Continuous Speech Emotion Recognition from Audio Segments with Supervised Learning and Reinforcement Learning Approaches

Mr. Fengbo Ma, Northeastern University

Fengbo Ma is a second-year master's student in Data Analytics Engineering at Northeastern University, the Department of Mechanical and Industrial Engineering. He holds a BME from Auburn University's Mechanical Engineering Department. Fengbo further enriched his academic journey with practical experience as an Alabama Board of Licensure for Professional Engineers and Land Surveyors (BELS) Engineering Intern.

Prof. Xuemin Jin, Northeastern University

Dr. Xuemin Jin is a teaching professor at the Department of Mechanical and Industrial Engineering at Northeastern University. He teaches two core courses for the Data Analytics Engineering Graduate Program, Data Management for Analytics and Data Mining in Engineering. His current research interests include emotion detection, remote sensing and atmospheric compensation. Before joining Northeastern University, Dr. Jin was a data scientist at State Street Corporation, a principal scientist at Spectral Sciences, Inc., a software engineer at eXcelon Corp, and a scientist at SerOptics, Inc. Dr. Jin received his Ph.D. in physics from University of Maryland at College Park. He was a postdoctoral at MIT and at TRIUMF Canada.

Continuous Speech Emotion Recognition from Audio Segments with Supervised Learning and Reinforcement Learning Approaches

1. Introduction

Emotion plays an important role in communications, conveying essential information beyond words. This is particularly evident in enhancing Human-Computer Interaction (HCI) and Speech Emotion Recognition (SER). The latter is a specialized area within Automatic Speech Recognition (ASR) and focuses on identifying human emotions, which is crucial to advancing HCI. Recognizing emotions in speech, such as anger or joy, allows AI systems to interpret and respond more effectively to human expressions.

Emotion recognition technology can be integrated into engineering education to improve learning efficiency and create a more responsive learning environment. Emotion speech recognition can be adopted to gauge students' emotional states during lectures, discussions, or assessments in a classroom environment. Immediate feedback can be provided to the instructor about the overall emotional engagement of the class or specific students. Implementing emotion recognition technology based on both speech and facial expression in online engineering courses can enhance the feedback and engagement mechanisms for students. Emotion recognition can be used to identify when students are struggling, bored, or disengaged. Based on these insights, the online learning platform can dynamically adapt the content, pace, or delivery method to re-engage students. Emotion-aware tutoring systems can detect when students are experiencing confusion or frustration. In response, the system can offer targeted assistance, explanations, or additional examples to support learning. The system might also encourage collaborative problem-solving or provide motivational messages to boost student confidence.

However, accurately capturing and assessing emotions can be challenging and potentially introduce biases. Conventional machine learning techniques such as Support Vector Machines (SVMs) have been employed in SER for their effectiveness in complex feature spaces. Deep learning approaches such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have achieved remarkable results in emotion detection from speech. These supervised machine learning approaches are usually applied post-speech and do not provide real-time recognition. This study explores Reinforcement Learning (RL) for real-time decision-making that simulates human cognitive processes. Our approach introduces a new neural network structure designed for rapid and accurate identification of speech emotional states, utilizing Mel-frequency Cepstral Coefficients (MFCC) features extracted from audio signals. The performance of the RL is evaluated using a consistent evaluation matrix to ensure accuracy and efficiency.

2. Speech Emotion Recognition (SER)

Speech emotion recognition is a subset of automatic speech recognition that focuses on identifying emotions from speeches. It involves analyzing a speaker's tone, pitch, tempo, and volume to determine their emotional state. This process is complex as it requires not only word recognition but also an understanding of the delivery that reflects various emotional states [1].

In utterance-level SER, emotions are classified for an entire spoken utterance, typically a complete thought or statement. Here the emotions are considered as attributes of the whole utterance, disregarding the temporal variations within it. The goal is to identify the dominant emotion conveyed in the utterance.

Frame-level SER delves into a more detailed analysis by breaking the speech into smaller segments, often milliseconds long [2]. This approach allows the detection of emotional changes within an utterance, providing a finer time granularity. It captures the dynamic nature of speech emotions, going beyond the scope of utterance-level classification by capturing discrete emotional changes over time. The two different methods are illustrated in Figure 1.

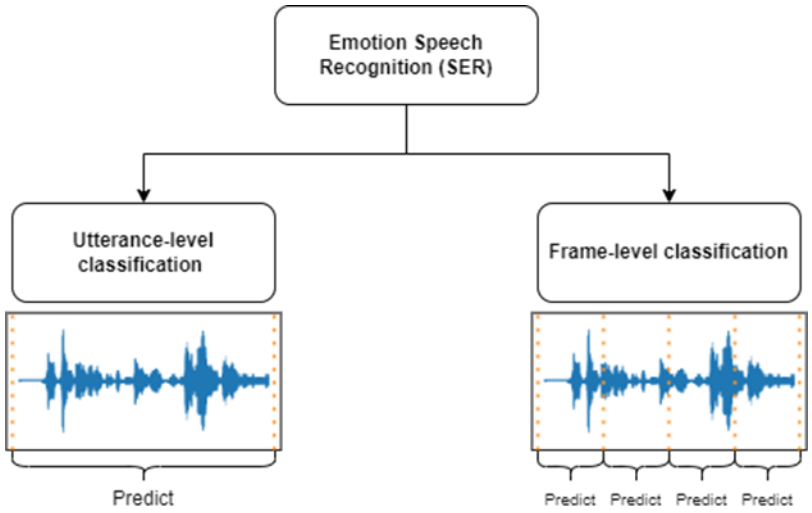

Figure 1: Speech Emotion Recognition (SER) methods

2.1 Mel-Frequency Cepstral Coefficient (MFCC)

Mel-frequency cepstral coefficients is a widely used feature extraction technique in the field of audio signal processing and speech recognition [3]. It was first proposed by S.B. Davis, and P. Mermelstein [4] in 1980. MFCC is crafted based on the auditory perception of humans, which typically does not register frequencies above 1 kHz.

