A NEURAL PROSODY ENCODER FOR END-TO-END DIALOGUE ACT CLASSIFICATION

Kai Wei [1*], *Dillon Knox* [2*], *Martin Radfar* [1], *Thanh Tran* [1], *Markus Müller* [1], *Grant P. Strimel* [1], *Nathan Susanj* [1], *Athanasios Mouchtaris* [1], *Maurizio Omologo* [1]

[1] Alexa Speech, Amazon, [2] University of Southern California

ABSTRACT

Dialogue act classification (DAC) is a critical task for spoken language understanding in dialogue systems. Prosodic features such as energy and pitch have been shown to be useful for DAC. Despite their importance, little research has explored neural approaches to integrate prosodic features into end-to-end (E2E) DAC models which infer dialogue acts directly from audio signals. In this work, we propose an E2E neural architecture that takes into account the need for characterizing prosodic phenomena co-occurring at different levels inside an utterance. A novel part of this architecture is a learnable gating mechanism that assesses the importance of prosodic features and selectively retains core information necessary for E2E DAC. Our proposed model improves DAC accuracy by 1.07% absolute across three publicly available benchmark datasets.

Index Terms— prosody, dialogue act, gating, pitch, end-to-end

1. INTRODUCTION

Dialogue acts (DAs) are speech acts that represent intentions behind a user's request to achieve a conversational goal [1]. Dialogue act classification (DAC) models aim to discriminate speech act units such as statement, question, backchannel, and agreement. For instance, when a user says "yes", DAC models are used to determine whether the user's intent is to agree with what the voice assistant system has said (DA: agreement) or to signal that the user is paying attention to the system (DA: backchannel).

Recent years have seen significant success in applying deep learning approaches to DAC [2–7]. These approaches use either transcripts [2, 3, 5] or a combination of transcript and audio [4, 7, 8] to predict DAs. However, relying on transcripts has three limitations: First, transcripts are not always available for a spoken dialogue system. Second, collecting oracle transcripts is expensive. Third, errors introduced from transcribing audio have been shown to decrease the performance of DAC significantly [9]. More recently, [6] introduced an end-to-end (E2E) DAC approach, where DAs are directly inferred from audio signals. This approach can address the limitations of using transcripts as the inputs. Yet, how to effectively model audio signals for E2E DAC is underexplored.

Prosody comprises the intonation, rhythm, and stress of spoken language. As highlighted in [10], it represents the non-lexical channel that serves a fundamental role in speech communication among humans. It captures the complex linguistic and semantic contents embedded in spoken language beyond words and their literal meanings [11]. At the syllable/word level, stressing on different syllables of a word can lead to different meanings (e.g., REcord vs. reCORD) [12]. At the sentence level, overall intonational contour contributes to characterize speaker's intention and communicative meanings (e.g., agreement vs. backchannel: yes vs. yes?) [13]. This highly intuitive linguistic phenomena inspired many works to explore ways to incorporate prosodic features for DAC [9, 14–19]. Early research primarily focused on conventional cumulative-statistics [14] and traditional machine learning approaches [15, 19]. Of note, [19] found that the location of the maximum F0 occurrence can effectively distinguish between questions and statements. A pitch contour rising on the second syllable of words such as *okay* can convey a topic shift, affirmation, or backchannel [20, 21]. Recently, neural modeling has emerged as a promising yet understudied approach to encode prosodic features. For instance, convolutional neural networks are used in [9] to model sentence-level prosodic features. However, little research has focused on neural approaches that fuse prosodic and spectral characteristics of audio signals at both syllable/word level and sentence level. Moreover, how best to integrate prosodic features for E2E DAC remains unexplored.

In this work, we propose a novel E2E neural architecture that takes into account the need for characterizing prosodic phenomena co-occurring at different levels inside an utterance. An essential part of this architecture is a learnable gating mechanism that assesses the importance of prosodic features and selectively retains core features necessary for E2E DAC. We compare our proposed model with previous E2E DAC models [4, 7] that only use spectral-based audio features. The results show that our models outperform the reference ones. Further, we compare our neural prosody encoder with the state-of-the-art prosody neural encoder [9] on three public benchmark datasets: DSTC2 [22], and DSTC3 [23], and Switchboard [24]. We show that our proposed model outperforms [9] on all these datasets. We also examine the effects of the gating mechanism and different prosodic features on our proposed model.

