

FlappyBird Competition System: 인공지능 수업의 경쟁 기반 평가 시스템의 구현

손의성^{*}, 김재경^{**}

FlappyBird Competition System: A Competition-Based Assessment System for AI Course

Eisung Sohn^{*}, Jaekyung Kim^{**}

ABSTRACT

In this paper, we present the FlappyBird Competition System (FCS) implementation, a competition-based automated assessment system used in an entry-level artificial intelligence (AI) course at a university. The proposed system provides an evaluation method suitable for AI courses while taking advantage of automated assessment methods. Students are to design a neural network structure, train the weights, and tune hyperparameters using the given reinforcement learning code to improve the overall performance of game AI. Students participate using the resulting trained model during the competition, and the system automatically calculates the final score based on the ranking. The user evaluation conducted after the semester ends shows that our competition-based automated assessment system promotes active participation and inspires students to be interested and motivated to learn AI. Using FCS, the instructor significantly reduces the amount of time required for assessment.

Key words: Online Judge, Competition-Based Learning, Gamification, AI Education, Automated assessment

1. 서 론

프로그래밍 수업에서 제출물을 수작업으로 평가하는 일은 교수자에게 많은 시간과 노력을 요구하는 일이다. Online Judge와 같은 자동 평가 시스템을 도입하는 것은 교수자의 평가 시간을 절약하는데 있어 유용한 접근이다. 학생들은 자동 평가 시스템을 통해 객관적인 평가 결과와 피드백을 얻을 수 있고 교수자는 수작업으로 평가에 할애해야 하는 시간을 질문 답변이나 개인 피드백과 같이 보다 더 유용한 교육 활동에 할애하여 결과적으로 교육의 질을 향상시킬

수 있기 때문이다.

그러나 기초 인공지능 과목에 있어서는 기존의 알고리즘 평가 중심의 Online Judge 시스템을 활용하는데 제약이 있다. 과목의 특성상 제출되는 문제는 알고리즘 자체 보다는 주어진, 혹은 제한된 알고리즘의 틀 안에서 어떠한 신경망 모델의 구조로 설계할 것인가, 또 정확도를 향상시키기 위해 어떻게 가장 적합한 하이퍼파라미터의 값을 찾을 것인가의 반복적인 시행착오 과정에 대한 합리적인 평가 방안이 필요하기 때문이다. 따라서 컴퓨터 프로그래밍 언어 교육에 활용되고 있는 기존의 Online Judge 방식과

※ Corresponding Author : Eisung Sohn, Address: (21983) Room 425, Libertas Hall A, 85 Songdogwahak-ro, Yeonsu-gu, Incheon, Korea, TEL : +82-32-749-3054, E-mail : esohn@yonsei.ac.kr

Receipt date : Feb. 23, 2021, Revision date : Apr. 16, 2021
Approval date : Apr. 20, 2021

^{*} University College, Yonsei University

^{**} University College, Yonsei University
kim.jk@yonsei.ac.kr

※ This research was supported by National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (2019R1G1A1010839).

는 다른 접근 방식의 자동 평가 시스템의 활용이 요구된다.

본 논문은 기초 인공지능 수업에 적용하기 위한 경쟁 기반 자동평가 시스템인 FlappyBird Competition System (FCS)을 제안하고 이를 비전공생을 대상으로 하는 대학의 교양 인공지능 수업에서 활용한 결과를 고찰한다.

YCS1003(인공지능의 이해와 활용)은 딥러닝 프로그래밍을 중심으로 인공지능의 전반에 대한 이해를 돕는 것을 목표로 하는 기초 인공지능 수업이다. Python과 NumPy를 이용하여 간단한 신경망을 구현하는 기본적인 개념에서 출발하여 PyTorch를 이용한 CNN의 구현을 통해 딥러닝 프로그래밍을 다루고 최종적으로는 OpenAI Gym 환경의 강화학습을 다루는 내용으로 구성되어 있다. 수업은 1주일에 2시간의 이론 수업과 1시간의 실습 수업으로 편성되며 이론 수업 중간 중간 코드를 실행하거나 문제를 푸는 등의 코딩 실습이 병행되는 실습 위주의 내용으로 진행된다. 제안 시스템이 적용된 학기말 프로젝트는 두 종류의 프로젝트 중 강화학습을 사용한 게임 AI를 제작하는 주제로 진행되며 PyGame으로 구현된 FlappyBird 게임 환경에서 인공지능 에이전트를 훈련하여 높은 점수를 획득하기 위한 시행착오의 과정에서 신경망의 특성을 학습하는 것을 목표로 한다. 최종 학점의 평가는 출석(10%), 실습과제(45%), Kaggle과제(15%), 프로젝트1(30%), 프로젝트 2(20%)로 이루어져 있는데 이 중 제안 시스템은 프로젝트 1항목에 적용되었다.