Essentially, the MFCC framework is constructed to mirror the variable critical bandwidth of the human ear across different frequencies, making it highly relevant for speech emotion recognition. By applying the Mel scale to audio windows and extracting Cepstral Coefficients through Discrete Cosine Transformation, one obtains a series of numerical arrays that constitute the extracted features. These arrays consist of highly independent Cepstrum coefficients

alongside their corresponding energy terms. This characteristic makes MFCC a solid option to extract features from audio signals in the field of SER [5]. Figure 2 shows the entire process of extracting MFCC features from audio signals.

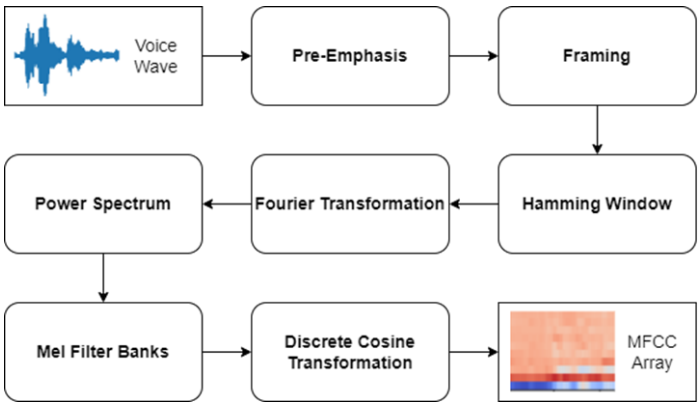

Figure 2: 1) Voice wave use as input; 2) Pre-Emphasis: Signal through high-pass filter; 3) Framing: Divide signal into 1ms frames; 4) Hamming Window: Clean out discontinuity signals due to framing; 5) Fourier Transformation: Apply Fourier Transformation; 6) Power Spectrum: Power Spectrum is computed to get the power of each frequency component; 7) Mel Filter Banks: Signals pass through a series of Mel-scale filters. The Mel scale is a perceptual scale that better represents human hearing; 8) Discrete Cosine Transformation (DCT): The DCT is applied to the log Mel spectrum. This step converts the log Mel spectrum into a time domain; 9) MFCC Array: MFCC coefficients are saved and output as an array

2.2 Reinforcement Learning

Reinforcement learning is a type of machine learning approach where an agent learns to make decisions by taking actions in an environment to achieve some notion of cumulative reward [6]. It is different from supervised machine learning in that correct input/output pairs are never presented, nor sub-optimal actions explicitly corrected. This learning process is similar to the way humans learn from the consequences of their actions. The agent aims to develop a strategy, or policy, that maximizes cumulative rewards over time. The feature of real-time decision-making in RL allows exploration and outcome interpretation, making it similar to how humans think [7]. This salient feature makes RL ideal for tasks needing quick and detailed understanding. RL nowadays has become a trending approach in the field of SER [8].

There have been some attempts in the realm of SER utilizing the power of reinforcement learning in recent years. The work of Lakomkin *et al*. [9] stands out for its innovative approach. Their study introduces a model called EmoRL that significantly enhances real-time emotion detection in speech by analyzing speech as it happens, without waiting for the end of an utterance. The authors employed deep reinforcement learning to train their model, optimizing for both accuracy and latency in emotion classification. This approach results in a model that competes in accuracy with established baseline models and offers quicker response times, making it particularly useful in scenarios where immediate emotional assessment is needed.

While both the EmoRL and our RL model strive for continuous emotion recognition, our study

approaches the analysis of each utterance from a different perspective. Instead of attempting to predict emotion without the full presence of an utterance, we segment each utterance into smaller frames and predict emotions within each frame, using a combination of reinforcement learning and other supervised learning techniques.

3. Methodology

To explore the feasibility of applying RL to the SER field, we first assume that the data is well-structured and labeled. We then knit the data into an environment for agents to solve as an RL approach. After the step of feature extraction and segmentation, the following steps are in parallel.

3.1 Feature Extraction and Segmentation

MFCC is used for feature extraction from original audio waves. We first extract 20 MFCC coefficients for each segment from windows of 50ms width using the librosa library in Python [10]. We then perform segmentation on audio to ensure each of the time-series waves have the same length.

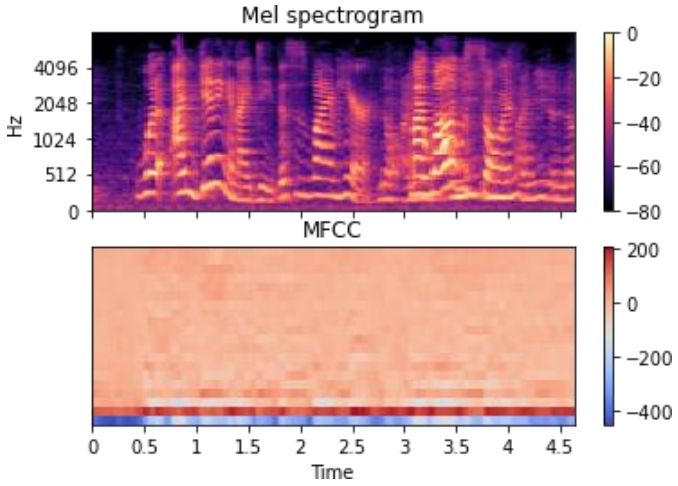

Figure 3: Visualization of MFCC array

Jain *et al*. [11] have found that the average auditory reaction time for adults is approximately 230 milliseconds. It is important to consider that an AI system performing a SER task requires additional time for classification. By trimming each audio segment to 50 milliseconds and allowing a buffer time for the AI system's response, the overall reaction time of the AI system is typically less than that of a human. Consequently, the SER task can be accomplished in real-time and continuously, making it efficient and effective.