2. PROPOSED MODELS

We formulate the problem of E2E DAC tasks as follows: The input is a sequence of raw audio with t time frames, $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_t\}$. Each x_t is converted to the logarithm of mel-scale filter bank energy (LFBE) features $L = \{\ell_1, \ell_2, ..., \ell_t\}$ and prosodic features $P = \{(\boldsymbol{e}_1, \boldsymbol{c}_1), ..., (\boldsymbol{e}_t, \boldsymbol{c}_t)\}$, where $\boldsymbol{e}_i \in \mathcal{R}^{|\boldsymbol{e}_i|}$ and $\boldsymbol{c}_i \in \mathcal{R}^{|\boldsymbol{c}_i|}$ denote energy and pitch features, respectively. Our goal is to correctly classify DAs for each audio input X, namely $\{y^{\text{diag.}}\}$. Figure 1 shows our proposed model. It consists of (i) a local prosodic infusion, (ii) an acoustic encoder, (iii) a global prosodic infusion, and (iv) a DA classifier. We detail each component below.

[2] Work done during author's internship at Amazon Alexa.

[*] Equal contribution.

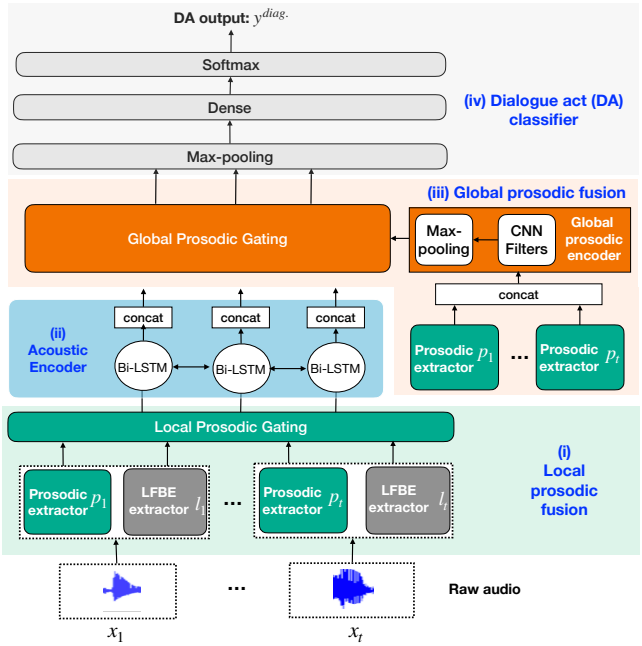


Fig. 1. Our proposed E2E DAC model.

2.1. Local prosodic fusion

The local prosodic infusion encodes prosodic features and combines them with the LFBE features via our local prosodic gating. We extract the LFBE features using Kaldi [25] with a window size of 25 ms, a frame rate of 10 ms, and a sampling frequency of 8 kHz. We describe prosodic extractor and local prosodic gating below:

2.1.1. *Prosodic extractors*

Input: We extract two types of basic prosodic features: energy and pitch. We focus on these two features as they were found to be most important for DAC [9, 19].

- Energy: For each audio frame $\boldsymbol{x}_i \in X$, the 3-dimensional energy features $\boldsymbol{e}_i$ are computed from the 40-mel frequency filter-bank using Kaldi [25], in the same manner as [9]. These features are (i) the log of total energy normalized by the maximum total energy of the utterance, (ii) the log of total energy in the lower 20 mel-frequency bands normalized by total energy, and (iii) the log of total energy in the higher 20 mel-frequency bands, normalized by total energy.
- Pitch: For each audio frame $\boldsymbol{x}_i \in X$, the 3-dimensional pitch features $\boldsymbol{c}_i$ are (i) the warped Normalized Cross Correlation Function (NCCF), (ii) log-pitch with Probability of Voicing (POV)-weighted mean subtraction over a 1.5-second window, and (iii) the estimated derivative of the raw log pitch [26].

