# Interactive Reinforcement Learning for Table Balancing Robot

**Haein Jeon**
Artificial Intelligence Robot Laboratory
Kyungpook National University
haeinjeon.knu@gmail.com

**Yewon Kim**
Artificial Intelligence Robot Laboratory
Kyungpook National University
yewonkim.knu@gmail.com

**Boyeong Kang**
Artificial Intelligence Robot Laboratory
Kyungpook National University
kby09@knu.ac.kr

## Abstract

With the development of robotics, the use of robots in daily life is increasing, which has led to the need for anyone to easily train robots to improve robot use. Interactive reinforcement learning(IARL) is a method for robot training based on human–robot interaction; prior studies on IARL provide only limited types of feedback or require appropriately designed shaping rewards, which is known to be difficult and time consuming. Therefore, in this study, we propose interactive deep reinforcement learning models based on voice feedback. In the proposed system, a robot learns the task of cooperative table balancing through deep Q-network using voice feedback provided by humans in real time, with automatic speech recognition(ASR) and sentiment analysis to understand human voice feedback. As a result, an optimal policy convergence rate of up to 96% was realized, and performance was improved in all voice feedback-based models.

## 1 Introduction

Service robots equipped with artificial intelligence technology are increasing in daily life. Examples include museum exhibition guide robot(Thrun et al., 1999), café-serving robot(Maxwell et al., 1999), and object carrying robot(Yokoyama et al., 2003). Robots increasingly perform tasks instead of or together with humans in various environments in daily life, and there has been an active research on robots that cooperate with humans(Calinon and Billard, 2007; Du et al., 2018).

Reinforcement learning (RL) —a robot learning technique– is a method in which an agent robot learns the action of obtaining maximum rewards through trial and error. In RL, rewards are generally given by agent action in a state, and if rewards are given through real-time human-agent interaction, it is called interactive reinforcement learning(IARL). Reward shaping(RS)(Ng et al., 1999)—an IARL method—is a technique in which a human trainer modifies reward functions by providing positive or negative feedback on the action of RL agents. In previous studies on IARL using natural language, the type of feedback is very limited using fewer than 10 feedbacks(Cruz et al., 2015; Tenorio-Gonzalez et al., 2010).To facilitate the use of robots, the need for a training system through various feedbacks is raised so that robot training can be naturally performed using various voice feedbacks.

Therefore, in this study, we propose an interactive deep RL model based on voice feedback to facilitate robot use. In the proposed system, a robot uses deep Q-networks(DQNs)(Mnih et al., 2013) to perform table balancing(Kim and Kang, 2020) tasks that require cooperation with humans and learns the RL policy through RS by human voice feedback. Using RS, a human trainer who collaborates table balancing task with robot and knows how to perform a task provides positive or negative feedback in real time about a robot's action via speech. Therefore, the agent provided with voice feedback learns the optimal policy—a policy that always leads to the balanced table state—faster and more naturally than when feedback is not used.

The rest of the paper is organized as follows. Section 2 explores the flow and limitations of prior IARL studies through related work, and Section 3 describes the proposed interactive deep RL system based on voice feedback. In Section 4, we describe the results of table balancing task training based on the proposed system, and compare the difference in learning performance against conventional DQN as a baseline and between voice feedback provision types. Finally, Section 5 concludes this study and suggests future research directions.

*Proceedings of Second International Combined Workshop on Spatial Language Understanding andGrounded Communication for Robotics*, pages 71–78
August 5–6, 2020. ©2021 Association for Computational Linguistics

## 2 Related Work

One of the strategies to improve learning performance in RL is that humans guide agents as external trainers. Representative examples include learning by imitation(Bandera et al., 2012), demonstration(Argall et al., 2009; Zhu and Hu, 2018), and by feedback. Among them, focusing on feedback-providing learning, we examine: (1) the design of IARL platforms that provide feedback through mouse or remote controls (Thomaz et al., 2006; Ullerstam and Mizukawa, 2004), (2) design of IARL algorithms(Knox and Stone, 2009; Griffith et al., 2013; Faulkner et al., 2020) and (3) studies of IARL through voice feedback(Tenorio-Gonzalez et al., 2010; Cruz et al., 2015). What these studies have in common is that RS reduces training time and fosters the robot or computer to learn the target action.