3.2 Conventional Machine Learning Models

To establish baseline for comparison, we first apply 3 conventional machine learning models and evaluate their performances. The first model is the supports vector machine (SVM). The

characteristic of data handling in a higher dimensional space [12] makes the SVM a solid approach in the field of SER. In this study, we use the Radial Basis Function (RBF) kernel SVM [13], which applies RBF kernel function and is very effective in capturing complex and non-linear relationships in data, especially data like audio transformed MFCCs. The RBF kernel is expressed as

$$K(x, x') = e^{-\frac{\|x-x'\|^2}{2\sigma^2}} \tag{1}$$

where x and x' are two data points in the input space, σ is a hyperparameter that determines the spread or width of the RBF kernel. In this study we also add a flatten step to convert two-dimensional MFCC input into a one-dimensional array.

The second model is deep neural networks (DNN). It is effective in discerning complex patterns in extensive datasets [14] and has advanced rapidly in the field of SER. DNN has brought significant advancements to SER, transforming how machines understand and respond to human emotions conveyed through speech. In this study, the two-dimensional arrays transformed by MFCC are used as input, passing through three hidden layers. We use Rectified Linear function (ReLU) as the activation function,

$$\text{ReLU}(z) = \max(0, z) \tag{2}$$

and the Sigmoid function as the output function to generate the binary prediction.

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{3}$$

The third model is called Long Short-Term Memory Networks (LSTM), a special kind of recurrent neural networks (RNN). This model can overcome limitations in traditional RNNs such as vanishing and exploding gradients. LSTM have complex design [15] and each cell contains multiple steps: Forget Gate, Input Gate, Output Gate, Cell State Update, Final Cell State, and Hidden State. The equation of each step is given below [16]

$$f_t = \sigma(x_t U^f + H_{t-1} W^f) \tag{4}$$

$$i_t = \sigma\left(x_t U^i + H_{t-1} W^i\right) \tag{5}$$

$$o_t = \sigma(x_t U^0 + H_{t-1} W^o) \tag{6}$$

$$\tilde{C}_t = \tanh(x_t U^g + H_{t-1} W^g) \tag{7}$$

$$C_t = \sigma\left(f_t C_{t-1} + i_t \tilde{C}_t\right) \tag{8}$$

$$h_t = tanh(C_t) o_t \tag{9}$$

The use of multiple LSTM hidden layers enhances the model's complexity, aligning it more closely with the principles of deep learning [17]. The layered architecture of these networks introduces a hierarchical system, where each layer addresses a segment of the overall task and relays its output to the subsequent layer. A combination of LSTM and dense layers theoretically can achieve a better performance [18].

3.3 Reinforcement Learning Methods

The reinforcement learning process is guided by rewards: positive feedback for beneficial actions and negative feedback for detrimental ones. In this study, we set up an environment to perform emotion recognition tasks. The RL framework is shown below in Figure 4:

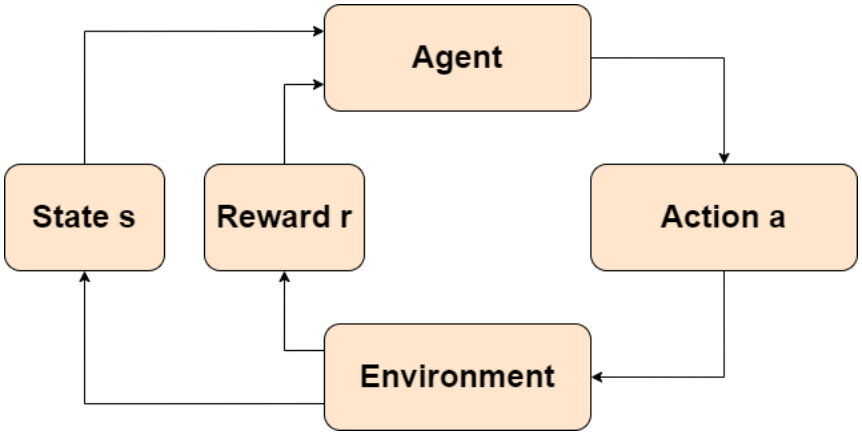

Figure 4. A framework of RL at given time step t

The agent interacts with the environment in discrete time steps t. The agent then makes an observation on State s over each t. Reward r is given to the agent by evaluating the result of the Action a acting upon the environment. The mathematical representation of maximizing the reward is the Bellman equation [19]

$$V(s) = \max_{\alpha} \sum_{s,r} r(s,a) + \gamma V(s') \tag{10}$$

where V is the current state value, $\gamma V(s')$ represents the discounted value of the next state. The reward at each step is calculated as

$$r_t = \sum_{i=0}^{t-1} r + r_{acc}. \tag{11}$$

We use the following two agent algorithms to the environment in our study. The first algorithm is

called Deep Q-Networks (DQN). It is an extension of Q-learning that uses deep neural networks to approximate the Q-value function. This works with discrete observation space and discrete action space. The key equation for DQN involves updating the weights of the neural network to minimize the loss function, which is typically the mean squared error between the predicted Q-values and the target Q-values. The function for DQN can be expressed as [20]

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha\big(r_{t+1} + \gamma Q'(s_{t+1}, a) - Q(s_t, a_t)\big). \tag{12}$$

