Reinforcement Learning To Adapt Speech Enhancement to Instantaneous Input Signal Quality

Rasool Fakoor *
Dept. of Computer Science and Engineering
Univ. of Texas at Arlington, TX 76019
rasool.fakoor@mavs.uta.edu

Xiaodong He, Ivan Tashev and Shuayb Zarar
Microsoft AI and Research
Redmond WA 98052
xiaohe,ivantash,shuayb@microsoft.com

Abstract

Today, the optimal performance of existing noise-suppression algorithms, both data-driven and those based on classic statistical methods, is range bound to specific levels of instantaneous input signal-to-noise ratios. In this paper, we present a new approach to improve the adaptivity of such algorithms enabling them to perform robustly across a wide range of input signal and noise types. Our methodology is based on the dynamic control of algorithmic parameters *via* reinforcement learning. Specifically, we model the noise-suppression module as a black box, requiring no knowledge of the algorithmic mechanics except a simple feedback from the output. We utilize this feedback as the reward signal for a reinforcement-learning agent that learns a policy to adapt the algorithmic parameters for every incoming audio frame (16 ms of data). Our preliminary results show that such a control mechanism can substantially increase the overall performance of the underlying noise-suppression algorithm; 42% and 16% improvements in output SNR and MSE, respectively, when compared to no adaptivity.

1 Introduction

Noise-suppression algorithms for a single-channel of audio data employ machine-learning or statistical methods based on the amplitude of the short-term Fourier Transform of the input signal (Ephraim and Malah, 1984; Ephraim and Trees, 1995; Boll, 1979; Xu et al., 2014). Although the approach we propose can be applied to the entire gamut of noise-suppression techniques, in this paper, we only illustrate its benefits with the classical algorithms for speech enhancement that are based on spectral restoration (Tashev et al., 2009). Such algorithms typically comprise four components (Tashev et al., 2009): (1) voice-activity detection, (2) noise-variance estimation, (3) suppression rule, and (4) signal amplification. The first two components help gather statistics on the target speech signal in the input audio, while the third and fourth components allow us to utilize these statistics to distill out the estimated speech signal. Despite these components being based on sound mathematical principles (Tashev et al., 2009), their performance is directly influenced by a sizeable set of parameters such as those that control the gain, geometry weighting, estimator bias, voice- and noise-energy thresholds *etc.* Consequently, the combined set of these parameters plays a critical role in achieving the best performance of the end-to-end speech-enhancement process. Needless to say, the numerical values of these parameters are heavily influenced by the input signal and noise characteristics. Furthermore, thanks to the complex interdependency between statistical models (Tashev et al., 2009), there is no known best value for these parameters that works well across all levels of input signal quality. Therefore any offline optimization process, such as the simplex method (Nash, 2000), is only a *sub-optimal solution* that tends to achieve good performance across only a small range of instantaneous input signal-to-noise ratios (SNRs). This is the status quo that we intend to break.

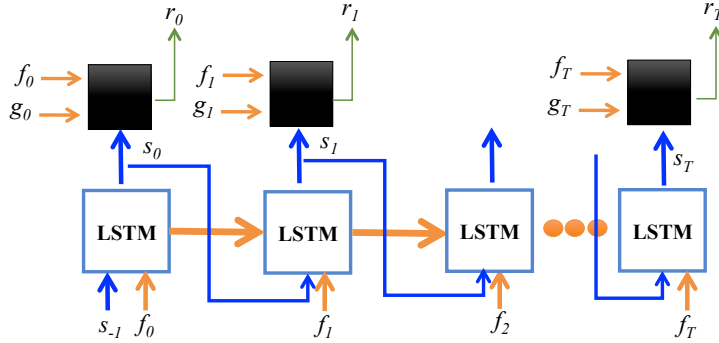*Work was done as an intern at Microsoft AI and Research, Redmond.

Figure 1: Proposed architecture. g_t and f_t are clean and noisy input frames, while s_t and r_t are action and reward values at time step t, respectively. At each time step, our model utilizes s_t and f_t to find the best set of control parameters that maximize r_t of the speech-enhancement black-box algorithm.

2   Proposed Approach

We develop data-driven techniques that allow us to adjust control parameters of a classical speech-enhancement algorithm (Tashev et al., 2009) dynamically at a frame level, depending just on simple feedback from the underlying algorithm. Specifically, we rely on reinforcement-learning (RL) based on a network of long-short term memory (LSTM) cells (Hochreiter and Schmidhuber, 1997). We show that our method can achieve the best performance of speech enhancement across a broad range of input SNRs.

