. of ASRU*, 2017.

[18] Y. Qian, R. Ubale, V. Ramanarayanan, P. Lange, D. Suendermann-Oeft, K. Evanini, and E.Tsuprun, "Towards end-to-end modeling of spoken language understanding in a cloud-based spoken dialog system," *In Proc. of SemDial*, 2017.

[19] D. Serdyuk, Y. Wang, C. Fuegen, A. Kumar, B. Liu, and Y. Bengio, "Towards end-to-end spoken language understanding," *In Proc. of ICASSP*, 2018.

[20] V. Ramanarayanan, D. Suendermann-Oeft, P. Lange, R. Mundkowsky, A. Ivanov, Z. Yu, Y. Qian, and K. Evanini, "Assembling the Jigsaw: How Multiple Open Standards Are Synergistically Combined in the HALEF Multimodal Dialog System," in *Multimodal Interaction with W3C Standards*. Springer, 2017, pp. 295–310.

[21] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation, Volume 9 (8), pp. 1735-1780*, 1997.

[22] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *ICASSP*, 2016.

[23] Y. Chung, C. Wu, C. Shen, H. Lee, and L. Lee, "Audio Word2Vec: unsupervised learning of audio segment representations using sequence-to-sequence autoencoder," *in Proc. of Interspeech*, 2016.

[24] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi speech recognition toolkit," *In Prof. of ASRU*, 2011.

[25] N. Dehak, P. Kenny, R. Dehak, P. Ouellet, and P. Dumouchel, "Front end factor analysis for speaker verification," *IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 19, no. 4, pp. 788798*, 2011.

[26] Y. Qian, X. Wang, K. Evanini, and D. Suendermann-Oeft, "Self-adaptive DNN for improving spoken language proficiency assessment," *in Proc. of Interspeech*, 2016.