Processing: We first concatenate energy $\boldsymbol{e}_i$ and pitch $\boldsymbol{c}_i$ for each audio frame $\boldsymbol{x}_i \in X$. Then, we transform the concatenated $\boldsymbol{e}_i$ and $\boldsymbol{c}_i$ using the linear projection $W^{\boldsymbol{ec}}$ with the $ReLU$ activation function.

$$\boldsymbol{p}_i = ReLU(W^{\boldsymbol{ec}}[\boldsymbol{e}_i; \boldsymbol{c}_i]) \tag{1}$$

Output: We produce $P = \{\boldsymbol{p}_1, \boldsymbol{p}_2, ..., \boldsymbol{p}_t\}$ as a stack of t local prosodic embeddings corresponding to t audio frames of the input audio X, with each $\boldsymbol{p}_i \in P$ computed by Eq. (1).

2.1.2. *Local Prosodic Gating*

High tone/energy sounds can appear in a few segments of the whole input audio. However, these sounds can not contribute equally to the E2E DAC task. Inspired by the gating mechanism in the LSTM architecture [27], we extend a local prosodic gating to selectively combine each local prosodic features $\boldsymbol{p}_i$ in Eq. (1) with LFBE features ℓ_i for each audio frame $\boldsymbol{x}_i$. The local prosodic gating provides a soft mechanism to allow the model to incorporate local prosodic features $\boldsymbol{p}_i$ when needed. Fig. 2(a) illustrates the architecture of our local prosodic gating, which operates as follows:

Input: A stack $P = \{\boldsymbol{p}_1, \boldsymbol{p}_2, ..., \boldsymbol{p}_t\}$ of local prosodic features and a stack $L = \{\ell_1, \ell_2, ..., \ell_t\}$ of local LFBE features.

Processing: A local prosodic gating score β_i is computed from the transformed $\boldsymbol{p}_i$, the transformed ℓ_i, and the interactive features between $\boldsymbol{p}_i$ and ℓ_i. We compute β_i as follows:

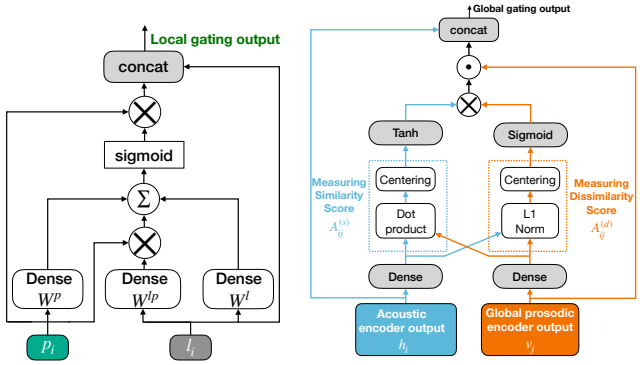
$$\beta_i = \sigma\big(W^p \boldsymbol{p}_i + W^l \ell_i + (W^{lp} \ell_i) \otimes \boldsymbol{p}_i\big), \tag{2}$$

where σ is the *sigmoid* function, $\otimes$ is the element-wise product operator, W^p, W^l, and W^{lp} are learnable parameters.

Output: A stack $A = \{\boldsymbol{a}_1, \boldsymbol{a}_2, ..., \boldsymbol{a}_t\}$ of local acoustic embeddings, where $\boldsymbol{a}_i$ is computed as follows:

$$\boldsymbol{a}_i = [\beta_i \otimes \boldsymbol{p}_i; \ell_i] \tag{3}$$

As shown in Eq. (2) and (3), when the local prosodic gating score $\beta_i \to 1$, $\boldsymbol{a}_i$ generalizes a simple concatenation between $\boldsymbol{p}_i$ and ℓ_i. In contrast, when $\beta_i \to 0$, $\boldsymbol{a}_i$ simply ignores prosodic signals $\boldsymbol{p}_i$ and only keep ℓ_i. Hence, our local prosodic gating provides a flexible mechanism to effectively fuse $\boldsymbol{p}_i$ with ℓ_i.



(a) Local Prosodic Gating. (b) Global Prosodic Gating.