프로젝트 컴피티션은 정해진 날짜와 시간에 열리는 6개의 컴피티션 라운드에 온라인으로 참여하는 방식으로 진행되며 준비기간 동안 설계하고 튜닝하고 훈련한 신경망 모델을 사용하여 자동으로 게임을 플레이하면서 각 라운드가 종료되는 시점에 서버에 결과를 제출하는 방식으로 참여가 이루어진다. 라운드 별로 결정된 순위는 K-means clustering 알고리즘을 활용하여 5개의 점수군으로 나누어지며 모든 컴피티션 라운드들에서 획득한 점수들을 합산하여 자동화된 평가 점수가 도출된다. 여기에 더하여 학생들이 제출한 코드와 축적된 데이터에 대한 간결한 정성 평가 점수를 더하여 최종 점수를 평가하는 방식이다.

학기가 종료된 후 수행된 학습자 만족도 평가를

통해서는 제안 시스템의 진행과 평가과정 전반의 만족도에 대해 응답자의 다수가 제안 시스템의 방식이 인공지능에 대한 흥미와 참여도를 높이는 데 유익하였다는 긍정적인 반응을 나타내었으며 제안 시스템이 인공지능 수업에서 교수자의 시간을 절약함과 동시에 기초 인공지능 수업에서 활용될 수 있는 다양한 평가 도구 중의 하나로써 활용되기에 유용하다는 것을 확인할 수 있었다.

2. 이론적 배경

2.1 Online Judge

학습 과정에서 많은 문제를 접해보고 연습하는 것은 프로그래밍 교육에서 매우 중요한 요소로 여겨진다. 그러나 제출된 소스코드를 하나씩 점검하고 실행하여 결과를 평가하는데는 많은 시간과 노력이 요구되기 때문에 이러한 평가체계를 자동화하려는 시도의 일환으로 Online Judge가 도입되었다[1].

Online Judge는 프로그램의 출력값 뿐 아니라 시간과 메모리 제약의 초과 여부, 런타임 에러 여부를 판단하여 학습자에게 피드백하기 때문에 학습자는 단시간에 자신의 제출물에 대한 평가를 확인할 수 있고 교수자는 과제의 평가에 대한 부담을 크게 덜 수 있기 때문에 학생들과의 질문 답변 등 추가적인 교육에 더 많은 역량을 쏟을 수 있다[2].

김미혜(2007)는 PAGS라는 프로그래밍 과제 평가 시스템을 제안하였다. 이는 웹을 기반으로 교수자가 프로그래밍 과제 평가 절차를 총괄적으로 관리할 수 있어 적절한 피드백과 함께 학생들의 과제를 일괄적으로 쉽게 평가하며 표절을 용이하게 검사할 수 있는 시스템이다[3].

정종광(2010)은 단순히 대회의 평가용으로만 이용되고 있는 기존의 Online Judge와는 달리 교육 현장에서 학생들의 프로그래밍 학습 및 평가 용도로 활용하기 위한 Online Judge를 제안하였다. 기존 유명 Online Judge 시스템의 대부분의 기능을 구현하고 추가된 Pre-compiler를 통해 과제 마다 주어진 제약 사항에 대한 선행 검사를 실시하는 초급용 평가 기능을 제공하여 기존 Online Judge에서 볼 수 없던 다양한 피드백을 제공한다는 점이 특징이다[4].

장원영(2014)은 기존 Online Judge 방식의 시스템이 채점 데이터 세트가 공개되지 않기 때문에 오답처

리의 원인을 확인할 수 없다는 문제를 인식하고 학습자의 메타인지와 동기를 촉진하도록 하는 자동평가 시스템을 제안하였다[5].