The second algorithm is the Proximal Policy Optimization (PPO). It is a policy gradient method that aims to improve the stability and reliability of policy updates such as DQN. The algorithm utilizes data efficiency, making the algorithm a popular choice in model engineering. The objective function for PPO-clip is given by [21]

$$\theta_{k+1} = arg\max_{\theta} E_{s,a\sim\pi_{\theta_k}}[\min\big(\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)} A^{\pi_{\theta_k}}(s,a),\ clip(\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)}, 1-\epsilon, 1+\epsilon) A^{\pi_{\theta_k}}(s,a)\big)] \tag{13}$$

where ϵ is a hyperparameter representing learning step of each update towards one direction and $\frac{\pi_\theta(a|s)}{\pi_{\theta_k}(a|s)} A^{\pi_{\theta_k}}(s,a)$ the surrogate advantage. It is a measure on how policy π_θ performs relative to the early time step policy π_{θ_k}.

4. Result

4.1 Data Preparation

We use the Interactive Emotional Dyadic Motion Capture (IEMOCAP) dataset [22] in our study. This dataset comprises five sessions of recorded dialogues between two actors, each representing a different gender. It spans 12 hours of audio-visual content featuring ten actors and is categorized with labels such as anger, happiness, sadness, neutral, surprise, fear, frustration, and excitement. Each entry, typically a few seconds long, is an utterance annotated by 3 reviewers.

In this study, we select only utterances that are classified as anger and neutral, totaling 3411 audio clips. Here anger is the class of interest and set as class 1 and neutral as class 0. This selection aligns with our goal of examining transitions from a neutral state to a negativity state, simulating scenarios where, detection is crucial for an AI's planning and reaction in collaboration with human responders. An application in engineering education is to detect students' negative feedback during a lecture.

The audio clips from the IEMOCAP are typically with a duration under 120ms. In a total of 3411

utterances, 1708 utterances are labeled as neutral, and the rest 1703 utterances anger. Audio utterances are fed into the MFCCs pipeline for feature extraction. To make segments of 50ms, we eliminate utterances with a duration of less than 50ms. For clips longer than 50ms but less than 100ms, we make segments based on only the first 50ms. For any clips with a duration longer than 100ms, trim to 100ms, we split it into the first half of 50ms and the second half of 50ms, keeping with the same label. After the segmentation, we have 4907 segments with a duration of 50ms, where 2338 segments are neutral and 2569 segments anger. The partitioning is shown in Figure 5.

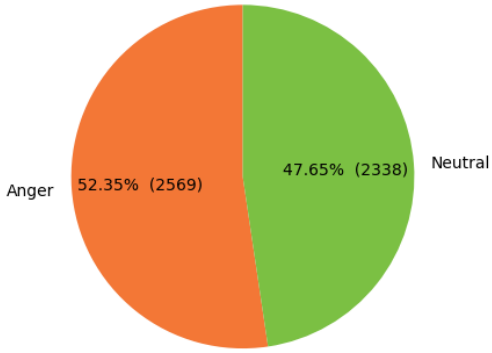

Figure 5. Data distribution by class after segmentation

4.2 Experiments

We executed several experiments concurrently to assess the performance differences between the conventional supervised machine learning models (SVM, DNNs, LSTM) and the reinforcement learning models. We designed scenarios to answer these two questions: 1). Is reinforcement learning a feasible approach for implementing real-time, continuous SER tasks? 2) Can reinforcement learning show comparable performance to SVM, DNN, and LSTM, which are commonly used in the industry and research fields of SER?

The entire workflow of this study is shown in Figure 6. With input data, we first treat the problem as a classic supervised machine learning problem and apply SVM, DNNs, and LSTM. These models are used to establish baselines that are used for evaluating the reinforcement learning models.

The dataset is nearly balanced between the two classes and is randomly split into a training set (70%), a development set (15%), and a test set (15%). Once the random split has been initialized, the training set is used as input for both conventional supervised machine learning models and the creation of the RL environment. The development set is utilized for grid searching hyperparameters for all models. The test set is exclusively used for reporting the final confusion matrices to prevent any data leakage.