Regarding methods that adopt hardware input devices, some approaches use a mouse or remote control to design an IARL platform(Thomaz et al., 2006; Ullerstam and Mizukawa, 2004). Thomaz et al. (2006) revealed that IARL can improve robot's learning efficiency in an interactive Q-learning platform for cooking simulation robots, where humans can use mouse scrolls to provide feedback for robot actions by giving a number between -1 and +1. In the study of Ullerstam and Mizukawa (2004), AIBO robots learned action sequences such as singing after hearing a command from a human feedback given by remote control. However, in these prior studies on the design of such an IARL platform, input hardware, such as a mouse and remote control, is required to provide human feedback, which is difficult to see as a natural interaction with human.

Studies on developing IARL algorithms using human feedback include TAMER (Knox and Stone, 2009), Advise (Griffith et al., 2013) and REPaIR algorithm (Faulkner et al., 2020). In TAMER—an interactive reinforcement learning algorithm proposed by Knox and Stone (2009)—an agent learns a human feedback function by receiving two evaluation signals of positive and negative from the human on their keyboards; it was tested in Tetris game and mountain car problem. In Advise proposed by Griffith et al. (2013), a human modifies an agent's action choice probability, i.e., the policy, by giving the agent binary feedback—positive or negative. As a result, Advise outperformed conventional RL algorithms on game tasks such as Pac-Man. Faulkner et al. (2020) proposed the REPaIR algorithm, which estimates the correctness of human feedback over time; virtual and physical robots performed tasks, such as putting a ball into the box in a simulation environment and grasping cup in the real world. They proved that the REPaIR algorithm matched or improved the performance of conventional Q-learning algorithms. However, these approachs that focused on feedback learning algorithms for IARL required the design of an appropriate shaping function, and additional time to calculate rewards or policies. Moreover, in the framework proposed in this study, natural language voice feedback is directly integrated into a reward so that the amount of additional computation required for DQN learning is relatively small.

Studies that investigated IARL using natural language speech voice feedback itself include dynamic RS (Tenorio-Gonzalez et al., 2010) and IARL through speech guidance(Cruz et al., 2015). Tenorio-Gonzalez et al. (2010) showed that robots can use human voice feedback in RL to learn navigation tasks by assigning specific scalar rewards to feedback vocabulary, such as +100 to ”excellent” and -10 to ”bad” in simulation environments. Cruz et al. (2015) used voice commands and automatic speech recognition(ASR) to transcribe input voice commands, and then compared the input sentence and predefined lists using Levenshtein distance for cleaning tasks of robot arm agents. However, in these approaches using voice feedback, the RS function was designed by assigning a static reward value to a list of very limited words and sentences defined in advance. Therefore, when a feedback vocabulary that has not been defined in the list is input, the agent may have difficulty in learning. Moreover, the framework proposed in this study analyzes the positive and negative degrees of input voice feedback using a pretrained sentiment analysis module and converts it into a reward value. Therefore, no matter what feedback phrase is input, the sentiment polarity of voice feedback can be analyzed and used for DQN RL.

Through the examination of prior studies, we can summarize that IARL ordinarily improves learning performance. However, most studies did not adopt a natural interaction method with humans by requiring hardware input devices such as a keyboard or mouse. Further, studies using voice feedback used a small number of feedbacks. In this current study, we designed an IARL system for natural

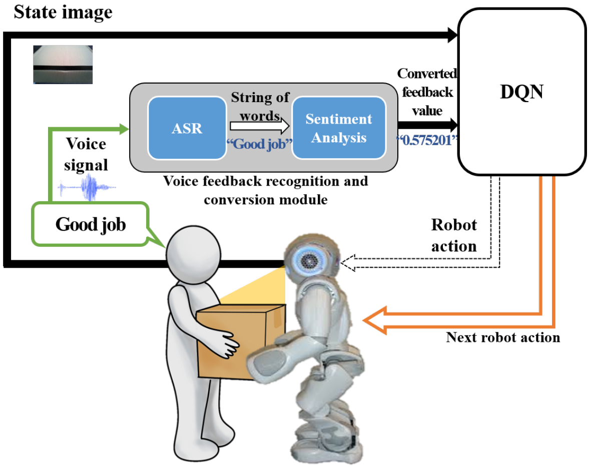


Figure 1: Interactive deep reinforcement learning model for table balancing based on human voice feedback

robot learning using voice feedback with ASR and sentiment analysis techniques to resolve these limitations.