Fig. 2. Architectures of local and global prosodic gating mechanisms.

2.2. Acoustic encoder

The acoustic encoder uses the local acoustic embeddings to produce global acoustic embeddings. Specifically, its **inputs** are the stack of fused local acoustic features $A = \{\boldsymbol{a}_1, \boldsymbol{a}_2, ..., \boldsymbol{a}_t\}$ from the local prosodic gating. We encode A using a n-layer Bi-LSTM acoustic encoder to learn the audio representations. The **outputs** are a stack $H = \{\boldsymbol{h}_1^{(n)}, \boldsymbol{h}_2^{(n)}, ..., \boldsymbol{h}_t^{(n)}\}$ of output hidden states at the last layer n computed as follows:

$$\begin{aligned} &h_i^{(k)} = W_h^{(k)} [\overrightarrow{LSTM}(h_i^{(k-1)}, \overrightarrow{h}_{i-1}^{(k)}); \overleftarrow{LSTM}(h_i^{(k-1)}, \overleftarrow{h}_{i+1}^{(k)})] \\ &\text{with } i \in [1, t],\ \overrightarrow{h}_0^{(k)} = \overrightarrow{0},\ \overleftarrow{h}_{t+1}^{(k)} = \overrightarrow{0},\ \text{and } h_i^{(0)} = \boldsymbol{a}_i \end{aligned} \tag{4}$$

where $\overrightarrow{h}_i^{(k)}$ and $\overleftarrow{h}_i^{(k)}$ are the hidden states at time frame i and layer k, which learn from *left-to-right* and *right-to-left*, respectively.

2.3. Global prosodic infusion

The global prosodic gating encodes prosodic features from the entire audio stream and fuses them with the acoustic encoder outputs via our proposed global prosodic gating mechanism (see Fig. 2(b)).

2.3.1. Global Prosodic Encoder

Inspired by [9], we design a 2-D CNN module to encode global prosodic features at varying timescales using multiple convolution filters. Each filter output is max-pooled, stacked, and flattened to output the global prosodic feature matrix V. We set the stride to 1 and use different kernel lengths (5, 10, 25, 50).

2.3.2. Global Prosodic Gating

We design a global prosodic gating layer based on [28] that learns in parallel a pair-wise similarity matrix and a pair-wise dissimilarity matrix between the global prosodic embedding matrix V and the acoustic encoder output matrix H. Under this dual affinity scheme, the pair-wise similarity matrix is followed by the *tanh* function, resulting in similarity scores between [-1, 1], which controls the *addition* and *subtraction* of V and H. The pair-wise dissimilarity matrix, on the other hand, is served as a gating mechanism that reduces V-H similarity scores to *zero* when prosodic information is not necessary. Our approach overcomes the limitation of the attention method [29,30] that uses the *softmax* operator by allowing *delete* or *subtract* V from H. We detail our proposed global gating below:
Input: Global prosodic embedding matrix V and acoustic encoder output matrix H.
Processing: As shown in Fig. 2(b), we first project each $\boldsymbol{h}_i \in H$ and $\boldsymbol{v}_j \in V$ into a space with the same dimension. This serves our goal of measuring affinity matrices between H and V.

$$\boldsymbol{h}'_i = W^{\boldsymbol{h}}\boldsymbol{h}_i, \quad \boldsymbol{v}'_j = W^{\boldsymbol{v}}\boldsymbol{v}_j \tag{5}$$

Next, we compute an affinity matrix $A^{(s)}$, which measures pair-wise similarities between H and V, where each entry $A^{(s)}_{ij}$ indicates a pair-wise similarity score between $\boldsymbol{h}_i \in H$ and $\boldsymbol{v}_j \in V$. $A^{(s)}_{ij}$ is measured as follows:

$$A^{(s)}_{ij} = \boldsymbol{h}'_i.\boldsymbol{v}'^{T}_j \tag{6}$$

Before computing *tanh*($A^{(s)}$), we want to ensure that $A^{(s)}$ has both positive and negative values, which encapsulates both the signal addition and subtraction. Thus, we first normalize $A^{(s)}$ to have a *zero* mean, then applying the *tanh* function on the normalized $A^{(s)}$.