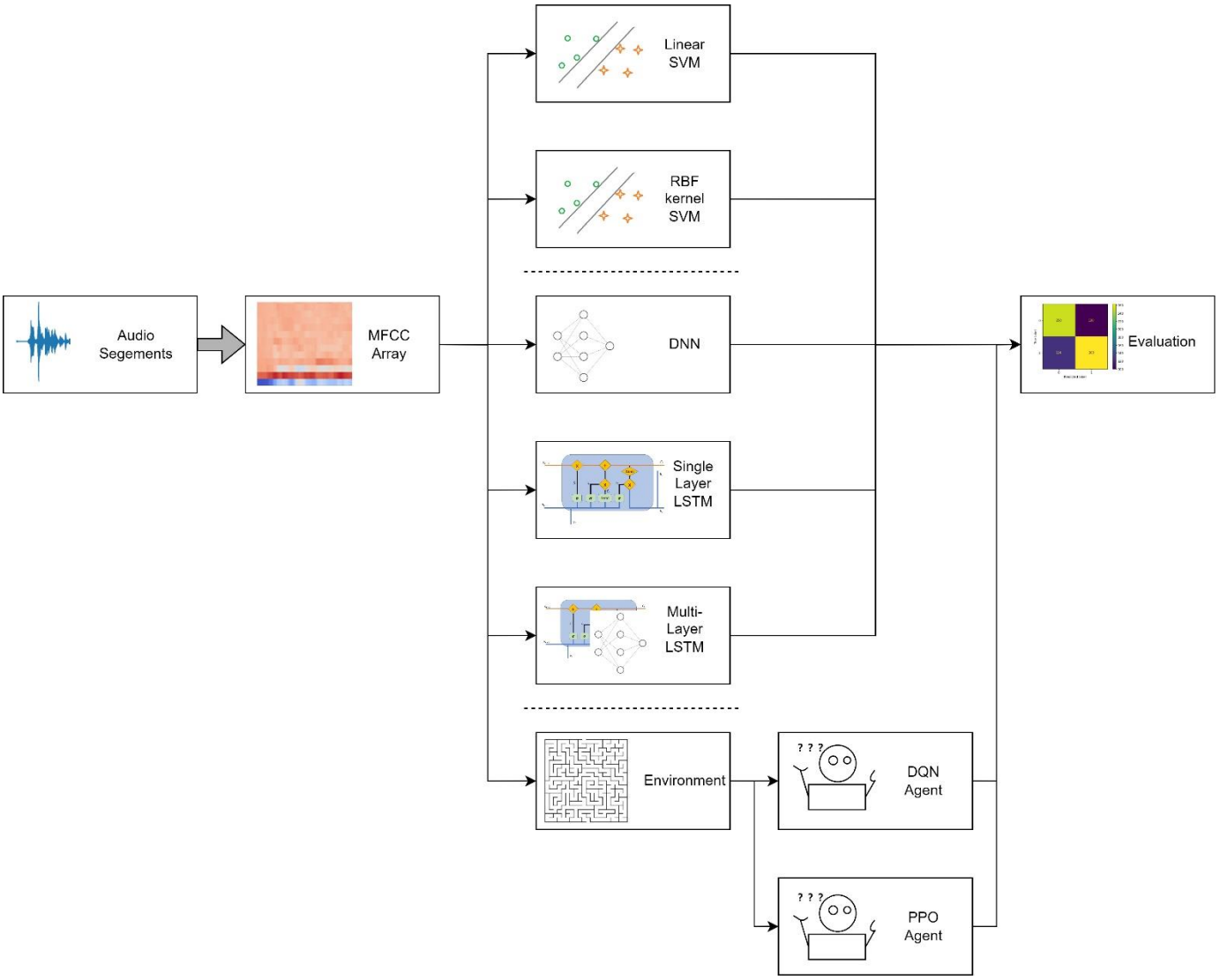

Figure 6. Experiment workflow used in this study

We adopt the standard performance metric for balanced binary classification problems: accuracy, F1 score, and precision. These metrics are particularly pertinent in the context of our research, which focuses on accurately identifying instances of anger emotion as class 1. Accuracy is essential as it reflects the overall correctness of the model in classifying both anger and neutral. The F1 score, a harmonic mean of precision and recall, is crucial in scenarios like ours where false negatives and false positives carry significant implications. We bring out precision by itself for its particularly relevant for our focus on anger detection.

The performance of the SVM models is summarized in Figure 7. Note that class 1 represents anger (class of interest), and class 0 neutral. Our study indicates that both the Linear SVM and RBF SVM have identical performance metrics on the test set.

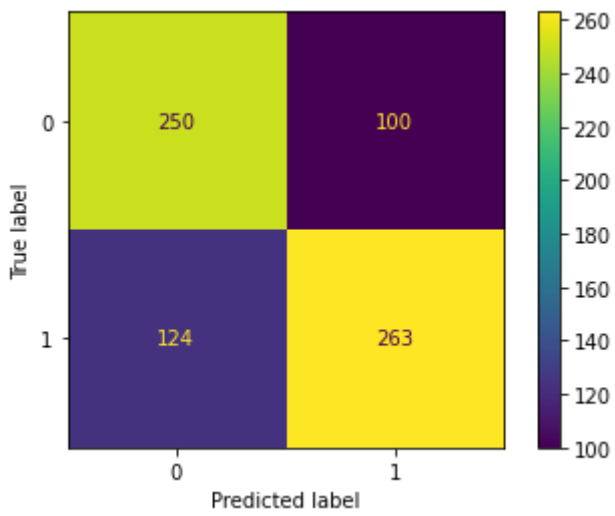

Figure 7. Confusion Matrix resulted from SVM models

The hyperparameters for deep learning approaches are set as follows: for the fully connected DNN model, we implemented three hidden layers following the flatten layer, each with 64 ReLU units, 64 ReLU units, and 4 ReLU units, respectively. This is followed by a single sigmoid function as the output layer. For the single-layer LSTM, we used 128 LSTM units after the flatten layer, leading to a sigmoid output layer. The multi-layer LSTM model stacks LSTM and NN layers within its hidden layers. It contains deeper hidden layers than the single-layer LSTM model. The structure of the multi-layer LSTM model starts similarly to the single-layer LSTM, with a flatten layer followed by 128 LSTM units and supplemented by additional layers of 64 ReLU units, 16 ReLU units, and 4 ReLU units, before connecting to the sigmoid output layer. Each of the ReLU units are attaching with L2 regularization to prevent overfitting. Adam are been chosen as optimizer for all three deep learning approaches.

The corresponding performance matrixes for deep learning methods are shown in Figure 8.