## 3 Proposed Method

In this section, we describe the proposed deep RL framework for table balancing robots based on voice feedback. The task that the robot aims to learn is to maintain balance when lifting a table cooperatively with a human. Figure 1 shows the overall work diagram of the proposed system.

First, the robot takes a table state image with a camera and forwards it to the DQN. Next,the robot drives the balancing action predicted by DQN through image analysis. Then, the robot receives evaluative feedback from humans on the executed action; the voice feedback is input via the robot's microphone, converted to numerical values by voice feedback recognition and conversion module, and then incorporated into the environmental rewards of the DQN algorithm. Through repetition of the above process, the robot learns a policy in which the sum of environmental rewards and human voice feedback are maximized, and because of the learning, the robot can perform a cooperative table balancing task. In this work, the robot that will learn the table balancing task is Softbank's NAO robot, and the table is a rectangular box with width, length, and height of 31, 23, 6cm respectively. In addition, the table states to be used for learning were imaged using the lower camera mounted on the NAO robot.

**Algorithm 1** Interactive Deep Q-Network Based on Voice Feedback

```
Initialize action-value function with random weights $\theta$
Initialize target action-value function $\hat{Q}$ with random weights $\theta^- = \theta$
for $episodes = 1, 20000$ do
    Initialize sequence
    for $t = 1, T$ do
        Get table state image $s_t = x_t$
        With probability $\epsilon$ select a random action $a_t$
        Otherwise select $a_t = \text{argmax}_{a \in A} Q_t(s_t, a_t)$
        Execute action $a_t$ and observe reward $r_t$ and image $x_{t+1}$
        if Human trainer provides voice feedback $f_t$ on state $s_t$ then
            Let $r_t \leftarrow r_t + f_t$
        end if
        $$y_t = \begin{cases} r_t & \text{if episode done at step } t+1 \\ r_t + \gamma \max_{a' \in A} \hat{Q}(s', a'; \theta^-)) & \text{otherwise} \end{cases} \quad (1)$$
        Perform a gradient descent step on
            $L(\theta) = \mathbb{E}[(y_t - Q(s_t, a_t; \theta_t))^2]$
        with respect to the network parameters $\theta$
        Every 5 steps reset $\theta^- = \theta$
    end for
end for
```

### 3.1 Deep Reinforcement Learning Process Based on Voice Feedback

The robot in the proposed system uses the DQN to recognize the table state image and output the table balancing action based on human voice feedback. A DQN combines Q-learning with a deep convolutional neural network to estimate a state–action value function (Q function) given an input image and action.

Depending on the degree of raising and balancing state of the table, the human action states are divided into five in our system: up $(s_{up})$, keep $(s_0)$, down $(s_{down})$, up a lot$(s_{upup})$ and down a lot $(s_{downdown})$. The subscripts of $s$ represent human actions. The robot executes the table balancing action $a$ by adjusting the knee joint drive value. Five robot actions are defined depending on the direction and degree of table movement: $a_{up}$, $a_{up}$, $a_0$, $a_{down}$, and $a_{down}$.

Algorithm 1 represents the training process of an interactive DQN based on voice feedback. This training process is identical to the DQN training process;an interactive voice feedback-based process is added after the robot action operation. The input state $s$ is a table image$(x_t)$,which is an RGB image of $128 \times 170$ size representing the balance status of the table imaged by the robot camera.

| **Agent action** | **Reward** |
|---|---|
| Reaching the target state | +0.5 |
| Returning undefined action | -0.5 |
| Reaching non-target states | -0.3 |

Table 1: DQN environmental reward model.