$$S = tanh[A^{(s)} - mean(A^{(s)})] \tag{7}$$

In the same manner, we formulate an affinity matrix $A^{(d)}$, which measures pair-wise dissimilarities between H and V.

$$A^{(d)}_{ij} = -\|\boldsymbol{h}'_i, \boldsymbol{v}'_j\|_{l_1}, \tag{8}$$

where $\|\cdot\|_{l_1}$ indicates the L_1 distance between two input feature vectors. From $A^{(d)}$, we formulate a gating matrix G, which acts as a mechanism to erase unnecessary global prosodic signals by:

$$G = \sigma[A^{(d)} - mean(A^{(d)})], \tag{9}$$

where σ is the *sigmoid* function. Since L_1 distance is non-negative, $\sigma(A^{(d)}) \in [0, 0.5]$. To ensure $G \in [0, 1]$, we normalize $A^{(d)}$ to have a *zero* mean (see Eq. 9).
Output: We produce a matrix F as the fusion of H and V by concatenating H with the attended V as follows:

$$F = [H; (S \otimes G)V] \tag{10}$$

Last, we apply the *max-pooling* operator on F to obtain a final representation vector f = *max-pooling*(F) of the input audio X.

2.4. Dialogue act classification

For each input audio X, we use the acoustic representation vector f as the output of the global prosodic infusion component and produce a DA distribution over all DAs (D) in the input dataset. The cross entropy loss for the input audio X is defined as:

$$\begin{aligned} \widehat{y}^{diag}_X &= \text{softmax}(W^f f) \\ \mathcal{L}_X &= -\sum_{d=1}^{D} y^{diag}_{X,d} \log\big(\widehat{y}^{diag}_{X,d}\big) \end{aligned} \tag{11}$$

3. EXPERIMENT SETTINGS

Datasets: We use three public benchmark datasets to train and evaluate our proposed models: DSTC2 [22], DSTC3 [23], and Switchboard Dialogue Act corpus (SwDA) [24, 31]. Table 1 provides descriptions for each dataset.

The DSTC2 is a standard dataset for tracking the dialogue state. Each utterance has a corresponding audio recording and the associated DAs. The DA is represented in a triple of the following form (actionType, slotName, slotValue). In this work, we treat each utterance in a dialogue as independent because our focus is to examine whether prosodic contexts of the current utterance is useful to its DAC or not. If an utterance only contains one DA label, we use that label. If an utterance contains more than one label, we combine all the labels for that utterance into a single label. In total, the DSTC2 has 15 unique DA labels.

The DSTC3 is an extension of DSTC2 to a broader domain without providing any further in-domain training data. We adopted the same labelling strategy as DSTC2. In total, the DSTC3 has 17 unique DA labels[1].

The SwDA is a collection of 1,155 five-minute telephone conversations between 543 speakers of American English. It was originally collected by [24]. The DAs were annotated as a part of the SWBD-DAMSL project [31]. We identified the corresponding audio for each annotated split using the unique conversation id for each utterance. In total, there are 42 unique DA labels[2].

Table 1. Number of utterances (hours) of each dataset

	Train	Validation	Test
DSTC2	12,930 (4.6)	1,437 (0.5)	9,116 (3.2)
DSTC3	10,870 (8.7)	1,551 (1.3)	3,100 (2.5)
SwDA	192,768 (289.1)	3,196 (4.8)	4,088 (6.7)

Baseline models: We build an E2E DAC model without any prosodic features (hereafter, baseline), and a model where prosodic information is concatenated with LFBEs at the local level (hereafter, local concat). We also report publicly available E2E DAC accuracy from [4] and [7]. Further, we evaluate our proposed encoder against the state-of-the-art prosody encoder [9].
Experimental Setup: We report test set accuracy of E2E DAC in all datasets. All experiments are implemented by using PyTorch [32]. Training is performed using the Adam optimizer [33] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The initial learning rate was set to 1e-4 and 5e-4 for the DSTC2 and DSTC3 datasets, respectively. We use a batch size of 32 and train for 60 epochs, with check-pointing based on validation loss. We run each experiment 10 times and report the mean and standard deviation of the accuracy score. The

[1] See DSTC2 and DSTC3 at https://github.com/matthen/dstc.