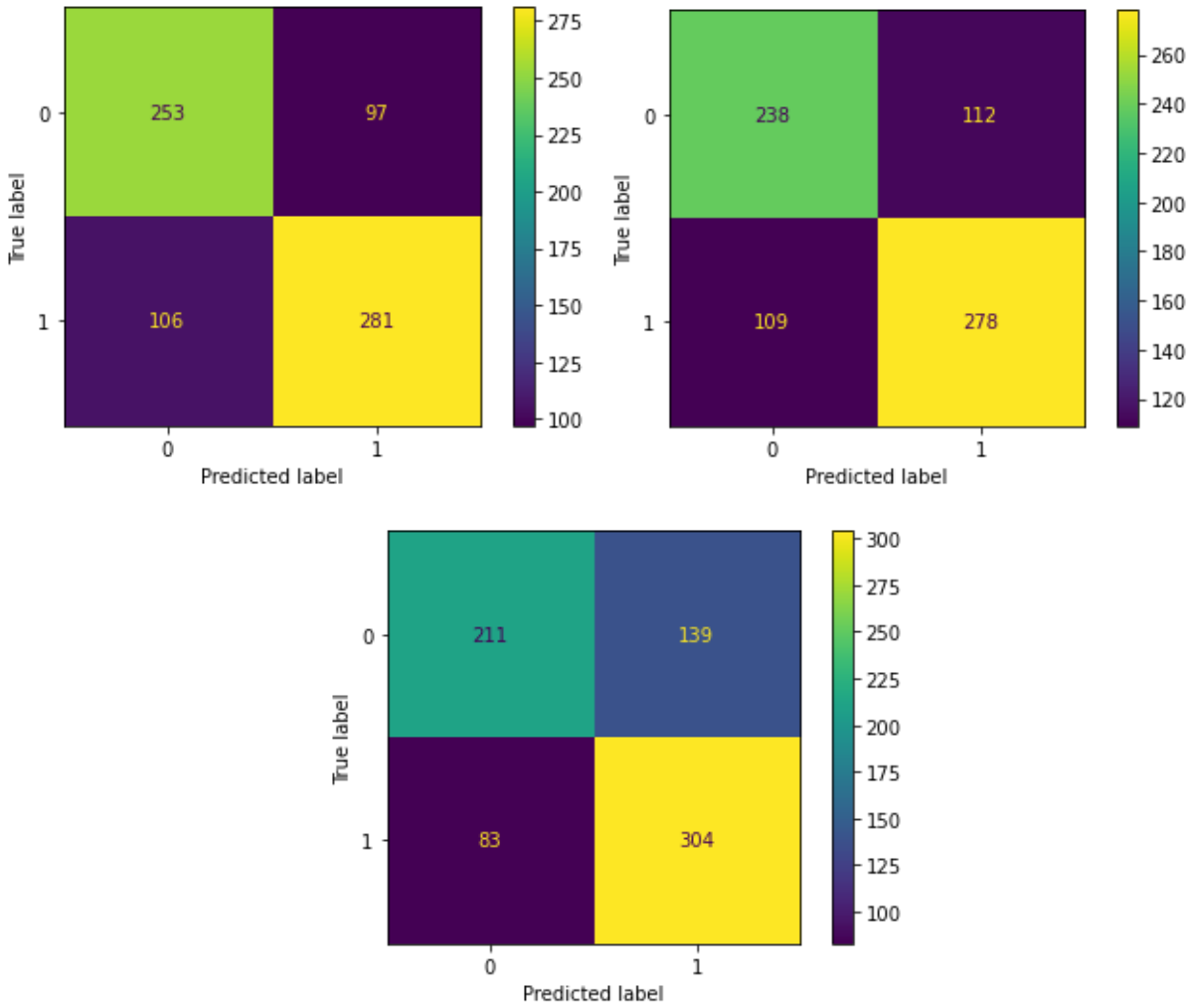

Figure 8. Confusion Matrixes for deep learning models (DNN-upper left, Single-layer LSTM-upper right, Multi-layer LSTM - bottom)

The training epoch vs. accuracy on test set for all three deep learning models is plotted in Figure 9. All three deep learning models have been trained over 2000 epochs. All three models can achieve over 70% accuracy over the test set.

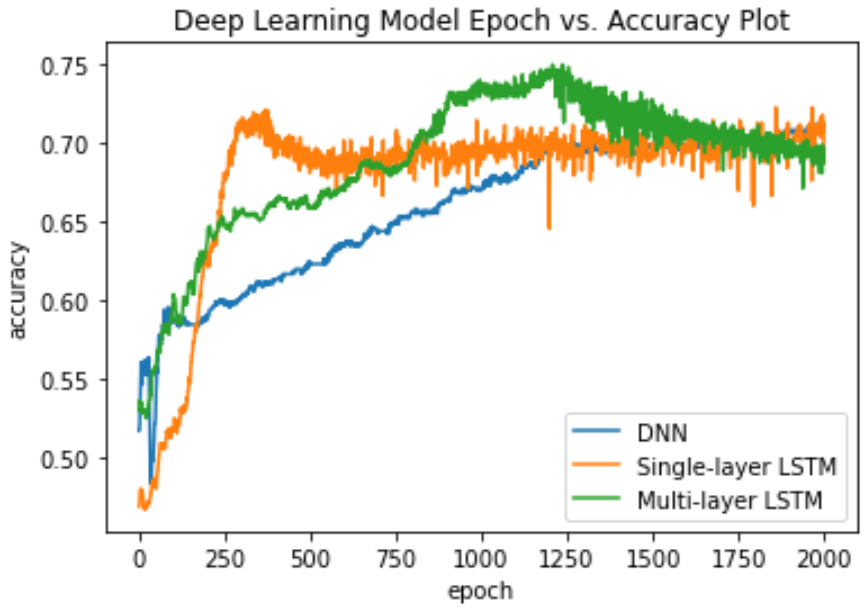

Figure 9. Accuracy vs. epoch for the deep learning models

Since we are using RL methods to target a supervised learning problem, the RL models share the same inputs. A custom reinforcement learning environment is set up for binary emotion classification tasks, utilizing gym framework [23]. This environment is tailored for training agents to accurately categorize data into one of the two classes. Segmented MFCC arrays are used as input for each observation.