The environment selects a table state image from the training dataset and feeds it to the robot, which is a DQN agent. The robot determines the action in the current time step according to the $\epsilon$-greedy policy, which selects a random action with a probability of $\epsilon$ for exploration. If no random action is selected, the agent chooses the action that maximizes the value of the Q function. The Q function that DQN aims to predict is as follows:

$$Q_{\pi}(s,a) = \mathbb{E}_{\pi}\sum_{t=1}^{\infty}\gamma^{t}r_{t} \tag{2}$$

where $r$ is the reward that the robot receives when it moves to the next state from the current state by performing the action. The Q function is represented as the expected value of the cumulative reward received when executing the action $a$ in state $s$, and $\gamma$ is the discount rate, which reduces the influence of the Q value in the future state.

After executing an action, the agent receives evaluative voice feedback from human and environmental rewards. Table 1 defines the environmental rewards of the proposed system. The environment provides a positive reward of +0.5 when the robot reaches the target state, the balancing maintenance state ($s_0$). A negative reward of $-0.3$ is given when the agent outputs an action that reaches a state other than the target. Finally the agent receives negative reward of $-0.5$ when returning an undefined action other than the one in the balancing task model in Kim and Kang (2020)'s work, such as returning $a_{down}$ while recognizing the human action state as $s_{upup}$.

Interactive voice feedback is a human speech evaluation of the robot's action. After checking the balance state of the table that has changed by the robot's action, the human provides positive voice feedback when the robot reaches the target state, and negative voice feedback otherwise. The provided voice feedback is converted into a numerical value through the voice feedback recognition and conversion module, and then added to the RL environment rewards. When the human provides voice feedback, the robot uses both feedback and environmental reward; and without feedback, the robot uses only environmental reward for learning. In Subsection 3.2, the voice feedback recognition and conversion module is described in depth.

In Algorithm 1, $\theta$ stands for the parameters of neural networks. DQN considers $y_t$ as a target and proceeds learning in a direction that reduces the error of $y_t$ and estimated $Q(s_t, a_t)$ by neural networks. Therefore, the DQN model is updated in every episode via the loss function $L(\theta)$, which computes the mean squared error. With a repetitive update of $\theta$ in the direction of minimizing $L(\theta)$, the Q function gets closer to the optimal state-action value function, and the agent learns the optimal action in the given state. Through this process, the robot can train DQN for table balancing with human voice feedback. To incorporate voice feedback in the DQN framework, we implemented voice feedback recognition and conversion module.

## 3.2 Voice Feedback Recognition and Conversion Modules

The voice feedback recognition and conversion module analyzed whether input voice feedback evaluated the robot's action positively or negatively. The voice feedback recognition and conversion module, shown in Figure 1, consisted of two processes: ASR and sentiment analysis.

First, the robot received an voice feedback signal from the microphone. ASR transcribed the signal into a character string and output it. We adopted Google Cloud speech-to-text as the ASR system, a cloud-based service that supported speech input and corresponding transcription in real time. This ASR system supports online streaming and offline voice audio processing, which was suitable for the agent's learning environment in our experimental setting.

Using a string of sentences obtained through ASR, sentiment analysis identified the positive and negative degrees of voice feedback phrases. The analyzed sentiment was returned in real value between $-1$ and 1 with positive and negative feedback being closer to +1 and $-1$. Moreover, if ASR could not correctly recognize speech signal, this module takes feedback as 'none' and only uses environmental reward. Google Natural Language API was used for sentiment analysis because of the ease of processing and modifying the sentiment analysis results in the implementation process.

| Feedback phrases | Converted value |
|---|---:|
| Well done | 0.8 |
| Fine | 0.6 |
| That is not how you do it | −0.699 |
| Try again | −0.5 |

Table 2: Examples of feedback phrase with converted numeric value.

## 4 Experimental Results

In this section, we discuss the construction of a feedback dataset for the experiment, evaluation of the voice feedback recognition and conversion module, and verification of the proposed interactive deep RL model through experiments.