2.2 경쟁기반 평가 시스템

경쟁을 도구로서 활용하는 것은 학습 환경에서 긍정적인 효과를 가지고 있는 것으로 여겨지는데 이는 학생들의 학습 활동에 있어 분명한 목표와 동기를 부여하기 때문이다[6].

Ibáñez 등 (2014)은 C 프로그래밍 언어를 가르치는데 있어 게임화를 사용한 연구들을 고찰하였고 경쟁의 방식이 학생들로 하여금 좀 더 높은 점수를 받고자 의욕적으로 임한다는 사실을 발견하였다[7].

Souza 등 (2017)은 소프트웨어공학 수업에서 순위표의 사용에 대해 연구하였는데 학생들은 순위표 상에서 자신의 위치를 다른 학생들과 비교하고 자신의 성과를 높이기 위해서 학습 습관을 바꾸는 등의 변화가 관찰되었다고 보고하였다[8].

Coore 등 (2019)은 프로그래밍 교육에 있어서 Hacker Rank라는 경쟁 플랫폼을 사용하여 자동 채점방식을 활용하였는데 실시간 순위를 보여주도록 하지만 순위를 직접 성적 평가에 사용하기보다 통과 갯수를 사용하도록 하였다. 학생들은 Hacker Rank를 사용하지 않은 경우보다 학습에 보다 동기력을 가지고 적극적으로 참여한다는 사실을 발견하였다[9].

학생들이 경쟁하는 환경에서는 장점도 있지만 반대로 부정적인 영향도 있을 수 있는데 순위가 뒤쳐짐으로 인한 부정적인 감정과 학습 경험에서의 즐거움이 감소되기 때문이다[10,11].

CHEN 등 (2013)은 경쟁 환경에서의 부정적인 감정을 극복하기 위한 방법으로 아바타와 같은 대리캐릭터를 사용하는 경쟁 방식, Surrogate Competition을 제안하였는데 이를 통해 직접적인 경쟁상황을 피하면서도 학생들의 학습 성과와 의욕을 높일 수 있었다[12].

기존의 경쟁 기반 평가 시스템이 주로 실행 결과의 일치나 실행 속도, 알고리즘의 효율성 등 일반적인 컴퓨터 프로그래밍 언어 교육과정에 적합한 방식인 반면 제안 시스템은 어떤 정답이나 정해진 방법을 활용하는지의 여부 보다는 신경망 구조의 설계와 가중치 훈련, 하이퍼파라미터 튜닝을 통해 반복적인 시행착오의 과정을 통해 성능을 향상시키는 인공지능

프로그래밍의 핵심 요소들을 평가하는데 보다 초점을 두어 설계되었다[13]. 자체적으로 성능을 점검하는 평가모드와 실시간으로 공개되는 순위와 그래프가 공개되는 Sandbox 컴피티션 모드를 통해 학생들이 장기간의 시행착오 과정 동안 계속해서 흥미와 도전의식을 가지고 참여하도록 유도하였으며 직접적인 경쟁으로 인한 부정적인 감정을 완화하기 위해서는 자유롭게 원하는 별명 등을 등록하여 참여할 수 있도록 배려하였다.

3. 시스템의 구현

3.1 전체 구조

FCS 시스템의 전체 구조는 Fig. 1과 같다. 학생들에게 배포되는 기본 코드는 크게 네 가지 모듈로 나뉘는데 하나는 PyGame을 사용하여 Python으로 구현된, OpenAI Gym과 연동되는 FlappyBird 게임 환경 모듈, 강화학습 알고리즘 DQL(Deep Q-Learning)이 구현되어 있는 에이전트 모듈, 신경망의 구조와 하이퍼파라미터를 수정하고 전체 프로그램을 실행하기 위한 메인 모듈, 그리고 훈련된 모델들을 관리하고 서버와 통신하는 헬퍼 모듈로 구성되어있다. 서버는 AWS Lambda를 이용하여 구현되었으며 REST API를 제공하여 학생들의 코드와 연동되도록 하였다. 컴피티션 페이지와 관리자 페이지는 AWS S3의 static web 방식으로 운영되며 내부적으로 Ajax를 활용하여 REST API를 호출하는 방식으로 구현되었다. 컴피티션 페이지는 Fig. 2와 같으며 라운드 별로 수집된 학생들의 제출 결과를 순위표와