The environment defines a discrete action space that contains binary actions, allowing the agent to choose between two actions 0 or 1. These actions represent the two possible emotions in the binary classification task. The observation space, on the other hand, is determined by the dimensions of the input samples. Thus, it is being considered as a discrete observation space. To prevent agent from "hard memorizing" the action for finite audio segments, we also applied randomization to shuffle the training set.

In each step of the environment, the agent is tasked with making a classification decision for the current segment. The reward mechanism is asymmetric: a correct prediction yields a reward of +1, while an incorrect prediction results in a relatively larger penalty of -5. This reward structure is designed to underscore the importance of accuracy in classification and to penalize errors more severely. At time step t, total reward r_t is the cumulative rewards from earlier steps, plus the prediction results r_{acc}, 1 or -5. After making a prediction, the environment advances to the next sample. An episode concludes after all samples in the dataset have been processed.

The performances of the RL models (DQN and PPO) are shown in Figure 10. Notice that the confusion matrix is not a commonly used evaluation method for the performance of RL methods. In this study, we treat the RL method as one possible solution towards a binary classification problem, so confusion matrix is a suitable measure of performance.

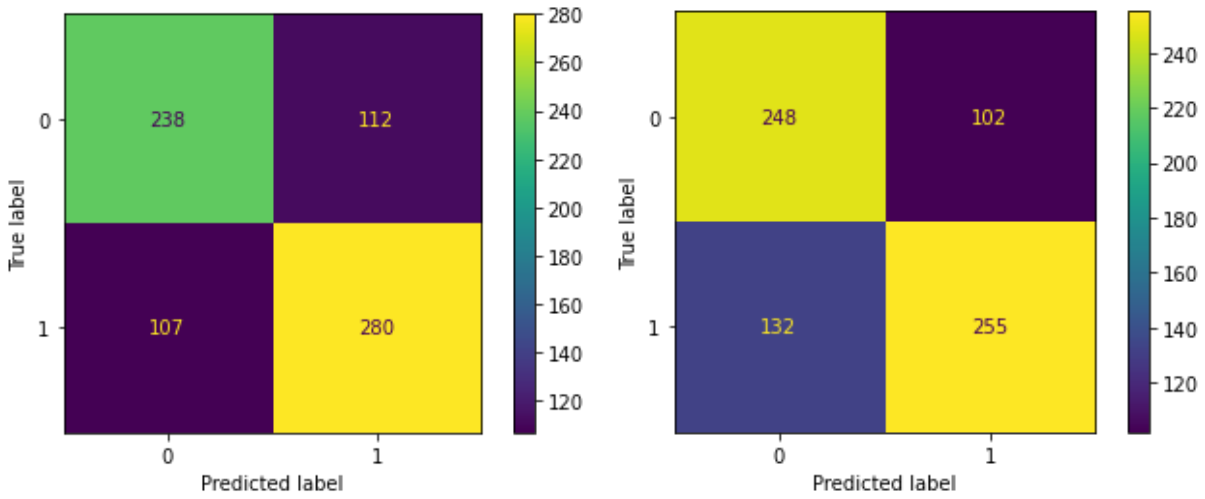

Figure 10. Confusion matrixes for RL models (DQN-left, PPO-right)

The accuracy, F1 score, and precision for all models in the study are summarized in Table 1. From this table, we observe that among the conventional machine learning algorithms, both LSVM and RSVM contain identical performance with 70% accuracy, a 70% F1-score, and slightly higher precision at 72%. This suggests a balanced performance in terms of both error minimization and positive class identification. For deep learning models, specifically DNN/RNN models, the multilayer DNN slightly outperforms others with 72% accuracy and F1-score, and 74% precision. This performance suggests that the multilayer DNN is more adept at recognizing complex patterns within this dataset, in contrast to both the single-layer and multi-layer LSTM networks, which display a slight reduction in their performance metrics. The RL models, DQN and PPO contain the unique capability of operating continuously, unlike the other models. DQN scores well with 70% accuracy and a notable 72% F1-score, indicating its effectiveness in balancing recall and precision. However, PPO performs barely worse with 68% accuracy and a 69% F1-score. This might be due to its more complex policy optimization approach, which is potentially less efficient with only thousands of records per class, typically considered short in training data. Despite these variations, there is no significant disparity in performance across the models; subtle differences exist in their accuracy, F1-score, and precision metrics. Notably, despite similar performance, RL models have the characteristic of continuously performing classifications once the model starts running. We can conclude that RL is a feasible solution for continuous SER.

Table 1. The accuracy, F1 score, and precision for all models over test data set

Algorithm	**Model**	**Accuracy**	**F1-score**	**Precision**	**Continuous**
SVM	Linear Support Vector Machine (LSVM)	70%	70%	72%	No
	Radial Basis Function Support Vector Machine (RSVM)	70%	70%	72%	No

DNN/RNN	Multilayer Deep Neural Networks (DNN)	72%	72%	74%	No
	Single-layer LSTM Networks	70%	70%	71%	No
	Multi-layer LSTM Networks	70%	70%	69%	No
RL	Deep Q-Networks (DQN)	70%	72%	71%	Yes
	Proximal Policy Optimization (PPO)	68%	69%	71%	Yes

5. Summary and Conclusion

In this study, we investigated the feasibility of continuous speech emotion recognition using reinforcement learning for real-time decision-making that simulates human cognitive processes. We introduced a new neural network structure designed for rapid and accurate identification of speech emotional states, utilizing MFCC features extracted from audio inputs. We used 3 supervised machine learning models to establish the performance baseline. The performance of the reinforcement learning models is then evaluated and compared against the baselines.

We found that the reinforcement learning models' performances are on par with the other supervised machine learning models, establishing their potential in speech emotion recognition. Moreover, the reinforcement learning models are effective for continuous real-time speech emotion recognition. We also noticed that accurate audio segmentation plays a crucial role in real-time speech emotion recognition.