### 4.1 Voice Feedback Dataset and Recognition Rate

First, we constructed the voice feedback phrase dataset to test the proposed DQN model from corpora. The corpora used to build the dataset were Sentiment lexicon (Hu and Liu, 2004), AFINN lexicon (Nielsen, 2011), and Classroom English(Hong and Sohn, 2013). A total of 100 feedback dataset phrases were extracted for experiments from the corpora, with 50 positive feedback phrases and 50 negative feedback. The feedback phrases were mainly short sentences or words that evaluated actions. Table 2 shows an example of some feedback phrases in the dataset and their converted sentiment analysis values which were incorporated in the RL reward function.

As a result of testing the recognition accuracy of Google Cloud speech-to-text, which is the ASR used in this study, the average sentence recognition rate was 86% using the built feedback phrase dataset. Three times of tests with the feedback phrase datasets on Google Natural Language APIs showed an average sentence recognition rate of 96%. An accuracy of less than 100% meant that the agent might receive an erroneous reward signal due to the malfunction of the voice feedback recognition and conversion module. In this study, all cases in which wrong rewards were given from malfunction of ASR or sentiment analysis were considered, and it was confirmed via experiments that using interactive voice feedback could foster the agent's target task learning despite such errors.

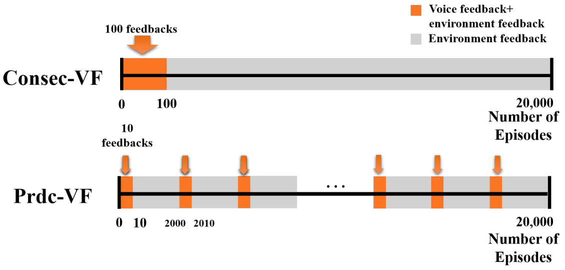


Figure 2: Comparison of Consec-VF and Prdc-VF model

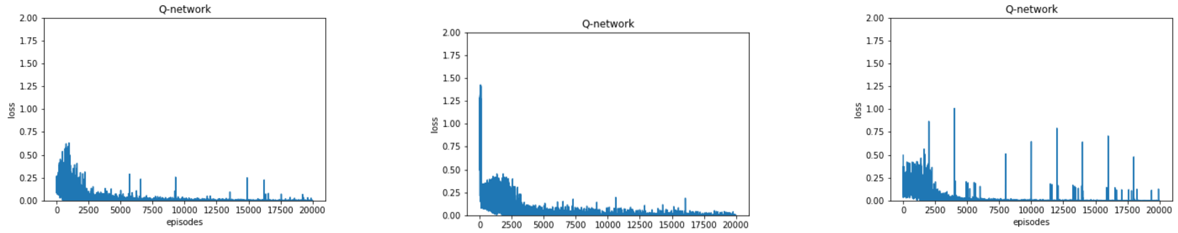
| Parameter | Value |
|---|---:|
| Learning rate $\alpha$ | 0.001 |
| Discount factor $\gamma$ | 0.9 |
| Epsilon $\epsilon$ | 20 |
| Number of episodes | 20,000 |
| Number of voice feedbacks | 100 |

Table 3: Hyperparameters of DQN training.

### 4.2 Interactive Voice Feedback DQN Model

In this paper, we employed two voice feedback models: consecutive voice feedback (Consec-VF) and periodical voice feedback (Prdc-VF) models (Figure 2). During the training, the human can provide (1) Consec-VF in the early stages of learning, or (2) Prdc-VF throughout learning. Consec-VF provided 100 consecutive feedback earlier in training, and Prdc-VF provided 10 feedbacks every 2,000 episodes. Training was conducted in simulation where random state images are given in every episode and human trainer provides voice feedback via microphone while observing the next state. We also run experiments on a physical NAO robot as a proof of concept, and robot training video can be found at this link. (http://air.knu.ac.kr/index.php/evolutionary-cooperative-robot-development-using-distributed-deep-reinforcement-learning) We compared the two feedback-providing models with conventional DQN without voice feedback as a baseline. Additional four optimizer comparison experiments were conducted on Consec-VF.