[2] The annotated DA train, validation, and test splits are available at https://github.com/NathanDuran/Switchboard-Corpus

Mann-Whitney U test [34] is used to determine the statistical significance level of the proposed model accuracy improvement. For the LSTM acoustic encoder, we use a three-layer Bi-LSTM acoustic encoder, with each layer containing 512 hidden units.

4. RESULTS

Overall Accuracy: Table 2 shows the overall accuracy of our proposed models and baselines on three benchmark datasets. We observe that just adding the pitch and energy prosodic information (local concat[3]) improves accuracy by 0.69% absolute for DSTC3 and 0.4% absolute for SwDA ($p < 0.05$). Further, our proposed model improves E2E DAC across all three benchmark datasets, with 0.39%, 1.65%, and 1.17% absolute increases in accuracy ($p < 0.05$) on DSTC2, DSTC3, and SwDA, respectively, suggesting the critical role of prosodic information in E2E DAC tasks.

Table 2. Overall model accuracy. * indicates a significant increase from the baseline (Mann-Whitney U test, $p < 0.05$)

	DSTC2	DSTC3	SwDA
Baseline	93.18±.52	91.01 ±.39	55.80±.56
Ortega et al. [4]	–	–	50.9
He et al. [7]	–	–	56.19
Local concat	93.23±.40	91.70±.51*	56.20±.48*
Our model	**93.57**±.30	**92.66**±.30*	**56.97**±.46*

The effects of local and global gating: Table 3 shows the effects of our proposed gating method. We individually removed gating at the global level and local level. We observe that adding the gating mechanism leads to improvements at both local and global levels. At global level, we found that adding the gating mechanism leads to 0.36% absolute improvement on average over the three datasets, and that the improvement is most pronounced for DSTC3 with 0.72% absolute improvement. At local level, adding gating leads to 0.30% absolute improvement on average over the three datasets, and that the improvement is most pronounced for SwDA with 0.51% absolute improvement. It is worth noting that our proposed prosody encoder method also outperforms the global encoder method proposed by [9].

Table 3. Effect of gating mechanisms on E2E DAC accuracy.

	DSTC2	DSTC3	SwDA
Our model	**93.57**±.30	**92.66**±.30	**56.97**±.46
Global encoder [9]	93.42±.33	91.94±.42	56.75±.31
Local gating	93.41±.29	91.91±.33	56.71±.40
Local concat	93.23±.40	91.70±.51	56.20±.48

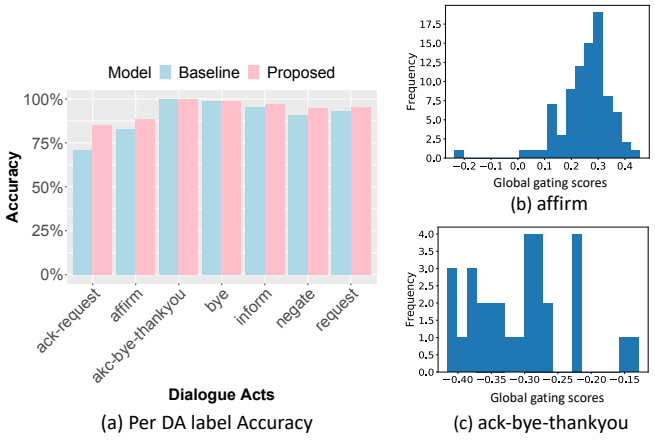
The effects of pitch and energy features: Table 4 shows the effects of pitch and energy features when training our best-performing model. Specifically, we investigate the performance when removing the energy feature group and when removing the pitch feature group. We observe pitch features are more critical than energy features for our proposed model. For example, the accuracy of our proposed model drops almost 1% in the absolute E2E accuracy for SwDA when pitch features are absent, whereas the accuracy of our proposed model drops only 0.6% when energy features are absent.