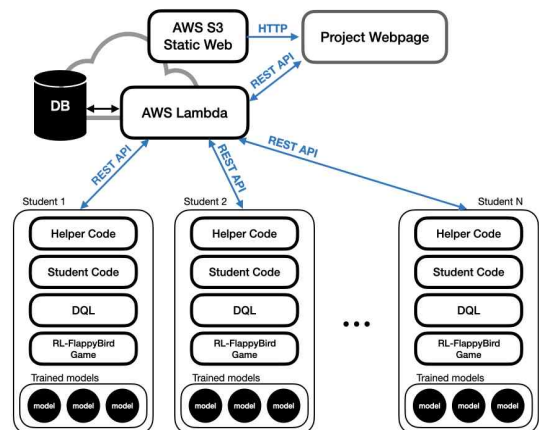


Fig. 1. Overview of the system.

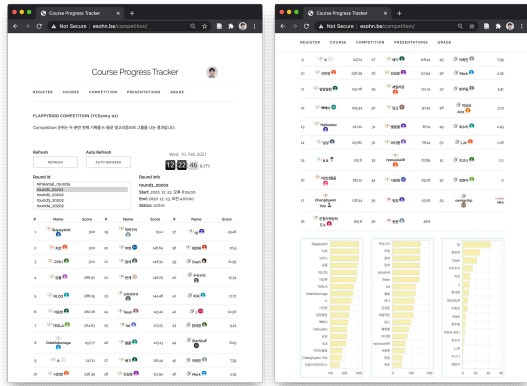


Fig. 2. Screenshot of the competition webpage.

그래프를 통해 실시간으로 확인할 수 있도록 하였다.

3.2 FlappyBird 게임 환경

FlappyBird는 OpenAI Gym 라이브러리와 연동되는 강화학습 환경으로 구현되었다. OpenAI Gym은 강화학습의 여러 알고리즘의 성능을 동일한 상황에서 비교하며 알고리즘을 개발하기 위한 라이브러리이다. OpenAI Gym에는 다양한 게임들이 제공되는데 FlappyBird는 이 OpenAI Gym 라이브러리에서 사용할 수 있는 써드 파티의 형태로 구현되었으며 그래픽스 등 기타 게임 요소들을 구현하기 위한 라이브러리로는 PyGame을 사용하였다.

강화학습 환경으로써 FlappyBird 게임에서 제공하는 상태는 Fig. 3에 보여지는 것처럼 4가지 값 (v_y , dx , dy_1 , dy_2)이며 이는 각각 FlappyBird의 y축 속도, 가장 가까운 파이프와의 x축 거리, 파이프 겹 판단과의 y축 거리, 파이프 겹 하단과의 y축 거리를

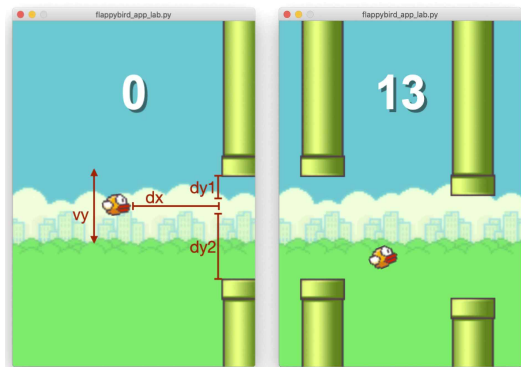


Fig. 3. Screenshot of the FlappyBird game.

정규화한 값이다. 학생들이 구성하게 될 신경망은 이 4개의 값을 입력으로 하도록 하며 출력으로는 1 (점프)과 0 (점프하지 않음)의 값을 출력하도록 모델링하게 된다. FlappyBird가 파이프를 하나 넘을 때마다 점수가 1점씩 상승하게 되며 FlappyBird가 땅이나 하늘에 닿거나 파이프에 닿게 되면 해당 에피소드는 종료된다.