We conducted 30 experiments for each model setting and evaluated the performance by calculating the optimal policy convergence rate after the training. Hyperparameter settings for training DQNs are shown in Table 3. All hyperparameter settings, except the number of voice feedbacks, were equally applicable to both the proposed IARL model and baseline model–DQNs.



(a) Baseline

(b) Consec-VF

(c) Prdc-VF

Figure 3: Loss graph of models

| Optimizer | Baseline | Consec-VF | Prdc-VF |
|---|---|---|---|
| SGD | 80% | **86%** | 80% |
| Adam | 73 % | **96%** | 60% |

Table 4: Optimal policy convergence rate of 3 experimental model

We analyze the difference in model performance by the two methods of providing interactive voice feedback: Consec-VF and Prdc-VF. Voice feedback was provided 100 times out of 20,000 episodes (Table 3), and other episodes only used environmental rewards from Table 1. The Consec-VF model is designed to intensively feed voice feedback at the beginning of learning to establish the initial learning direction, whereas Prdc-VF model is designed to reflect human feedback steadily in the overall learning process so that human feedback could be consistently reflected.

Table 4 shows the results of experiment with two optimizers by applying the hyperparameter settings of Table 3 to the two voice feedback models and baseline DQNs. First, for the Consec-VF model, the optimal policy convergence rate was 86% and 96% when SGD and Adam optimizers were used, showing higher performance than the baseline with optimal policy convergence rates of 80% and 73% , respectively. Particularly, the convergence rate of 96% where 29 of 30 experiments learned optimal policies with Adam optimizer showed that combining Consec-VF with DQN significantly improved model performance.

Moreover, the Prdc-VF model showed lower performance than the Consec-VF and baseline models, which could be analyzed by training loss graphs. Figure 3 shows the training loss of the baseline, Consec-VF, and Prdc-VF models. In Figure 3-(a) and -(b), the loss stably converged to zero in the Consec-VF baseline model. However, in the Prdc-VF model in Figure 3-(c), loss spikes were observed during the training process. We analyzed that the intermittent intervention of voice feedback interfered with the convergence of losses during the training, resulting in a lower performance of the Prdc-VF model compared with others.

| Optimizer | Baseline | Consec-VF |
|---|---|---|
| SGD | 80% | **86%** |
| Adam | 73 % | **96%** |
| Adagrad | 43 % | **56%** |
| Adadelta | 63 % | **76%** |

Table 5: Optimal policy convergence rate of the baseline and Consec-VF models using four different optimizers

Experiment results showed that the Consec-VF model learned optimal policies better than baseline and Prdc-VF models. As in-depth experiments, we examine the results of the experiment by adding Adagrad, Adalta optimizers to the Consec-VF model to ensure that the use of Consec-VF consistently leads to model learning performance. Table 5 shows the optimal policy convergence rate after 30 experiments on the Consec-VF and baseline model on four optimizers. In all experiments Consec-VF showed improved optimal policy learning compared to the baseline DQN. These experiment results indicated that incorporating interactive voice feedback into DQN for table balancing tasks improved model learning performance in all optimizer settings.

## 5 Conclusion

In this study, we proposed an interactive deep RL model based on voice feedback for table balancing robot. The proposed system suggests DQN incorporating human voice feedback using ASR and sentiment analysis, where feedback given by humans are incorporated into the reward function. Experiment results show that the Consec-VF model, which provides Consec-VF early in learning, achieves an optimal policy convergence rate higher than the baseline model in all optimizer settings. There are several areas of extensions of our approach. Future direction for our work includes incorporating multimodal feedback to DQN using various robot sensors. We could also focus on deepening model optimization technique that improves learning performance of interactive RL model in varying settings. Robot could also learn when to use feedback and when to discard it or incorporate text semantics such as guiding robot behavior.

## Acknowledgments

This research was supported by the National Research Foundation of Korea funded by the Korean Government under grant NRF-2019R1A2C1011270 and by the BK21 FOUR (Fostering Outstanding Universities for Research) funded by the Ministry of Education, Department of Artificial Intelligence, Kyungpook National University, Korea (I20SS7610128).