The neural-network model that we propose is shown in Fig. 1. As mentioned before, it treats the classical speech-enhancement algorithm (Tashev et al., 2009) as a black box. Suppose s_t and r_t represent the action proposed by our network model and the reward signal that is available to it from the algorithm at an instance of time t. We set up an objective function as follows:

$$J^{\pi}(\theta) = \mathbb{E}_{\pi_\theta} \left[\sum_{t=0}^{T} r_t(s_t) \right] \tag{1}$$

where π_θ is the policy that our model tries to learn. The goal of this function is to maximize the expected reward over time. Thus, in order to solve Eq. (1), we employ the REINFORCE algorithm (Williams, 1992; Zaremba and Sutskever, 2015; Ranzato et al., 2016; Mnih et al., 2014; Ba et al., 2015; Sutskever et al., 2014). At each time step, our network picks an action at a given state of the model (i.e. given it policy π_θ) that causes some change to the set of control parameters that are applied to the black box. In other words, each parameter of the speech-enhancement algorithm (Tashev et al., 2009) is mapped to an action. Thus, an action can result in an increase or decrease of the parameter value by a specific step size. It could also lead to no change in the parameter values. Note that within the black-box algorithm, we also utilize the clean signal g_t in addition to the noisy frame at every time step. The reason we do this is because once the underlying speech-enhancement algorithm (Tashev et al., 2009) is done with the denoising process, it relies on the ground-truth clean signal g_t to compute a score that represents the reward function for the RL agent. g_t is not used within the black box in any other way.

3   Experimental Results

We evaluated the performance of our methodology with single-channel recordings based on real user queries to the Microsoft Cortana Voice Assistant. We split studio-level clean recordings into training, validation and test sets comprising 7500, 1500 and 1500 queries, respectively. Further, we mixed these clean recordings with noise data (collected from 25 different real-world environments), while accounting for distortions due to room characteristics and distances from the microphone. Thus, we convolved the resulting noisy recordings with specific room-impulse responses, and scaled them to achieve a wide input SNR range of 0-30 dB. Each (clean and noisy) query has on average more than 4500 audio frames of spectral amplitudes, each lasting 16 ms. We applied a Hann weighting window

Method	SNR(dB)	LSD	MSE	WER	SER	PESQ
Noisy Data	15.18	23.07	0.04399	15.4	25.07	2.26
Baseline (Tashev et al., 2009)	18.82	22.24	0.03985	14.77	25.93	2.40
Proposed (Unbiased Estimation)	**26.16**	**21.48**	**0.03749**	17.38	31.87	2.40
Proposed (Biased Estimation)	**26.68**	**21.12**	**0.03756**	18.97	32.73	2.38
Clean Data	57.31	1.01	0.0	2.19	7.4	4.48

Table 1: The proposed RL approach improves MSE, LSD and SNR with no algorithmic changes to the baseline speech-enhancement process, except frame-level adjustment of the control parameters.

to the frames allowing accurate reconstruction with a 50% overlap. These audio frames in the spectral domain formed the features for our algorithm. Since we utilized a 512-point short-time Fourier Transform (STFT), each feature vector was a positive real number of dimensionality 256. To train our network model, we employed a first-order stochastic gradient-based optimization method, Adam (Kingma and Ba, 2015), with a learning rate that was adjusted on the validation set. Furthermore, we used a single layer LSTM with 196 hidden units and number of steps equal to the number of frame. We trained it with a batch size of 1 given each of input file has different number of frames.

To compute the reward function, we employed the negated mean-squared error (MSE) between the ground-truth clean signals g_t and denoised input signals $\hat{g}_t$ as follows:

$$r_t = -\|g_t - \hat{g}_t\|_2^2. \tag{2}$$

Further, to avoid instability during training, we normalized this reward function to lie between $[-1, 1]$. Our underlying speech-enhancement algorithm (Tashev et al., 2009), i.e black-box, was based on a generalization of the decision-directed approach, first defined in (Ephraim and Malah, 1984). To quantify the performance of speech enhancement, we employed the following metrics:

- Signal-to-noise ratio (SNR) dB
- Log spectral distance (LSD)
- Mean squared error in time domain (MSE)
- Word error rate (WER)
- Sentence error rate (SER)
- Perceptual evaluation of speech quality (PESQ)

A larger value is desirable for the first and last metrics, while a lower value is better for the rest. To compute the WER and SER, we employed a production-level automatic speech recognition (ASR) algorithm, whose acoustical model was trained separately on a different dataset that had similar statistics as our training examples. Thus, the ASR algorithm was not re-trained during our speech-enhancement experiments. Results of evaluating our model on the test data are shown in Table 1. In the baseline approach (Tashev et al., 2009), we utilized a non-linear solver to find the set of algorithmic parameters that achieved the best score for a multi-variable function that equally weighted all of the above metrics. This unconstrained optimization was performed *offline* once across the training data, which resulted in a parameter set that achieved the best trade-off for all metrics and feature vectors across the input SNR range. This parameter set was held constant when the speech-enhancement algorithm (Tashev et al., 2009) was applied to the test audio frames. However, in the proposed approaches (third and fourth rows in the table), the parameter set was adjusted depending on the action proposed by our RL meta-network. The third row corresponds to utilizing LSTM-based RL alone on top of the baseline speech-enhancement algorithm [also known as the unbiased estimator that utilizes the reward function of Eq. (2)]. While in the fourth row, we add an additional step of reducing the variance of gradient estimation during REINFORCE (also known as the biased estimation).