3.3 신경망 설계 및 하이퍼파라미터 튜닝

학생들이 가장 먼저 해야 할 일은 주어진 게임 환경의 관측값들을 입력으로 하는 가장 적합한 형태의 신경망을 설계하는 것이다. 또한 신경망의 구조 및 형태를 설계함과 동시에 가장 좋은 성능을 보이는 하이퍼파라미터를 찾는 과정을 동시에 수행하게 된다. 주어진 코드를 실행하면 나타나는 메인메뉴는 Fig. 4에서 확인할 수 있으며 학습자가 수정할 수 있는 하이퍼파라미터의 종류는 Table 1에 나열되어 있다. 기본적인 신경망의 훈련은 (T) Train from scratch and Save 메뉴를 통해서 이루어지게 되는데 훈련 에피소드 수 (H2)를 지정한 후 H5~H13까지의 정해진 하이퍼파라미터 값으로 한번의 훈련을 수행한다. 하이퍼파라미터 튜닝은 (H) Hyperparameter Tuning 메뉴를 통해서 수행할 수 있는데 이를 통해 높은 성능을 내는 H5~H11의 하이퍼파라미터 값을 찾는 과정을 반복하게 된다. 하이퍼파라미터 튜닝은 횟수 H4 만큼 H5~H11에 지정된 범위 안에서 랜덤하게 대입하게 되며 에피소드 횟수는 H3만큼 수행하는 방식으로 이루어진다. 하이퍼파라미터 튜닝과 훈련을 반복하는 가운데 시스템은 텐서보드를 위한 로그를 기록하는데 텐서보드를 사용하면 Fig. 5의 Parallel

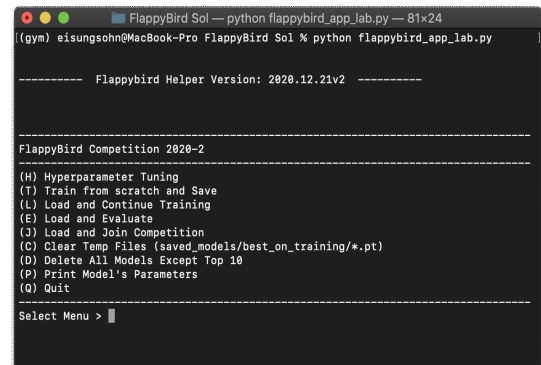


Fig. 4. Screenshot of the main menu.

Table 1. List of hyperparameters.

H#	Hyperparameter	Meaning
H1	NUM_EP_EVAL	Number of episodes on evaluting
H2	NUM_EP_TRAIN	Number of episodes on training
H3	NUM_EP_TUNE	Number of episodes on hyperparameter tuning
H4	NUM_ATTEMPTS_TUNE	Number of attempts on hyperparameter tuning
H5	LEARNING_RATE	Learning rate
H6	BATCH_SIZE	Batch size
H7	DROPOUT	Dropout ratio
H8	GAMMA	Reward discount rate
H9	REPLAY_MEMORY_CAPACITY	Replay Memory capacity
H10	OPTIMIZER_ID	Optimizer type (Adam, SGD, ...)
H11	MODEL_ID	Model ID
H12	IS_RANDOM_GAP	Whether using random pipe gap
H13	GAP_SIZE	Pipe gap size

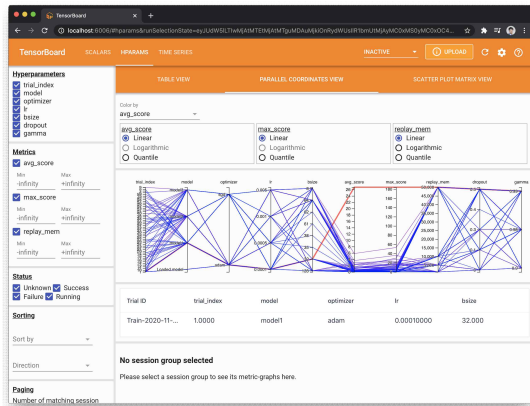


Fig. 5. Parallel coordinate view of TensorBoard.