## References

Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. 2009. A survey of robot learning from demonstration. *Robotics and autonomous systems*, 57(5):469–483.

JP Bandera, JA Rodriguez, L Molina-Tanco, and A Bandera. 2012. A survey of vision-based architectures for robot learning by imitation. *International Journal of Humanoid Robotics*, 9(01):1250006.

Sylvain Calinon and Aude Billard. 2007. Active teaching in robot programming by demonstration. In *RO-MAN 2007-The 16th IEEE International Symposium on Robot and Human Interactive Communication*, pages 702–707. IEEE.

Francisco Cruz, Johannes Twiefel, Sven Magg, Cornelius Weber, and Stefan Wermter. 2015. Interactive reinforcement learning through speech guidance in a domestic scenario. In *2015 international joint conference on neural networks (IJCNN)*, pages 1–8. IEEE.

G. Du, M. Chen, C. Liu, B. Zhang, and P. Zhang. 2018. Online robot teaching with natural human–robot interaction. *IEEE Transactions on Industrial Electronics*, 65(12):9571–9581.

Taylor A Kessler Faulkner, Elaine Schaertl Short, and Andrea L Thomaz. 2020. Interactive reinforcement learning with inaccurate feedback. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7498–7504. IEEE.

Shane Griffith, Kaushik Subramanian, Jonathan Scholz, Charles L Isbell, and Andrea L Thomaz. 2013. Policy shaping: Integrating human feedback with reinforcement learning. Georgia Institute of Technology.

Seonmi Hong and Jungmi Sohn. 2013. *Classroom English*. Hankukmunhwasa.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 168–177.

Yewon Kim and Bo-Yeong Kang. 2020. Cooperative robot for table balancing using q-learning. *The Journal of Korea Robotics Society*, 15(4):404–412.

W Bradley Knox and Peter Stone. 2009. Interactively shaping agents via human reinforcement: The tamer framework. In *Proceedings of the fifth international conference on Knowledge capture*, pages 9–16.

Bruce A Maxwell, Lisa A Meeden, Nii Addo, Laura Brown, Paul Dickson, Jane Ng, Seth Olshfski, Eli Silk, and Jordan Wales. 1999. Alfred: The robot waiter who remembers you. In *Proceedings of AAAI workshop on robotics*, pages 1–12. AAAI Press.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Andrew Y Ng, Daishi Harada, and Stuart Russell. 1999. Policy invariance under reward transformations: Theory and application to reward shaping. In *Icml*, volume 99, pages 278–287.

Finn Årup Nielsen. 2011. A new anew: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint arXiv:1103.2903*.

Ana C Tenorio-Gonzalez, Eduardo F Morales, and Luis Villasenor-Pineda. 2010. Dynamic reward shaping: training a robot by voice. In *Ibero-American conference on artificial intelligence*, pages 483–492. Springer.

Andrea Lockerd Thomaz, Cynthia Breazeal, et al. 2006. Reinforcement learning with human teachers: Evidence of feedback and guidance with implications for learning performance. In *Aaai*, volume 6, pages 1000–1005. Boston, MA.

S. Thrun, M. Bennewitz, W. Burgard, A. B. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. 1999. Minerva: a second-generation museum tour-guide robot. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, volume 3, pages 1999–2005 vol.3.

Mans Ullerstam and Makoto Mizukawa. 2004. Teaching robots behavior patterns by using reinforcement learning: how to raise pet robots with a remote control. In *SICE 2004 Annual Conference*, volume 1, pages 143–146. IEEE.

Kazuhiko Yokoyama, Hiroyuki Handa, Takakatsu Isozumi, Yutaro Fukase, Kenji Kaneko, Fumio Kanehiro, Yoshihiro Kawai, Fumiaki Tomita, and Hirohisa Hirukawa. 2003. Cooperative works by a human and a humanoid robot. In *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, volume 3, pages 2985–2991. IEEE.

Zuyuan Zhu and Huosheng Hu. 2018. Robot learning from demonstration in robotic assembly: A survey. *Robotics*, 7(2):17.