From Table 1, we see that the proposed models show better performance on MSE, LSD and SNR, improving them by up to 16%, 4%, and 42%, respectively. However, they do not show improvement on the other metrics (WER, SER and PESQ). In fact, these results are expected because our RL network only employs a measure for signal distortion as the reward function [see Eq. (2)]. Thus, it is able to only optimize metrics that are related to this measure (*i.e.*, MSE, SNR and LSD). The

difficulty of including WER, SER and PESQ in the RL optimization process lies in the fact that these metrics do not provide a direct way of quantifying representation error. There is also no good signal-level proxy for them that can be computed with a low processing cost, which is necessary to train the RL algorithm in practical amounts of time. Thus, although our network already deals with a hard and complex optimization problem due to the black-box optimization and policy gradients, as part of future work, we are continuing to investigate different methods of incorporating WER, SER and PESQ functions into the RL reward signal.

4 Discussion and Future Work

In this ongoing work, we proposed a method based on black-box optimization and RL to deal with the problem of adapting speech-enhancement algorithms to varying input signal quality. Our work is related to hyperparameter optimization in deep learning and machine learning[2], which has been extensively studied in the literature. Methods like random search (Bergstra and Bengio, 2012), Bayesian optimization with probabilistic surrogates (*e.g.*, Gaussian processes (Snoek et al., 2012; Henrández-Lobato et al., 2014)) or deterministic surrogates (*e.g.*, radial basis functions (Ilievski et al., 2017)) have been used to find the best setup for the model hyperparameters. However, once these methods find a set of parameters for a given model offline, the set typically remains fixed throughout the inference process. In contrast, our RL approach adaptively changes parameters of the underlying (data-driven or analytical) algorithm at inference time, achieving the best performance under all input signal conditions. Furthermore, in this particular paper, we demonstrated how to apply our dynamic parameter-adaptation technique to the problem of speech enhancement Tashev et al. (2009). To the best of our knowledge, black-box optimization using reinforcement learning for real-time application such as speech enhancement has not been conducted before, the previous work (Chen et al., 2017) only studies a simple synthetic task. Based on experiments with real user data, we showed that our RL agent is very effective in changing algorithmic parameters at a frame level, enabling existing speech-enhancement algorithms to adapt to changing input signal quality and denoising performance. However, there are still hurdles that need to be overcome in the design of a reliable reward function that helps us achieve the best algorithmic performance across a diverse range of metrics including WER, SER and PESQ. We intend to address this challenge in future work, in addition to reducing the overhead of RL computation during training.

References

Ba, J., Mnih, V., and Kavukcuoglu, K. (2015). Multiple object recognition with visual attention. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305.

Boll, S. (1979). Suppression of acoustic noise in speech using spectral subtraction. *IEEE Transactions on acoustics, speech, and signal processing*, 27(2):113–120.

Chen, Y., Hoffman, M. W., Colmenarejo, S. G., Denil, M., Lillicrap, T. P., Botvinick, M., and de Freitas, N. (2017). Learning to learn without gradient descent by gradient descent. In *Proceedings of the International Conference on Machine Learning (ICML)*, volume 70, pages 748–756.

Ephraim, Y. and Malah, D. (1984). Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 32(6):1109–1121.

Ephraim, Y. and Trees, H. L. V. (1995). A signal subspace approach for speech enhancement. *IEEE Transactions on speech and audio processing*, 3(4):251–266.

Henrández-Lobato, J. M., Hoffman, M. W., and Ghahramani, Z. (2014). Predictive entropy search for efficient global optimization of black-box functions. In *Neural Information Processing Systems (NIPS)*, pages 918–926.

---

[2]Here, hyperparameters refer to training parameters such as learning rate, decay function or a network structure such as the number of layers, number of hidden units, type of layers *etc.*

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computing*, 9(8):1735–1780.

Ilievski, I., Akhtar, T., Feng, J., and Shoemaker, C. (2017). Efficient hyperparameter optimization of deep learning algorithms using deterministic rbf surrogates. In *AAAI Conference on Artificial Intelligence (AAAI)*.

Kingma, D. P. and Ba, J. (2015). Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Mnih, V., Heess, N., Graves, A., and Kavukcuoglu, K. (2014). Recurrent models of visual attention. In *Neural Information Processing Systems (NIPS)*, pages 2204–2212.

Nash, J. C. (2000). The (dantzig) simplex method for linear programming. *Computing in Science and Engineering*, 2(1):29–31.

Ranzato, M., Chopra, S., Auli, M., and Zaremba, W. (2016). Sequence level training with recurrent neural networks. *Proceedings of the International Conference on Learning Representations (ICLR)*.

Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. In *Neural Information Processing Systems (NIPS)*, pages 2951–2959.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Neural Information Processing Systems (NIPS)*, pages 3104–3112.

Tashev, I., Lovitt, A., and Acero, A. (2009). Unified framework for single channel speech enhancement. In *2009 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, pages 883–888.

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Journal of Machine Learning*, 8(3-4):229–256.

Xu, Y., Du, J., Dai, L.-R., and Lee, C.-H. (2014). An experimental study on speech enhancement based on deep neural networks. *IEEE Signal processing letters*, 21(1):65–68.

Zaremba, W. and Sutskever, I. (2015). Reinforcement learning neural turing machines. *CoRR*, abs/1505.00521.