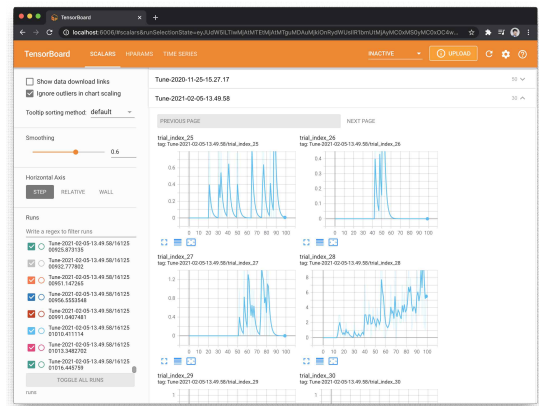


Fig. 6. Graphs generated during hyperparameter tuning.

coordinates view에서 어떤 하이퍼파라미터의 조합이 좋은 성능을 내는지 확인할 수 있으며 Fig. 6에서 처럼 개별 훈련 및 하이퍼파라미터 튜닝 도중 이루어졌던 학습의 추이에 대한 그래프를 분석하는 과정을 거치게 된다.

또한 시스템은 훈련 도중 중간결과와 훈련 종료 후 최종결과를 모델 파일로 저장하는데 저장된 모델을 불러와서 성능을 검증하기 위해 (E) Load and Evaluate 메뉴를 실행하거나 저장된 모델을 기본으로 새로운 하이퍼파라미터로 시도하고자 할 경우 (L) Load and Continue Training을 실행할 수 있다. 최종적으로 저장된 모델을 불러와서 공식 컴퍼티션에 참여할 때는 (J) Load and Join Competition 메뉴

를 실행한다.

3.4 컴퍼티션 및 평가

정식 컴퍼티션 라운드는 Table 2와 같이 총 6개의 라운드로 구성되며 각 라운드 마다 파이프의 수직 갭 크기가 고정 값으로 지정되거나 일정 범위에서 무작위로 발생하도록 지정되어 있다. 각 라운드마다 100개의 에피소드를 실행하여 평균 점수를 서버에 업로드하게 되며 이를 바탕으로 순위가 산정된다.

정해진 순위를 기준으로 점수를 평가하기 위해서 제안 시스템은 K-means Clustering 알고리즘을 사용하여 5개의 점수군으로 나누게 된다. 게임의 최대 스코어가 지나치게 높을 경우 에피소드에 걸리는 시

Table 2. List of competition rounds.

Round ID	Pipe Gap	Random Gap	Max Score
round1	125	False	300
round2	135	False	300
round3	145	False	300
round4	155	False	300
round5	165	False	300
round6	120~170	True	300

Table 3. Points by tier.

Tier	Points	Tier	Points
Tier 1	5.0	Tier 4	3.5
Tier 2	4.5	Tier 5	3.0
Tier 3	4.0	Tier 6	2.0

행시간이 지나치게 길어지는 문제가 있기 때문에 현재 모든 라운드는 300점을 최고 스코어로 하여 에피소드를 중단하도록 하였다. 이 때문에 K-means clustering 알고리즘을 사용하기 전 최고점인 300점을 따로 분리하여 가장 높은 배점을 할당하고 나머지 점수군에 대해서 높은 순서대로 점수를 할당하는데 각 점수군별 배점은 Table 3과 같다.

4. 학습자 평가

제안 시스템은 2020년 1학과 2학기 동안 코로나-19로 인한 비대면 온라인 수업으로 진행되는 환경에서 활용되었다. 학기 종료 후 진행된 설문문을 통해 제안 시스템을 사용한 평가방식의 만족도와 효용성,

공정성에 대한 학생들의 반응을 조사하였으며 총 58명이 학생들이 설문문에 응답한 결과가 Table 4와 Fig. 7에 제시되어 있다. 문항별 응답의 평균분포는 4.4에서 4.6909로 대체적으로 높게 나타났으며 크론바흐알파(Cronbach Alpha) 신뢰도는 0.84로 양호한 신뢰도 범위에 있음을 알 수 있다. 학생들은 제안 시스템을 통하여 시행되었던 프로젝트1 항목의 진행에 대해 전반적으로 높은 만족도를 나타내었으며, 제안 시스템을 통해 AI의 개념을 학습하는데 도움이 되었고 경쟁을 통해서 동기 부여가 되었으며 또한 제안 시스템이 공정하고 합리적인 평가 방식이라는 인식을 나타내었다. 주관식으로 제시된 기타 의견으로는 학생들이 간의 학습배경에 대한 편차를 고려하여 상대평가가 아닌 절대평가를 도입하는 것에 대한 의견과 현재보다 라운드의 구성을 보다 다양하게 하는 것, 그리고 저장된 모델을 이어서 훈련하는 과정에서 좋은 모델을 선별하기 어려운 점등에 대한 개선 의견들이 제시되었다.