[3]Note that in the case of *Local concat*, the architecture collapses into a set of Bi-LSTM layers (i.e., without global prosodic fusion and without any gating), and its inputs only consist of the concatenation of LFBEs and prosody extractor outputs, frame by frame. In the case of *Local gating*, the right portion of the architecture in Fig.1 (i.e., without global prosodic fusion) is not activated. In the case of *Baseline*, the architecture receives only LFBEs features as inputs.

Table 4. Effect of different prosodic information.

	DSTC2	DSTC3	SwDA
Our model	**93.57**±.30	**92.66**±.30	**56.97**±.46
w/o energy	93.37±.12	92.53±.24	56.38±.49
w/o pitch	93.30±.28	92.25±.27	55.99±.27

Fig. 3. DA label-wise accuracy and distribution of gating scores.



DA label-wise analysis: Fig. 3 shows the DA label-wise accuracy and gating scores for our proposed model on the DSTC3 dataset. The *ack-request* gains the most improvement, with a 14.26% absolute increase in accuracy (see Fig. 3(a)). After listening to the audio, we found that the *ack* and *request* sequences are connected without any pause, with the *ack* consisting of only one word (e.g., okay). This finding is consistent with [20,21]'s linguistic observation where affirmative cue words such as *okay* can be distinguished by rising pitch. Our proposed gating mechanism was expected to solve complex situations, such as multiple dialogue acts, with one localized in a short portion of the utterance. To verify, we show the distributions of the global gating scores for *affirm* (the DA class where our model outperforms the base model and also contains mostly affirmative cue words such as *yes*) and *ack-bye-thankyou* (the DA class where our model and the base model perform similarly) in Figure 3(b) and Figure 3(c), respectively. The gating scores of *affirm* is mostly distributed in [0.00, 0.40], indicating that prosodic features contribute positively and help improve our model's performance. In contrast, the scores of *ack-bye-thankyou* are mostly distributed in [-0.50, -0.1], suggesting that prosodic features contribute negatively and do not help the performance. This suggests the need of our gating mechanisms to correctly recognize DA labels.

5. CONCLUSION

In this work, we introduced a novel neural model architecture to incorporate prosodic features into an acoustic encoder for E2E DAC. We focused on pitch and energy features and integrated them at both local level and global level. Our experiments show that our proposed approach provides improvements over the state-of-the-art solutions. In the future, we plan to pave the way for integrating the proposed neural architecture with other prosody-related acoustic cues, such as speaking-rate.

6. REFERENCES

[1] J. Austin, *How to do things with words*, 1962.

[2] Q. Tran, I. Zukerman, and G. Haffari, "A hierarchical neural model for learning sequences of dialogue acts," in *ACL*, 2017, vol. 1, pp. 428–437.

[3] Y. Ji, G. Haffari, and J. Eisenstein, "A latent variable recurrent neural network for discourse relation language models," *NAACL*, p. 332–342, 2016.

[4] D. Ortega and N. Vu, "Lexico-acoustic neural-based models for dialog act classification," in *ICASSP*, 2018, pp. 6194–6198.

[5] S. Shen and H. Lee, "Neural attention models for sequence classification: Analysis and application to key term extraction and dialogue act detection.," in *Interspeech 2016*, 2016, pp. 2716–2720.

[6] V. Dang, T. Zhao, S. Ueno, H. Inaguma, and T. Kawahara, "End-to-end speech-to-dialog-act recognition," in *Interspeech 2020*, 2020, pp. 3910–3914.

[7] X. He, Q. Tran, W. Havard, L. Besacier, I. Zukerman, and G. Haffari, "Exploring textual and speech information in dialogue act classification with speaker domain adaptation," *Proceedings of the Australasian Language Technology Association Workshop 2018*, pp. 61–65, 2018.

[8] F. Julia, K. Iftekharuddin, and A. Islam, "Dialog act classification using acoustic and discourse information of maptask data," *International Journal of Computational Intelligence and Applications*, pp. 289–311, 2010.

[9] Trang Tran, *Neural Models for Integrating Prosody in Spoken Language Understanding*, Ph.D. thesis, University of Washington, 2020.