We conclude that reinforcement learning's ability to continually integrate feedback greatly enhances speech emotion recognition tasks in practical settings. However, the current data, derived from controlled lab environments, differs significantly from more complex and noisy real-world data. Future work should include testing the model's robustness with diverse data sources and exploring its applicability to a broader range of emotions. Integrating this approach with other automatic speech recognition techniques, like Speech to Text, could further improve the SER performance.

Emotion recognition has tremendous potential in engineering education. Emotion recognition can be used to assess the overall learning experience and satisfaction of students. Feedback on emotional engagement can guide instructors in refining their teaching methods or improving course content. By incorporating emotion recognition technology into engineering education, institutions can create more adaptive and engaging learning environments that better cater to the needs and emotional states of individual students.

Finally, it is important to point out that emotional data can be sensitive, in particular when a model can constantly monitor and classify one's emotions based on speeches. If the model is being misused, it could lead to manipulation or unfairness. The collection and analysis of emotional data with real-world data can be seen as an invasion of privacy. People may not consent to or be aware that their emotions are being monitored and analyzed by AI systems.

While continuous speech emotion recognition has potential benefits, it is also crucial to address ethical concerns through rigorous standards, transparent practices, and careful consideration of privacy and consent.

Bibliography

[1] R. Elbarougy, "Extracting A Discriminative Acoustic Features from Voiced Segments for Improving Speech Emotion Recognition Accuracy," *International Journal of Advanced Research in Computer Science and Electronics Engineering,* vol. 8, no. 9, pp. 39-44, 2019.

[2] I. Trabelsi, D. B. Ayed, and N. Ellouze, "Improved frame level features and SVM supervectors approach for the recogniton of emotional states from speech: Application to categorical and dimensional states," *arXiv preprint arXiv:1406.6101,* 2014.

[3] J. de Lope and M. Graña, "An ongoing review of speech emotion recognition," *Neurocomputing,* vol. 528, pp. 1-11, 2023, doi: 10.1016/j.neucom.2023.01.002.

[4] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE transactions on acoustics, speech, and signal processing,* vol. 28, no. 4, pp. 357-366, 1980.

[5] S. Bedoya-Jaramillo, E. Belalcazar-Bolaños, T. Villa-Cañas, J. Orozco-Arroyave, J. Arias-Londoño, and J. Vargas-Bonilla, "Automatic emotion detection in speech using mel frequency cesptral coefficients," in *2012 XVII Symposium of Image, Signal Processing, and Artificial Vision (STSIVA)*, 2012: IEEE, pp. 62-65.

[6] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *Journal of artificial intelligence research,* vol. 4, pp. 237-285, 1996.

[7] X. Huang, W. Wu, and H. Qiao, "Connecting model-based and model-free control with emotion modulation in learning systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems,* vol. 51, no. 8, pp. 4624-4638, 2019.

[8] K. Zhang, Y. Li, J. Wang, E. Cambria, and X. Li, "Real-time video emotion recognition based on reinforcement learning and domain knowledge," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 32, no. 3, pp. 1034-1047, 2021.

[9] E. Lakomkin, M. A. Zamani, C. Weber, S. Magg, and S. Wermter, "Emorl: continuous acoustic emotion classification using deep reinforcement learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018: IEEE, pp. 4445-4450.

[10] B. McFee *et al.*, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, 2015, vol. 8, pp. 18-25.

[11] A. Jain, R. Bansal, A. Kumar, and K. D. Singh, "A comparative study of visual and auditory reaction times on the basis of gender and physical activity levels of medical first year students," (in eng), *Int J Appl Basic Med Res,* vol. 5, no. 2, pp. 124-7, May-Aug 2015, doi: 10.4103/2229-516x.157168.

[12] H. Bhavsar and M. H. Panchal, "A review on support vector machine for data classification," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET),* vol. 1, no. 10, pp. 185-189, 2012.

[13] B. Scholkopf *et al.*, "Comparing support vector machines with Gaussian kernels to radial basis function classifiers," *IEEE transactions on Signal Processing,* vol. 45, no. 11, pp. 2758-2765, 1997.

[14] O. I. Abiodun *et al.*, "Comprehensive review of artificial neural network applications to pattern recognition," *IEEE access,* vol. 7, pp. 158820-158846, 2019.
[15] S. Bordoni and S. Giagu, "Convolutional neural network based decoders for surface codes," *Quantum Information Processing,* vol. 22, no. 3, p. 151, 2023.
[16] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation,* vol. 9, no. 8, pp. 1735-1780, 1997, doi: 10.1162/neco.1997.9.8.1735.
[17] R. Pascanu, C. Gulcehre, K. Cho, and Y. Bengio, "How to construct deep recurrent neural networks," *arXiv preprint arXiv:1312.6026,* 2013.
[18] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *2013 IEEE international conference on acoustics, speech and signal processing*, 2013: Ieee, pp. 6645-6649.
[19] R. E. Bellman and S. E. Dreyfus, *Applied dynamic programming*. Princeton university press, 2015.
[20] H. van Hasselt, A. Guez, and D. Silver, "Deep Reinforcement Learning with Double Q-Learning," *Proceedings of the AAAI Conference on Artificial Intelligence,* vol. 30, no. 1, 03/02 2016, doi: 10.1609/aaai.v30i1.10295.
[21] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347,* 2017.
[22] C. Busso *et al.*, "IEMOCAP: Interactive emotional dyadic motion capture database," *Language resources and evaluation,* vol. 42, pp. 335-359, 2008.
[23] G. Brockman *et al.*, "Openai gym," *arXiv preprint arXiv:1606.01540,* 2016.