5. 결 론

본 논문에서는 인공지능 수업을 위한 경쟁 기반의 평가 시스템인 FlappyBird Competition System을 구현하고 이를 기반으로 학기말 프로젝트를 진행한 결과를 제시하였다.

제안 시스템은 Online Judge로 대표되는 자동화된 평가방식의 장점을 가져오면서 인공지능 수업의 학습과 평가 항목에 적합한 기능들을 제공한다. 강화 학습을 이용하여 게임 AI를 고도로 훈련하는 것을 주제로 주어진 시간동안 신경망 구조의 설계와 가중

Table 4. Student survey results.

Survey questions	SD	D	N	A	SA	Var	Mean
I think the evaluation method of FlappyBird Competition is suitable as the evaluation method of a non-face-to-face environment.	0	0	4	14	37	0.3855	4.6000
I think the evaluation method of FlappyBird Competition was reasonable and fair.	0	1	3	13	38	0.4582	4.6000
I think FlappyBird Competition keep me motivated and challenged	0	4	5	11	35	0.8582	4.4000
I think FlappyBird Competition was helpful to learn AI concepts.	0	1	4	11	39	0.4945	4.6000
Overall, I am satisfied with the evaluation of FlappyBird Competition.	0	2	2	7	44	0.5045	4.6909

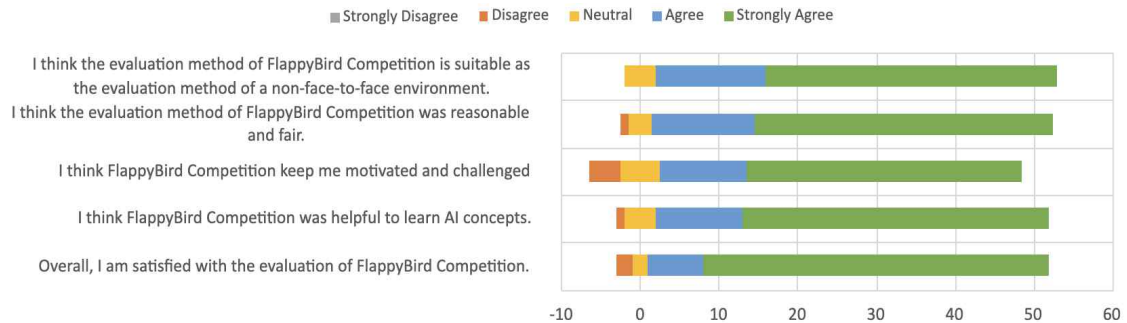


Fig. 7. Chart of the student survey results.

치 학습, 그리고 하이퍼파라미터 튜닝의 과정에서 겪게 되는 시행착오의 과정을 학생들이 흥미와 동기력을 가지고 주도적으로 참여하도록 유도하는 방식이다. 제안 시스템은 실시간 컴피티션에서 결정된 순위를 토대로 자동으로 점수군을 나누고 모든 라운드에서 획득한 점수들을 합산하는 방식으로 평가 결과를 제공한다. 학생들이 이 시행착오의 과정을 통해 축적된 데이터와 제출된 코드에 대한 간결한 정성 평가를 기본 점수로 하여 제안 시스템에서 얻은 정량 평가결과를 주로 활용하여 평가를 수행하였다.

학습자 평가를 통해서 제안된 시스템을 사용한 평가방식이 프로젝트 준비기간동안 실시간 경쟁을 통해 학생들에게 학습에 대한 흥미와 의욕을 고취시키는 면에서, 인공지능의 개념을 이해하도록 돕는 면에 있어서 유용한 방식임을 확인할 수 있었다. 또한 제안 시스템에서 제공하는 자동화된 평가 방식은 수업에서 평가에 할애되는 교수자의 시간적 부담을 덜고 수업 내용의 전달과 학생들과의 상호작용에 보다 많은 시간을 할애하는 측면에 도움이 될 것으로 기대된다.