[10] S Wallbridge, P. Bell, and C. Lai, "It's not what you said, it's how you said it: discriminative perception of speech as a multichannel communication system," in *Interspeech 2021: The 22nd Annual Conference of the International Speech Communication Association*, 2021.

[11] D. Dahan, "Prosody and language comprehension," *Wiley Interdisciplinary Reviews: Cognitive Science*, 2015.

[12] N. Ward, *Prosodic Patterns in English Conversation*, 2019.

[13] K. Honda, "Physiological factors causing tonal characteristics of speech: from global to local prosody," in *Speech Prosody 2004, International Conference*, 2004.

[14] E. Shriberg, A. Stolcke, D.Jurafsky, N. Coccaro, M. Meteer, R. Bates, P. Taylor, K. Ries, R. Martin, and C. Ess-Dykema, "Can prosody aid the automatic classification of dialog acts in conversational speech," *Language and Speech*, vol. 41, pp. 443–492, 1998.

[15] M. Zimmermann, "Joint segmentation and classification of dialog acts using conditional random fields.," in *INTERSPEECH*, 2009, pp. 864–867.

[16] S. Quarteroni, A. Ivanov, and G. Riccardi, "Simultaneous dialog act segmentation and classification from human-human spoken conversations," in *2011 ICASSP*, 2011, pp. 5596–5599.

[17] J. Ang, Yang Liu, and E. Shriberg, "Automatic dialog act segmentation and classification in multiparty meetings," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, 2005, vol. 1, pp. 1061–1064.

[18] A. Stolcke, N. Coccaro, R. Bates, P. Taylor, C. Ess-Dykema, K. Ries, E. Shriberg, D. Jurafsky, R. Martin, and M. Meteer, "Dialogue act modeling for automatic tagging and recognition of conversational speech," *Computational Linguistics*, vol. 26, no. 3, pp. 339–373, 2000.

[19] H. Arsikere, A. Sen, A. Prathosh, and V. Tyagi, "Novel acoustic features for automatic dialog-act tagging," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 6105–6109.

[20] A. Gravano, S. Benus, H. Chavez, J. Hirschberg, and L. Wilcox, "On the role of context and prosody in the interpretation of 'okay'," in *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 2007, pp. 800–807.

[21] S. Benus, A. Gravano, and J. Hirschberg, "The prosody of backchannels in american english," *ICPhS*, 2007.

[22] H. Matthew, T. Blaise, and W. Jason, "The second dialog state tracking challenge," in *Proceedings of the 15th annual meeting of the special interest group on discourse and dialogue (SIGDIAL)*, 2014, pp. 263–272.

[23] H. Matthew, T. Blaise, and W. Jason, "The third dialog state tracking challenge," in *2014 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2014, pp. 324–329.

[24] J. Godfrey, E. Holliman, and J. McDaniel, "Switchboard: telephone speech corpus for research and development," in *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1992, vol. 1, pp. 517–520.

[25] D. Povey, A. Ghoshal, G. Boulianne, N. Goel, M. Hannemann, Y. Qian, P. Schwarz, and G. Stemmer, "The kaldi speech recognition toolkit," in *In IEEE 2011 workshop*, 2011.

[26] P. Ghahremani, B. BabaAli, D. Povey, K. Riedhammer, J. Trmal, and S. Khudanpur, "A pitch extraction algorithm tuned for automatic speech recognition," in *ICASSP*, 2014.

[27] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[28] T. Yi, L. Tuan, Z. Aston, W. Shuohang, and H.Cheung, "Compositional de-attention networks," in *Advances in Neural Information Processing Systems*, 2019.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, vol. 30, pp. 5998–6008.

[30] D. Bahdanau, K. Cho, and B. Yoshua, "Neural machine translation by jointly learning to align and translate," in *ICLR*, 2015.

[31] D. Jurafsky, E Shriberg, and D. Biasca, "Switchboard swbd-damsl shallow-discourse-function annotation coders manual, draft 13. technical report," 1998, pp. 97–02.

[32] P. Adam et al., "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems*. 2019.

[33] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[34] P. McKnight and J. Najab, "Mann-whitney u test," *Corsini Encyclopedia of Psychology*, pp. 1–1, 2010.