향후 보다 넓은 범위의 난이도를 가진 게임 환경들을 도입하여 컴피티션을 다양화하는 것과 현재 정성적인 평가에 의존하는 요소들, 예를들어 신경망 구조의 설계와 하이퍼파라미터 튜닝의 분석과정에 투입하는 노력에 대한 자동화된 평가 기능들을 추가하는 것, 그리고 강화학습 이외의 주제에 대한 학습과 경쟁의 환경을 제공하는 측면에서 제안 시스템을 확장해 나갈 필요가 있다.

REFERENCE

[1] A. Kurnia, A. Lim, and B. Cheang, "Online

Judge," *Computers & Education*, Vol. 36, No. 4, pp. 299-315, 2001.

[2] Y. Luo, X. Wang, and Z. Zhang, "Programming grid: A Computer-aided Education System for Programming Courses based on Online Judge," *Proceedings of the 1st ACM Summit on Computing Education in China on First ACM Summit on Computing Education in China*, pp. 1-4, 2001.

[3] M. Kim, "Design and Implementation of an Automatic Grading System for Programming Assignments," *Journal of Internet Computing and Services*, Vol. 8, No. 6, pp. 75-85, 2007.

[4] J. Jeong, *Design and Construct of Programming Assessment System based on Online Judge for a Science High School student*, Master's thesis of Korea National University of Education, 2010.

[5] W. Chang and S. Kim, "Development and Application of Algorithm Judging System: Analysis of Effects on Programming Learning," *The Journal of Korean Association of Computer Education*, Vol. 17, No.4, pp. 45-57, 2014.

[6] E. Cagiltay, E. Ozcelik, and S. Ozcelik, "The Effect of Competition on Learning in Games," *Computers & Education*, Vol. 87, No. 1, pp. 35-41, 2015.

[7] M. Ibáñez, A. Di-Serio, and C. Delgado-Kloos, "Gamification for Engaging Computer Science Students in Learning Activities: A Case Study," *IEEE Transactions on Learning Te-*

- chnologies, Vol. 7, No. 3, pp. 291-301, 2014.
- [8] A. Souza, F. Constantino, F. Veadó, and L. Figueiredo, "Gamification in Software Engineering Education: An Empirical Study," *2017 IEEE 30th Conference on Software Engineering Education and Training*, pp. 276-284, 2017.
- [9] D. Coore and D. Fokum, "Facilitating Course Assessment with a Competitive Programming Platform," *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, pp. 449-455, 2019.
- [10] P. Vorderer, T. Hartmann and C. Klimmt, "Explaining the enjoyment of playing video games: The role of competition," *Proceedings of the Second International Conference on Entertainment Computing*, pp. 1-9, 2003.
- [11] M. Vansteenkiste and L. Deci, "Competitively Contingent Rewards and Intrinsic Motivation: Can Losers Remain Motivated?," *Motivation and Emotion*, Vol. 27, No. 4, pp. 273-299, 2003.
- [12] H. Chen and Y. Chen, "A Surrogate Competition Approach to Enhancing Game-based Learning," *ACM Transactions on Computer-Human Interaction*, Vol. 20, 2013.

- [13] H. Rim, "A Study on Teaching using Website Code.org in Programming Education based on Computational Thinking," *Journal of Korea Multimedia Society*, Vol. 20, No. 2, pp. 382-395, 2017.



손 의 성

2012년 연세대학교 컴퓨터과학과
(공학박사)
2012년 ~ 2014년 연세대학교 소프트웨어응용연구소 연구원
2015년 ~ 2018년 애니펜 수석연구원

2018년 ~ 현재 연세대학교 학부대학 조교수
관심분야: SW교육, 딥러닝, 컴퓨터그래픽스



김 재 경

2007년 연세대학교 컴퓨터과학과
(공학박사)
2009년 연세대학교 컴퓨터과학과
연구교수
2014년 연세대학교 컴퓨터과학과
객원교수

2017년 ~ 현재 연세대학교 학부대학 조교수
관심분야: SW교육, 데이터과학